

ИНФОРМАТИКА ЖӘНЕ АҚПАРАТ ТЕХНОЛОГИЯЛАРЫ

Орта білім беретін мекемелердің 10-сыныбы
оқушыларына арналған оқулық

1-басылымы

Өзбекстан Республикасы Халыққа білім беру министрлігі
бекіткен

Ташкент – 2017

УЎК 004(075.32)

КБК 32.81я722

Т 14

**Тайлаков Норбек Исакулович
Ахмедов Акром Бурхонович
Пардаева Мехринисо Данияровна
Абдуғаниев Абдували Абдулхаевич
Мирсанов Уралбой Мухаммадиевич**

Жалпы редакциясын басқарған – педагогика ғылымдарының докторы,
профессор Н.И.Тайлаков.

Пікір жазғандар:

А.Абдуқодиров – Ташкент мемлекеттік педагогикалық университетінің “Информатиканы оқыту әдістемесі” кафедрасы профессоры, педагогика ғылымдарының докторы;

С.Жуманазаров – ТҚХББҚҚДБЖИ-дің “Табиғи және анық пәндер бойынша білім беру” кафедрасының меңгерушісі, техника ғылымдарының кандидаты, доцент;

С.Эштемиров – Самарқанд Мемлекеттік университетінің “Информатика” кафедрасы доценті, физика-математика ғылымдарының кандидаты;

Б.Каримов – Ташкент қаласындағы 90-жалпы орта білім беретін мектептің жоғары санатты информатика және ақпарат технологиялары пәні оқытушысы.

Шартты белгілер:



– сабақтың басталуы;



– есінде сақта;



– сұрақтар мен тапсырмалар;



– үйге тапсырма.

Республикалық мақсатты кітап қорының қаржысы есебінен басылды.

ISBN 978-9943-4860-6-5

© Extremum-press, 2017

АЛҒЫ СӨЗ

Құрметті оқушылар!

Бұл оқулық Өзбекстан Республикасы Министрлер Кабинетінің 2017 жылғы 6 сәуірдегі “Жалпы орта білім мен орта арнаулы, кәсіптік-техникалық білімнің мемлекеттік білім стандарттары туралы” 187-санды қаулысымен бекітілген жалпы орта білімнің мемлекеттік білім стандартының, сондай-ақ жалпы орта білімнің информатика және ақпарат технологиялары пәні бойынша біліктілік талаптарының негізінде жаратылған.

Оқулықтың алғашқы тарауында Сендер төменгі сыныптарда өткен Microsoft Office Excel электронды кестесінде жұмыс істеуді жалғастырып, MS Excel 2010-да амалдар және формулалармен, функциялар кітапханасымен жұмыс істеуді, сондай-ақ логикалық және математикалық функцияларды пайдаланып, амалдарды орындауды үйренесіңдер. Оқулықтың екінші тарауында MS Access 2010 дерекқоры туралы түсінікке ие болып, оның басқару жүйелерімен, соның ішінде, дерекқорын жасау, дерекқорында кестелерді өзара байланыстыру, белгілі бір шаблон бойынша деректер іздеу және оларды қайта өңдеу тәсілдерімен танысасың. Оқулықтың үшінші тарауында Delphi бағдарламалау ортасында қосымшалар жарату негіздерімен танысасың. Атап айтқанда, қосымша терезесін пайда ету, оған дайын басқару объектілерін орналастыру және интербелсенді қосымшаларды жарату мүмкіндіктерін игересің. Бір сөзбен айтқанда, Сендер оқулықтағы білімді игеріп, үздіксіз түрде өздеріңді рухани және интеллектуалдық тұрғыдан дамытуға, кемелденуге ұмтылып, танымдық дағдыларыңды өз беттеріңмен тұрақты түрде арттырасыңдар және өз іс-әрекеттеріңді бағалау мүмкіндігіне ие боласыңдар деген үміттеміз.

Авторлар



I ТАРАУ. MS EXCEL 2010-ДА АМАЛДАР МЕН ФОРМУЛАЛАР

**СЕН БҰЛ ТАРАУДЫ ОҚЫП ТӨМЕНДЕГІ БІЛІМ, ДАҒДЫ,
БІЛКІТІЛІК ПЕН КОМПЕТЕНЦИЯҒА ИЕ БОЛАСЫҢ:**

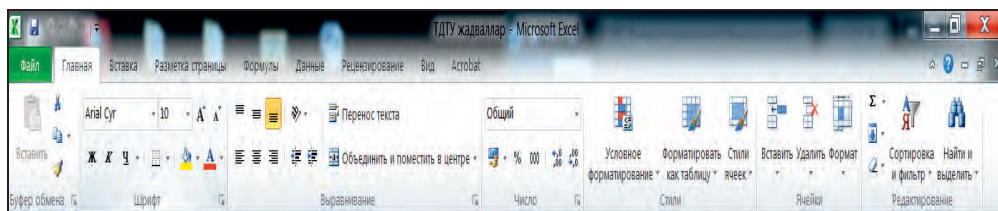
- Microsoft Excel 2010 электронды кестесінің мүмкіндіктерімен танысу;
- түрлі амалдарды орындау, өрнектерді есептеу;
- ұяшықтарға сілтеме (салыстырмалы, абсолют және аралас сілтеме) жасау;
- сілтемеден пайдаланып амалдарды орындауда көшірудің артықшылығын пайдалану;
- қарапайым және күрделі функциялардың графигін жасау;
- басқа параққа немесе кітапқа сілтеме жасау;
- функциялар кітапханасын пайдалану;
- функциялар аргументі терезесін, формула жолын пайдалану;
- функция атауын жазып орналастыруда автотолтырудың мүмкіндіктерін пайдалану;
- логикалық, математикалық, статистикалық функцияларға қатысты мысалдарды шешу тәсілдері жайлы білімге ие боласың.





1-САБАҚ. ҚАРАПАЙЫМ ӨРНЕКТЕРДІ ЕСЕПТЕУ

MS Excel 2010 электронды кестесінде жаңа интерфейсі негізінде **таспа** (ағылшын тілінде “Ribbon”) деп аталатын және негізгі терезенің жоғары жағында көп парақты өріс орналасқан.



MS Excel 2010 таспасының әрбір бөлігінде белгілі бір міндеттерді орындауға арналған түймешіктер кешені жинақталған:

- **Главная** – кестелерге деректерді енгізуге және редакциялауға;
- **Вставка** – кестеге сурет, диаграмма сияқты объектілерді орналастыруға;
- **Разметка страницы** – кесте парақтарын басып шығару үшін парақтардың шекарасын, тағы басқа да параметрлерін белгілеуге;
- **Формулы** – кестелердегі есептеулерді орындайтын түрлі формулаларды пайдалануға;
- **Данные** – кестелердің бағандары мен жолдарындағы деректерді сүзу, сұрыптау, нұсқасын көбейту, қажетсіздерін жоғату мен сыртқы ақпаратты енгізуге;
- **Рецензирование** – мәтіндік кателерді түзетуге және құжаттарды редакциялауға;
- **Вид** – кестелерді экранда сан алуан түрде көрсетуге сәйкестеуге арналған.

Басты терезенің жоғары сол бұрышында қол жетімді панел бар болып, онда есте сақтау, соңғы команданың күшін жою сияқты көп пайдаланылатын командаларды жылдам және оңай орындауға болады.

Осы панелге жаңа командаларды енгізу арқылы оның мүмкіндігін арттыруға болады.

Таспаның төменінде электронды кестенің жұмыс өрісі бар. Кесте көрінісіндегі файл MS Excel 2010-да кітап (Книга) деп аталады.

Ал кітап парақтардан (Лист) құралған. Стандартты жағдайда олар үшеу болып, қажеттілік туындаса көбейтуге немесе азайтуға болады.

MS Excel 2010-ның стандарт парағында әріптермен белгіленген 16348 баған және цифрлармен өрнектелген 1 048 576 жол бар, бағандар А әрпімен басталып XFD-мен аяқталады. Парақтағы жеке алынған әрбір ұяшық өз атауына және мекенжайына: **A3, C45** ие болады. Қарастырылатын ұяшықты белсенділендіру оның үстіне тышқанның сол түймешігін басу арқылы орындалады. Кестенің төменінде жағдай жолы бар болып, онда жұмыс өрісіне тиісті мәліметтер берілген.

Кестеге мән енгізіліп, амалдар орындалған болса, тышқан арқылы (мысалы, **B2:F5**) блоктарды ажыратып алуға болады.

1-жаттығу. **B1** ұяшығына 231-ді, **B2** ұяшығына 569-ды енгізіп, олардың қосындысын **B3**-те көрсет.

Орындау: **B3** ұяшығына “=” белгісін жазу арқылы MS Excel 2010 енгізу жағдайына өтеді. Содан соң тышқанның сол түймешігі **B1** ұяшығының үстіне басылғанда, осы мекенжай **B3**-те пайда болады. Келесі адымда “+” белгісі енгізіліп, кейін **B2** ұяшығының үстіне тағы да тышқанның сол түймешігі басылады. Нәтижеде **B3**-те **B1+B2** жазуы пайда болады. **Enter** пернесі басылған соң, орындалған амалдың нәтижесі **B3**-те пайда болып, Формулалар жолында **=B1+B2** деген өрнек жазылады.

	B3 =B1+B2												
	A	B	C	D	E	F	G	H	I	J	K	L	M
1		231											
2		569											
3		800											

MS Excel 2010-дағы тағы басқа амалдар тиісті тақырыптарда жаттығу ретінде көрсетіледі.

MS Excel 2010-да жұмыс істеу үдерісінде түрлі қателер пайда болуы мүмкін. Көп кездесетін қателердің сипаттамасына тоқталамыз:

- ##### – дерек ұяшыққа сыймаған.
- #ДЕЛ/0 – есептеу формуласында 0 цифрына бөлу жағдайы бар.
- #ЗНАЧ! – формулаға мүмкін болмаған өрнектер қатысуда, мысалы: кирилл шрифтіндегі айнымалы пайдаланылған болуы мүмкін.

■ **#ИМЯ?** – MS Excel формуладағы айнымалының атауын анықтай алмады.

■ **#ЧИСЛО!** – формулада функция айнымалысы немесе мән қате пайдаланылуда.

■ **#ССЫЛКА!** – ұяшыққа қате сілтеме жасалған.

■ **#ПУСТО!** – ұяшыққа қате аралықтағы мән енгізілген.

2-жаттығу. Төмендегі амалдарды орында:

■ **A1** ұяшығына 23 санын, ал **B1**-ге 0-ді енгіз және **C1** ұяшығында **A1/B1** амалын орындап, нәтижеге талдау жаса.

■ **A1** ұяшығына “Қате” мәтінді, ал **A2**-ге 10-ды енгіз және **A3** ұяшығында **A1*A2** амалын орындап, пайда болған қатенің себебін тап;

■ **A1** ұяшығына 1 санын, ал **B1**-ге 1-ді енгіз және **A3** ұяшығында **A1A/B1** амалын орындап, пайда болған қатенің себебін анықта;

C1		fx		=A1A/B1									
	A	B	C	D	E	F	G	H	I	J	K		
1		1	#ИМЯ?										
2													

■ **A1** ұяшығына 1-ді, ал **B1**-ге 2-ні енгіз, **C1**-де **A1+B1**-ді есепте, **D1** ұяшығына 15-ті енгіз, **E1**-ге 12-ні енгіз, **F1**-де **D1–E1**-ді есепте және **G1**-де **C1*F1**-ді есепте;

G1		fx		=C1*F1									
	A	B	C	D	E	F	G	H					
1	1	2	3	15	12	3	9						
2													

Жоғарыдағы жаттығуларда арифметикалық амалдарды MS Excel 2010-ның көмегімен жылдам және оңай орындау мүмкіндігі бар екені белгілі болды. MS Excel 2010 электронды кестесінің калькулятордан айырмашылығы – ол күрделі өрнектерді есептеуде жалпылау мүмкіндігіне ие. Бұл мүмкіндіктерді кейінгі сабақтарда қарастырамыз.



ЕСІНДЕ САҚТА !

MS Excel 2010-ның стандарт парағында әріптермен белгіленген 16348 баған және цифрлармен өрнектелген 1 048 576 жол бар.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. MS Excel кестелерінде деректердің қандай түрлері бар?
2. **A2D:H9** жазуынан қандай қате табуға болады?
3. Берілген үш санның қосындысын есепте.
4. Берілген екі санды көбейтіп, көбейтіндінің төртінші дәрежесін тап.



ҮЙГЕ ТАПСЫРМА

1. MS Excel 2010 электронды кестесінің көмегімен $(a+b)*(c+d)$ өрнегі айнымалыларының ерікті мәндерін есепте.
2. Берілген төрт санды екі жұпқа бөліп, олардың әрбірінің бөліндісін тап және бөлінділерді өзара көбейт.



2-САБАҚ. ҰЯШЫҚҚА СІЛТЕМЕ: САЛЫСТЫРМАЛЫ, АБСОЛЮТ ЖӘНЕ АРАЛАС

Microsoft Excel 2010 электронды кестелерінің көмегімен арифметикалық амалдардан тыс ұяшықтарға түрлі сілтемелер жасау арқылы практикалық тапсырмаларды сапалы шешуге болады.

1-жаттығу. Сыныптарда пәндерді игеру көрсеткіштерін анықта.

	A	B	C	D	E	F	G	H	I	J	K	L
1	Сыныптарда пәндерді игеру көрсеткіштері											
2	Сынып- тар	Оқушы саны	Пәндер				Игеру көрсеткіштері					
3			Қазақ тілі	Мате- матика	Физи- ка	Шет тілі	Химия	Қазақ тілі	Мате- матика	Физи- ка	Шет тілі	Химия
4	10-а	32	27	22	25	30	27					
5	10-б	33	29	26	27	24	28					
6	10-с	31	23	25	28	21	26					
7	Жалпы	96	79	73	80	75	81					

Орындау:

■ бірінші жолдағы ұяшықтар  **Объединить и поместить в центре** түймешігінің көмегімен біріктіріліп, оған “Сыныптарда пәндерді игеру көрсеткіштері” мәтіні жазылады;

■ **A2:A3** ұяшықтары біріктіріліп, “Сыныптар” жазуы ұяшықтың орталығына жазылады;

■ **B2:B3** ұяшықтары біріктіріліп “Оқушы саны” жазуы ұяшықтың орталығына жазылады;

■ **C2:G2** ұяшықтары біріктіріліп, “**Пәндер**” жазуы ұяшықтың орталығына жазылады;

■ **H2:L2** ұяшықтары біріктіріліп, “**Игеру көрсеткіші**” жазуы ұяшықтың орталығына жазылады;

■ **C3:G3** және **H3:L3** ұяшықтарына пәндердің аттары жазылады;

■ **B4:B6** ұяшықтарына сыныптардағы оқушылар саны енгізіледі;

■ **C4:C6**-дан **G4:G6**-ға дейін пәнді игерген оқушы саны енгізіледі;

■ **B4:B6** тышқанның сол түймешігін баса отырып белгіленеді, MS Excel 2010-ның таспасындағы **Σ** түймешігі басылады, нәтижеде **B7**-де оқушылардың жалпы саны пайда болады;

■ **C7:G7** ұяшықтарында пәндерді игерген оқушылардың барлық сыныптар бойынша жалпы санын оңай анықтау үшін **B7** ұяшығының төменгі оң жақ бұрышына тышқанның сол түймешігін баса отырып жылжытып, керекті мәндерді пайда етеміз. Ұяшыққа осылай сілтеме жасау тәсілі **салыстырмалы сілтеме** деп аталады;

■ **10-а** сыныбында қазақ тілін игеру көрсеткішін анықтау үшін **H4** ұяшығына “=” белгісін енгізіп, пәнді игерген оқушылардың санын **C4**-ті сыныптағы оқушылардың жалпы санына **B4**-ке бөліп, шыққан нәтижені 100-ге көбейтеміз. Игеру көрсеткішін әрдайым сыныптағы оқушылардың санына бөлінетіндіктен **B4**-ті өзгермейтін ету үшін, осы мекенжайға **\$B\$4** көрінісінде, яғни **абсолют сілтеме** жасалады;

■ қалған сыныптар және басқа пәндер бойынша игеру көрсеткіштерін оңай анықтау үшін тышқанның сол түймешігін баса отырып **B** баған және **7** жол бойынша жылжытамыз.

Осылай салыстырмалы және абсолют сілтеме жасау тәсілдерін қолданып, қарастырылып жатқан мәселені шешу өте оңай орындалды. Нәтижеде төмендегі электронды кесте пайда болды:

	A	B	C	D	E	F	G	H	I	J	K	L
1	Сыныптарда пәндерді игеру көрсеткіштері											
2	Сынып- тар	Оқушы саны	Пәндер				Игеру көрсеткіштері					
3			Қазақ тілі	Мате- матика	Физи- ка	Шет тілі	Химия	Қазақ тілі	Мате- матика	Физи- ка	Шет тілі	Химия
4	10-а	32	27	22	25	30	27	84,4	68,8	78,1	93,8	84,4
5	10-б	33	29	26	27	24	28	87,9	78,8	81,8	72,7	84,8
6	10-с	31	23	25	28	21	26	74,2	80,6	90,3	67,7	83,9
7	Жалпы	96	79	73	80	75	81	82,3	76	83,3	78,1	84,4

Абсолют және салыстырмалы сілтемелер біргелікте қолданылса **аралас сілтеме** деп аталады.

MS Excel 2010-ның тағы басқа да қолданыстағы бағдарламалармен баламалылығын одан әрі жақсарту мақсатында құжаттарды жадыға сақтау үшін жаңа XML (eXtensible Markup Language – белгіленудің кеңейтілген тілі) форматы жасалған. Мұнда сығымдау технологиясын қолдану есебіне файлдардың өлшемі кішірейді және кестелерді қалаған операциялық жүйелерде пайдалану мүмкіндігі туындады.



ЕСІНДЕ САҚТА !

MS Excel 2010-дағы файлдар “.xlsx” кеңейтпесіне ие.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Абсолют және салыстырмалы сілтеменің ерекшелігін айт.
2. MS Excel 2010 электронды кестелерін жадыға сақтау үшін қандай жаңа технологиялар қолданылады?
3. MS Excel 2010 электронды кестесімен сыныптағы пәндерді игерудің мониторингін жүргізу үшін деректеме дайында.



ҮЙГЕ ТАПСЫРМА

1. Абсолют және салыстырмалы сілтеме жасауға қатысты амалдарды орында.
2. Сынып оқушыларының жоқтамасын анықтайтын кесте жаса.



3-САБАҚ. СІЛТЕМЕНІ ПАЙДАЛАНЫП АМАЛ ОРЫНДАУДА КӨШІРУДІҢ АРТЫҚШЫЛЫҒЫ

Сілтемені пайдаланып амалды орындауда көшіруді пайдалану арқылы орындалатын амалдарды біршама оңайлату мүмкіндігі бар. Сілтемелерді пайдаланып көшірудің артықшылығын төмендегі жаттығуларда қарастырамыз

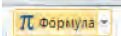
1-жаттығу. a санын 9-шы дәрежесіне дейін есептеу.

Орындау:

- тоғыз баған мен алты жолдық кесте жасаймыз;

■ екінші жолдағы ұяшықтар  түймешігі арқылы біріктіріліп, “Берілген санның дәрежелерін есептеу” мәтіні жазылады;

■ А бағанына a^9 санының мәні енгізіледі;

MS Excel-дің **Вставка - вставка формулу**  тізбектілігі таңдалып, ***a*-ның** төмендегі дәрежелерін жазамыз:

	A	B	C	D	E	F	G	H	I
1									
2	Берілген санның дәрежелерін есептеу								
3	a	a^2	a^3	a^4	a^5	a^6	a^7	a^8	a^9
4	2								
5	3								
6	4								
7	5								

■ **B4** ұяшықта берілген санның квадраты, яғни **= \$A4* A4** амалы орындалады. Осы ұяшықтың төменгі оң жақ бұрышына тышқанның нұсқағышын әкеліп, сол түймешігін баса отырып баған бойынша төменге қарай жылжытылады және **B7** ұяшығында тоқтатылады;

■ **B7** ұяшығының төменгі оң жақ бұрышына тышқанның нұсқағышы әкелініп, сол түймешігін баса отырып **B7:I7** аралығында жылжытылады. Нәтижеде төмендегідей көрініс пайда болады:

	A	B	C	D	E	F	G	H	I
1									
2	Берілген санның дәрежелерін есептеу								
3	a	a^2	a^3	a^4	a^5	a^6	a^7	a^8	a^9
4	2	4	8	16	32	64	128	256	512
5	3	9	27	81	243	729	2187	6561	19683
6	4	16	64	256	1024	4096	16384	65536	262144
7	5	25	125	625	3125	15625	78125	390625	1953125



ЕСІНДЕ САҚТА !

MS Excel 2010-да енгізілген мәтін ұяшыққа сыймаса, **Перенос текста** (мәтінді тасымалдау) түймешігі пайдаланылады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. MS Excel 2010 электронды кестесінде көшірудің ерекшеліктерін түсіндіріп бер.

2. MS Excel 2010 электронды кестесінде мәтінді көшіру қалай жасалады?
3. 3 санының 10 дәрежесіне дейін есептейтін кесте жаса.



ҮЙГЕ ТАПСЫРМА

1. Оқу құралдарын сатып алу кестесін жаса.
2. $1+2^2+3^3+4^4$ өрнегін MS Excel 2010-да есепте.



4-САБАҚ. ҚАРАПАЙЫМ ЖӘНЕ КҮРДЕЛІ ФУНКЦИЯЛАРДЫҢ ГРАФИКТЕРІ

MS Excel 2010-да қарапайым және күрделі функциялардың графиктері таспаның **Вставка** қойыншасының **Диаграммы** тобындағы **График** түймешігі арқылы жасалады. Төмендегі жаттығуларда қарапайым және күрделі функциялардың графикін жасаймыз.

1-жаттығу. $z=a+bx+cx^2+dx^3$ функциясының графикін жасау.

Орындау:

▪ **A1:D1** ұяшықтарына **a,b,c,d** коэффициентінің төмендегідей мәндері енгізіледі:

	A	B	C	D
1	-1	5	-35	65

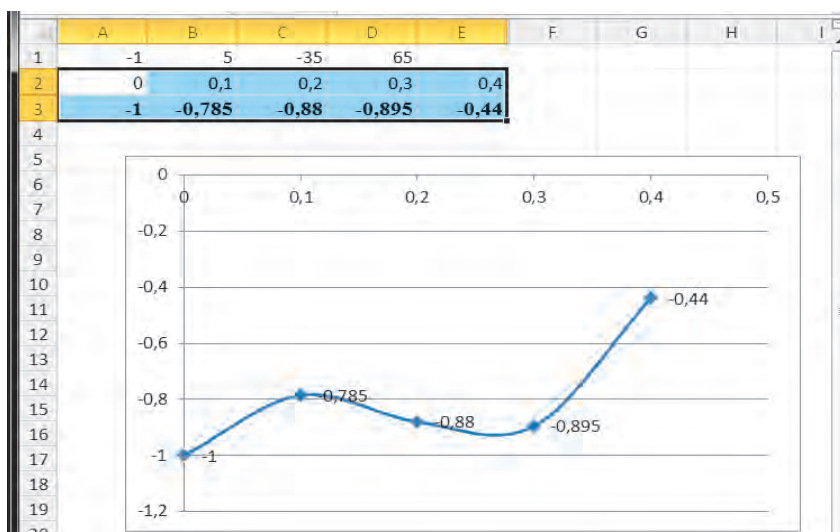
▪ **A2:E2** ұяшықтарына x -тің 0; 0,1; 0,2; 0,3; 0,4 мәндері енгізіледі;

▪ **A3:E3** ұяшықтарына сәйкесінше x -тің мәндері үшін, мысалы, **A3-те** қарастырылған $=\$A1+\$B1*A2+\$C1*A2*A2+\$D1*A2*A2*A2$ формуласының $x=0$ -дегі мәні туындайды. Қалған нүктелерге салыстырмалы сілтеменің көмегімен жалғастырылады;

▪ **A2:E3** ұяшықтарының блогы ажыратылып алынады;

A3:E3 ұяшықтары белгіленіп ажыратылып алынған соң, MS Excel 2010-ның таспасынан **Вставить-Диаграммы-Точечная** тізбектілігі орындалады.

Пайда болған графикке x -тің мәндерін шығару үшін графиктің үстіне тышқанның оң түймешігі басылып, **Добавить подписи данных** тармағы таңдалады. Нәтижеде төмендегідей терезе пайда болады:



2-жаттығу. $z=a+bx+cy$ және $z=ax^2+bxу+cy^2$ функцияларының графигін жасау.

Орындау:

- **A1:C1** ұяшықтарына a, b, c коэффициентінің мәндері:

Сол жақтағы сызба үшін

	A	B	C
1	-1	2	3

Оң жақтағы сызба үшін

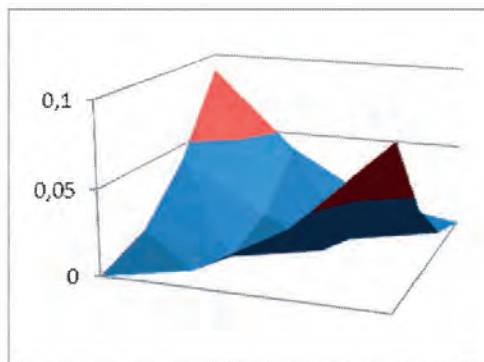
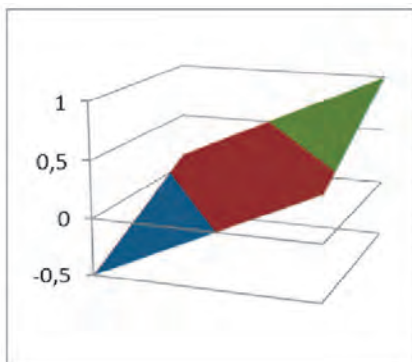
	A	B	C
1	1	-2	1

енгізіледі.

- **A2:E2** ұяшықтарына x -тің 0; 0,1; 0,2; 0,3; 0,4 мәндері енгізіледі;
- **A2:A6** ұяшықтарына y -тің 0; 0,1; 0,2; 0,3; 0,4 мәндері енгізіледі;
- **A3:E7** ұяшықтарында сәйкесінше x -тің мәндері үшін, мысалы

A4-те қарастырылатын $=\$A1*A2*A2+\$B1*A2*A4+\$C1*A4*A4$ формуласының $x=0$ мен $y=0,1$ -дегі мәні пайда болады. Қалған нүктелерге **салыстырмалы** сілтеменің көмегімен жалғастырылады. Мұндағы **\$** әрпі осы ұяшыққа **абсолют** (өзгермейтін) сілтеме екенін білдіреді;

▪ **A3:E7** ұяшықтары белгіленіп ажыратылып алынған соң MS Excel 2010-ның панеліндегі **Вставить** қойыншасының **“таспасындағы”** **Поверхность** түймешігін басып, өрнектің сызбасын пайда етеміз:



ЕСІНДЕ САҚТА !

MS Excel 2010-да функция графигін жасау **Вставить** қойыншасының **Диаграммы** тобында жүзеге асырылады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. MS Excel 2010-да график жасау мүмкіндіктері туралы әңгімелеп бер.
2. Функция графигін жасау үшін қандай тәсіл пайдаланылады?
3. Сыныптағы оқушылардың “Информатика және ақпарат технологиялары” пәнін игеру көрсеткіштерінің графигін жаса.



ҮЙГЕ ТАПСЫРМА

1. $y=3x+4$ функциясының графигін жаса.
2. Сыныптағы оқушылардың “Математика” пәнін игеру көрсеткіштерінің графигін жаса.



5-САБАҚ. БАСҚА ПАРАҚҚА НЕМЕСЕ КІТАПҚА СІЛТЕМЕ ЖАСАУ

MS Excel 2010-ның тағы бір қолайлылығы мынада, басқа парақтарға немесе кітаптарға сілтеме жасалып, олардағы деректермен байланыс тудыруға болады. Төмендегі жаттығуларда парақтар мен кітаптарға сілтеме жасау тәсілдерін қарастырамыз.

1-жаттығу. Сауда орындарындағы деректерге негізделіп, оргтехникамен сауда жасайтын фирма бойынша жалпылама кесте түзу.

Орындау:

- әрбір сауда орны бойынша, мысалы

1 Маркет – 1 параққа

	A	B	C	D
1	оргтехника саудасы Маркет 1			
2	Тауар атауы	бағасы	саны	сома
3	монитор	200	12	2400
4	процессор	60	31	1860
5	пернетақта	25	28	700
6	винчестр	95	14	1330
7	жалпы			6290

2 Маркет – 2 параққа

	A	B	C	D
1	оргтехника саудасы Маркет 2			
2	Тауар атауы	бағасы	саны	сома
3	монитор	200	15	3000
4	процессор	60	25	1500
5	пернетақта	25	35	875
6	винчестр	95	13	1235
7	жалпы			6610

3 Маркет – 3 параққа

	A	B	C	D
1	оргтехника саудасы Маркет 3			
2	Тауар атауы	бағасы	саны	сома
3	монитор	200	22	4400
4	процессор	60	20	1200
5	пернетақта	25	21	525
6	винчестр	95	23	2185
7	жалпы			8310

деректерін енгіземіз.

- 4 параққа жалпыланған деректі жасау үшін **В3** ұяшығына **=Лист1!В3+Лист2!В3+Лист3!В3** жазуын жазамыз.

■ **В3**-те “=” белгісінен кейін 1 парақтағы **В3** ұяшығына тышқанның сол түймешігі басылады, нәтижеде **Лист1!В3**, сосын + және 2 параққа **Лист2!В3** жазылады және 4 парақтағы **В4, В5, В6, С3, С4, С5, С6, D3, D4, D5, D6, D7** ұяшықтарына көшірме жасалып, төменгі оң бұрышына тышқанның сол түймешігін басып жасалады. Нәтижеде төмендегідей терезе пайда болады:

	A	B	C	D
1	Фирманын оргтехника саудасы			
2	Тауар атауы	бағасы	саны	сума
3	монитор	200	49	9800
4	процессор	60	76	4560
5	пернетақта	25	84	2100
6	винчестр	95	50	4750
7	жалпы			21210

2-жаттығу. Оқушылардың пәндер бойынша алған бағаларын кітаптардағы кестелерінің көмегімен жинақтау.

Орындау:

■ оқушылардың пәндерден алған бағаларының кітап кестесінде байланысын тудыру үшін олардың әрбір пән бойынша бағалары жеке жұмыс кітаптарына жазылады (математика, физика, информатика):

Математика		
	A	B
1	Р/с	Оқушының аты-жөні
2	1	Әлиев Әлішер Әнуарұлы
3	2	Мирсанов Оралбай Мұхаммедұлы
4	3	Қожабаев Пархат Дәулетбайұлы
5	4	Дәулетбаев Бекзат Пархатұлы
6	5	Мұхаммедов Асетбек Оралұлы

Физика		
	A	B
1	Р/с	Оқушының аты-жөні
2	1	Әлиев Әлішер Әнуарұлы
3	2	Мирсанов Оралбай Мұхаммедұлы
4	3	Қожабаев Пархат Дәулетбайұлы
5	4	Дәулетбаев Бекзат Пархатұлы
6	5	Мұхаммедов Асетбек Оралұлы

■ оқушылардың осы үш кестедегі бағаларын көрсететін жеке жалпы пәндер жұмыс кітабында жалпы кесте жасалады:

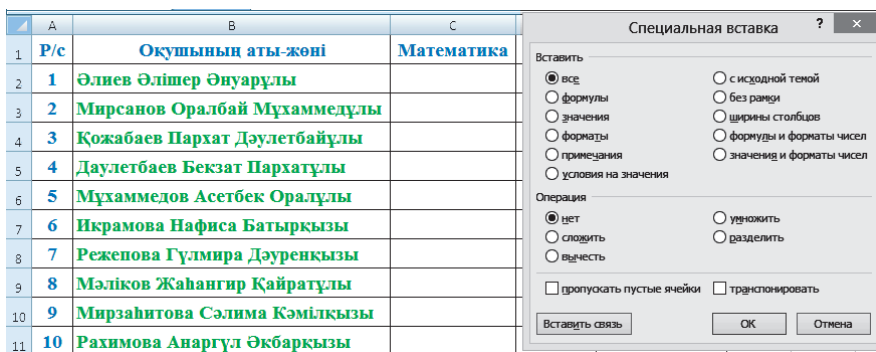
Информатика

	A	B	C
1	Р/с	Оқушының аты-жөні	Информатика
2	1	Әлиев Әлішер Әнуарұлы	4
3	2	Мирсанов Оралбай Мұхаммедұлы	5
4	3	Қожабаев Пархат Дәулетбайұлы	5
5	4	Дәулетбаев Бекзат Пархатұлы	5
6	5	Мұхаммедов Асетбек Оралұлы	5

Жалпы пәндер.xlsx

	A	B	C	D	E
1	Р/с	Оқушының аты-жөні	Математика	Физика	Информатика
2	1	Әлиев Әлішер Әнуарұлы			
3	2	Мирсанов Оралбай Мұхаммедұлы			
4	3	Қожабаев Пархат Дәулетбайұлы			
5	4	Дәулетбаев Бекзат Пархатұлы			
6	5	Мұхаммедов Асетбек Оралұлы			

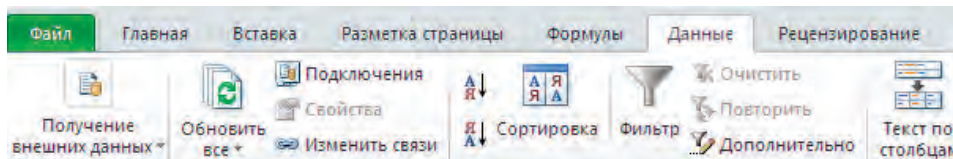
■ жеке пән (жалпы пәндер кесте файлы) енгізілген жұмыс кітабы ашылып, бағалар енгізілген ұяшықтар белгіленіп көшірме жасалады (Копировать). Жалпы жұмыс кітабындағы сәйкес ұяшықтар белгіленіп, Арнайы кірістіру (Специальная вставка) бөлімі таңдалып, Байланысты кірістіру (Вставить связь) түймешігі басылады:



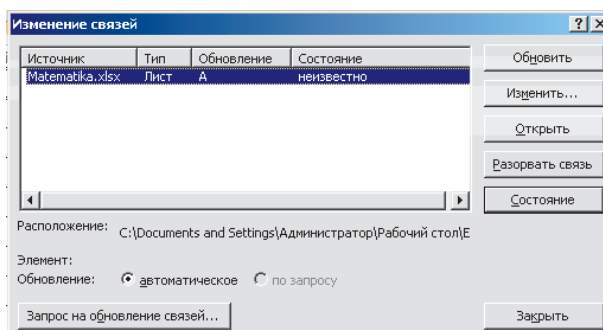
Нәтижеде төмендегідей байланыс туындайды:

C2 [Математика.xlsx]Математика!C2					
Жалпы пәндер.xlsx					
	A	B	C	D	E
1	Р/с	Оқушының аты-жөні	Математика	Физика	Информатика
2	1	Әлиев Әлішер Әнуарұлы	3		
3	2	Мирсанов Оралбай Мұхаммедұлы	5		
4	3	Қожабаев Пархат Дәулетбайұлы	5		
5	4	Дәулетбаев Бекзат Пархатұлы	5		
6	5	Мұхаммедов Асетбек Оралұлы	5		

■ барлық жұмыс кітаптарының арасында байланыс пайда етілген соң, байланыс пайда болған жұмыс кітабының меню жолынан **Данные** қойыншасы таңдалып, **Подключения – изменить связи** тізбектілігі таңдалады:



■ пайда болған терезеден **Обновить** түймешігі таңдалып, жұмысты аяқтау үшін **Заккрыть** түймешігі таңдалып терезе жабылады.



ЕСІНДЕ САҚТА !

Кітаптарды өзара байланыстыру үшін меню жолынан **Данные** қойыншасы таңдалып, **Подключения – изменить связи** тізбектілігі таңдалады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Парақтарды байланыстыру тәсілдерін түсіндір?
2. Кітаптарды байланыстыру тәсілдерін түсіндір?
3. 10 сынып оқушыларының барлық пәндерден алған бағаларына жеке кестелер жаса және оларды жалпы кестеге біріктірі.



ҮЙГЕ ТАПСЫРМА

1. Үшінші парақтың **B4** ұяшығында 1 парақтың **C3** ұяшығындағы және 2 парақтың **D6** ұяшығындағы сандардың қосындысын есепте.
2. Оқушылардың тарих, химия, физика, математика пәндерінен алған бағаларын жеке файлдарға сақтаңдар және оны “Жалпы” атты файлға біріктіріңдер.



6-САБАҚ. MS EXCEL-ДЕГІ ФУНКЦИЯЛАР КІТАПХАНАСЫ

MS Excel 2010-да дайындалатын деректі кестелер мәтінмен немесе санмен толтырылатыны бұрын айтылған. Кейде ұяшықтардағы

деректер бойынша кейбір амалдарды орындау қажеттілігі туындайды, мұндай жағдайда формулаларды пайдаланамыз.

Бұл үшін MS Excel 2010 таспасындағы **Формулы** қойыншасы пайдаланылады. Осы қойыншада мәндермен, өрнектермен және ұяшықтармен амалдар орындауға арналған **Функциялар кітапханасы** орналасқан. Кітапханадағы түрлі міндеттерді орындауға арналған функциялардың кейбірімен танысамыз

Математикалық функциялар

р/с	Функция	Міндеті
1.	ФАКТР	Аргумент ретінде берілген бүтін сан факторларын анықтайды
2.	ABS	Аргумент мәнінің модулін есептейді.
3.	LN	Санның натурал логарифмін есептейді.
4.	EXP	Санның экспонентасын есептейді.
5.	SIN	Санның синусын есептейді.
6.	COS	Санның косинусын есептейді.
7.	TAN	Санның тангенсін есептейді.
8.	НОД	Екі және одан үлкен бүтін сандардың ең үлкен ортақ бөлгішін есептейді.
9.	НОК	Екі және одан үлкен бүтін сандардың ең кіші ортақ бөлінгішін есептейді.
10.	ОКРУГЛ	Шын санды жуықтайды.
11.	СУММ	Екі және одан үлкен сандардың қосындысын есептейді.
12.	ПРОИЗВЕД	Екі және одан үлкен сандардың көбейтіндісін есептейді.
13.	ЗНАК	Теріс сан болса -1-ді, оң болса 1-ді, нөл болса 0-ді шығарады.
14.	КОРЕНЬ	Санның квадрат түбірін есептейді.

1-жаттығу. Берілген 6,10, 30, 60 сандарының ең үлкен ортақ бөлгішін есепте.

Орындау:

- **A1** ұяшығына 6 санын, **B1** ұяшығына 10 санын, **A2** ұяшығына 30 санын, **B2** ұяшығына 60 санын енгіземіз;
- **C4** ұяшығына **=НОД(A1:B2)** формуласын енгіземіз;
- **Enter** пернесі басылып, нәтижеде мына терезе пайда болады:

The screenshot shows an Excel spreadsheet with the following data:

	A	B	C	D	E	F	G	H
1	6	10						
2	30	60						
3								
4			2					

The formula bar at the top shows the formula **=НОД(A1:B2)** entered in cell C4.



ЕСІНДЕ САҚТА !

MS Excel 2010-дағы *алгебралық амалдардың белгілері*:
 + (қосу); - (азайту); * (көбейту); / (бөлу); ^ (дәрежеге шығару).



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. **СУММПРОИЗ** функциясын пайдаланып (5;10), (15;20), (25;30) сан жұптықтары көбейтінділері қосындысын есепте.
2. Берілген -99, -23, -6, 0, 10, 56, 77, 89, 123, 345, 678, 2345 сандарының ішінен ең үлкені мен ең кішісінің қосындысы мен көбейтіндісін есепте.
3. -7,8; -9,67865; -4,6; -3,8; 7,8 сандарының көбейтіндісін есепте және **ОКРУГЛ** функциясы көмегімен 2 цифрға дейін жуықта.



ҮЙГЕ ТАПСЫРМА

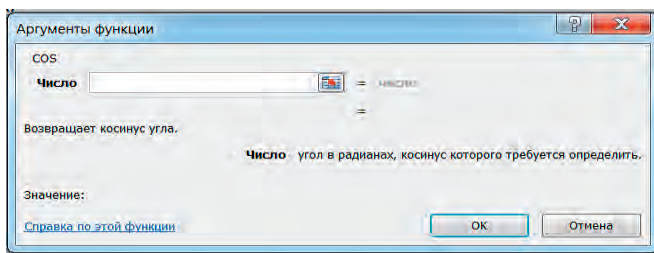
1. 1, 2, 3, 4, 5, 6, 8, 9 сандарының квадрат түбірлерінің қосындысын есепте.
2. 15, 20, 45, 62, 94, 112 сандарының ең кіші ортақ бөлінгішін табындар.
3. 4,10 сандарының факториалын есепте.



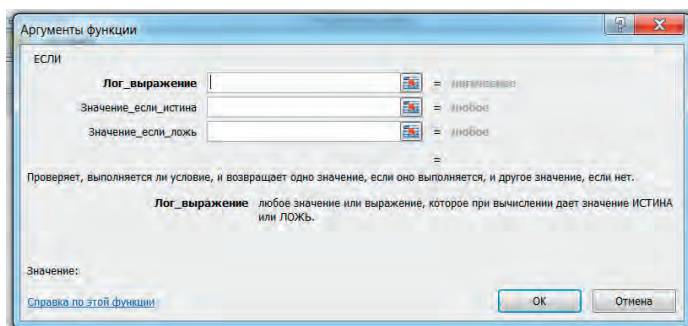
7-САБАҚ. ФУНКЦИЯ АРГУМЕНТІ ТЕРЕЗЕСІ

MS Excel 2010-да пайдаланылатын функциялар мазмұн тұрғысынан бірінен-бірі шұғыл ерекшеленгені үшін өз кезегінде олардың аргументтері де түрлі көріністе берілуі мүмкін.

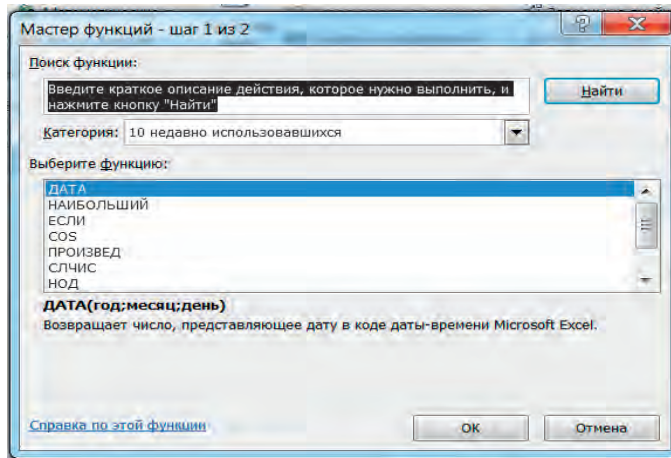
Мысалы, **Математические функции** бөліміндегі **cos(x)** функциясында аргумент бір шын мәннен құралады және бұрыштың мәні радианда беріледі:



Логические функции бөліміндегі **ЕСЛИ** функциясындағы аргумент логикалық өрнек болып, оның тек екі мәніне (“шын” немесе “жалған”) орай екі түрлі нәтижеге ие боламыз:



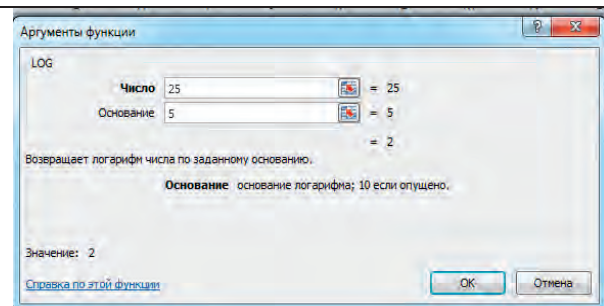
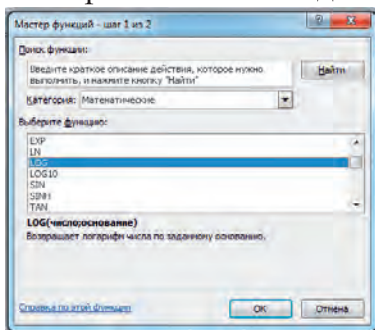
Мастер функций терезесіне алдымен қажетті функцияны енгізіп **Найти** түймешігінің көмегімен іздеп табылады, кейін осы функция таңдалып, аргументтеріне мәндер беріледі. Сосын таңдалған функция қажетті ұяшыққа орналастырылады:



1-жаттығу. $\log_5 25$ өрнегінің мәнін есепте.

Орындау:

- MS Excel 2010 таспасынан  түймешігі таңдалады;
- **Мастер функций** терезесінен логарифмді есептейтін **Log** функциясы таңдалады;
- **Число** жолына 25 саны, **Основание** жолына 5 саны енгізіліп, **OK** түймешігі басылады:



ЕСІНДЕ САҚТА!

MS Excel 2010-да функцияларды пайдалану үшін формулалар жолынан  түймешігі таңдалады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. **Мастер функций** қандай міндетті орындайды?
2. Функциялардың аргументтері туралы әңгімелеп бер.
3. 5, 10, 15, 20, 25, 30 сандарының натурал логарифмдерінің қосындысын есептеп, нәтижені 3 цифрға дейін жуықта.
4. **Мастер функций**-ді пайдаланып, берілген 4, 6, 10, 23 сандарының экспонентасын есепте.
5. **Мастер функций**-ді пайдаланып, берілген 15, 20, 45 62, 94, 112 сандарының ең кіші ортақ бөлінгішін тап.



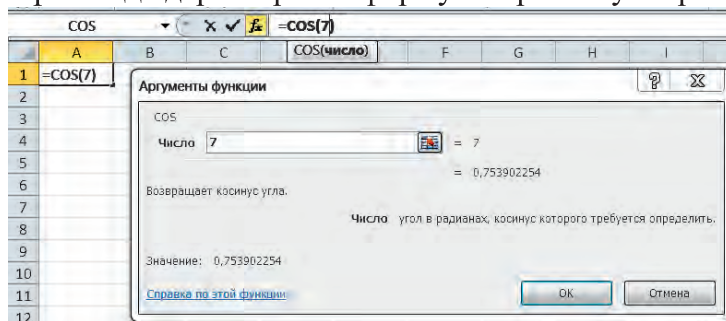
ҮЙГЕ ТАПСЫРМА

1. **Мастер функций**-ді пайдаланып, берілген 15, 20, 45 62, 94, 112 сандарының ең үлкен ортақ бөлгішін тап.
2. **Мастер функций**-ді пайдаланып, берілген 10 санының факториалын есепте.
3. **Мастер функций**-ді пайдаланып, 2,4567 санының синусын есепте.



8-САБАҚ. ФОРМУЛАЛАР ЖОЛЫН ПАЙДАЛАНУ

Формулалар жолы ұяшықтағы мәннің формуласын көрсетуге немесе осы ұяшыққа деректер мен формулалар енгізуге арналған:



Мұндағы түймешігі амалды теріске шығару, түймешігі амалды орындау, түймешігі кітапханаға сілтеме жасау міндетін атқарады.

1-жаттығу. Төмендегі деректі кестені дайында. Есеп-қисап істерін формулалар жолын пайдаланып орында. **Кәсіпорынның іссапар қаражаттары** (мың сум есебінде):

	A	B	C	D	E	F	G
1	Р/с	Баратын жері	Жол бағасы	Күндер саны	Күнделікті қаражат	Адамдар саны	Жалпы қаражат
2	1.						
3	2.						
4	3.						
5	4.						

Осы мысалдағы “Жалпы қаражат” бағаны төмендегі формула арқылы есептеледі:

“Жалпы қаражат” = (2 * “Жол қаржысы” + “Күндер саны” * “Күнделікті қаражат”) * “Адамдар саны”

Орындау:

- баған мен жолдардың кеңдігі жеткілікті болмағаны үшін оны қажетті мөлшерде өзгертеміз;
- ұяшықтарды қажетті деректермен толтырамыз:

	A	B	C	D	E	F	G
1	Р/с	Баратын жері	Жол бағасы	Күндер саны	Күнделікті қаражат	Адамдар саны	Жалпы қаражат
2	1.	Ташкент	80	5	80	4	
3	2.	Бұхара	86	4	50	5	
4	3.	Киев	1860	12	160	4	
5	4.	Мәскеу	1780	10	400	6	

■ **Файл** қойыншасынан **Сохранить как** тармағын таңдаймыз **Имя файла** жолына файлдың атын, мысалы “comras.xlsx” деп жазып, **Сохранить** түймешігін басамыз.

■ дисктен кестені жүктеу үшін **Файл** қойыншасынан **Открыть** тармағын таңдаймыз. **Файл** тізімінен қажетті файлды таңдап (атап айтқанда, “comras.xlsx” файлын) тышқанның түймешігі басылады;

- **G2** ұяшығына төмендегі **=(2*C2+D2*E2)*F2** формуласын енгіземіз;
- қалған жолдарға да дәл осы сияқты формулаларды енгіземіз:

	A	B	C	D	E	F	G
1	Р/с	Баратын жері	Жол бағасы	Күндер саны	Күнделікті қаражат	Адамдар саны	Жалпы қаражат
2	1.	Ташкент	80	5	80	4	$=(2*C2+D2*E2)*F2$
3	2.	Бұхара	86	4	50	5	
4	3.	Киев	1860	12	160	4	
5	4.	Мәскеу	1780	10	400	6	

Нәтижеде төмендегі кесте пайда болады:

	A	B	C	D	E	F	G
1	Р/с	Баратын жері	Жол бағасы	Күндер саны	Күнделікті қаражат	Адамдар саны	Жалпы қаражат
2	1.	Ташкент	80	5	80	4	2240
3	2.	Бұхара	86	4	50	5	1860
4	3.	Киев	1860	12	160	4	22560
5	4.	Мәскеу	1780	10	400	6	45360



ЕСІНДЕ САҚТА!

Формулалар жолы ұяшықтағы мәннің формуласын көрсетуге немесе осы ұяшыққа деректер мен формулалар енгізуге арналған.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Формулалар жолының міндетін түсіндір.
2. Формулалар жолын пайдаланып, үш ұяшыққа берілген сандардың үшінші дәрежелерінің қосындысын есепте.
3. Бірінші парақтағы формулалар жолына $=(2*C3+D4*E2)*\text{Лист}2!F2$ өрнегін енгіз және қажетті ұяшықтарға мәндерді беріп нәтижені есепте.



ҮЙГЕ ТАПСЫРМА

1. Формулалар жолын пайдаланып берілген төрт санның көбейтіндісін есепте.
2. Формулалар жолын пайдаланып $=A3-B2*D4+(C3/D2)$ формуласын жаз.



9-САБАҚ. МӘТІНДІК ФУНКЦИЯЛАР

MS Excel 2010-да ұяшықтардағы мәтіндермен жұмыс істеу үшін арнаулы функциялар бар. Олар төмендегілер:

1.	Функция	Міндеті
2.	БАТТЕКСТ	Санды мәтінге алмастырады.
3.	ДЛСТР	Мәтінді жолдағы белгілердің санын анықтайды.
4.	ЗАМЕНИТЬ	Белгіленген мәтін бөлігін жаңа мәтінге ауыстырады.
5.	ЗНАЧЕН	Мәтін көрінісіндегі санды санға ауыстырады.
6.	ПОВТОР	Мәтінді n рет қайталайды.
7.	СЦЕПИТЬ	Бірнеше мәтінді бір мәтінге өткізеді.

1-жаттығу. Үш ұяшықтағы оқушылардың фамилиясы, аты, әкесінің аты жазылған деректерді бір ұяшыққа біріктіру.

Орындау: Бұл мысалды шешудің екі түрлі тәсілі бар:

Бірінші тәсіл:

- **B** бағанына оқушының фамилиясы, **C** бағанына аты, **D** бағанына әкесінің аты енгізіледі;

- **E1** ұяшығына **=СЦЕПИТЬ(B1;" ";C1;" ";D1)** формуласы енгізіліп, **Enter** пернесі басылады;

- қалған ұяшықтарға осы формуланың көшірмесі жасалады. Нәтижеде төмендегі терезе пайда болады:

Е3		fx		=СЦЕПИТЬ(B3;" ";C3;" ";D3)	
Книга1					
	A	B	C	D	E
1	1	Маматов	Уәли	Әлиевич	Маматов Уәли Әлиевич
2	2	Намазов	Сәлім	Саматович	Намазов Сәлім Саматович
3	3	Мирсанов	Оралбай	Мұхаммедович	Мирсанов Оралбай Мұхаммедович

Екінші тәсіл:

- кестені жоғарыдағы реттілікпен толтырып аламыз;

- **E1** ұяшығына **=B1&" "&C1&" "&D1** формуласы енгізіліп, **Enter** пернесі басылады.

Оқушының фамилиясы **Лист1** парағының **B1** ұяшығында, аты **Лист2** парағының **B1** ұяшығында, әкесінің аты **Лист3** парағының **B1** ұяшығында берілген болса, осы деректерді **Лист4** парағында пайда ету

үшін қалаған ұяшыққа **=СЦЕПИТЬ(Лист1!B1;" ";Лист2!B1;" ";Лист3!B1)** формуласы енгізіліп, **Enter** пернесі басылады.



ЕСІНДЕ САҚТА !

MS Excel электронды кестесіндегі **B2:C5** өрнегі ұяшықтар блогын білдіреді. Ол 8 ұяшықты блокка алады. Егер **B2:C5** көрінісінде болса 2 ұяшықты белгілейді.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Мәтіндік функциялардың міндеттерін түсіндір?
2. **С** бағанында оқушылардың фамилиясы, аты, әкесінің аты, **D** бағанында оқушылардың туылған жылы берілген. Оларды **E** бағанында біріктір.



ҮЙГЕ ТАПСЫРМА

1. **ДЛСТР** функциясының көмегімен “Өзбекстан – тәуелсіз Отан” мәтінінде неше белгі бар екенін анықта.
2. **ЗАМЕНИТЬ** функциясына өз бетіңше қандай да бір амал түзіп, оны шеш.



10-САБАҚ. ЛОГИКАЛЫҚ ФУНКЦИЯЛАР

Кейбір амалдар мен мысалдарды шешуде есептеулер бірқатар шарттарға байланысты болуы мүмкін. Мұндай жағдайда логикалық шартты функцияларды пайдалануға болады.

Логикалық функциялар

р/с	Функция	Міндеті
1.	ЕСЛИ	Егер логикалық өрнектің мәні ШЫН болса, бірінші шарт, керісінше болса екінші шарт орындалады.
2.	И	Егер логикалық өрнектердің барлығының мәні шын болса, функцияның мәні ШЫН, кері жағдайда функцияның мәні ЖАЛҒАН болады.
3.	ИЛИ	Егер логикалық өрнектердің қандай да бір мәні шын болса, функцияның мәні ШЫН, кері жағдайда

		функцияның мәні ЖАЛҒАН болады.
4.	НЕ	Шын мәнді жалғанға, жалғанды шынға айналдырады.

1-жаттығу. Сыныпты бес пәннен 20 балдық жүйеде бағалағанда олардың игеру көрсеткішін талдайтын кесте жарату.

Орындау: Осы тапсырманы орындау алгоритмі төмендегідей:

- **В** бағанына сынып оқушыларының аты-жөні енгізіледі;
- **С** бағанына әдебиет пәнінен алынған балдары енгізіледі;
- **Д** бағанына алгебра пәнінен алынған балдары енгізіледі;
- **Е** бағанына физика пәнінен алынған балдары енгізіледі;
- **Ғ** бағанына химия пәнінен алынған балдары енгізіледі;
- **Г** бағанына информатика пәнінен алынған балдары енгізіледі;
- **Н** бағанына алынған балдардың қосындысы есептеледі;
- **ІЗ** және **І5** ұяшықтарына төмендегі формула енгізіледі:

=ЕСЛИ(С17<55;"қанағаттанарсыз";ЕСЛИ(С17<71;"қанағаттанарлы";ЕСЛИ(С17<86;"жақсы";"үлгілі"))))

	A	B	C	D	E	F	G	H	I
1	P/c	Пәндер							
2		Аты-жөні	Әдебиет	Алгебра	Физика	Химия	Инф.	Жалпы	Талдау
3	1	Ахмедов Н	20	12	18	19	18	87	үлгілі
4	2	Әзімов С	6	7	2	3	9	27	қанағаттанарсыз
5	3	Бақытов А	15	10	18	18	13	74	жақсы
6	4	Мұхтаров Х	20	6	7	19	18	70	қанағаттанарлы
7	5	Нұрымов Х	20	3	7	19	18	67	қанағаттанарлы



ЕСІНДЕ САҚТА !

Кейбір амалдар мен мысалдарды шешуде есептеулер бірқатар шарттарға байланысты болса, логикалық шартты функцияларды пайдалануға болады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. **И** функциясының міндетін түсіндіріп бер.
2. **ИЛИ** функциясының міндетін түсіндіріп бер.
3. Сауда маркетінде сатушыларға төмендегі шарт қойылған:

- сатушы ең аз еңбекақының 5 есесі мөлшерінде сауда жасаса 5 пайыздық үстемеге ие болады;
 - сатушы ең аз еңбекақының 10 есесі мөлшерінде сауда жасаса 7,5 пайыздық үстемеге ие болады;
 - сатушы ең аз еңбекақының 20 есесі мөлшерінде сауда жасаса 10 пайыздық үстемеге ие болады;
 - сатушы ең аз еңбекақының 20 есесінен астам мөлшерде сауда жасаса 12 пайыздық үстемеге ие болады.
- Осы тапсырманы логикалық есептеу кестесін жаса.



ҮЙГЕ ТАПСЫРМА

1. 10 пәннен 10 балдық жүйеде бағаланғанда оқушылардың игеру көрсеткішін талдайтын кесте жаса.
2. **ИЛИ** функциясына қатысты өз бетіңше мысал келтір және оны шеш.



11-САБАҚ. ПЫСЫҚТАУҒА АРНАЛҒАН ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

Төмендегі тапсырмаларды орында:

1. Ұйымның бөлімдерінде қызмет сапары қаражаттарының есебі (мың сум есебінде).

Р/с	Бөлімдер	Жылдар			Жалпы
		2015	2016	2017	
1.	Техникалық қамту	131,5	14,6	152,6	298,7
2.	Бухгалтерия	141,6	112,7	114,6	368,9
3.	Кадрлар бөлімі	128,4	153,3	143,8	425,5
	Жалпы	401,5	280,6	411	1093,1

2. Ұйым қызметкерлерінің еңбекақысынан салықты айыру есебі (мың сум есебінде).

Р/с	Аты-жөні	Еңбекақы	15 % салық	1 % сақтандыру	Қолға алатын
1.	Ахатов С.	1680			

2.	Сындаров А.	1360			
3.	Усманов А.	1460			
Жалпы		4500			

3. Ұйымның “кіріс-шығыс” операцияларынан салыққа тарту есебі (мың сум есебінде).

Р/с	Операция тақырыбы	Жалпы сома	1 % сақтандыру	18 % қосымша құн салығы	Қалды
1.	Еңбекақы	26500			
2.	Сатып алу	32500			
3.	Сату	41800			
	Жалпы	100800			

4. Тұрғын үй құратын ұйым өнімінің өзіндік құнын анықтау (мың сум есебінде).

Р/с	Жұмсалған өнім	Өзіндік құны	18 % қосымша құн салығы	Қалды
1.	Цемент	214,3		
2.	Шебен	216,4		
3.	Су	48,9		
	Жалпы	479,6		

5. Кәсіпорынның сатылған өнімінен алынатын салықтың есебі (мың сум есебінде).

Р/с	Өнім атауы	Саны	Бағасы	18 % қосымша құн салығы
1.	Кірпіш	2790	0,6	
2.	Блок	6170	1,2	
3.	Есік	1070	700	
	Жалпы	10030	701,8	



12-САБАҚ. БАҚЫЛАУ ЖҰМЫСЫ

Өтілген тақырыптарда алынған дағдыларды пайдаланып төмендегі мысалдарды шеш, шыққан нәтижелерді графикалық көріністе бейнелеңдер:

1-вариант

1. Жеке тұлғалардың жинақ банкіне қойған сомасынан алған пайдасының есебі (мың сум есебінде).

Р/с	Аты-жөні	Қойылған сома	9 % жылдық пайда	Жалпы
1.	Сүйеров А.	11500		
2.	Базаров А.	9500		
3.	Әлиев А.	14400		
4.	Омаров С.	13500		

2. Жалпы қаражаттар есебі (мың сум есебінде).

Р/с	Қаражат мақсаты	1-күндік қаражат сомасы	2-күндік қаражат сомасы	% есебіне өсім
1.	Таңғы ас	12	15	
2.	Түстену	15	17	
3.	Кешкі тамақ	14	18	
	Жалпы			

2-вариант

1. Мектеп оқытушыларының оқу жылы жүктемесінің орындалуы.

Р/с	Аты-жөні	Жоспар	Орындады	% есебінде
1.	Әлімов А.	780	780	
2.	Әлімұратов Б.	806	798	
3.	Сүлейменов А.	810	804	
4.	Әзімбаев Б.	678	678	

2. Кәсіпорын қызметкерлерінің еңбекақысын жергілікті коэффициент пен зиянды ескеріп есептеу (мың сум есебінде).

Р/с	Аты-жөні	Еңбекақы	Жергілік- ті коэфф.	Зиян	Жалпы
1.	Назаров О.	1400	0,05	0,2	
2.	Айдаров Р.	1300	0,05	0,2	

3.	Маматов Б.	1280	0,05	0,2	
4.	Қасымов Р.	1170	0,05	0,2	
	Жалпы				

3-вариант

1. Ұйым бөлімінің тоқсандық есебі (мың сум есебінде).

Р/с	Тоқсан	Тауар көлемі (тонна есебінде)	1 тонна өзіндік құны	Жалпы
1.	I тоқсан	205,6	22400	
2.	II тоқсан	207,7	21800	
3.	III тоқсан	208,8	22400	
1 тонналық тауардың орташа бағасы				

2. Кәсіпорынның жалпы айналымынан жылдың әр айы бойынша жергілікті бюджетке қаржы өткізуінің есебі (мың сум есебінде).

Р/с	Айлар	Айналым сомасы	5 % жергілікті бюджетке бөлу
1.	Январь	12000000	
2.	Февраль	22600000	
3.	Март	32600000	

4-вариант

1. Мектепте “Информатика және ақпарат технологиялары” пәні бойынша игеру көрсеткіштері.

Р/с	Сынып	Оқушы саны	Тапсырған- дар	% есебінде игеру
1.	9-а сынып	18	14	
2.	9-б сынып	19	15	
3.	10-а сынып	17	15	

3. Фирманың үстіміздегі жылдың I тоқсанында өнімін сатудан алған табысы (мың сум есебінде).

Р/с	Операция мазмұны	Айлар			Жалпы
		Январь	Февраль	Март	
1.	Сату	205,7	304,8	215,6	

2.	Сатып алу	218,7	189,6	151,9	
3.	Әкелу шығыны	40,5	21,4	24,5	
4.	Табыс				



13-САБАҚ. МАТЕМАТИКАЛЫҚ ФУНКЦИЯЛАР

MS Excel 2010 электронды кестесінде экономикалық, инженерлік мәселелерді шешуде кеңінен қолданылатын математикалық функцияларды үйренеміз.

1-жаттығу. MS Excel 2010-ның математикалық функцияларын пайдалана отырып $y=\arccos x$, $y=\cos x$, $y=e^x$, $y=\csc x$ функцияларының аргументінің түрлі мәндерін есепте.

Орындау:

- A1:C1 ұяшықтарына

	S1		f _x
	A	B	C
1	-0,9	0,2	-0,9

мәндерін енгіземіз;

▪ **Формулы** қойыншасының **Математические** тобындағы **acos**, **cos**, **exp** функцияларын пайдаланып A2:A5 ұяшықтарында сәйкес өрнектерді пайда етеміз. Мысалы: A4-ке **=EXP(A1)** және A5-ке **=EXP(A4)** енгізіледі, өйткені $\csc x$ күрделі функция.

▪ **Салыстырмалы сілтеме** жасау тәсілін пайдаланып ұяшықтарға амалды жалғастырамыз:

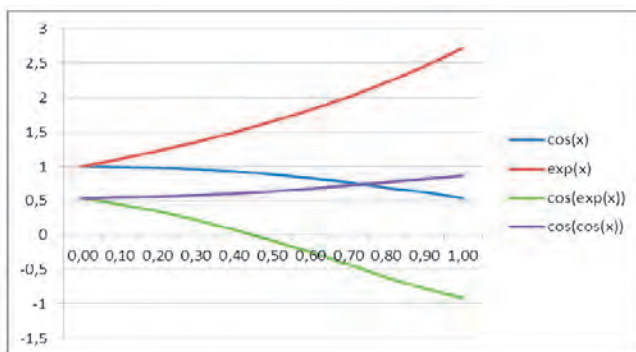
	S1		f _x
	A	B	C
1	-0,9	0,2	-0,9
2	2,690566	1,369438	2,690566
3	0,62161	0,980067	0,62161
4	0,40657	1,221403	0,40657
5	0,918483	0,342328	0,918483

Осылайша қалаған күрделі функцияларды есептеп шығаруға болады.

2-жаттығу. x -тің 0, 0,1, 0,2, 0,3, 0,4, 0,5, 0,6, 0,7, 0,8, 0,9, 1 мәндеріне $y=\cos x$, $y=e^x$, $y=\cos e^x$, $y=\cos(\cos x)$ функциялар графигін жаса.

Орындау:

- **F3:P3** ұяшықтарына x -тің мәндері енгізіледі;
- **F4** ұяшығына **Формулы** қойыншасының **Математические** тобынан **cos** функциясы таңдалып, оның аргументі үшін **F3** ұяшығына сілтеме жасалады;
- **F4** ұяшығында есептелінетін өрнек **G4:P4** блогына **салыстырмалы сілтеме** жасау арқылы игеріледі;
- **F5** ұяшығына $\exp(F3)$, **F6** ұяшығына $\cos(\exp(F3))$, **F7** ұяшығына $\cos(\cos(F3))$ өрнектері енгізіледі және сәйкесінше **F5:P5**, **F6:P6**, **F7:P7** блоктарына **салыстырмалы сілтеме** жасау арқылы игеріледі;
- **Вставка** қойыншасының **График** тобының тиісті тармағын таңдау арқылы қарастырылған функциялардың графигі жасалады:



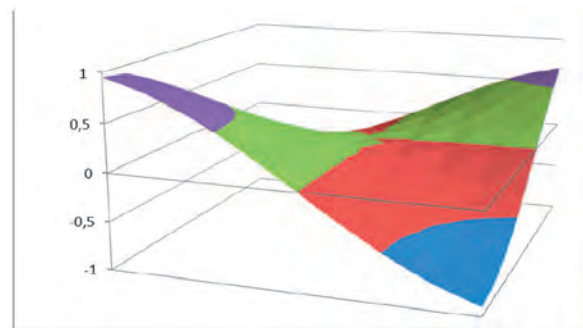
3-жаттығу. $z=\cos x \cos y$, функциясына сәйкес кестені жаса .

Орындау:

- **A1:J1** ұяшықтарына мәндер енгізіледі;
- **A2:A10** ұяшықтарына тиісті мәндер енгізіледі;
- **B2** ұяшығына **=COS(\$A\$1)*COS(B1)** өрнегі жазылады;
- **B2**-ден **B10** ұяшығына дейін аралас сілтеме жасауды пайдаланып формула пайда етіледі;
- **B10**-нан **J10** ұяшығына дейінгі өрнек аралас сілтеме жасауды пайдаланып пайда етіледі. Нәтижеде мына терезе пайда болады:

J10		=COS(A9)*COS(\$J\$1)								
	A	B	C	D	E	F	G	H	I	J
1	0,00	0,30	0,60	0,90	1,20	1,50	1,80	2,10	2,40	2,70
2	0,3	0,955336	0,825336	0,62161	0,362358	0,070737	-0,2272	-0,50485	-0,73739	-0,90407
3	0,6	0,912668	0,788473	0,593847	0,346174	0,067578	-0,21705	-0,4823	-0,70446	-0,86369
4	0,9	0,788473	0,681179	0,513037	0,299067	0,058382	-0,18752	-0,41667	-0,6086	-0,74616
5	1,2	0,593847	0,513037	0,386399	0,225245	0,043971	-0,14123	-0,31382	-0,45837	-0,56198
6	1,5	0,346174	0,299067	0,225245	0,131303	0,025632	-0,08233	-0,18293	-0,2672	-0,3276
7	1,8	0,067578	0,058382	0,043971	0,025632	0,005004	-0,01607	-0,03571	-0,05216	-0,06395
8	2,1	-0,21705	-0,18752	-0,14123	-0,08233	-0,01607	0,051621	0,114702	0,167537	0,205407
9	2,4	-0,4823	-0,41667	-0,31382	-0,18293	-0,03571	0,114702	0,25487	0,37227	0,456417
10	2,7	-0,70446	-0,6086	-0,45837	-0,2672	-0,05216	0,167537	0,37227	0,543749	0,666657

- **B2:J10** блогы ажыратылып алынады;
- **Вставка-Другие-Поверхность** тізбектілігі таңдалып, қарастырылып жатқан функциялардың графигі жасалады;
- пайда болған графиктен маңызды болмаған жазулар өшіріледі. Нәтижеде төмендегі график пайда болады:



ЕСІНДЕ САҚТА !

Функция графиктерін жасау үшін MS Excel 2010-ның ұяшықтарына функция мәндері енгізіліп, **Вставка** қойыншасының **Диаграммы** тобындағы **График** түймешігі пайдаланылады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Функция графигін жасау тізбектілігін түсіндіріп бер.
2. $y = \sin x$ функциясының графигін жаса.
3. $y = \cos^2 x$ функциясының графигін жаса.



ҮЙГЕ ТАПСЫРМА

1. $y = \sin^2 x$ функциясының графигін жаса.
2. $y = x^2$ және $y = x^3$ функцияларының графиктерін бір терезеде жаса.



14-САБАҚ. КӨБЕЙТІНДІНІ ЕСЕПТЕУГЕ ҚАТЫСТЫ ФУНКЦИЯЛАР

Көбейтінді математикалық өрнектерде кеңінен таралған амалдардың бірі болып, біз осы сабақта көбейтінді қатысқан есептеулердің ерекшеліктерімен танысамыз:

1-тәсіл. Ұяшыққа = белгісі қойылып қажетті өрнектер көбейтіледі;

2-тәсіл. Көбейтіндіні есептеу функциясынан пайдаланылады.

1-жаттығу. MS Excel 2010-да төмендегі кесте бар:

	A	B	C	D	E	F
1	0	10				
2	0,3	11				
3	0,6	12				
4	0,9	13				
5	1,2	14				
6	1,5	15				
7	1,8	16				
8	2,1	17				
9	2,4	18				
10	2,7	19				

- C бағанға A және B бағандағы сандардың көбейтіндісін есепте;
- D бағанға A және B бағандағы сандардың қосындысын есепте;
- E бағанға A, B, C, D бағандағы сандардың көбейтіндісін есепте;
- F бағанға A, B, C, D, E бағандағы сандар қосындысын есепте;

Орындау:

▪ C1 ұяшығында A1:B1 блогының көбейтіндісі = ПРОИЗВЕД(A1:B1) пайда етіледі;

▪ D1 ұяшығында A1:B1 блогының қосындысы =A1+B1 пайда етіледі;

▪ E1 ұяшығында **A1:D1** блогының көбейтіндісі = **ПРОИЗВЕД(A1:D1)** пайда етіледі;

▪ F1 ұяшығында **A1:E1** блогының қосындысы =**СУММ(A1:E1)** пайда етіледі;

▪ Жоғарыдағы формулалар **C2:F10** ұяшықтарына сәйкес түрде көшірме жасалады. Нәтижеде төмендегі терезе пайда болады:

	A	B	C	D	E	F
1	0	10	0	10	0	20
2	0,3	11	3,3	11,3	123,1	149
3	0,6	12	7,2	12,6	653,2	685,6
4	0,9	13	11,7	13,9	1903	1942
5	1,2	14	16,8	15,2	4290	4337
6	1,5	15	22,5	16,5	8353	8409
7	1,8	16	28,8	17,8	14764	14828
8	2,1	17	35,7	19,1	24343	24417
9	2,4	18	43,2	20,4	38071	38155
10	2,7	19	51,3	21,7	57108	57202



ЕСІНДЕ САҚТА !

Берілген сандардың көбейтіндісін есептеу үшін **ПРОИЗВЕД** функциясы пайдаланылады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Берілген сандардың көбейтіндісін есептеудің неше тәсілін білесің?
2. Берілген 20 санының көбейтіндісін **ПРОИЗВЕД** функциясын пайдаланбай есепте.
3. Төменгі 10, 20, 40, 80, 160 сандарының қосындысын 40 санына бөл.



ҮЙГЕ ТАПСЫРМА

1. 10,20, 40, 80, 160 сандарының көбейтіндісінен 56 санын айыр.
2. **A, B, C** бағандарына қалаған **20** сан енгіз. **A** және **C** бағандарындағы сандардың көбейтіндісінен **B, C** бағандардағы сандардың айырмасын **D** бағанда пайда ет.



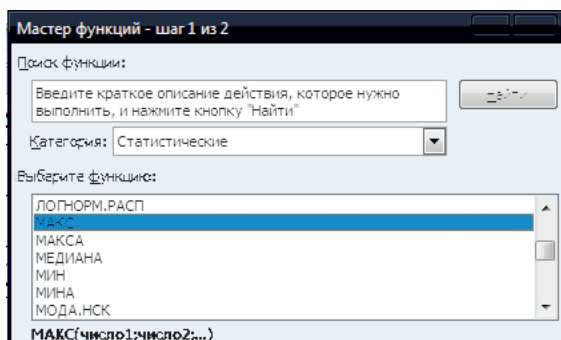
15-САБАҚ. СТАТИСТИКАЛЫҚ ФУНКЦИЯЛАР

MS Excel 2010-ның мүмкіндіктерінің тағы бірі – статистикалық функцияларды пайдаланып тиісті мысалдарды шешу саналады.

Статистикалық функцияларды пайдаланудың екі тәсілі бар:

1-тәсіл. Функцияларды ұяшыққа тікелей енгізу арқылы;

2-тәсіл. Формулаар жолынан  түймешігін таңдау арқылы, бұл жағдайда төмендегі терезе ашылады:



MS Excel 2010-да статистикалық функциялар көп болып, төменде кейбір функциялармен және олардың міндеттерімен танысамыз:

Р/с	Функция	Міндеті
1.	МАКС	Берілген сандардың ең үлкенін есептейді.
2.	МИН	Берілген сандардың ең кішісін есептейді.
3.	СРЗНАЧ	Берілген сандардың орташа арифметигін есептейді.
4.	СЧЁТЕСЛИ	Шартты қанағаттандыратын блоктың бос емес ұяшықтарының санын есептейді.

1-жаттығу. A1:C5 блогындағы сандардың ең үлкені мен ең кішісінің орташа арифметигін D8 ұяшығында пайда ету керек.

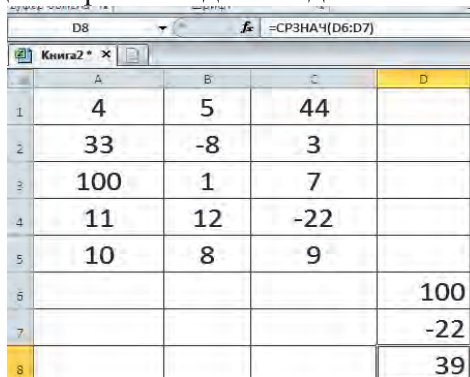
Орындау:

- A1:C5 блогына қалаған сандарды енгіземіз;
- D6 ұяшығына төмендегі формуланы енгіземіз (берілген сандардың ең үлкенін табу формуласы): **=МАКС(A1:C5);**

■ **D7** ұяшығына төмендегі формуланы енгіземіз (берілген сандардың ең кішісін табу формуласы): **=МИН(A1:C5);**

■ **D8** ұяшығына төмендегі формуланы енгіземіз (берілген сандардың орташа арифметигін табу формуласы): **=СРЗНАЧ(D6:D7);**

Нәтижеде төмендегі терезе пайда болады:



	A	B	C	D
1	4	5	44	
2	33	-8	3	
3	100	1	7	
4	11	12	-22	
5	10	8	9	
6				100
7				-22
8				39



ЕСІНДЕ САҚТА !

Статистикалық функцияларды пайдалану екі түрлі тәсілде жүзеге асырылады:

1-тәсіл. Функцияларды ұяшыққа тікелей енгізу арқылы;

2-тәсіл. Менюдің формулалар жолынан  түймешігін таңдау арқылы.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Статистикалық функциялардың **МАКС** функциясын таңдаудың тізбектілігін түсіндіріп бер?
2. Берілген 10 санның ішінен ең үлкен санды тап. Жаттығуды екі тәсілде де орында.



ҮЙГЕ ТАПСЫРМА

1. 4, 6, 8, 9, 10 сандарының орташа арифметигін есепте.
2. Өз бетіңше **СЧЁТЕСЛИ** функциясының көмегімен қандай да бір жаттығуды ойлап тап және оны орында.



16-САБАҚ. ПЫСЫҚТАУҒА АРНАЛҒАН ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

MS Excel 2010-да:

1. $y = \ln 2x + \sin x$ функциясының графигін жаса.
2. 1, 3, 5, 7, 9 сандарының көбейтіндісінен 2, 4, 6, 8, 10 сандарының көбейтіндісін айыр.
3. Берілген 20 санның қосындысын, басқа 10 санның көбейтіндісінің нәтижесіне салыстырмалылығын тап.
4. 10, 20, 30, 40 сандарының кубының түбірін есепте.
5. 900 санының 34 пайызын есепте.
6. 1, 2, 3, 4, 5, 6, 7, 8, 9 сандарының квадраттарының орташа арифметигін есепте.
7. Төмендегі өрнекті есепте: $345 - 45 * 5678 + 12 : 456 - 89 + (34 - 8 * 9)$.
8. $\sqrt[4]{625} - \sqrt[3]{\frac{8}{3}} - 2 * 6$ өрнегін есепте. 9. $\frac{2}{3} + \frac{4}{5} + \frac{6}{7} + \frac{8}{9}$ өрнегін есепте.
10. $\frac{2}{3} + \frac{4}{5} + \frac{6}{7} + \frac{8}{9}$ өрнегін есепте. 11. $\frac{1}{2} - \frac{1}{3 \cdot 4} + \frac{1}{5 \cdot 6} - \frac{1}{7 \cdot 8} + \frac{1}{9 \cdot 10}$ өрнегін есепте.
12. $\frac{\log_3 9}{2} - \frac{\sqrt[3]{67}}{9} + 3 \cdot \frac{5}{\sqrt{36}}$ өрнегін есепте.
13. $\sqrt{2 + \sqrt{2 + \sqrt{2 + \sqrt{2 + \sqrt{2}}}}}$ өрнегін есепте.



17- САБАҚ. MS EXCEL-ДІҢ КӨМЕГІМЕН КЕЙБІР МЫСАЛДАРДЫ ШЕШУ

MS Excel 2010-ның көмегімен теңдеулерді және теңдеулер жүйесін график тәсілде шешу мүмкіндігі бар. Ол үшін теңдеулер немесе теңдеулер жүйесі функциялар көрінісіне келтіріледі. Графикті талдау арқылы теңдеулердің тұспалдық шешімі анықталады.

1-жаттығу. $x^4 - 2x^3 - 3x^2 + 1 = 0$ теңдеуінің шешімі жатқан аралықты тап.

Орындау:

- A1 ұяшығына x , A2-ге $f(x)$ енгізіледі;
- B1:F1-ге x -тің мәндері енгізіледі;

- B2-ге $=B1^4-2*B1^3-3*B1^2+1$ формуласын енгіземіз;
- C2:F2 блогындағы ұяшықтарға жоғарыдағы формуланың көшірмесі көшіріп алынады. Нәтижеде төмендегі кесте пайда болады:

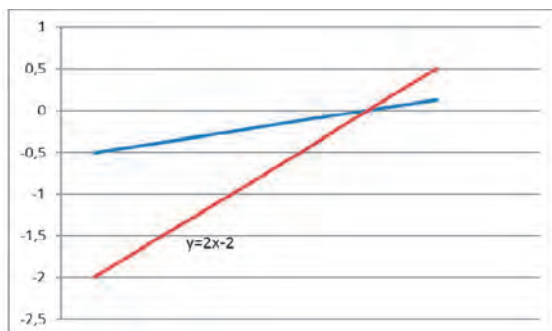
F2		fx =F1^4-2*F1^3-3*F1^2+1				
	A	B	C	D	E	F
1	x	0	1	2	3	4
2	f(x)	1	-3	-11	1	81

Кестеден қарастырылып жатқан теңдеудің шешімі (0;1) және (2;3) аралығында екені белгілі болды. Қарастырылып жатқан аралықты одан да нақтылау есебіне ізделінген шешімге жеткілікті жақындауымыз мүмкін.

2-жаттығу. $\begin{cases} x-2y-1=0 \\ 2x-y+1=0 \end{cases}$ теңдеулер жүйесін шеш.

Шешу. Осы теңдеулер жүйесін шешу үшін төмендегі екі функцияның графикін қарастырамыз: $\begin{cases} y=(x-1)/2 \\ y=2x+1 \end{cases}$

Қарастырылған теңдеулер жүйесінің шешімін табу үшін осы функциялардың ординаталары тең болған x -тің мәндерін іздейміз:



Графикте $x=1$ болғанда, $y=0$ болатыны көрініп тұр.



ЕСІНДЕ САҚТА !

MS Excel 2010-ның көмегімен теңдеулерді және теңдеулер жүйесін график тәсілде тұспалдап шешу мүмкіндігі бар.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Теңдеулерді график тәсілінде шешудің тізбектілігін түсіндіріп бер.
2. Адамның өмір сүрген күнін есептейтін кесте жарат.
3. Квадрат теңдеуді шеш.



ҮЙГЕ ТАПСЫРМА

1. Сынып оқушыларының “Информатика және ақпарат технологиялары” пәнін игеру көрсеткішін диаграмма арқылы көрсет.
2. Бір жылдық қаражатыңды есептейтін кесте жарат.



18-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

1. MS Excel-де шеңбердің ауданын есепте.
2. $\frac{2 - |0.2 + 3|}{1 + 2^3}$ өрнегін есепте.
3. $y = \sin 2x$ функциясының графигін жаса.
4. Төмендегі кестедегі сандардың 15% есепте:

Р/с	Токсандар	Айналым қаржы	Жергілікті бюджетке 15% бөлу
1.	I	1175000	
2.	II	211800	
3.	III	3151000	
4.	IV	4151000	
	Жалпы:		

5. Жылдамдықтың жол мен уақытқа қатысын есепте және оның графигін жаса.

S	v	t
12	5	2,4
10	3	3,333333
5	4	1,25
2,5	5	0,5



19-САБАҚ. БАҚЫЛАУ ЖҰМЫСЫ

1-вариант

1. Бірінші парақтың (Лист1) **B3** ұяшығындағы санның **C8** ұяшығындағы санға бөліндісін екінші парақтың (Лист2) **D9** ұяшығында есепте.
2. MS Excel-де төмендегі өрнекті есепте: $2*(6-5)/5$.
3. MS Excel-де түрлі радиустағы шеңберлердің ұзындығын есепте.
4. $y=\sin x$ функциясының графигін жаса.

2-вариант

1. Бірінші парақтың (Лист1) **B1** ұяшығындағы санның **C3** ұяшығындағы санға бөліндісін екінші парақтың (Лист2) **D3** ұяшығында есепте.
2. MS Excel-де кубтың көлемін есепте.
3. Төмендегі өрнекті есепте: $(6:8)-9:81$.
4. $y=\ln x$ функциясының графигін жаса.

3-вариант

1. **B1** ұяшығына оқушының фамилиясын, **C1** ұяшығына атын, **D1** ұяшығына әкесінің атын енгіз. **E1** ұяшығына осы деректерді біріктірі.
2. **A1:D9** диапазонындағы сандардың ең кішісін тап.
3. Төмендегі өрнекті есепте: $(6-8)*6+1$.
4. $y=x^3$ функциясының графигін жаса.

4-вариант

1. Квадрат функцияның графигін жаса.
2. **A2:E10** диапазонындағы сандардың көбейтіндісін есепте.
3. Төмендегі өрнекті есепте: $\sqrt{9} + \sqrt{81} - 4$.
4. $y=x^3 - 1$ функциясының графигін жаса.



II ТАРАУ. ДЕРЕКҚОР

**СЕН БҰЛ ТАРАУДЫ ОҚЫП ТӨМЕНДЕГІ БІЛІМ, DAҒДЫ,
БІЛІКТІЛІК ПЕН КОМПЕТЕНЦИЯҒА ИЕ БОЛАСЫҢ:**

- Дерекқор туралы ұғым және оларды басқару жүйелері;
- MS Access 2010-ның негізгі элементтері мен аландарының ерекшеліктері;
- MS Access 2010-да дерекқор жасау;
- MS Access 2010-да кестелерді өзара байланыстыру;
- MS Access 2010-да деректерді берілген шаблон бойынша іздеу және қайта іздеу;
- MS Access 2010-да дерекқорды жасауда және редакциялауда формаларды пайдалану;
- MS Access 2010-да математикалық амалдарды орындау.





20-САБАҚ. ДЕРЕКҚОР ТУРАЛЫ ҰҒЫМ

Деректерді пайдалану, яғни оларды белгілі бір технологияның негізінде жинау, сақтау, қайта іздеу және ұзату маңызды мәселелердің бірі болып табылады. Деректер ресурстары оларды пайдалану максималды түрде қолайлы және пайдалы болатындай етіп ұйымдастырылуы қажет. Мысал шешуге әрекеттену, деректер іздеу үдерістеріндегі жаңаша жанасым жаңа технологияны өмірге әкелді. Бұл технологияның негізін деректер қоры мен банктері құрайды.

Дерекқор (ДҚ) – компьютер жадына енгізілген белгілі бір құрылымға ие, бір-бірімен өзара байланысқан және реттелген деректер кешені.

Деректер моделі – деректердің өзара байланысқан құрылымдары мен олардың үстінде орындалатын операциялардың жинағы.

Белгілі бір ақпаратты машинаның ішкі ортасына орналастыру үшін деректердің түрлі құрылымдары мен модельдерін пайдалануға болады. Оның қай бірін таңдау ДҚ-ны жарататын пайдаланушының міндетіне жүктелген болып, бұл көп факторларға байланысты.

ДҚ-ны құрудың негізгі тәсілдері төмендегілерді құрайды:

- Иерархиялық (шежіре) модель – бұл модельде деректер ағаш сияқты көріністе сақталады. Жұмыс істеу бірлігі жазу түрінде болады;
- Тармақтық модель – бұл модельде де ағаш сияқты байланыс болғанымен, түйіндердегі тармактануда шектелу жоқ;
- Реляциялық (өзара байланысты) модель – бұл модельде деректер кесте көрінісінде сақталады.

Деректерді екі түрлі тәсілді пайдалана отырып алуға болады. Олар: жүйеленбеген және жүйеленген.

Жүйеленбеген – деректерді бейнелеу тәсілі туралы келісім болмаса, бұл жағдайда олар жүйеленбеген деп айтылады.

Жүйеленген – деректерді бейнелеу тәсілі туралы келісім болса, бұл жағдайда олар жүйеленген деп айтылады.

Деректердің жүйеленбегеніне мынадай мысалды келтіруге болады: оқушы (Фамилиясы, аты, әкесінің аты, мекенжайы, телефон нөмірі) туралы ақпаратты құрайтын деректерге төмендегідей жазылған

ақпаратты көрсетуге болады: Фамилиясы – **Назаров**, Аты – **Надыр**, Әкесінің аты – **Ахмедович**, Мекенжайы – **Науаи қаласы**, Телефон нөмірі: **+998947777777**.

Осы деректерді кесте көрінісінде келтірсек жүйеленген дерекке айналады:

Р/с	Фамилиясы	Аты	Әкесінің аты	Мекенжайы	Телефон нөмірі
1.	Назаров	Надыр	Ахмедович	Науаи қаласы	+998947777777



ЕСІНДЕ САҚТА !

Дерекқор (ДҚ) – компьютер жадына енгізілген белгілі бір құрылымға ие, бір-бірімен өзара байланысқан және реттелген деректер кешені.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Дерекқор дегенде нені түсінесің?
2. ДҚ-ның моделі туралы мәлімет бер.
3. Жүйеленген дерек дегенде нені түсінесің?
4. Сыныптағы оқушылардың аты-жөні, туылған жылы, телефон нөмірі, пәндерден алған бағалары бойынша жүйеленбеген деректерді қалыптастыр.



ҮЙГЕ ТАПСЫРМА

1. 10-сыныптағы оқулықтардың авторлары мен басылған жылы бойынша жүйеленбеген деректерді қалыптастыр.
2. Сыныптағы оқушылардың аты-жөні, туылған жылы бойынша жүйеленген деректерді қалыптастыр.



21-САБАҚ. ДЕРЕҚҚОРДЫ БАСҚАРУ ЖҮЙЕЛЕРІ

ДҚ-ны құру, оларға қосымша деректерді енгізу және деректерді пайдалану үшін арнайы ДҚ-лармен жұмыс істейтін бағдарламалық қамтамасыз ету өнімдерін жарату қажет болды. Мұндай бағдарламалар кешені дерекқорды басқару жүйелері деп аталады.

Дерекқорды басқару жүйесі (ДҚБЖ) – пайдаланушылар тарапынан ДҚ-ды жарату, толықтыру және біргелікте қолдану үшін арналған бағдарламалық құралдар жүйесі.

ДҚБЖ-ларға MS Access, OpenOffice.org Base, Cache, IMS, Firebird, MySQL сияқты бағдарламалық құралдар мысал болады.

ДҚБЖ-ның негізгі ерекшеліктері – олар тек деректерді енгізу мен сақтауда қолданылатын процедуралар құрамын ғана емес, олардың құрылымын да бейнелейді. Деректерді өзінде сақтайтын және ДҚБЖ арқылы басқарылатын файл бұрын деректер банкі, кейіннен “Дерекқор” деп айтыла бастады.

ДҚ-ны жаратуда төмендегілерге көңіл бөлу қажет:

- деректердің түрі, көрінісі оларды пайдаланатын бағдарламаларға байланысты болмауы қажет;
- дерекқордағы қажетті деректі білу немесе іздеу үшін қандай да бір бағдарламаны құруға қажеттілік туындамауы керек.

Қазіргі уақытта дерлік барлық ДҚБЖ-лар реляциялық модельдер негізінде құрылатынын айтып өткен жөн. Сондықтан Microsoft корпорациясы да кеңінен таралған бағдарламалық құралдарды жаратуда. Бұл бағдарламалық құралдар қалаған салада жоғары деңгейдегі кәсіби құжаттарды даярлау мүмкіндігін береді. Соның бірі – ДҚ-лармен жұмыс істеуге арналған Microsoft Access бағдарламасы.



ЕСІНДЕ САҚТА!

Дерекқорды басқару жүйесі (ДҚБЖ) – пайдаланушылар тарапынан ДҚ-ны жарату, толықтыру және біргелікте қолдану үшін арналған бағдарламалық құралдар жүйесі.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. ДҚБЖ-ға сипаттама бер.
2. ДҚ-ны жарататын қандай бағдарламалық құралдарды білесің?
3. ДҚ-ның модельдері туралы әңгімелеп бер.
4. Сыныптағы оқушылардың аты-жөні, мекенжайы туралы деректерді қалыптастыр.



ҮЙГЕ ТАПСЫРМА

1. Пәндерге қатысты кітаптардың аты, баспадан шыққан жылы мен авторлары туралы деректерді қалыптастыр.
2. Еліміздің облыстары, олардың жер көлемі мен тұрғындарының саны туралы деректерді қалыптастыр.

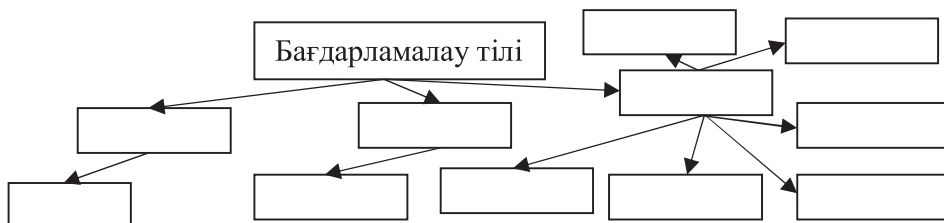


22-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

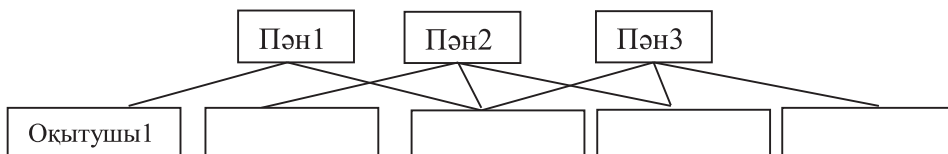
1. Сынып оқушылары туралы ДҚ жарату үшін модельдер жарат.
2. Төмендегі кестені MS Excel 2010-да жарат және оны сынып оқушылары туралы деректермен толықтыр:

Р/с	Аты-жөні	Пәндерден алған бағалары				
		Математика	Физика	Химия	Биология	Тарих
1.						
2.						
3.						

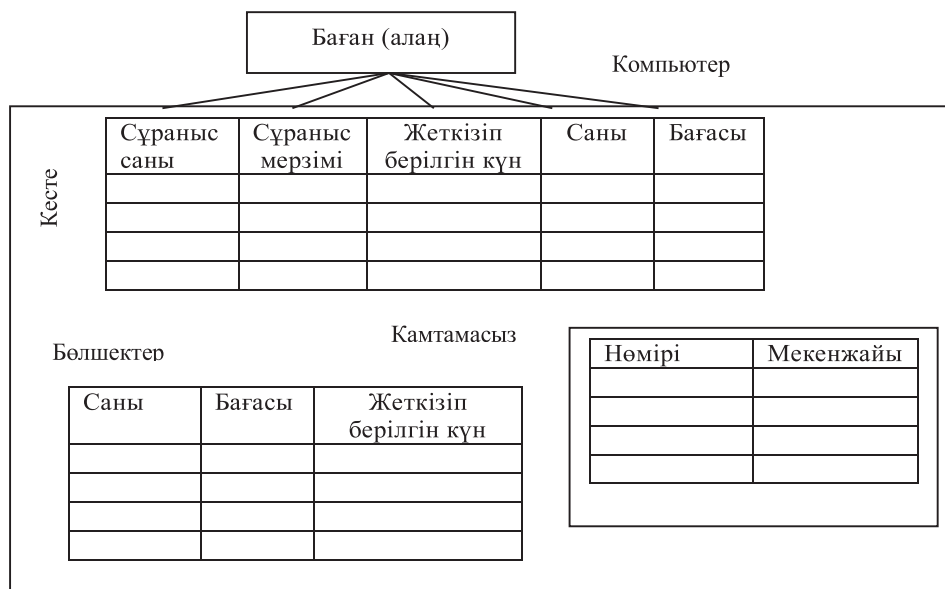
3. Төмендегі иерархиялық модельдің сызбасын қағазға түсіріп, ұяшықтарды толықтыр:



4. Төмендегі тармақтық модельдің сызбасын қағазға түсіріп, ұяшықтарды толықтыр:



5. Төмендегі реляциялық модельдің сызбасын қағазға түсіріп, ұяшықтарды толықтыр:



6. Иерархиялық модельдерге мысалдар айт және олардың сызбасын жаса.

7. Тармақтық модельдерге мысалдар айт және олардың сызбасын жаса.

8. Реляциялық модельдерге мысалдар айт және олардың сызбасын жаса.



23-САБАҚ. MS ACCESS 2010-НЫҢ НЕГІЗГІ ЭЛЕМЕНТТЕРІ МЕН АЛАҢДАРЫНЫҢ ЕРЕКШЕЛІКТЕРІ

Microsoft Access 2010 ДҚБЖ-ның реляциялық түріне жатады, қазіргі уақытта кеңінен таралған дерекқорды басқару жүйелеріне енеді және ол ДҚ-ны жарату, ДҚ-да деректерді сақтау, іздеу мен қайта іздеуді автоматтандыруға арналған.

MS Access 2010-ның тағы бір қосымша артықшылығы оның Microsoft Office-тің құрамына енетін MS Word, MS Excel сияқты бағдарламалармен өзара бір ортада екендігінде. Бұл бағдарламаның

қосымшаларында жаратылған деректер бір қосымшадан екінші қосымшаға оңай импортталауы және экспортталуы көзделген.

MS Access 2010-ның көмегімен қарапайым пайдаланушы үлкен және күрделі ДҚ-ны *бағдарламалық қамтамасыз етудің* ақ жарату және қолдану мүмкіндігіне ие болады.

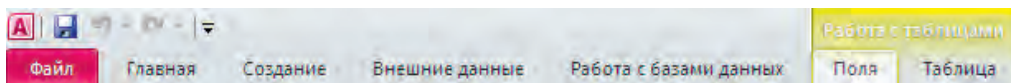
MS Access 2010-ды жүктеудің үш түрлі тәсілін қарастырайық:

- пайдаланушының жұмыс столындағы таңбашаның көмегімен;
- бағдарламаның таңбашасының көмегімен;
- басты менюдің көмегімен.

MS Access 2010-мен жұмыс істеу

MS Access 2010-ның интерфейсінің жоғарғы жолында деректермен түрлі амалдарды орындауға арналған меню орналасқан.

Ол **Файл**, **Главная**, **Создание**, **Внешние данные**, **Работа с базами данных**, **Поля**, **Таблица** қойыншаларынан құралған.



Файл – деректерді жадта сақтау, файлдарды таңдау, деректерді қағазға басып шығару, бағдарламаның параметрлерін өңдеу мүмкіндігіне ие.

Главная – кестедегі деректерді сүзу, сұрыптау, қайта өңдеу сияқты бірқатар істерді жүзеге асыру мүмкіндігін береді.

Создание – кестелер, сұраунамалар, пішіндер, есеп беру, модульдер, макростар жарату сияқты істерді жүзеге асырады.

Внешние – жаратылған деректерді импорт, экспорт жасау және деректерді электронды мекенжайға жіберу сияқты істерді жүзеге асырады.

Работа с базами данных – макростар жарату, жаратылған кестелерді өзара байланыстыру, SQL сервер қорымен байланыстыру сияқты істерді жүзеге асырады.

Поля – алаң түрлерін, қасиеттерін өзгертеді, кесте пішімдерін редакциялайды, математикалық функцияларды пайдаланады.

Таблица – макростар жаратады, кестелерді өзара байланыстыру ерекшеліктерін орнатады.

MS Access 2010-ның негізгі элементтерін төмендегілер құрайды:

- **Таблицы** – деректерді сақтау үшін қызмет етеді;
- **Запросы** – деректерді таңдау шарттарын беруге, оларға өзгертулер енгізуге қызмет етеді;
- **Формы** – деректерді көруге және редакциялауға қызмет етеді;
- **Страницы** – HTML (гипермәтін) пішіміндегі файлдар, олар MS Access 2010-да Internet Explorer браузерінің көмегімен деректерді көру үшін қолданылады;
- **Отчеты** – деректерді жалпыландыру және басып шығару мүмкіндігін береді;
- **Макрос** – бір немесе бірнеше амалдарды автоматты түрде орындайды.

MS Access 2010-дағы алаңдар деректерді логикалық құрастырудың элементар бірлігі болып саналады.

MS Access 2010-да бейнеленетін алаң түрлері

Түрі	Сипаттамасы
Мәтіндік (Текстовый)	Алаң түрлі мәтіндерден құралған болып, оның ұзындығы (көлемі) 255 белгіден аспауы қажет.
МЕМО алаңы (поле МЕМО)	МЕМО алаңында көлемі 65535 белгіден көп болмаған мәтіндік және сандық деректер сақталады.
Сандық (Числовой)	Алаң түрі сандық мәндерді сақтайды, олардың диапазоны алаң өлшемі параметрімен анықталады.
Мерзім/уақыт (Дата/время)	Алаң мәні мерзім мен уақыттан құралған (8 байт) және 100-ден 9999-ға дейінгі жыл диапазонын енгізуге болады.
Ақшалық (Денежный)	Алаң мәні сандар болып, оған бүтін нүктесінен 15 разряд солға және 4 разряд оңға сандар сақтауға болады.
Есептегіш (Счетчик)	Кестеге жаңа деректер қосылғанда автоматты түрде бір мәнге артып барады.
Логикалық (Логический)	Алаң “Иә” немесе “Жоқ” мәндерін сақтайды. MS Access-те “1” – “Иә” және “0” – “Жоқ”

	жағдайында қолданылады
OLE объекті алаңы (Поле объекта OLE)	Алаңға OLE сервері тарапынан қайта өңделген объектілер орналастырылады.
Гиперсілтеме (Гиперссылка)	Бұл алаң қатты дискінің басқа файлына гипербайланысты жаратады.
Қосу (Вложение)	Математикалық функциялармен жұмыс істеу мүмкіндігін жаратады.



ЕСІНДЕ САҚТА !

MS Access 2010-ның файлдары “.accdb” кеңейтпесіне ие.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. MS Access 2010 ДҚБЖ-ның қай түріне жатады?
2. MS Access 2010-дағы объектілердің негізгі элементтерін түсіндіріп бер?
3. MS Access 2010-дағы алаң түрлері туралы айт?
4. MS Access 2010-ды іске қосып, оның негізгі терезесінде жаңа дерекқорды (Новая база данных) жарат.



ҮЙГЕ ТАПСЫРМА

1. MS Access 2010-да жаңа дерекқор жарат. Дерекқордың атын “оқушылар” деп қойып, оны D дискіне сақта.
2. MS Access 2010-да бейнеленетін алаң түрлерін талдап шығыңдар.



24-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

1. MS Access 2010-ды төмендегі тәсілдерді пайдаланып іске қос:
 - пайдаланушының жұмыс столындағы таңбашаның көмегімен;
 - MS Access 2010-ның таңбашасының көмегімен;
 - басты менюдің көмегімен

2. MS Access 2010-ның менюсіндегі **Файл, Главная, Создание, Внешние данные, Работа с базами данных, Поля, Таблица** қойыншаларының мүмкіндіктерін түсіндіріп бер.

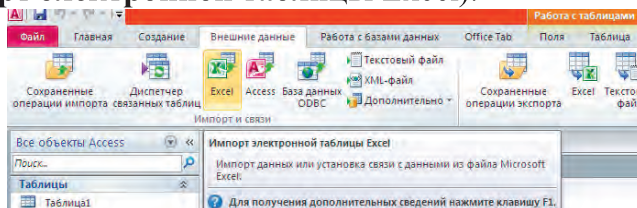
3. MS Access 2010-дағы негізгі элементтерінің міндеттерін түсіндіріп бер.

4. MS Access 2010-дағы алаң түрлерінің міндеттерін түсіндіріп бер.

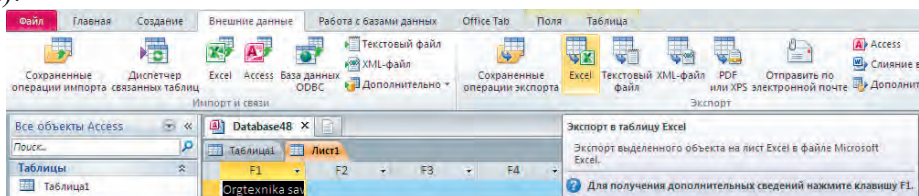
5. MS Excel 2010-да төмендегі кестені жарат:

Р/с	Фамилиясы	Аты	Туылған жылы	Телефон нөмірі
1.				
2.				
3.				
4.				

Пайда болған кестені MS Access 2010-ға импортта (**Внешние данные-Импорт электронной таблицы Excel**):



6. MS Access 2010-да жаратылған файлды жүкте және оны MS Excel 2010-қа экспортта (**Внешние данные-Экспорт в таблицу Excel**):

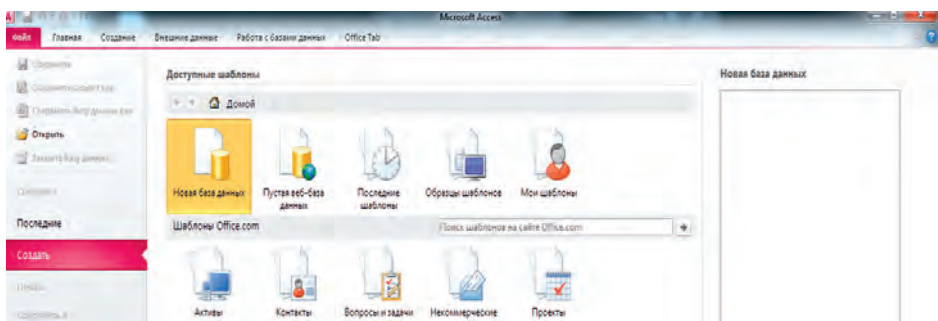


7. MS Access 2010-да жаратылған қандай да бір файлды **Файл-Открыть** тізбектілігі арқылы жүкте.

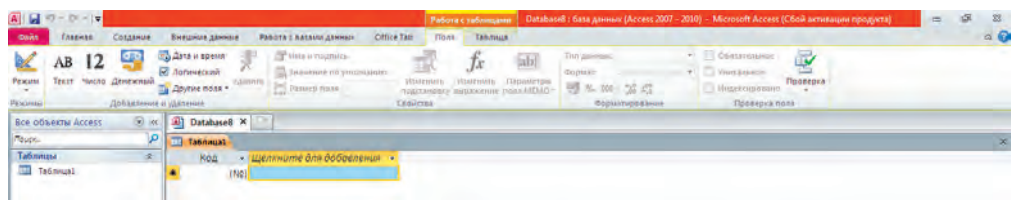


25-САБАҚ. MS ACCESS 2010-ДА ДЕРЕКҚОР ҚҰРУ

MS Access 2010-да дерекқорды құру үшін Microsoft Office бағдарламаларының арасынан Microsoft Access 2010-ды жүктейміз:



Пайда болған терезеден **Новая база данных** бөлімін таңдаймыз. Нәтижеде төмендегі терезе пайда болады:



MS Access 2010-да кесте жаратып, оған деректерді енгізуді үш түрлі тәсілмен жүзеге асыруға болады:

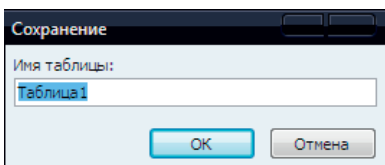
1. Деректерді тікелей енгізу.
2. Режим-Конструктор.
3. Создание-Конструктор таблиц.

MS Access 2010 терезесінде бір ғана кесте жарату үшін **Режим-Конструктор** тізбектілігінен пайдалануға болады. Ал бірнеше кестені жаратып, оларды өзара байланыстыру үшін менюдің **Создание** қойыншасынан **Конструктор таблиц** түймешігін басу қажет.

1-жаттығу. MS Access 2010-да оқушылар туралы деректер қалыптастыру тәсілі.

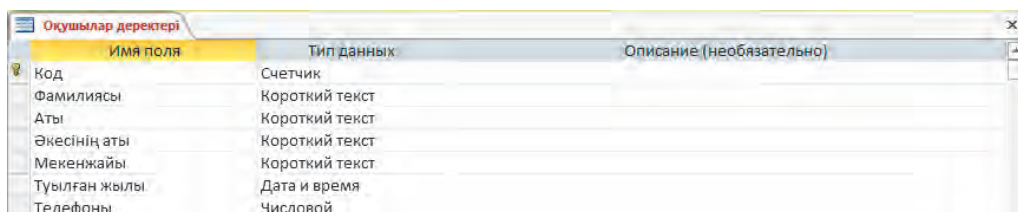
Орындау:

■ MS Access 2010-ның **Режим-Конструктор** тізбектілігі таңдалады. Нәтижеде төмендегі терезе пайда болады:

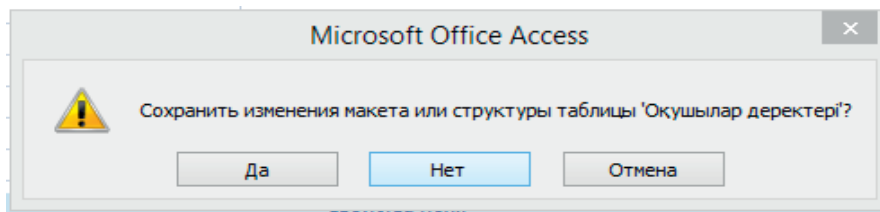


■ **Имя таблицы** жолына “Оқушылар деректері” атымен сақтаймыз және кестені төмендегідей қалыптастырамыз:

Оқушының идентификациялық нөмірі	Сандық (числовой)
Оқушының фамилиясы	Мәтіндік (текстовой)
Оқушының аты	Мәтіндік (текстовой)
Оқушының әкесінің аты	Мәтіндік (текстовой)
Оқушының мекенжайы	Мәтіндік (текстовой)
Оқушының туылған жылы	Уақыт (Дата/время)
Оқушының телефоны	Сандық (числовой)

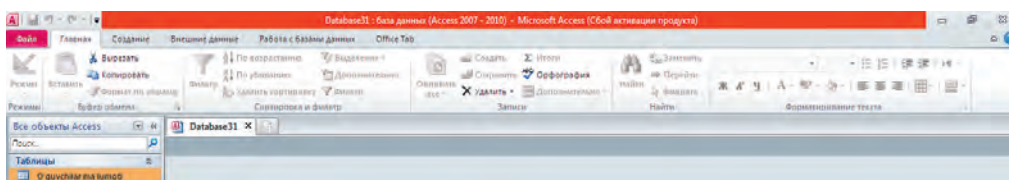


■ бұл терезеден шығамыз (терезенің бұрышындағы x белгісі таңдалады). Нәтижеде төмендегі терезе пайда болады:



■ терезедегі **Да** түймешігі таңдалады (“Оқушылар деректері” атты кесте пайда болады).

■ оқушылар туралы дерекқорды қалыптастыру үшін, “Оқушылар деректері” дейілген жазудың үстіне тышқанның түймешігі екі рет басылады:



■ пайда болған дерекқор оқушылар туралы деректермен толықтырылады:

Код	Фамилиясы	Аты	Әкесінің аты	Мекенжайы	Туылған жылы	Телефоны
1	Махмұтов	Ахмет	Нажімович	Науаи қ., М. Тараби к. 117	23.09.2003	944826749
2	Махмұтова	Анарғұл	Ахатовна	Науаи қ., Науаи к. 23.26	06.09.2004	932456789

■ жаратылған дерекқорды компьютер жадына сақтаймыз (**Файл-Сохранить базу данных как**).

MS Access 2010-да жаратылған кестенің баған жазуын өзгерту мүмкіндігі бар. Ол үшін **Главная-Режим-Конструктор** тізбектілігі таңдалады.



ЕСІНДЕ САҚТА!

MS Access 2010-да кесте жаратып, оған деректерді енгізуді үш түрлі тәсілмен жүзеге асыруға болады: 1. Деректерді тікелей енгізу. 2. Режим-Конструктор. 3. Создание-Конструктор таблиц.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. MS Access 2010-ның менюсіндегі қойыншаларды санап өт?
2. MS Access 2010-ның **Конструктор таблиц** түймешігі қайсы қойыншада орналасқан?
3. MS Access 2010-да **Режим-Конструктор** тізбектілігін таңдап, оқушылардың аты-жөні, мекенжайы, телефон нөмірі туралы ДҚ жарат.



ҮЙГЕ ТАПСЫРМА

1. MS Access 2010-ды үш түрлі тәсілмен іске қос.
2. **Создание-Конструктор таблиц** тізбектілігін таңдап, оқушылардың аты-жөні, мекенжайы, телефон нөмірі туралы ДҚ жарат.



26-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

1. MS Access 2010-да **Режим-Конструктор** тізбектілігін таңдап, оқулықтардың аты, авторы, баспадан шыққан жылы, көлемі туралы деректерді құрайтын кесте жарат.

2. MS Access 2010-да **Создание-Конструктор таблиц** тізбектілігін таңдап, үш кесте жарат (бірінші кестеде Өзбекстан Республикасы облыстары, олардың жер көлемі, екінші кестеде облыстар тұрғындарының саны, зауыт-фабрикалары, үшінші кестеде Ташкент қаласы жер көлемі, тұрғындарының саны, зауыт-фабрикалары туралы деректер болсын).

3. MS Access 2010-да **Главная** қойыншасының **Итоги** түймешігі арқылы бағандардағы 10 санның ең үлкенін, ең кішісін, орташа арифметигін және қосындысын есепте.

4. MS Access 2010-да төмендегі кестені жарат:

Таблица2							
Код	Фамилиясы	Аты	Әкесінің аты	Мекенжайы	Математ	Физика	Информат
1	Назаров	Надыр	Қасымович	Ташкент қаласы	4	3	5
2	Маматов	Уәли	Әліқұлович	Қыбырай ауданы	4	4	4
3	Әшірова	Уәзипа	Ахметовна	Науаи облысы	3	3	5
4	Шәріпова	Шарапат	Тасмұратовна	Самарқанд қаласы	5	5	3
* (№)							
Итог					4	3,75	4,25



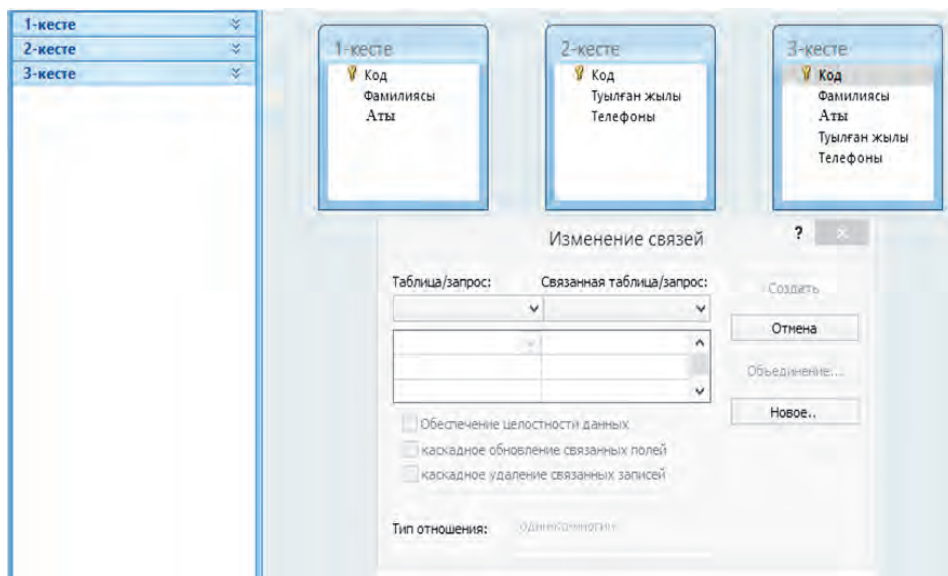
27-САБАҚ. MS ACCESS 2010-ДА КЕСТЕЛЕРДІ ӨЗАРА БАЙЛАНЫСТЫРУ

MS Access 2010-да ДҚБЖ кестелерін бір-бірімен байланыстырудың маңызы ерекше зор. Өйткені, ДҚБЖ бірнеше кестелерді жарату және олардағы деректерді бір жерге жинақтауға пайдаланылады. MS Access 2010-да кестелерді байланыстыру үшін **Работа с базами данных** қойыншасының **Схема данных** түймешігі таңдалады. Кестелерді байланыстыру және осы кестелерді сұраулардың жәрдемімен бір ғана кестеге біріктіру үшін **Создание** қойыншасының **Конструктор запросов** түймешігі таңдалады.

1-жаттығу. Түрлі деректерге ие болған үш кестені байланыстыру мәселесін қарастырайық.

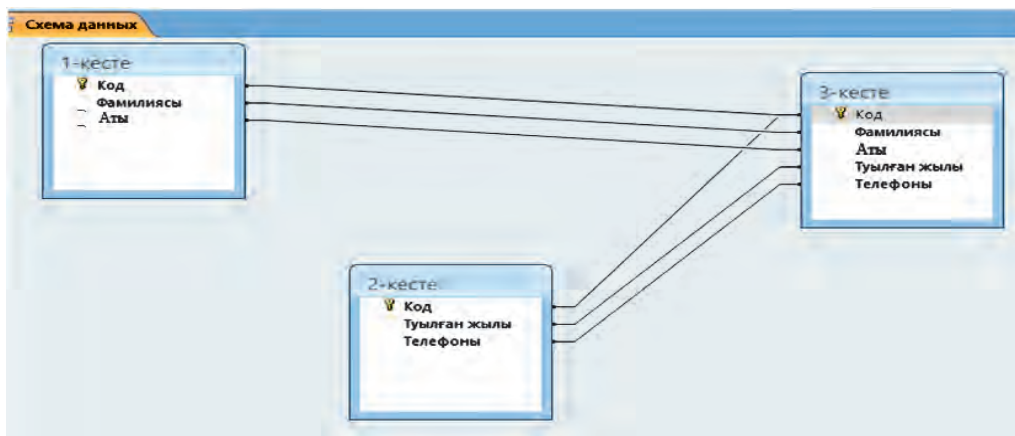
Орындау:

- **Создание** қойыншасының **Конструктор таблиц** түймешігі арқылы үш кесте жаратып, оларды “1-кесте”, “2-кесте”, “3-кесте” деген атпен сақтаймыз. Соң **Работа с базами данных** қойыншасының **Схема данных** түймешігін басып, **Добавление таблицы** терезесін жарат, онда әрбір кестені белгілеп, **Добавить** түймешігін таңдаймыз;
- Пайда болған кестелерді өзара байланыстыру үшін менюдегі **Изменить связи** бөлімін таңда. Нәтижеде мына терезе пайда болады:



- терезеден **Новое** түймешігі тандалып, **Создание** атты жаңа терезе пайда болады. Терезенің **Левая таблица** жолынан “1-кестені”, **Правая таблица** жолынан “3-кестені” таңдаймыз. **Левый столбец** және **Правый столбец** жолында “Код” таңдалып, **ОК** түймешігі басылады;
- **Изменение связей** терезесінен “1-кестенің” бірінші жолына “Код”, екінші жолына “Фамилиясы”, үшінші жолына “Аты” таңдалып **Создать** түймешігі басылады;

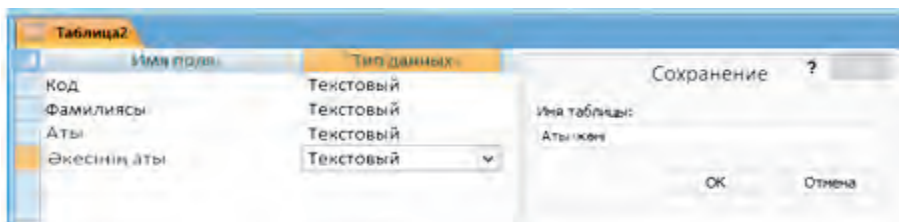
Осылай орындалған тізбектілікті “2-кесте” мен “3-кесте” үшін де орындалады. Нәтижеде төмендегі терезе пайда болады:



2-жаттығу. Оқушылар туралы деректерді үш кестеде қалыптастыр және оларды сұраулардың көмегімен бір ғана кестеге біріктірі.

Орындау:

- бірінші кестеге оқушылардың фамилиясы, аты, әкесінің аты туралы деректер енгізіледі;
- екінші кестеге оқушылардың мекенжайы, телефон нөмірі, сыныбы туралы деректер енгізіледі;
- үшінші кестеге оқушылардың информатика, математика және физика пәндерінен алған бағалары туралы деректер енгізіледі;
- бірінші кестені жасау үшін MS Access 2010-ды іске қосып, менюдің **Создание** қойыншасынан **Конструктор таблиц** түймешігі таңдалады. Пайда болған кестені төмендегідей толықтырамыз:



- жаратылған кестені “Аты-жөні” атауымен сақтаймыз;
- екінші кестені жасау үшін менюдің **Создание** қойыншасынан **Конструктор таблиц** түймешігі таңдалады. Пайда болған кестені төмендегі тәртіппен толықтырамыз:

Имя поля	Тип данных
Код	Текстовый
Мекенжайы	Текстовый
Телефоны	Числовой
Сыныбы	Текстовый

Сохранение

Имя таблицы:
Деректер

ОК Отмена

- кестені “Деректер” атауымен сақтаймыз;
- үшінші кестені жасау тәртібі де жоғарыдағы тізбектілікпен жүзеге асырылады және төмендегі тәртіппен толықтырылады:

Имя поля	Тип данных
Код	Текстовый
Информатика	Числовой
Математика	Числовой
Физика	Числовой

Сохранение

Имя таблицы:
Пәндер

ОК Отмена

- кестені “Пәндер” атауымен сақтаймыз және кестелерді төмендегі тәртіппен толықтырамыз:

а)

Код	Фамилиясы	Аты	Әкесінің аты
1	Назаров	Ысмайыл	Махмұтович
2	Қасымов	Уәли	Әлімович
№			

ә)

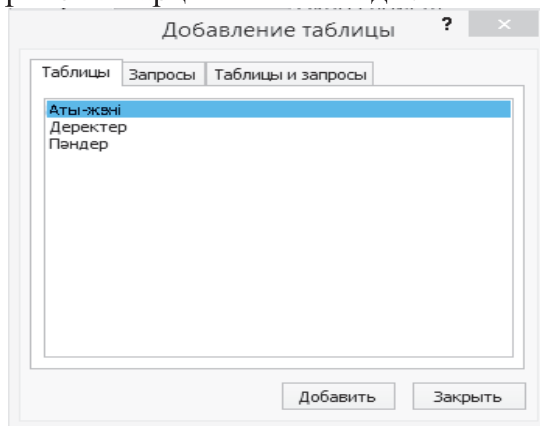
Код	Мекенжайы	Телефоны	Сыныбы
1	Науаи	944826749	10 А
2	Самарқант	4387778	10 Б
№			

б)

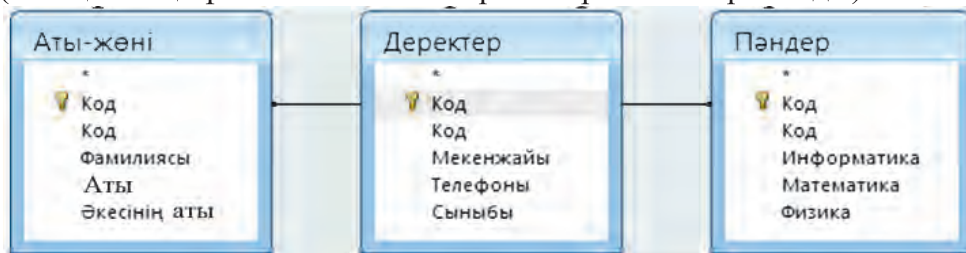
Код	Информатика	Математика	Физика
1	5	4	3
2	3	4	4
№			

■ кестелер деректермен толықтырылған соң, менюдің **Создание** қойыншасынан **Конструктор запросов** түймешігі таңдалып, төмендегі тізбектілік орындалады:

- а) “Аты-жөні” кестесі таңдалып, **Добавить** түймешігі таңдалады.
- ә) “Деректер” кестесі таңдалып, **Добавить** түймешігі таңдалады.
- б) “Пәндер” кестесі таңдалып, **Добавить** түймешігі таңдалады және **Заккрыть** түймешігі арқылы шығылады.



■ Пайда болған үш пішінді төмендегі тәртіппен байланыстырамыз (тышқанның түймешігін баса отырып сырғытып тартылады):



■ үш кестедегі тиісті алаң мен шарттарды таңдаймыз. Ол үшін **Поле** жолының бірінші бағанына тышқанның нұсқағышы әкелініп сол түймешігі басылса, төмендегідей көрініс пайда болады:

Поле:	Фамилиясы
Имя таблицы:	Аты-жөні
Сортировка:	
Вывод на экран:	☒
Условие отбора:	
или:	

- пайда болған кесте төмендегідей деректерге толықтырылады:

Поле:	Фамилиясы	Аты	Әкесінің аты	Мекенжайы	Телефоны	Сыныбы	Информатика	Математика	Физика
Имя таблицы:	Аты-жөні	Аты-жөні	Аты-жөні	Деректер	Деректер	Деректер	Пәндер	Пәндер	Пәндер
Сортировка:									
Вывод на экран:	☒	☒	☒	☒	☒	☒	☒	☒	☒
Условие отбора:									

- менюден **Выполнить** командасы таңдалады. Нәтижеде төмендегі терезе пайда болады:

Фамилиясы	Аты	Әкесінің аты	Мекенжайы	Телефоны	Сыныбы	Информатика	Математика	Физика
Назаров	Ысмайыл	Махмұтович	Науаи	944826749 10 А		5	4	3
Қасымов	Уәли	Әлімович	Самарқант	4387778 10 Б		3	4	4

Кестеге көңіл бөліп қарасақ, үш кестедегі деректер сұрау арқылы бір ғана кестеге біріктірілген.



ЕСІНДЕ САҚТА !

MS Access 2010-да кестелерді байланыстыру үшін **Работа с базами данных** қойыншасының **Схема данных** түймешігі таңдалады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. MS Access 2010-да кесте жаратудың неше түрлі тәсілін білесің?
2. MS Access 2010-да кестелерді басқару тәсілдерін түсіндіріп бер?
3. MS Access 2010-да сұраулардың көмегімен жасалатын кестедегі деректерді өзгертуге бола ма?
4. Оқушылар туралы деректерді үш кестеде қалыптастыр және оларды сұраудың көмегімен бір ғана кестеге біріктір.



ҮЙГЕ ТАПСЫРМА

1. Түрлі деректерге ие болған төрт кестені **Схема данных** түймешігі арқылы байланыстыр.
2. Оқушылар туралы деректерді төрт кестеде қалыптастыр және оларды сұраудың көмегімен бір ғана кестеге біріктір.



28-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

1. “Информатика және ақпарат технологиялары” пәніне қатысты әдебиеттер, олардың авторлары, баспадан шыққан жылдары туралы ДҚ жарат және сұрау арқылы ДҚ-дан тиісті деректерді ізде.

2. Операциялық жүйелер туралы ДҚ жарат. Жаратылған ДҚ-да операциялық жүйелердің атауы, жаратылған жылы, қандай типке тиісті екені сияқты деректерді сақта.

3. Процессор туралы ДҚ жарат. Жаратылған ДҚ-да Процессордың жасалған жылы, жасаған фирманың аты, жұмыс істеу жылдамдығы, түрі сияқты деректер сақталсын.

4. Жады түрлері туралы ДҚ жарат. Жаратылған ДҚ-да жады түрлері, деректерді оқу және жазу жылдамдығы, сыйымдылығы сияқты деректер сақталсын.

5. Сервистік бағдарламалар туралы ДҚ жарат. Жаратылған ДҚ-да бағдарламаның аты, олардың міндеттері, қайсы операциялық жүйемен жұмыс істейтіні сияқты деректер сақталсын.

6. Антивирус бағдарламалары туралы ДҚ жарат. Жаратылған ДҚ-да бағдарламаның аты, олардың міндеттері, қайсы операциялық жүйемен жұмыс істейтіні, ерекшеліктері сияқты деректер сақталсын.

7. Өзің жайлы деректерді бес кестеде қалыптастыр және оларды біріктірі.

8. Кітаптар туралы деректерді үш кестеде қалыптастыр және оларды сұраудың көмегімен бір ғана кестеге біріктірі.



29-САБАҚ. MS ACCESS 2010-ДА ДЕРЕКТЕРДІ БЕРІЛГЕН ШАБЛОН БОЙЫНША ІЗДЕУ ЖӘНЕ ҚАЙТА ӨНДЕУ

MS Access 2010-да деректерді берілген шаблон бойынша іздеу және қайта өңдеу үшін менюдің **Главная** қойыншасынан **Выделение** түймешігі немесе кесте үстіне тышқанның оң түймешігі басылып, **Текстовые – фильтры – Ровнр...** тізбектілігінің көмегімен жүзеге асыруға болады. Кестедегі деректерді аты-жөні, мекенжайы сияқты деректер бойынша іздеуімізге болады. Іздеп табылған деректерді жою үшін менюдің **Главная** қойыншасынан **Фильтр** түймешігі таңдалады.

1-жаттығу. 10-сынып оқушыларының (аты-жөні, мекенжайы, телефон нөмірі, туылған жылы) дерекқорын жарату және оларды іздеу, қайта өңдеу.

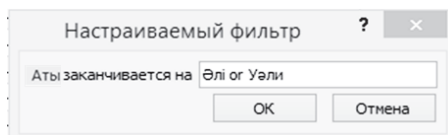
Орындау:

■ MS Access 2010-да оқушылар туралы деректерді мына параметрлер бойынша қалыптастырамыз:

Деректеме						
Код	Фамилиясы	Аты	Мекенжайы	Телефон нөмірі	Сыныбы	Туылған күні
1	Оспанов	Әлі	Науаи қ. М. Тараби к. 120 үй, 2 пәтер	944826749	10 А	12.07.2003
2	Қасымов	Уәли	Науаи қ. Науаи к. 28 үй, 23 пәтер	934567890	10 Б	03.03.2003
3	Қуандықов	Пархат	Самарқант о., Пахташы ауд., Бақыт махалласы	912346789	10 А	07.02.2002
4	Шадманов	Дәурен	Самарқант о., Пахташы ауд., Шыназ махалласы	907865434	10 А	06.10.2003
5	Нәжімов	Тәліп	Самарқант о., Пахташы ауд., Лашын махалласы	935678909	10 А	07.09.2002
6	Әсетова	Жанар	Самарқант о., Пахташы ауд., Лашын махалласы	945670090	10 А	12.04.2004
7	Намазов	Нышан	Науаи қ. Науаи к. 29 үй, 19 пәтер	912345696	10 А	04.05.2003
8	Лүтпіллаев	Ләзиза	Науаи қ. Науаи к. 34 үй, 2 пәтер	903458976	10 А	08.09.2002
9	Нышанов	Ахмет	Науаи қ. Науаи к. 36 үй, 5 пәтер	945678912	10 А	05.03.2003
10	Ахметов	Ахрар	Науаи қ. Науаи к. 1 үй, 6 пәтер	907865432	10 Б	04.07.2004
11	Қасымов	Әлі	Науаи қ. Науаи к. 1 үй, 6 пәтер	917652345	10 А	08.04.2004

Осы кестедегі деректерден “Әлі” атты оқушыны іздеу үшін “Әлі” деген жазуды белгілеп, менюдің **Главная** қойыншасынан **Выделение – Равно – Әлі** тізбектілігі таңдалады. Нәтижеде бағдарлама кестедегі “Әлі” атты оқушылар туралы деректерді іздеп табады. Егер бізге кестеден “Әлі және Уәли” атты оқушыларды іздеу сұралған болса, оқушылардың аты жазылған ұяшыққа тышқанның оң түймешігі басылып, **Текстовый фильтр – Заканчивается на..** тізбектілігі таңдалады және **Настраииваймый фильтр** жолына “**Әлі or Уәли**” тіркесі енгізіліп, **OK** түймешігі таңдалады.

Деректеме						
Код	Фамилиясы	Аты	Мекенжайы	Телефон нөмірі	Сыныбы	Туылған күні
1	Оспанов	Әлі	Науаи қ. М. Тараби к. 120 үй, 2 пәтер	944826749	10 А	12.07.2003
2	Қасымов	Уәли	Науаи қ. Науаи к. 28 үй, 23 пәтер	934567890	10 Б	03.03.2003
3	Қуандықов	Пархат	Самарқант о., Пахташы ауд., Бақыт махалласы	912346789	10 А	07.02.2002
4	Шадманов	Дәурен	Самарқант о., Пахташы ауд., Шыназ махалласы	907865434	10 А	06.10.2003
5	Нәжімов	Тәліп	Самарқант о., Пахташы ауд., Лашын махалласы	935678909	10 А	07.09.2002
6	Әсетова	Жанар	Самарқант о., Пахташы ауд., Лашын махалласы	945670090	10 А	12.04.2004
7	Намазов	Нышан	Науаи қ. Науаи к. 29 үй, 19 пәтер	912345696	10 А	04.05.2003
8	Лүтпіллаев	Ләзиза	Науаи қ. Науаи к. 34 үй, 2 пәтер	903458976	10 А	08.09.2002
9	Нышанов	Ахмет	Науаи қ. Науаи к. 36 үй, 5 пәтер	945678912	10 А	05.03.2003
10	Ахметов	Ахрар	Науаи қ. Науаи к. 1 үй, 6 пәтер	907865432	10 Б	04.07.2004
11	Қасымов	Әлі	Науаи қ. Науаи к. 1 үй, 6 пәтер	917652345	10 А	08.04.2004



Нәтижеде төмендегі терезеде Әлі және Уәли атты оқушылар туралы деректер пайда болады:

Деректеме						
Код	Фамилиясы	Аты	Ұ	Мекенжайы	Телефон нөмірі	Туылған күні
1	Оспанов	Әлі		Науаи қ. М. Тараби к. 120 үй, 2 пәтер	944826749	10 А 12.07.2003
2	Қасымов	Уәли		Науаи қ. Науаи к. 28 үй, 23 пәтер	934567890	10 Б 03.03.2003
11	Қасымов	Әлі		Науаи қ. Науаи к. 1 үй, 6 пәтер	917652345	10 А 08.04.2004

Реттелген деректерді жою үшін менюдің **Главная** қойыншасының **Фильтр** түймешігі таңдалады.

MS Access 2010-да деректерді сұраулардың көмегімен іздеу мүмкіндігі де бар. Осы үдерісті менюдің **Создание** қойыншасының **Конструктор запрос** түймешігі арқылы да жүзеге асыруға болады. **Конструктор запрос**-тың көмегімен бір кестеден бірнеше сан алуан түрлі сұрауларды (кестелерді) жаратуға болады. Әрбір сұрау негізгі кестеден деректердің қандай да бір бөлігін ажыратып алады.

Сұраудың жұмыс істеуінің арқасында негізгі кестеден нәтижелік кесте пайда болады. Бұл кесте жалпы деректен сұрауға қатысты бөлігін қамтыған болады.

Сұраудың жұмыс істеу үдерісінде берілгендер реттелуі, сүзілуі, қосылуы, бөлінуі, өзгеруі мүмкін, сонымен қатар кестелердегі деректер өзгермейді. Бұл сұраулардың негізгі қасиеттерінің бірі болып табылады. Сұраулардың тағы бір қасиеттерінің бірі – олардың нәтижелік есеп-қисабын орындауы.

MS Access 2010-да сұраулардың бірнеше түрлері бар:

- таңдау негізінде сұрау, үлгі негізіндегі сұрау;
- параметрлік сұрау;
- жоғарыда көрсетілген сұраулардың кең таралғаны – таңдау негізіндегі сұрау.

Бұл сұрауды пайдаланудағы мақсат – сұрау шарты бойынша бейнеленетін нәтижелік кестені жасаудан құралған.

2-жаттығу. Жоғарыда келтірілген 1-жаттығудағы оқушылар дерекқорынан, қандай да бір оқушыны аты бойынша ізде.

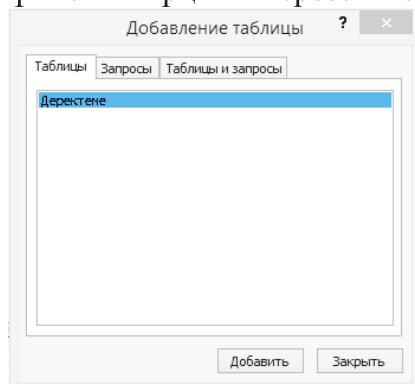
Орындау:

- оқушылар туралы төмендегідей дерекқор жасалған болсын:

Деректеме							
Код	Фамилиясы	Аты	Мекенжайы	Телефон нөмірі	Сыныбы	Туылған күні	
1	Оспанов	Әлі	Науаи қ. М. Тараби к. 120 үй, 2 пәтер	944826749	10 А	12.07.2003	
2	Қасымов	Уәли	Науаи қ. Науаи к. 28 үй, 23 пәтер	934567890	10 Б	03.03.2003	
3	Қуандықов	Пархат	Самарқант о., Пахташы ауд., Бақыт махалласы	912346789	10 А	07.02.2002	
4	Шадманов	Дәурен	Самарқант о., Пахташы ауд., Шыназ махалласы	907865434	10 А	06.10.2003	
5	Нәжімов	Тәліп	Самарқант о., Пахташы ауд., Лашын махалласы	935678909	10 А	07.09.2002	
6	Әсетова	Жанар	Самарқант о., Пахташы ауд., Лашын махалласы	945670090	10 А	12.04.2004	
7	Намазов	Нышан	Науаи қ. Науаи к. 29 үй, 19 пәтер	912345696	10 А	04.05.2003	
8	Лүтпіллаев	Ләзиза	Науаи қ. Науаи к. 34 үй, 2 пәтер	903458976	10 А	08.09.2002	
9	Нышанов	Ахмет	Науаи қ. Науаи к. 36 үй, 5 пәтер	945678912	10 А	05.03.2003	
10	Ахметов	Ахрар	Науаи қ. Науаи к. 1 үй, 6 пәтер	907865432	10 Б	04.07.2004	
11	Қасымов	Әлі	Науаи қ. Науаи к. 1 үй, 6 пәтер	917652345	10 А	08.04.2004	

менюдің **Создание** қойыншасынан **Конструктор** запрос түймешігі таңдалады;

Добавление таблицы терезесінен **Добавить** түймешігі таңдалып, **Закрыть** түймешігі арқылы терезе жабылады:



- пайда болған терезені төмендегі тәртіппен толықтырамыз:

Поле:	Код	Фамилиясы	Аты	Мекенжайы	Телефон нөмірі	Телефон нөмірі	Сыныбы	Туылған күні
Имя таблицы:	Деректеме	Деректеме	Деректеме	Деректеме	Деректеме	Деректеме	Деректеме	Деректеме
Сортировка:								
Вывод на экран:	☒	☒	☒	☒	☒	☒	☒	☒
Условие отбора:			[Издеу]					

осы терезенің қайсы параметр бойынша іздеу қажет болса сол бағанының **Условие отбора** жолы белгіленіп, “[]” жақшаның ішіне қандай да бір сөз енгізіледі. Мысалы, оқушылардың аты бойынша іздейтін сұрау жарату үшін “Аты” бағанына “[Издеу]” сөзі енгізіледі, менюдегі **Выполнить** командасы таңдалған соң, мына терезе шығады:

Введите значение параметра ?

Издеу

Әлі

ОК Отмена

Мұнда біз терезеге ізделінетін оқушының атын енгізіп, **ОК** түймешігін таңдаймыз. Нәтижеде кестедегі “Әлі” атты оқушылар туралы дерек пайда болады:

Код	Фамилиясы	Аты	Мекенжайы	Телефон нөмірі	Сыныбы	Туылған күні
1	Оспанов	Әлі	Науаи қ. М. Тараби к. 120 үй, 2 пәтер	944826749	10 А	12.07.2003
11	Қасымов	Әлі	Науаи қ. Науаи к. 1 үй, 6 пәтер	917652345	10 А	08.04.2004



ЕСІНДЕ САҚТА !

MS Access 2010-да дерекқордағы берілген шаблон бойынша іздеу және қайта өңдеу үшін менюдің **Главная** қойыншасынан **Выделение** түймешігі таңдалады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Дерекқорда берілген шаблон бойынша іздеу қалай жүзеге асырылады?
2. Сұрыпталған деректер қалай жойылады?
3. Кестедегі үш фамилияны бір уақытта іздеп сұрыпта.
4. Оқушылардың дерекқорын жарат және онда төмендегі тізбектілікті жарат:
 - а) Фамилиясы бойынша іздейтін ДҚ жарат;
 - ә) Телефон нөмірі бойынша іздейтін ДҚ жарат;
 - б) Сыныбы бойынша іздейтін ДҚ жарат;



ҮЙГЕ ТАПСЫРМА

1. 10-сыныптағы оқулықтардың дерекқорын жарат және олардың авторлары арқылы іздейтін бағдарлама жарат.
2. 10-сыныптағы оқулықтарының деректері кестесін жүкте және менюдегі **Найти** командасы арқылы (Оқулықтың атын, баспадан шыққан жылын, авторын) ізде.



30-САБАҚ. MS ACCESS 2010-ДА ДЕРЕКҚОРДЫ ЖАРАТУДА ЖӘНЕ РЕДАКЦИЯЛАУДА ФОРМАЛАРДЫ ПАЙДАЛАНУ

Формалар ДҚ-ға жаңа деректерді енгізу және бар деректерді қарастыру үшін пайдаланылады.

Форма деректерді енгізу үшін арналған алаңдары болған электронды бланк көрінісіне ие. Бұл алаңдарға енгізілген деректер тікелей ДҚ-ның кестесіне қосылады.

Жалпы алғанда, **форма** пайда етпей-ақ деректерді кестеге енгізуге болады. Бірақ кестеге деректерді **форма** арқылы енгізу біршама қолайлы болып табылады. Кестеге деректерді енгізу үшін түрлі **формалар** түзіледі.

MS Access 2010-да ДҚ-ға жаңа деректерді енгізудің екі түрлі тәсілі бар:

- тікелей ДҚ кестесіне енгізу;
- арнайы түзілген **формалар** арқылы.

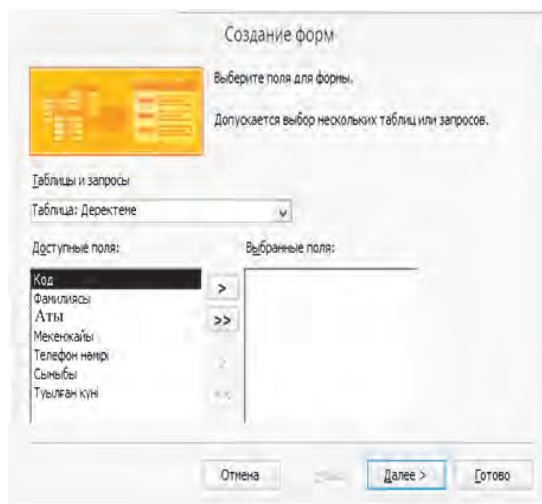
1-жаттығу. Оқушылар туралы дерекқор үшін **форма** жарат және **форманың** көмегімен деректерді енгіз.

Орындау:

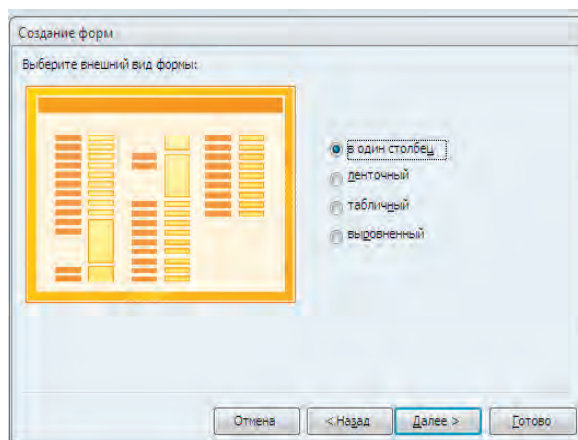
■ “Деректеме” атты кесте жаратылады және ол белгіленіп, менюдің **Создание** қойыншасынан **Мастер форм** түймешігі таңдалады;

■ пайда болған терезеден “>>” белгісін таңдап, **Далее** түймешігі арқылы келесі терезеге өтеміз;

■ терезеде **форма** терезесін төрт түрлі көрініске өткізу мүмкіндігі бар:



- в один столбец (бір баған);
- летночный (таспа сияқты);
- табличный (кестелі);
- выровненный (тегістелген).
- осы терезеден **в один столбец** түймешігі белгіленіп, **Далее** түймешігін таңдаймыз;



- келесі терезеден **Открыть форму для просмотра и ввода данных** түймешігі таңдалады;
 - **Готов** түймешігін басамыз.
- Нәтижеде төмендегі терезе пайда болады:

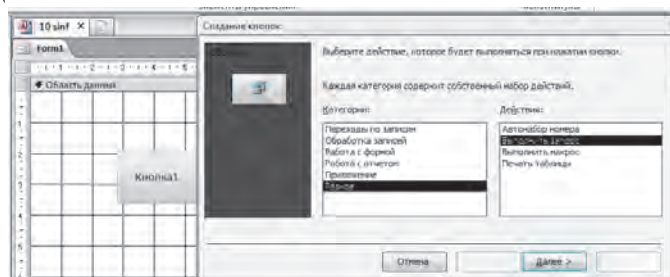
Пайда болған форма терезесінің көмегімен “Деректеме” атты кестеге деректерді енгізу мүмкіндігі бар.

2-жаттығу. 29-сабақта берілген оқушылар туралы деректерді форма арқылы ізде.

Орындау:

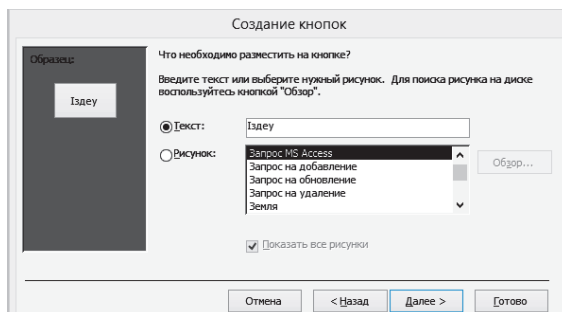
- “Аты арқылы іздеу” атымен сақталған кестені белгілейміз және **Конструктор форм** түймешігін таңдаймыз;

■ **Конструктор** қойыншасынан **Кнопка** түймешігін таңдаймыз және оны форма терезесіне орналастырамыз. Нәтижеде форма терезесі пайда болады:

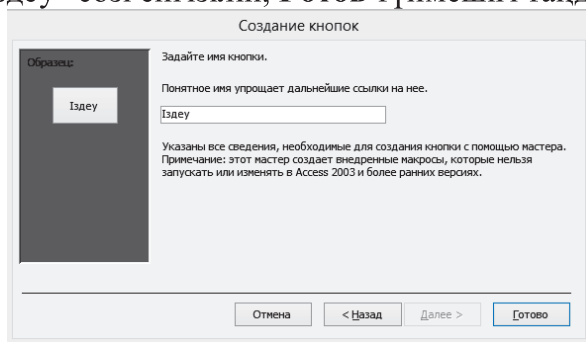


■ пайда болған бірінші терезедегі **Разное** командасы таңдалып, екінші терезеден **Выполнить запрос** командасы белгіленіп, **Далее** түймешігін таңдаймыз;

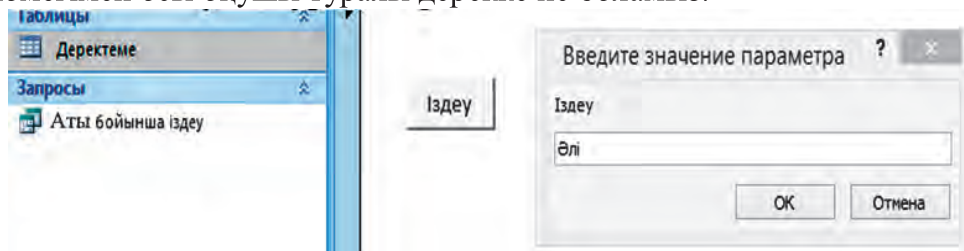
■ терезенің **Текст** жолына “Іздеу” сөзі енгізіледі және **Далее** түймешігін таңдау арқылы келесі терезеге өтеміз:



■ терезедегі **Понятное имя упрощает дальнейшие ссылки на нее** жолына “Іздеу” сөзі енгізіліп, **Готов** түймешігі таңдалады:



Пайда болған бағдарламаны іске қосу үшін **Режим** тобынан **Режим формы** түймешігі таңдалады. Нәтижеде жаратылған бағдарлама жұмыс істеу жағдайына өтеді. Терезедегі “Іздеу” түймешігін таңдау арқылы кестедегі қажетті оқушының атының көмегімен осы оқушы туралы дерекке ие боламыз:



Нәтижеде төмендегідей көріністегі, осы **Әлі** атты оқушылар туралы деректер терезеде пайда болады:

Код	Фамилиясы	Аты	Мекенжайы	Телефон нөмірі	Сыныбы	Туылған күні
1	Оспанов	Әлі	Науаи қ. М. Тараби к. 120 үй, 2 пәтер	944826749	10 А	12.07.2003
11	Қасымов	Әлі	Науаи қ. Науаи к. 1 үй, 6 пәтер	917652345	10 А	08.04.2004

Егер форма терезесіне қосымша өзгерту енгізу қажет болса, оны **Режим** тобының **Конструктор** түймешігін таңдау арқылы жүзеге асыруға болады.



ЕСІНДЕ САҚТА !

MS Access 2010-да ДҚ-ға жаңа деректерді енгізудің екі түрлі тәсілі бар: 1. Тікелей ДҚ кестесіне енгізу. 2. Арнайы түзілген **формалар** арқылы енгізу.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. MS Access 2010-да форма жаратудың тізбектілігін түсіндіріп бер?
2. MS Access 2010-да ДҚ-ға жаңа деректерді енгізудің неше түрлі тәсілі бар?
3. Форма терезесіне түймешікті қалай орналастыруға болады?
4. **Мастер форм**-ды пайдаланып форма жарат.
5. Жаратылған форма терезесі арқылы кестеге деректерді енгіз.



ҮЙГЕ ТАПСЫРМА

1. Математика пәніне қатысты әдебиеттер, олардың авторлары, баспадан шыққан жылдары туралы ДҚ жарат және сұрау арқылы ДҚ-дан тиісті деректі ізде.
2. Кестедегі деректерді форма арқылы қағазға басып шығаратын түймешік жарат.



31-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

1. Сынып оқушылары туралы кесте жарат (фамилиясы, аты, телефон нөмірі).
2. Жаратылған кестеде төмендегі істерді орында: фамилиясы бойынша; аты бойынша; телефон нөмірі бойынша сұрыпта.
3. Жаратылған кестедегі оқушыларды фамилиясы бойынша ретте.
2. 29-30-сабақтарда орындалған жаттығуды төмендегідей бөлімдер бойынша іздейтін сұраулар жарат: фамилиясы; мекенжайы; туылған күні; телефон нөмірі; сыныбы бойынша.
3. **Конструктор форм**-ды пайдаланып форма жарат.
4. Жаратылған форма терезесіне диаграмма орнат.
5. Формаға сурет орнат.



32-САБАҚ. MS ACCES 2010-ДА МАТЕМАТИКАЛЫҚ АМАЛДАР ОРЫНДАУ

MS Access 2010-да дерекқор жаратып, олардың көмегімен кейбір математикалық есеп-қисап істерін орындауға тура келеді. Осы істерді жүзеге асыру математикалық және статистикалық функциялардың көмегімен жүзеге асырылады. Математикалық есептеу істерін менюдің **Создание** қойыншасындағы **Конструктор запросов** түймешігі арқылы жүзеге асыруға болады.

MS Acces 2010-да да MS Excel 2010-дағы сияқты математикалық және қатыстылық амалдарын, математикалық және статистикалық функцияларды, логикалық амалдарды пайдалану мүмкіндіктері бар.

MS Access 2010-дағы математикалық амалдар

1.	+	Қосу.
2.	-	Азайту.

3.	*	Көбейту
4.	/	Бөлу.
5.	^	Дәрежеге шығару.

MS Access 2010-да математикалық қатыстылық амалдары

1.	>	Үлкен.
2.	<	Кіші.
3.	=	Тең.
4.	>=	Үлкен немесе тең.
5.	<=	Кіші немесе тең.
6.	<>	Тең емес.

MS Access 2010-да математикалық функциялар

p/c	Функция	Міндеті
1.	Abs	Санның модулін есептейді.
2.	sqr	Санның квадрат түбірін есептейді.
3.	cos	Санның косинусын есептейді.
4.	sin	Санның синусын есептейді.
5.	tan	Санның тангенсін есептейді.
6.	atn	Санның арктангенсін есептейді.
7.	log	Санның логарифмін есептейді.
8.	Rnd	0 мен 1 аралығында қалаған санды таңдау.
9.	int	Шын санның бүтін бөлігін алу.

MS Access 2010-да статистикалық функциялар

p/c	Функция	Міндеті
1.	max	Бағандағы ең үлкен мәнді табады.
2.	min	Бағандағы ең кіші мәнді табады.
3.	Sum	Баған мәндерінің қосындысын есептейді.

MS Access 2010-да логикалық амалдар

p/c	Функция	Міндеті
1.	NOT	Жоқ
2.	AND	Және
3.	OR	Немесе

1-жаттығу. Үш бағанда берілген сандардың қосындысын төртінші бағанға жаз.

Орындау:

- кесте төмендегідей жаратылады:

Таблица2	
Имя поля	Тип данных
A	Числовой
B	Числовой
C	Числовой
A+B+C	Числовой

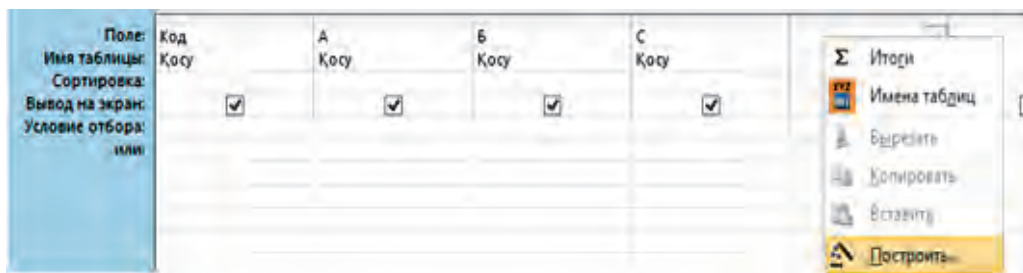
- пайда болған кестені “Қосу” атауымен сақтаймыз;
- сақталған кесте жүктеліп, төмендегі тәртіппен толықтырылады:

Все объекты Access		Қосу				
Таблицы		Код	A	B	C	A+B+C
Қосу		1	7	7	7	
		2	5	5	5	
		3	4	4	4	
		4	3	3	3	
		5	2	2	2	
		6	1	1	1	

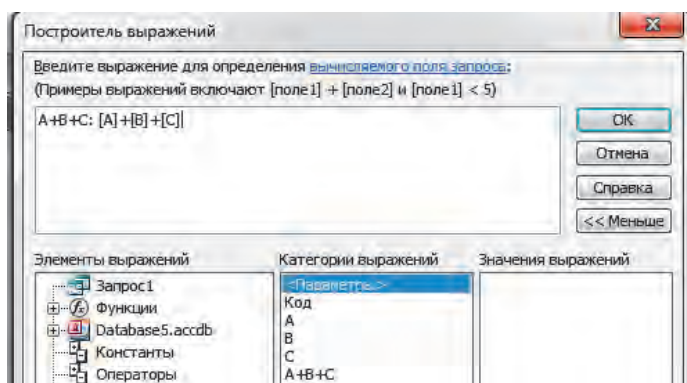
- толықтырылған кестені жауып, меню жолынан **Создания-Конструктор запросов** тізбектілігі таңдалады;
- “Қосу” атты кесте белгіленіп, **Добавить** түймешігі басылады.
- жұмыс терезесін төмендегідей етіп толықтырамыз:

Поле:	Код	A	B	C	
Имя таблицы:	Қосу	Қосу	Қосу	Қосу	
Сортировка:					
Вывод на экран:	☒	☒	☒	☒	☐
Условие отбора:					
или:					

- кестенің келесі бағанына тышқанның оң түймешігін басып контекстік менюдің **построить** түймешігі таңдалады:



■ пайда болған терезе екі бөліктен құралып, бірінші терезеге математикалық есептеу істерінің формулалары енгізіледі. Екінші терезеде математикалық, статистикалық функциялар мен логикалық амалдарды қолдану тәсілдері беріледі. Берілген үш бағандағы сандардың қосындысы төртінші бағанда пайда болуы үшін бірінші айнаға төмендегі кодты енгіземіз: $A+B+C$: $[A]+[B]+[C]$



■ орындалған амалдарды “Сандарды қосу” атымен сақтап, кестені жүктесек, төмендегі көрініс пайда болады:

Все объекты Access		Сандарды қосу				
Таблицы		Код	А	Б	С	A+B+C
Косу		1	7	7	7	21
Запросы		2	5	5	5	15
Сандарды қосу		3	4	4	4	12
		4	3	3	3	9
		5	2	2	2	6
		6	1	1	1	3

Осы кестенің үш бағанындағы сандардың қосындысы төртінші бағанда бейнеленген.



ЕСІНДЕ САҚТА !

MS Access 2010-да математикалық есептеу істері **Создание-Конструктор запросов** тізбектілігі арқылы орындалады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. MS Access 2010-да қандай математикалық амалдар бар?
2. MS Access 2010-да қандай қатыстылық амалдары бар?
3. MS Access 2010-да математикалық функцияларды жаз.
4. MS Access 2010-да статистикалық функцияларды жаз.
5. MS Access 2010-да логикалық амалдарды жаз.
6. Берілген төрт санның көбейтіндісін есептейтін кесте жарат.
7. Бағандағы сандардың түбірін есептейтін кесте жарат.



ҮЙГЕ ТАПСЫРМА

1. Бағандағы сандардың косинусын есептейтін кесте жарат.
2. Бағандағы сандардың орташа мәнін есептейтін кесте жарат.



33-САБАҚ. ҚАЙТАЛАУҒА ҚАТЫСТЫ ТАПСЫРМАЛАР

1. MS Access 2010-дағы дерекқор туралы әңгімелеп бер.
2. ДҚБЖ туралы әңгімелеп бер.
3. MS Access 2010-да сыныптағы оқушылар туралы деректерді сақтайтын ДҚ-ның файлын жарат.
4. MS Access 2010-да сыныптағы оқушылар туралы кестеден төрт оқушыны фамилиясы бойынша ізде.
5. MS Access 2010-да сыныптағы оқушылар туралы кестеден төрт оқушыны туылған күні бойынша ізде.
6. Кітаптар туралы деректі үш кестеде жарат және оларды сұраудың көмегімен бір кестеге біріктірі.
7. Оқушылар туралы деректерді бес кестеде жарат және оларды сұраудың көмегімен бір кестеге біріктірі.
8. Өзің туралы деректерді үш кестеде жарат және оларды бір кестеге біріктірі.

9. MS Access 2010-да сыныптағы оқушылар туралы деректерді іздейтін ДҚ жарат.

10. MS Access 2010-да көп сабақ қалдырған оқушыларды анықтайтын бағдарлама жарат.

11. Берілген екі санның айырмасын баған бойынша есептейтін кесте жарат.

12. Бағандағы сандардың квадратын есептейтін кесте жарат.

13. Бағандағы сандардың синусын есептейтін кесте жарат.

14. Төмендегі амалдарды орында:

а) $3*(5-7+8*6+9)$. ә) $4*(7-5+12*3-4)$.



34-САБАҚ. БАҚЫЛАУ ЖҰМЫСЫ

1-вариант

1. Сыныптағы қыздар туралы дереккөз жарат және ондағы деректер бойынша сұрыптау істерін жүргіз.

2. MS Access 2010-да төмендегі түрлі түстегі кестені жаса:

Код	Фамилиясы	Аты	Информатика	Физика	Математика	Химия	Тарих
1.	Оралбаев	Әсетбек	3	5	4	3	5
2.	Батырова	Умида	4	4	3	4	4
3.	Ахметов	Әзіз	2	3	4	4	2
4.	Кәрімова	Нәзипа	4	3	3	2	5

3. MS Access 2010-да **max** функциясын пайдалануды түсіндіріп бер.

4. MS Access 2010-да мына өрнекті шеш: $5^2 + 6 - 4 * 6 + \frac{9}{4}$.

2-вариант

1. Төрт кесте жарат және оларды өзара байланыстыр.

2. MS Access 2010-да төмендегі түрлі түстегі кестені жарат:

Код	Фамилиясы	Аты	Туылған жылы	Мекенжайы	Телефон нөмірі
1.	Оралбаев	Әсетбек	25.06.2002	Науаи қаласы	678-34-09
2.	Батырова	Умида	6.09.2001	Ташкент қаласы	234-56-01
3.	Ахметов	Әзіз	5.03.2003	Бұхара облысы	234-12-32

4.	Кәрімова	Нәзипа	6.09.2001	Самарқант облысы	345-00-12
----	----------	--------	-----------	---------------------	-----------

3. MS Access 2010-да **min** функциясын пайдалануды түсіндіріп бер.

4. MS Access 2010-да мына өрнекті шеш: $4 - 8 * 5^2 + 9$.

3-вариант

1. Сыныптағы оқушылардың фамилиясының көмегімен іс жүзіндік бағдарлама жарат.

2. MS Access 2010-да төмендегі кестені жарат:

Код	Фамилиясы	Аты	Туылған жылы	Мекенжайы	Телефон нөмірі
1.	Оралбаев	Әсетбек	25.06.2002	Науаи қаласы	678-34-09
2.	Батырова	Умида	6.09.2001	Ташкент қаласы	234-56-01
3.	Ахметов	Әзіз	5.03.2003	Бұхара облысы	234-12-32
4.	Кәрімова	Нәзипа	6.09.2001	Науаи қаласы	345-00-12

3. MS Access 2010-да **tan** функциясын пайдалануды түсіндіріп беріңдер.

4. MS Access 2010-да мына өрнекті шеш: $5^3 + 7 + 4 * 6 - \frac{5}{7}$.

4-вариант

1. Оқушылар туралы деректерді енгізетін форма терезесін жарат.

2. MS Access 2010-да төмендегі кестені жарат:

Код	Фамилиясы	Аты	Информатика	Физика	Математика	Химия	Тарих
1.	Оралбаев	Әсетбек	3	5	4	3	5
2.	Батырова	Умида	4	4	3	4	4
3.	Ахметов	Әзіз	2	3	4	4	2
4.	Кәрімова	Нәзипа	4	3	3	2	5

3. MS Access 2010-да **log** функциясын пайдалануды түсіндіріп беріңдер.

4. MS Access 2010-да мына өрнекті шеш: $2^3 - 6 + 4 * 6 + \frac{7}{4}$.

Borland

Delphi™
Enterprise

III ТАРАУ. DELPHI ОРТАСЫНДА ҚОСЫМШАЛАР ЖАРАТУ

СЕН БҰЛ ТАРАУДЫ ОҚЫП ТӨМЕНДЕГІ БІЛІМ, ДАҒДЫ,
БІЛКІТІЛІК ПЕН КОМПЕТЕНЦИЯҒА ИЕ БОЛАСЫҢ:

- Қосымшалар жаратудың заманалық тәсілдері;
- Delphi 7 ортасының интерфейсі;
- Delphi 7 ортасында қосымша терезесін жарату;
- Қосымшада басқару түймешігін пайдалану;
- Қосымшалардағы **ShowMessage** терезесі;
- Қосымшадағы терезеге деректерді орналастыру;
- Басқару объектілерінің белсенділігі мен көрінбеуі;
- Қосымшаларға деректер енгізу;
- Деректердің түрін өзгерту;
- Қосымшаларда жалаушаларды пайдалану;
- Қосымшадағы радиотүймешіктер тобын пайдалану;
- **ListBox** және **ComboBox** объектілері;
- **Memo** басқару объектісі;
- Delphi 7 ортасында графикамен жұмыс істеу;
- **Timer** объекті және оны пайдалану;
- Суретке басқа объектілерді орналастыру;
- Графикалық және анимациялық мүмкіндіктерді пайдалану.

Borland Delphi 7



35-САБАҚ. ҚОСЫМШАЛАРДЫ ЖАРАТУДЫҢ ЗАМАНАЛЫҚ ТӘСІЛДЕРІ

1. Заманалық қосымшалар. Заманалық компьютер операциялық жүйеге ие екені белгілі. Операциялық жүйе **қосымшалары** іс жүзіндік міндеттерді орындайды. Оларға офистік қосымшалар: **MicroSoft Word** (қысқа түрінде **W** – **MS Word**), **X** – **MS Excel**, **A** – **MS Access**, **P** – **MS Power Point**-тарды мысал ретінде айтуға болады.

Бұл қосымшалардың ерекшелігі интербелсенділігінде болып, олардың көмегімен жүздеген, тіпті мыңдаған амал орындалады. Ал мұндай күрделі қосымшалардың өздері қалай жаратылады екен?

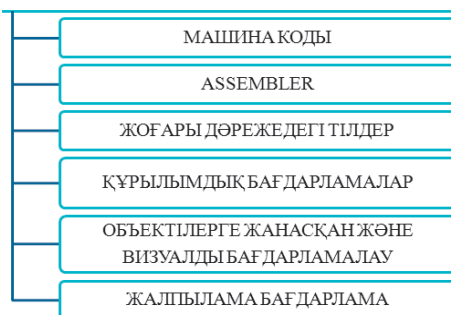
Бүгінде қосымшаларды жарататын бағдарламалық құралдардың мүмкіндіктері өте үлкен. Олармен танысу үшін бағдарламалау тарихына назар аударайық.

2. Бағдарламалаудың даму кезеңдері. Қосымшаларды (іс жүзіндік бағдарламаларды) жарату құралдарының дамуын төмендегі кезеңдерге бөлуге болады:

Алғашқы компьютерлердегі бағдарлама тікелей **микропроцессор командаларының** (машина коды) тізбектілігі көрінісінде жазылған. Бұл жағдай бағдарламалау үшін өте үлкен күш пен уақытты талап еткен, бағдарламадағы қателікті табу қиын болған. Мұны оңайлату мақсатында микропроцессор командасы үшін қысқа атаулар енгізіліп, арнаулы бағдарлама мұндай атауларды машина кодына (микропроцессор командаларына) тәржімалаған. Пайда болған бағдарлама коды тікелей компьютерде орындалған. Мұндай бағдарламалау **Assembler** тілінде бағдарламалау деп аталды.

Компьютерлерді жасау кеңейген соң (үшінші ұрпақ компьютері), оларда **операциялық жүйе** пайда болды. Мұндай компьютерлерді бағдарламалау үшін **жоғары дәрежедегі бағдарламалау тілдері** жаратылды. Енді бағдарламалау машина кодына тікелей байланысты болмайтын болды. Бағдарламалау тілі барған сайын адамдар

БАҒДАРЛАМАЛАУДЫҢ ДАМУ КЕЗЕҢДЕРІ



арасындағы тілдесу сияқты болып бара жатты. Мысалы, бұл тілдерде егер $x > 0$ болса, онда $y = \ln(x)$ сияқты тіркестерді пайдалануға болатын еді. Бағдарлама мәтінін компьютер түсінетін машина кодына тәржималауды арнайы жасалған және **translyator** деп аталатын бағдарлама орындаған. Нәтижеде бағдарламалау біршама жеңілдеп, компьютерде шешілетін мысалдардың көлемі кеңейе түсті.

Келесі кезеңде компьютерде шешілетін мысал бірнеше шағындау және оңайлау мысалдарға бөлінген. Қажет болса оларды да шағындау мысалдарға бөлген. Бұл бір мәселе бойынша бірнеше, кейде ондаған бағдарламашылардың біргелікте жұмыс істеуіне мүмкіндік берді. Бағдарламалық құралдардың, яғни қосымшалардың сапасы одан әрі артты, бағдарламалар күрделеніп, көбірек мүмкіндіктерге ие болды. Мұндай бағдарламалау **құрылымдық бағдарламалау** деп аталды. Біз білетін бағдарламалау тілі – **Turbo Pascal** осындай тілдердің қатарына енеді. Мысалды шағындау мысалдарға бөлу және бағдарламалау процедуралар мен функциялардың көмегімен жүзеге асырылды.

Бағдарламалаудың келесі басқышы **объектілерге жанасқан бағдарламалау** деп аталады. Бұл кезең жекеменшік компьютерлердің кең таралуымен және оларда жұмыс істеуді одан әрі оңай да қолайлы ету мақсатында жаратылған графиктік операциялық жүйелермен (қазірде кең таралған Windows те солардың қатарынан) байланысты.

Біз ақпаратты қайта өңдеу тәсілін белгілі дерек деп атауға болатынын білеміз. Әрқандай ақпаратқа объект деп қарастыруға болады. Turbo Pascal-да деректер өзгертін және өзгермейтін болып бөлінеді. Оларды қайта өңдеу тәсілдері (қайта өңдеу алгоритмдері) ерекше процедуралар және функциялар көрінісінде болады.

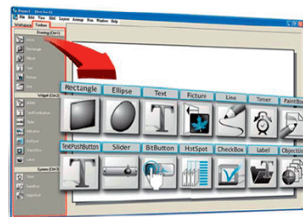
Объектілерге арналған бағдарламалауда бұрын біріктірілмеген осы екі элементті біріктіру мүмкіндігі пайда болды. Арасындағы айырмашылық жойыла бастады. Ақпаратпен жұмыс істеу қолайлы және оңай болып қалды. Бір бағдарлама бойынша ондаған, жүздеген, тіпті мыңдаған бағдарламашылар бірге істейтін болды. Ал қолданбалы бағдарламалық құралдардың мүмкіндіктері шұғыл артты.

Жаратылып жатқан бағдарламалық құралдың өзін де ақпарат деп қарастыруға болады. Демек, бағдарлама кодын жаратуда объектілерді

пайдалануға болады. Мысалы, әрбір бағдарламаның өз терезесі бар. Бағдарлама терезесінің биіктігі мен ені сияқты қасиеттері (терезе объекті қасиеттері) бар, терезені жарату, жабу, орнын және өлшемдерін өзгерту сияқты қайта өңдеу тәсілдері (терезе объекті тәсілдері) арқылы бағдарлама терезесімен істеуді оңайлатуға болады.

Терезе деген объект жаратылған соң, бағдарлама терезесімен істеу осы объектінің қасиеттерін қажетті түрде орнату және объект тәсілдерін қажетті жерге пайдалану түрінде өте оңай орындалды. Нәтижеде бағдарламалау одан әрі оңай болып, бағдарламалаудың сапасы тағы да артты.

Енді қосымша жарату үшін басқалардың жаратқан дайын объектілерін бағдарламаға енгізу және олардың қасиеттерін қажеттілігіне орай орнату жеткілікті болды. Мұндай бағдарламалау **визуалды бағдарламалау** деп аталды және ол Lego-мен ойыншық жасау сияқты болып қалды.



Қазіргі уақытта **жалпылама бағдарламалау** тәсілі кең таралуда. Оның мағынасын төмендегідей түсінуге болады. Бағдарламада үшбұрыш, төртбұрыш, көпбұрыш, шеңбер сияқты объектілер және олардың ауданы деген қасиеттері болсын. Олардың әрбірін пайдаланудың орнына геометриялық пішіннің ауданы деген қасиетін пайдалануға болады. Бағдарламаның өзі әңгіме қайсы геометриялық пішін жайлы екенін қажетті объектінің қажетті қасиетін пайдаланып анықтайды. Нәтижеде бағдарламалау тағы да оңай әрі арзан болды. Енді бағдарламаны қалаған операциялық жүйе үшін және оның интерфейсін қажетті тілде жарату мүмкіндігі туды. Оларды өзгертуге көп уақыт пен күш талап етілмейтін болды.

3. Заманалық бағдарламалау орталары. Қазіргі уақытта әлемде он миллионнан астам бағдарламашылар бар, олардың екі миллионы кәсіби, қалғаны әуесқой бағдарламашылар. Әрине олардың жұмыс құралдары, яғни пайдаланатын бағдарламалау орталары да әртүрлі.

Бүгінде кең таралған бағдарламалау орталарында негізінен үш бағдарламалау тілі: **Si**, **BASIC**, **Pascal** көбірек қолданылады. Кәсіби және тәжірибелі әуесқой бағдарламашылар **C** (Си) мен кейінгі нұсқасы

C++ және C#-ті пайдаланса, пайдаланушылардың басым көпшілігі тек өзінің мәселесін шешетін бағдарламалауға Pascal-ды қолданады. BASIC (анығында, Visual BASIC) Microsoft компаниясы өнімі офистік қосымшалардағы, басқа да компаниялардың бірқатар өнімдеріндегі, графиктік редакторлардағы істерді автоматтандыруға қолданылады.

Күшті бәсекелестік және пайдаланушыларды тарту үшін қазіргі уақыттағы кең таралған бағдарламалау орталары бірнеше бағдарламалау тілдерін пайдалану мүмкіндігін беруде. Сондай-ақ соңғы кезде жаратылған бағдарламалау тілдерінің көбісі жоғарыдағы тілдердің бірін пайдаланып жатқандарға арнап жаратылған. Мысалы, кең таралған веб бағдарламалау тілдері Java, Java Script, ASP, PHP, Python-дар C++ мен C#-ке, кең таралған компьютер математикасы пакеттері MatLab, MathCAD, Maple-дердің бағдарламалау тілдері Pascal-ға, MacroMedia Flash атты анимациялық графика жарататын қосымшаның бағдарламалау тілі Action Script Visual BASIC-ке ұқсас.

4. Delphi бағдарламалау ортасы. Delphi (Делфи) ежелгі дәуірдегі атақты әулие жасаған ғибдатхана орналасқан грек қалашығының аты болып, сол әулие сияқты бағдарламалау ортасының атақ-даңқы да көпшіліктің арасына кең таралды. Оның дерлік ширек ғасырда 25 нұсқасы жаратылып, олар үш топқа бөлінеді:

Delphi 1 – Delphi 8 (1995-2003 ж.ж.);

Delphi 2005 – Delphi 2010 (2005-2009 ж.ж.);

Delphi XE 1 – Delphi XE 10 (2010-2017 ж.ж.).

Олардың бәрінде негізгі (кейде жалғыз) бағдарламалау тілі Object Pascal болғанымен, көпшілігінде C++, Assembler-де, соңғы нұсқаларында Java-да, кейбір версияларында PHP-де де бағдарлама жазуға болады. Соңғы версиялары Windows, Android, IOS қосымшаларын жарата алады.



ЕСІНДЕ САҚТА !

1. Визуалды бағдарламалау ең заманалық бағдарламалау тәсілі.
2. Delphi-ді әлемдегі бірнеше миллион бағдарламашы пайдаланады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Бағдарламалаудың даму кезеңдерін санап өт.
2. Заманалық бағдарламалау орталары туралы не білесің?
3. **Turbo Pascal**-да а) мән беру операторы; ә) шартты оператор қатысқан бағдарламалар түзіндер.
4. **Turbo Pascal**-да а) тармақталатын оператор; ә) циклды оператор қатысқан бағдарламалар түзіндер.



ҮЙГЕ ТАПСЫРМА

1. **Delphi** бағдарламалау ортасының неше нұсқасы жасалған және олар неше топқа бөлінеді?



36-САБАҚ. DELPHI БАҒДАРЛАМАЛАУ ОРТАСЫ

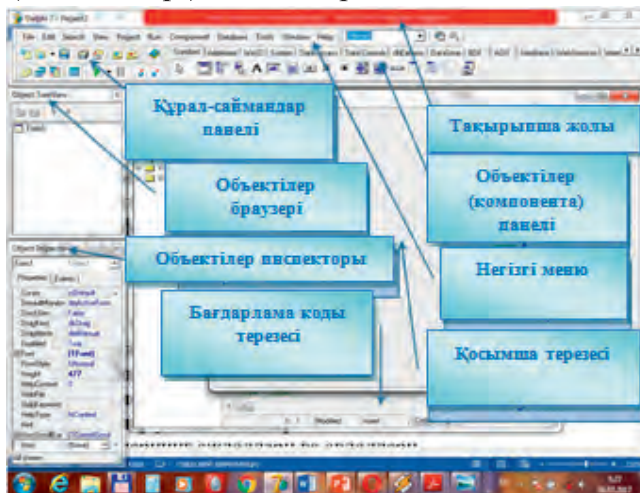
1. **Delphi** интерфейсін төмендегі бөлімдер құрайды:

1. Тақырыпша жолы.
2. Негізгі меню.
3. Құрал-саймандар панелі.
4. Басқару объектілері (компоненталар) панелі.
5. Басқару объектілері браузері.
6. Басқару объектілері (элементтері) инспекторы.
7. Жаратылатын

қосымша терезесі
(форма).

8. Жаратылатын
қосымша коды терезесі.

Delphi ортасы басты терезесінің тақырыпша жолы стандарт көрініске ие болып, онда қосымшаның аты, онда жаратылған жоба аты және терезені жасыру, өлшемін өзгерту және



оны жабу үшін қызмет ететін үш түймешік орналасқан.

2. Delphi-дің негізгі менюсінде 11 бөлім бар:

- | | |
|-------------------------------|--|
| 1. File (Файл). | 7. Component (Басқару объектілері). |
| 2. Edit (Редакциялау). | 8. Database (Дерекқор). |
| 3. Search (Іздеу). | 9. Tools (Құрал-саймандар). |
| 4. View (Көрініс). | 10. Window (Терезе). |
| 5. Project (Жоба). | 11. Help (Көмек). |
| 6. Run (Іске қосу). | |

3. Delphi құрал-саймандар панелі мен объектілер (компоненттер) панелі. Құрал-саймандар панелінде көп істетілетін командалардың түймешіктері орналасқан болып, бұл оларды негізгі менюге кірмей-ақ пайдалану мүмкіндігін береді. Құрал-саймандар панелінде оннан астам түймешіктер бар, оларға жаңаларын қосуға, кейбіреулерін алып тастауға болады.

Компонента деп жаратылатын қосымшаға орналастыруға арналған түрлі басқару элементтеріне (объектілеріне) айтылады. Визуалды бағдарламалау осы басқару элементтерін қосымшаның терезесіне орналастыру, оның қасиеттерін өзгерту арқылы жүзеге асырылады. **Delphi**-де жүздеген басқару элементі бар, олар компоненталар панеліндегі жиырмадан астам топтамаларға орналастырылған.

4. Delphi терезелері. Delphi-дің бас терезесінен басқа да терезелері бар, олардың бас терезенің ішіндегі орналасуын өзгертуге, оларды жасыруға, экранға шығаруға, өлшемдерін өзгертуге болады. Бұл терезелердің біріншісі **форма терезесі** деп аталады және онда жаратылатын қосымшаның негізгі терезесінің көрінісі бейнеленеді. Бұл терезенің өлшемін өзгерту арқылы жаратылатын қосымша терезесінің өлшемін өзгертуге болады. Оған компоненталар панелінен қажетті компоненталар тышқанның көмегімен тартып өткізіледі және олардың өзара орналасуы жоспарланады. Бұл терезені жаңа қосымшаның терезесі немесе қарапайым түрде **қосымша терезесі** деп атаймыз. Оның өлшемдерін өзгертуге, оған жаңа басқару элементтерін орналастыруға және осы элементтердің орналасуын өзгертуге болады.

Екінші маңызды терезе **бағдарлама коды** (бағдарлама мәтіні) терезесінде қосымша терезесіндегі өзгерістер көрсетіледі. Мысалы, біз

қосымша терезесінің өлшемін өзгертсек, бағдарламаның мұны орындайтын операторы оның мәтініне қосылады. Қосымша терезесіне жаңа басқару элементтері орналасса, бұл элементтерді сипаттайтын операторлар автоматты түрде бағдарлама мәтініне қосылады.

Бағдарлама коды терезесінде бағдарламашы істейтін негізгі жұмыс – басқару элементтерін оқиғаларға байланыстырғанда элементтердің осы оқиға болғанда бағдарлама оған қалай жауап қайтаруы қажет екенін көрсетіп беру. Мұнда әрбір оқиға үшін бір процедура жаратылады және осы процедура мәтінін бағдарламашы жазады.

Визуалды бағдарламалауда бағдарламашының негізгі міндеті – бағдарламада пайдаланылатын басқару элементтерінің, яғни объектілердің қажетті қасиеті мен тәсілін пайдалану. Мұнда жүзден астам объектінің ондаған қасиеті мен тәсілінің қажеттісін жылдам табуы маңызды. Бұл істе **Объект инспекторы**ның орыны өте маңызды.

Қосымша терезесінде қандай да бір объект таңдалғанда осы объектінің барлық қасиеттері объект инспекторында бейнеленеді. Қажетті қасиеттің аты мен мәнін есте сақтаудың, іздеудің қажеті жоқ.

Ірі және күрделі қосымшаларды жаратуда онда пайдаланылатын объектілердің саны шұғыл артып кетеді. Қосымшадағы объектілердің қажеттісін табудың оңай жолы **Object TreeView** (Объектілердің ағаш сияқты тізімі) деп аталатын арнаулы терезені пайдалану. Бұл терезе **Объектілер браузері** деп те аталады. Объектілердің бірі екіншісін қамтыған болуы мүмкін. Сондықтан объектілердің ағаш сияқты (шақтаған) тізімі ең қолайлы тізім болып табылады.

Delphi-де тағы бірнеше негізгі және көптеген тілдесу терезелері бар, олармен кейінірек, қажеттілік туындағанда танысамыз.

5. Delphi-де жобаларды жарату мен сақтау. **Delphi**-де жаратылатын қосымшалар **проекттер** (жобалар) деп аталады және арнаулы **Projects** папкасында сақталады. Олар **Pascal**-да жаратылған қосымшалардан күрделірек, оларды сақтауға бірнеше файл қажет.

Pascal-да жаратылған бағдарламаны сақтауға бір файл жеткілікті және онда жаратылған барлық бағдарламаларды бір папкада сақтауға болады. **Delphi**-де олай етуге болмайды, өйткені түрлі жобалардың аты бірдей файлдарының бірінің орнына екіншісін сақтап қояды. Нәтижеде

жобалар қате істеп, арасында қайшылықтар пайда болады. Сондықтан жаңа жобамен істеуден бұрын осы жоба үшін жаңа папка ашу қажет.

Delphi-де бар жоба негізінде жаңа жоба жаратуда да ықтият болу керек. Жобаға өзгерту енгізілуімен бұл өзгерістер жоба файлдарында автоматты түрде сақталады және бұрынғы жобаны қайта тіктеуге болмайды. Сондықтан бар жоба негізінде жаңа жоба жаратудан алдын бар жобаны жаңа атты басқа папкаға көшіру керек.

Delphi жобаларын қосымшалар деп атауға келісіп алайық.



ЕСІНДЕ САҚТА !

Delphi-де жүздеген дайын басқару объектілері бар.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. **Delphi**-дің компоненталар панелінде нелер орналасқан?
2. **Delphi**-дің қосымша терезесінің міндеті не?
3. **Delphi** қосымшалары **Pascal**-дағы бағдарламалаудан айырмашылығы неде?
4. **Turbo Pascal**-да стандартты функция мен процедура қатысқан екі бағдарлама жаса.
5. **Turbo Pascal**-да жаңа функция мен процедура қатысқан екі бағдарлама жаса.



ҮЙГЕ ТАПСЫРМА

1. **Delphi**-дің интерфейсінің бөлімдерін жазып ал.
2. **Delphi**-дің негізгі менюсінің бөлімдерін жазып ал.



37-САБАҚ. ҚОСЫМША ЖӘНЕ ОНЫҢ ТЕРЕЗЕСІ

1. **Delphi**-ді іске қосу және онда қосымша жарату. **Delphi**-ді іске қосу үшін **Windows**-тың жұмыс столындағы **Пуск** (Басқару) түймешігіне тышқанның нұсқағышын әкеліп басамыз. Пайда болған менюдегі **Все программы** (Барлық бағдарлама) тармағын таңдаймыз. Пайда болған менюдегі **Borland Delphi 7** тармағы белсенділендіріледі.

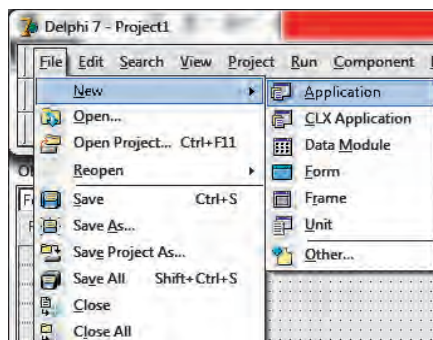
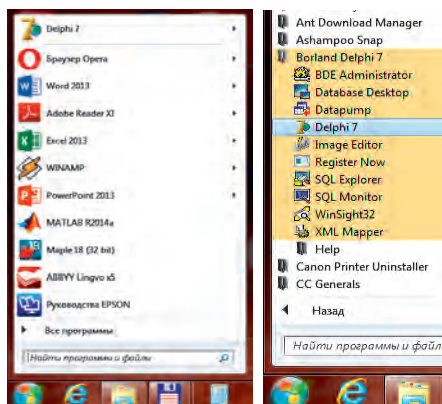
Сосын тізімде Delphi-ге қатысты жаңа тармақтар пайда болады. Олардан Delphi 7-ні таңдаймыз. Экранда Delphi бағдарламалау ортасының терезелері пайда болады және жаңа қосымша жаратылады.

Егер менюдің Delphi 7 тармағына тышқанның оң түймешігін басып контекстік менюді пайда етсек, онда **Закрепить в меню Пуск** (Бастау менюсіне орналастыру) немесе **Закрепить на панели задач** (Амалдар панеліне орналастыру) тармақтарын таңдасақ, Delphi 7-нің таңбашасы пайда болады және оны іске қосу оңайға соғады. Егер менюдің Delphi 7 тармағын тышқанмен тартып, жұмыс столының басты жеріне әкеліп, тышқанның түймешігін босатсақ, таңбаша жұмыс столында пайда болады.

Delphi-де жаңа қосымша жарату үшін оның негізгі менюсінің **File** (Файл) бөлімінің **New** (Жаңа) тармағын және пайда болған жаңа менюдің **Application** (Қосымша) тармағын таңдаймыз. Егер алдыңғы қосымша сақталмаған болса, Delphi осы туралы ескертеді және оны сақтауды ұсынады. Мұнда ескі қосымшаны сақтап қоюға немесе оны сақтамастан жаңа қосымшаға өтуге болады.

2. Delphi-де қосымшаны сақтау және жүктеу. Delphi-де қосымшаны сақтаудың бірнеше тәсілі бар. Бұл командалардың барлығы негізгі менюдің **File** (файл) бөлімінде. Олар:

1. **Save**
2. **Save as ...**
3. **Save Project as ...**
4. **Save All.**



Олардың біріншісі **Save** (Сақтау) командасы қосымшаны ағымдағы атымен сақтайды. Екіншісі жаңа атпен сақтайды. Үшіншісі

қосымшаны жаңа жерге, ал төртіншісі қосымшаның барлық файлдарын сақтауға қызмет етеді.

3. Қосымша терезесі мен оның қасиеттерін орнату. Қосымша терезесінің негізгі қасиеттері – оның өлшемі мен компьютердегі орны. Қасиеттерді **Delphi**-дің қосымша терезесінің өлшемін тышқанның көмегімен оның төменгі оң жақ бұрышын жылжытып өзгертуге болады. Бірақ мұның басқаша жолы да бар.

Тышқанды қосымша терезесінің (**Form1**) үстіне әкеліп оның сол түймешігін басамыз және Объект инспекторы (**Object Inspector**) терезесінде қажетті қасиеттердің мәнін өзгертеміз. Бұл терезеде 50-ден астам қасиеттер әліппе тәртібімен келтірілген. Қажетті қасиет осы терезеде көрінбесе, оң жақтағы вертикаль жылжығышты тышқанмен жылжытып, оны экранға шығару мүмкін.

Қосымша терезесінің кеңдігін **Width** қасиетіне 400-ді енгізіп өзгертеміз. Ол үшін **Width** қасиетін тышқанмен таңдап, 400 санын енгіземіз. Сол тәрізді, қосымша терезесінің биіктігін **Height** қасиетінің көмегімен 300-ге тең етіп аламыз. Осы амалдарды қысқаша төмендегі көріністе бейнелейміз, оны кейін кеңінен пайдаланамыз:

```
Form1.Width:=400;
```

```
Form1.Height:=300;
```

Қосымша терезесінің тақырыпшасын алдын ала берілген **Form1**-ден **Менің бірінші қосымшам**-ға өзгерту үшін

```
Form1.Caption:= 'Менің бірінші қосымшам';
```

-ды орындаймыз.

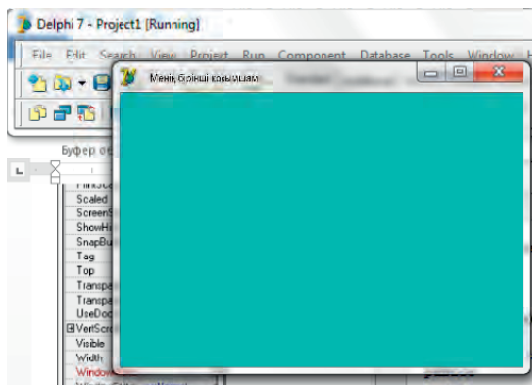
```
Form1.Color:=
```

'ClAqua' көмегімен қосымша терезесінің фонын өзгертеміз.

```
Form1.Left:=100 және
```

```
Form1.Top:=50
```

қасиеттерінің көмегімен қосымша терезесінің сол жағынан және жоғарысынан қалатын жерді 100 және 50-ге тең етіп орнатамыз.



Орындалған өзгертулерді көру үшін қосымшаны іске қосамыз. Ол үшін үш түрлі тәсіл бар.

1. Негізгі менюдің **Run** (Іске қосу) бөлімінің **Run** тармағын таңдау.
2. Құрал-саймандар панеліндегі  түймешігін басу.
3. Пернетақтадағы **F9** түймешігін басу.

Қосымшамен жұмыс істеуді аяқтау үшін оның жоғарғы оң жақ бұрышындағы **×** түймешігін басамыз.



ЕСІНДЕ САҚТА !

Form1 – жаратылатын қосымша терезесі болып, оның 50-ден астам қасиеті бар.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. **Delphi** қосымшасы терезесінің негізгі қасиеттерін атап өт.
2. **Delphi** қосымшасы қандай тәсілдермен іске қосылады?
3. **Delphi** қосымшасының тақырыпшасына өзіңнің атыңды жаз.
4. **Delphi**-де өлшемі 600-ге 800 болған қосымша терезесін жарат.
5. **Delphi**-де фонның түсі сары болған қосымша терезесін жарат.



ҮЙГЕ ТАПСЫРМА

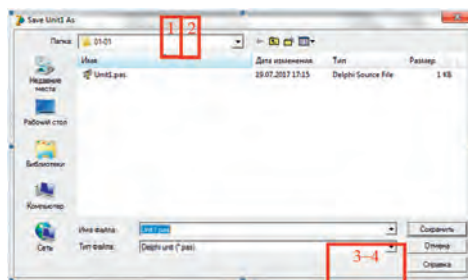
1. **Delphi**-де жаңа қосымша қалай жаратылады?
2. **Delphi**-де қосымшаны сақтаудың неше түрлі тәсілі бар?



38-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

1. **Delphi**-ді іске қос.

1.1. Жаңа қосымша жаратып, оның терезесінің тақырыпшасын **Caption** қасиетінің көмегімен **01_02**-ге өзгерт. Қосымшаны негізгі менюдің **File** (Файл) бөліміндегі **Save** (сақтау) тармағын таңдап сақта. Қосымша бірінші рет сақталғаны үшін экранда сақтау тілдесу терезесі пайда болады.



1.1.1. Тілдесу терезесіндегі ағымдағы папка ретінде бұрынғы қосымшаның **01_01** папкасы көрсетілген. Суретте көрсетілген 1-түймешікті басып, одан шығып кетеміз.

1.1.2. Қосымша үшін жаңа папка жарату үшін суреттегі 2-түймешікті басамыз.

1.1.3. Ағымдағы папкада жаңа **Новая папка** деп аталған папка пайда болып, ол ерекшеленіп тұрады. Оны **01_02** деп атап, суреттегі **Сохранить** (сақтау) түймешігінің орнында пайда болған **Открыть** (ашу) түймешігін басамыз.

1.1.4. Жаңа папкада ұсынылған атпен қосымшаны сақтау үшін **Сохранить** (сақтау) түймешігін басамыз.

1.2. Қосымшаны іске қосып жұмыс істеуін тексереміз және қосымша терезесін жабамыз.

1.3. Негізгі менюдің файл бөліміндегі **Close all** (барлығын жабу) тармағын тандап, қосымшаны жабамыз.

2. Алдыңғы сабақта берілген жаттығуларды компьютерде орында.

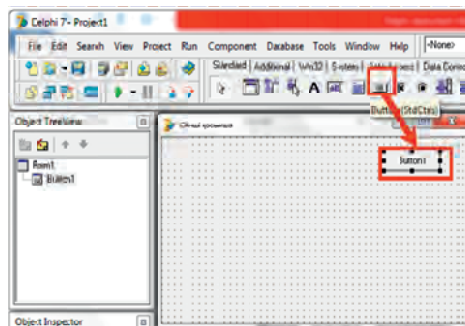


39-САБАҚ. БАСҚАРУ ТҮЙМЕШІГІ

1. Қосымша терезесіне басқару түймешігін орнату. Алдыңғы сабақтағы бірінші қосымшамыз ешқандай міндет атқармайды. Енді осы кемшілікті жоямыз.

Жаңа қосымша жаратамыз және оның өлшемін 255-ке 400 етіп орнатамыз. Ал тақырыпшаны **екінші қосымша** деп өзгертеміз. Қосымша терезесіне басқару объекті: басқару түймешігін орнатамыз.

Құрал-саймандар панелінің стандартты топтамасындағы басқару түймешігін тышқанмен тандап, соң тышқанды қосымша терезесі **Form1**-дің басқару түймешігі тұруы қажет болған жерге әкеліп тағы бір рет басамыз. Егер оның орнын өзгерту керек болса, оны тышқанмен жылжытып басқа жерге өзгертеміз.

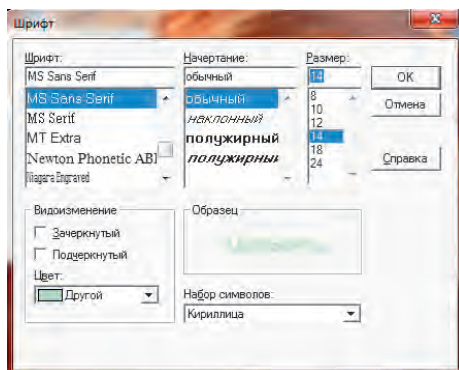


2. Басқару түймешігінің қасиеттерін орнату. Қосымша терезесінің қасиеттері болғанындай оған орнатылған басқару түймешігінің де өзіндік қасиеттері бар. Оларды да өзгертуге болады. Екі объектінің көп қасиеті бірдей міндетті атқарғаны істі оңайлатады.

Алдымен басқару түймешігі: **Button1**-ді оның үстіне тышқанның сол түймешігін басып таңдаймыз. Объектілер диспетчерінде (**Object TreeView**) **Button1** ерекшеленіп тұрады және Объекті инспекторы (**Object Inspector**) терезесінде **Button1** объектінің қасиеттерінің тізімі пайда болады. Оған алдымен **Caption** қасиетінде түймешіктің тақырыпшасын ‘++-ке’ өзгертеміз. Бұл төмендегідей жазылады:

```
Button1.Caption:='++'
```

Түймешік тақырыпшасы біршама кіші болғаны үшін оны үлкейтеміз. Ол үшін **Font** қасиетінен пайдаланамыз. Оны таңдағанда **Шрифт** тілдесу терезесі пайда болады. Оның **Размер** алаңының астындағы тізімнен **14**-ті таңдап, **OK** түймешігін басамыз:



3. Басқару түймешігіне оқиғаларды байланыстыру. **Button1** түймешігі қосымшада қандай да бір міндетті орындауы қажет. Оны басқанда қосымша өз терезесінің өлшемін 16 мен 9-ға үлкейтеді.

Тапсырманы орындау осы түймешікті екі рет басумен басталады. Бағдарлама коды терезесі пайда болып, оның меңзер тұрған жеріне

```
Form1.Width:=Form1.Width+16;  
Form1.Height:=Form1.Height+9;
```

жолын енгіземіз. **F9** түймешігін басып, қосымшаны іске қосамыз және оның дұрыс жұмыс істеуін тексеріп көреміз.

ЕСІНДЕ САҚТА !

Қосымшадағы **Button1** басқару түймешігі басылғанда **TForm1.Button1Click** процедурасы іске қосылады және ондағы операторлар орындалады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Басқару түймешігі қосымша терезесіне қалай өткізіледі?
2. Басқару түймешігінің қасиеттері қай жерде өзгертіледі?
3. Басқару түймешігі таңдалғанда оның не істейтіні қайда көрсетіледі?
4. **Delphi**-де жаңа қосымша жарат. Оған төрт басқару түймешігін енгіз. Олардың біріншісін **Жарат**, екіншісін **Жүкте**, үшіншісін **Сақта**, төртіншісін **Аяқта** деген жазумен экранға шығар.



ҮЙГЕ ТАПСЫРМА

1. Басқару түймешігінің негізгі қасиеттерін көшіріп жаз.



40-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

1. **Delphi**-де жаңа қосымша жарат және оны **01_03** папкасына сақта.
 - 1.1. Қосымшаның тақырыпшасын **сәлем, әлем!** деп өзгерт.
 - 1.2. Қосымша терезесіне 3 басқару түймешігін бір бағанға орнат.
 - 1.3. Бірінші түймешікті екі рет бас. Экранда пайда болған оның **TForm1.Button1Click** процедурасына **begin**-нен кейін **TForm1.Caption:='salom, '+TForm1.Caption;**-дарды енгіз.
 - 1.4. Екінші түймешікті екі рет басып, оның процедурасын төмендегідей көрініске келтір:


```
procedure TForm1.Button2Click(Sender: TObject);
begin
  TForm1.Left:= TForm1.Left+10; end;
```
 - 1.5. Үшінші түймешікті екі рет басып, оның процедурасын төмендегідей көрініске келтір:


```
procedure TForm1.Button3Click(Sender: TObject);
begin
  TForm1.Top:= TForm1.Top+10; end;
```
 - 1.6. Қосымшаны іске қосып, түймешіктер қандай міндет орындауын бақыла. Қосымша терезесін жауып, оның барлық файлдарын сақтап қой. Қосымшаның барлық файлдарын жап.
2. Алдыңғы сабақта берілген жаттығуды компьютерде орында.



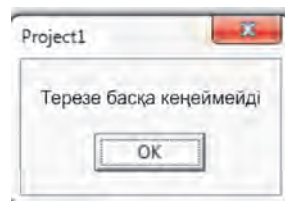
41-САБАҚ. SHOWMESSAGE ТЕРЕЗЕСІ

1. Екінші қосымшаның кемшілігі. Екінші қосымшаны іске қосып, ондағы “++” түймешігін мың рет бассақ, қосымша экранының ені 16000 пикселден артып кетеді. Бұл бағдарламашы үшін үлкен қателік. Оны жою үшін қосымша терезесінің ені 600 пикселден артса, басқа үлкейтпеуге келісіп аламыз. Бірақ пайдаланушы түймешікті басса да, онда ешқандай өзгеріс болмаса, бағдарламаның дұрыс істеуіне күмән тууы мүмкін.

2. ShowMessage процедурасы. Мұның алдын алу үшін қосымша өзі терезесінің ені 600-ден артып кеткені туралы экранда ерекше терезеде хабарландырсын.

Бұл істі **ShowMessage** арқылы орындаймыз. Екінші қосымшадағы **Button1** түймешігін екі рет басып, пайда болған процедурадағы бұрынғы екі операторды төмендегідей қайта жазып шығамыз:

```
If Form1.Width<600 then begin  
Form1.Width:=Form1.Width+16;  
Form1.Height:=Form1.Height+9; End else  
ShowMessage('Терезе басқа кеңеймейді');
```



Қосымшаны іске қосып, оның түймешігін бірнеше рет бассақ, экранға төмендегі жазу шығады. Ондағы **OK** түймешігін баспағанымызға дейін, қосымша өз жұмысын тоқтатып тұрады.

3. IntToStr функциясы. Қосымшаның жұмысын одан әрі жақсартамыз. Қосымша өз терезесінің өлшемдерін өзгерткенде олар қандай болғаны бізге белгісіз болып қалады. Оны қосымша терезесіне шығарудың мүмкіндігі бар ма? Қазірге дейін біз мұның бір ғана жолын білеміз: терезенің өлшемдерін терезе тақырыпшасына шығару. Ол үшін

```
Form1.Caption:=Form1.Height+'*'+Form1.Width;
```

операторын пайдаланамыз. Бірақ бұл операторды пайдалануға болмайды, өйткені онда бір түрдегі айнымалыға екінші түрдегі айнымалыны енгізуге әрекет болуда. **Delphi**-де бұл қателікке соқтырады.

Бүтін сан түріндегі шаманы мәтін жолы түріндегі шамаға өткізу үшін **IntToStr** (**integer to string** – бүтіннен жолға тіркесінен алынған) функциясын пайдаланамыз:

```
Form1.Caption:=' ('+IntToStr(Form1.Height)+'*'+IntToStr(Form1.Width)+' )';
```

Оған қосымшаның шын атын да қосып қоямыз:

```
Form1.Caption:='Үшінші қосымша'+ Form1.Caption;
```

Button1 түймешігін екі рет басып, бағдарлама коды терезесін шығарамыз және соңғы екі жолды оған қосып қоямыз:

```
If Form1.Width<600 then begin  
Form1.Width:=Form1.Width+16;  
Form1.Height:=Form1.Height+9; End else  
ShowMessage('Терезе басқа кеңеймейді');  
Form1.Caption:=' ('+IntToStr(Form1.Height)+'*'+IntToStr(Form1.Width)+' )';  
Form1.Caption:='Үшінші қосымша ' + Form1.Caption;
```

Қосымшаны іске қосып, оның тақырыпшасының өзгеруін көреміз:



ЕСІНДЕ САҚТА !

ShowMessage жаңа тілдесу терезесін жаратады және ол жабылмағанға дейін қосымша өз жұмысын тоқтатады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Екінші қосымшаның кемшілігі неде?
2. **ShowMessage** қандай міндетті атқарады?
3. **IntToStr** функциясы қандай міндетті атқарады?
4. Үшінші қосымшаны сондай өзгерт, ол а) қосымша терезесінің ені 800-ден, ә) қосымша терезесінің биіктігі 450-ден артса экранда ескертуші жазу пайда болсын.



ҮЙГЕ ТАПСЫРМА

1. Қосымша терезесінің негізгі қасиеттерін есіңе түсір және дәптеріңе көшіріп жазып ал.



42-САБАҚ. БАҚЫЛАУ ЖҰМЫСЫ

Төмендегі бақылау жұмысы варианттары берілген:

1-вариант

1. Машина кодымен және assembler-де бағдарламалау.
2. **Delphi**-дің негізгі менюсі.
3. Объектілердің **Caption** қасиеті.

2-вариант

1. Жоғары дәрежедегі бағдарламалау тілдері.
2. **Delphi**-дің интерфейсі.
3. Объектілердің **Left** және **Top** қасиеттері.

3-вариант

1. Визуалды бағдарламалау.
2. **Delphi**-дің терезелері.
3. Объектілердің **Width** және **Height** қасиеттері.

4-вариант

1. Жалпылама бағдарламалау.
2. **Delphi**-де қосымша жарату және оны сақтау.
3. **ShowMessage** процедурасы.



43-САБАҚ. ҚОСЫМША ТЕРЕЗЕСІНЕ ДЕРЕКТІ ОРНАЛАСТЫРУ

1. **Label басқару объектісі және оның қасиеттері.** Әрқандай қосымша қандай да бір деректі пайдаланушыға қосымша терезесіне немесе басқа тәсілмен шығарып беруі керек. Ол үшін бірқатар басқару объектілері жаратылған болып, олардың ең қарапайымы **Label** (нысана, белгі, жазу) объекті. Бұл объект объектілер панелінің Стандарт бөлімінде бесінші болып орналасқан.

Осы объект қосымша терезесінің объекті **Form**-ның “атасы” болғаны үшін оның көп қасиетін мұра етіп алған. Мысалы, **Label** объектінің **Width** (кеңдік), **Height** (биіктік), **Left** (сол), **Top** (жоғары)

қасиеттері объект иелеген жердің кеңдігін, биіктігін, одан терезенің сол және жоғары шекарасына дейінгі арақашықтықты анықтайды.

Label-дің негізгі қасиеті **Caption** (Тақырыпша). Бұл объект экранға қандай мәтін жолының шығуын анықтайды.

2. Label басқару объектінің қасиеттерін бағдарламада өзгерту.

Label объектінің де басқа объектілер сияқты өзінің атауы бар, ол **Name** атты қасиетте сақталады. Оны объектімен және оның қасиеттерімен бірге пайдаланамыз. Сондықтан оны қосымша жұмыс істегенде өзгерту проблема туындатуы мүмкін. Объектілердің **Name** қасиеті тек осы объектілер қосымша терезесіне қойылған кезде өзгертіледі.

Объектілер қосымшаға қойылуымен (мұны объектінің көшірмесін жарату деп те айтады) оған автоматты түрде атау беріледі. Мысалы, **Label** объектінің көшірмелеріне кезек-кезегімен **Label1**, **Label2**, **Label3** атаулары беріледі. Бұрынғы қосымшалардағы **Form1**, **Button1** атты объектілерді есіңе түсір.

Объектінің **Name** қасиетінің мәні басқа қасиеттерді пайдалану үшін қос атауларды жаратуда қолданылады: объект_атауы.қасиет_атауы. Мысалы, **Form1.Width**, **Button1.Caption**, **Label1.Left**. Осы ретте проблемаға кездеспеу үшін қазірше тек автоматты түрде берілген атауларды пайдаланамыз.

3. Терезе өлшемдері қосымшасы.

Алдыңғы сабақтағы қосымшаны қайта жаратамыз. Қосымша терезесі өлшемдерін үлкейту және кішірейту үшін екі **Button**-ды, терезе өлшемдерін көрсету үшін екі **Label**-ді пайдаланамыз.

Алдымен қосымша терезесінің қасиеттерін үйренеміз

```
Form1.Caption:='Төртінші қосымша';  
Form1.Height:=225; Form1.Width:=400;  
Form1.Left:=200; Form1.Top:=100;
```

Жоғарыдағы төрт объектіні қосымша терезесіне жайғастырамыз және олардың қасиеттерін орнатамыз:

```
Button1.Caption:='++';  
Button2.Caption:='--';  
Label1.Caption:='Терезенің биіктігі: ';  
Label2.Caption:='Терезенің ені: ';
```

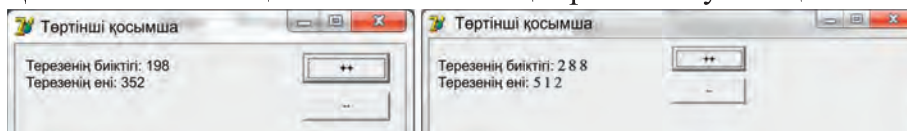

Қосымша терезесіндегі **Button1**-ді екі рет басып, экранда пайда болған бағдарлама коды терезесінің меңзер тұрған жеріне төмендегі операторларды енгіземіз:

```
Form1.Width:=Form1.Width+16;  
Form1.Height:=Form1.Height+9;  
Label1.Caption:='Терезенің биіктігі:'+IntToStr(Form1.Height);  
Label2.Caption:=' Терезенің ені:'+IntToStr(Form1.Width);
```

Қосымша терезесіндегі **Button1**-ді екі рет басып, экранда пайда болған бағдарлама коды терезесінің меңзер тұрған жеріне төмендегі операторларды енгіземіз:

```
Form1.Width:=Form1.Width+16;  
Form1.Height:=Form1.Height+9;  
Label1.Caption:='Терезенің биіктігі:'+IntToStr(Form1.Height);  
Label2.Caption:='Терезенің ені:'+IntToStr(Form1.Width);
```

Қосымшаны іске қосамыз және оның жұмыс істеуін бақылаймыз:



ЕСІНДЕ САҚТА !

Label объекті қосымша терезесіне мәтін жолын енгізу үшін қолданылады және осы мәтін оның **Caption** қасиетінде сақталады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. **Label** басқару объекті қасиеттері қосымшада қалай өзгертіледі?
3. Төртінші қосымшаның биіктігі мен енін сәйкесінше 5 және 8 кадаммен өзгерт.
3. Төртінші қосымшаның терезесі енінің мәні 500-ден 800-ге дейін болсын.
4. Төртінші қосымшаның түймешіктеріндегі жазуларды **a->A** мен **A->a**-ға өзгерт.



ҮЙГЕ ТАПСЫРМА

1. **Label** басқару объектінің міндетін түсіндір.
2. **Label** басқару объектінің негізгі қасиеттерін санап айт.

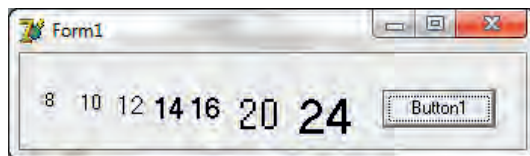
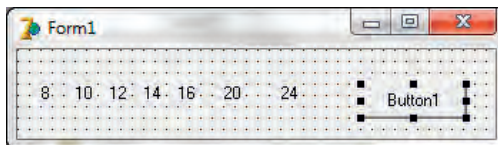


44-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

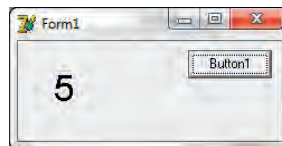
1. Жаңа қосымша жарат және оны **03_01** папкасына сақта.
 - 1.1. Қосымшаға жеті **Label** және бір **Button** объектіін орнат.
 - 1.2. **Label**-дердегі жазуларды қасиетінің көмегімен сәйкесінше 8, 10, 12, 14, 16, 20 мен 24-ке өзгерт.
 - 1.3. **Button1** түймешігін екі рет басып, оның процедурасына төмендегі операторларды енгіз:

```
Label1.Font.Size:=8; Label2.Font.Size:=10;  
Label3.Font.Size:=12; Label4.Font.Size:=14;  
Label5.Font.Size:=16; Label6.Font.Size:=20;  
Label7.Font.Size:=24;
```

- 1.4. Қосымшаны іске қос және түймешікті басып, нәтижесін көр:



2. Жаңа қосымша жарат және оны **03_02** папкасына сақта. Оған біреуден **Label** мен **Button** объектілерін орнат. **Button**-ды басқанда **Label**-дегі жазу шрифті өлшемі екі бірлікке артсын:



3. Алдыңғы сабақтағы жаттығуларды компьютерде орында.



45-САБАҚ. БАСҚАРУ ОБЪЕКТІЛЕРІНІҢ БЕЛСЕНДІЛІГІ МЕН КӨРІНБЕУІ

1. Басқару объектілерінің Visible қасиеті. Компьютер вирусын табу неліктен қиын? Өйткені олар көрінбейді, себебі олардың өз терезесі жоқ. Жаратылып жатқан қосымшаларды да көрінбейтін етіп қоюға бола ма? Әрине, болады.

Ол үшін әрбір басқару объектінде, соның ішінде қосымша терезесінде де болатын **Visible** (көрінетін) қасиетін пайдалану керек. Ол логикалық түрдегі қасиет болып, тек **True** (Шын) немесе **False** (Жалған) мәндерін қабылдайды.

Объектінің **Visible** қасиеті алдымен шын болады және объект терезеде көрініп тұрады. Егер осы қасиетті жалған етіп орнатсақ, объект терезеде көрінбейді. Оның орнын басқа мақсатта пайдалануға болады.

Егер екі немесе одан көп басқару объектілерін бір уақытта пайдаланылмаса, оларды тек бірінің ғана **Visible** қасиетін шын, қалғанын жалған етіп, қосымша терезесінің бір жеріне қойып, орынды үнемдеуге болады. Мұны визуал бағдарламалауда кең қолданады.

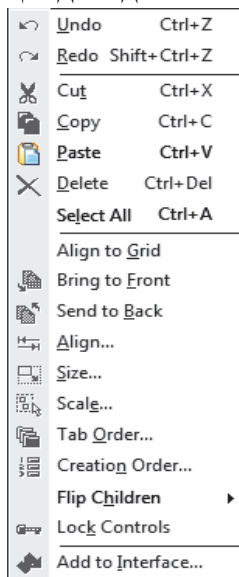
2. Басқару объектінің Enabled қасиеті.

Суретте **Delphi 7** негізгі менюсі **Edit** (Редакциялау) бөлімі көрсетілген. Оның көптеген тармақтары көмескіленгені көрініп тұр. Бұл олардың белсенді емес екенін, пайдалануға болмайтынын білдіреді.

Мысалы, менюдің төртінші тармағы **Copy**-ді (көшірме жасау) пайдалану үшін алдымен мәтіннің бір бөлігін белгілеп алу қажет. Мәтіннің бір бөлігі белгіленсе оны көшірме жасауға болады және менюдің **Copy** тармағы белсенділендіріліп, тоқ реңде пайда болып, осы амалды орындауға болады.

Басқару объектілерінің **Enabled** (белсенді) қасиеті осы мақсатқа қызмет етеді. Осы қасиет те логикалық мәндерді қабылдайды. Егер оның мәні шын болса, объект белсенді, жалған болса, объект пассив жағдайда болады. Бастапқыда барлық объектілерге бұл қасиеттің мәні шын болып, олар белсенді жағдайда болады. Егер бұл қасиетті жалған етіп орнатсақ, объект терезеде көрінсе де, ол көмескіленіп қалады. Оны уақытша пайдалануға болмайды.

3. Visible және Enabled қасиеттерін пайдалану. Объектілердің осы қасиеттерін пайдаланып, соңғы қосымшаның жұмысын жақсартамыз. Оның терезесінің кеңдігі 600-ден артса, **Button1** түймешігінің **Enabled** қасиетін жалған етіп орнатамыз. Нәтижеде терезені одан әрі кеңейтуге болмайды. Бастапқыда екінші түймешік **Button2** қосымша терезесінде көрінбесін (**Button2.Visible=False**).



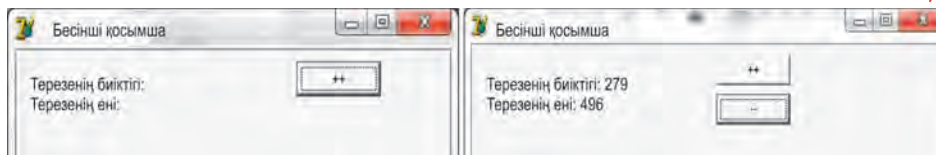
Бірінші түймешіктің көмегімен қосымша терезесін кеңейтеміз. Ол 600-ден артып кетуімен екінші түймешік терезеде пайда болады да (**Button2.Visible:=True**) бірінші түймешік көмескіленіп қалады (**Button1.Enabled:=False**). Енді тек екінші түймешіктің көмегімен қосымша терезесінің кеңдігін азайтуға болады. Кеңдік 600 пикселден аздауымен бірінші түймешік тағы да белсенді жағдайға өтсін (**Button1.Enabled:=True**). Терезенің кеңдігі 400-ден азайса, екінші түймешік тағы жоқ болсын, терезені одан әрі кіші етуге болмасын.

Бірінші түймешік басылғанда не істеу қажеттігін көрсету үшін оны 2 рет басамыз және мензер тұрған жерге төмендегі кодты енгіземіз:

```
If Form1.Width<600 then  
begin  
Form1.Width:=Form1.Width+16;  
Form1.Height:=Form1.Height+9; End else  
begin  
Button2.Visible:=True;  
Button1.Enabled:=False; end;
```

Екінші түймешікті 2 рет басып мына бағдарлама бөлігін енгіземіз:

```
If Form1.Width>400 then  
begin  
Form1.Width:=Form1.Width-16;  
Form1.Height:=Form1.Height-9; End else  
Button2.Visible:=False;  
if Form1.Width<600 then Button1.Enabled:=True;
```



ЕСІНДЕ САҚТА !

Әрбір визуалды объект белсенді немесе пассив, көрінбейтін немесе көрінетін жағдайда болады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. **Visible** қасиеті қандай міндетті атқарады?
2. **Enabled** қасиеті қайсы объектілерде кездеседі?
3. Бесінші қосымшаның терезесінің биіктігі мен ені сәйкесінше 5 және 8 қадаммен өзгерсін.
4. Бесінші қосымшаның терезесі енінің мәні 500-ден 800-ге дейін болсын.
5. Бесінші қосымшаның түймешіктеріндегі жазуларды **a→A** мен **A→a**-ға өзгерт.



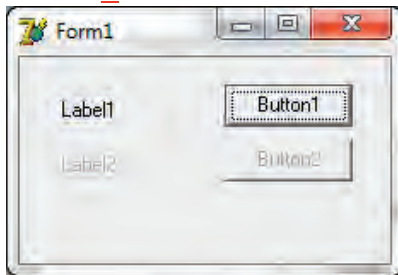
ҮЙГЕ ТАПСЫРМА

1. **Visible** қасиеті қайсы объектілерде кездеседі?
2. **Enabled** қасиеті қандай міндетті атқарады?



46-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

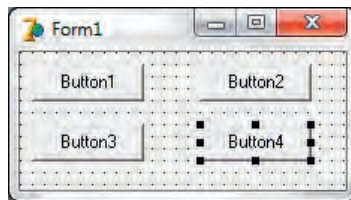
1. **Delphi**-де жаңа қосымша жарат және оны **04_01** папкаға сақта.
 - 1.1. Қосымша терезесінде үш **Label** мен үш **Button** объектіні орналастыр.
 - 1.2. Олардың біріншілерінің **Enabled** (белсенді) және **Visible** (көрінетін) қасиеттерін өзгертпе. Екіншілеріндегі **Enabled** (белсенді) қасиеттерін, ал үшіншілеріндегі **Visible** (көрінетін) қасиеттерін **False** (жалған) етіп орнат.



- 1.3. Қосымшаны іске қосып, ондағы өзгерістерді бақыла.

- 1.4. Қосымшаны сақта және оны жап.

2. Жаңа қосымша жарат және оны **04_02** папкаға сақта.



- 2.1. Оған төрт **Button** объектісін екі баған етіп орналастыр.

- 2.2. Олардың біріншісі екінші бағандағылардың, ал екіншісі бірінші бағандағылардың белсенділігін, үшінші мен төртіншілері көрінбейтіндігін әр кез керісінше өзгертсін.

- 2.3. Бірінші түймешікті екі рет басып, оның процедурасына:

```
Button2.Enabled:=not Button2.Enabled;  
Button4.Enabled:=not Button4.Enabled;
```

екінші түймешік процедурасына

```
Button1.Enabled:=not Button1.Enabled;  
Button3.Enabled:=not Button3.Enabled;
```

үшінші түймешік процедурасына

```
Button2.Visible:=not Button2.Visible;  
Button4.Visible:=not Button4.Visible;
```

төртінші түймешік процедурасына

```
Button2.Visible:=not Button2.Visible;  
Button4.Visible:=not Button4.Visible;
```

операторын енгіз.

2.4. Қосымшаны іске қосып, дұрыс жұмыс істеуін тексер.

2.5. Қосымшаны сақтап, оны жап.

3. Алдыңғы сабақтағы жаттығуларды компьютерде орында.



47-САБАҚ. ҚОСЫМШАҒА ДЕРЕКТІ ЕНГІЗУ

1. Қосымшаға мәтін жолын енгізу. Edit басқару объекті мен оның қасиеттері. Қосымша деректі шығарумен бірге оған деректі енгізу мүмкіндігіне де ие болуы керек. **Delphi** ортасында мұның ең қарапайым тәсілі қосымша терезесіне **Edit** (Енгізу алаңы) басқару объектінің енгізу. Бұл объект басқару объектілерінің стандарт панелінде алтыншы болып орналасқан және оның таңбашасы **Ab**.

Бұл объектінің көп қасиеті қосымша терезесі объекті **Form**-дікі сияқты. Онда да **Left**, **Top**, **Width**, **Height** қасиеттері бар. Онда **Caption** қасиеті жоқ. Сондықтан ол әдетте **Label** объектімен бірге істетіледі. Оның негізгі қасиеті **Text** (Мәтін) болып, оған енгізілетін мәтін жолы осы қасиетте сақталады. Егер осы қасиетке қандай да бір мәтін жолы мәнін берсек, бұл жол оның енгізу алаңшасында бейнеленеді.

Енгізу алаңы объектінің тағы бір маңызды қасиеті **ReadOnly** (тек оқуға ғана) деп аталады. Бұл қасиеттің мәні логикалық шама болып, тек **True** (шын) немесе **False**-ті (жалғанды) қабылдайды. Бастапқыда бұл қасиеттің мәні **False** (жалған) деп орнатылған. Сондықтан оған деректі енгізуге болады.

Егер бұл қасиеттің мәні **True**-ге (шынға) өзгертілсе, оған пайдаланушы тарапынан дерек енгізілмейді. Бірақ қосымша ондағы деректі өзгерте алады. Бұл не үшін қажет, деген сұрақ туындауы табиғи. **Edit** объектінің **Label**-ден айырмашылығы – рамкаға алынған және оған енгізілген дерек әдемірек көрінеді. Егер біз одан деректі шығару үшін пайдаланбақшы болсақ, осы деректің пайдаланушы тарапынан өзгертілмеуін қамтамасыз етуіміз керек.

2. Edit басқару объектінің басқа қасиеттері. **Edit** объектінің тағы бір пайдалы қасиеті **MaxLength** деп аталады. Оның көмегімен алаңға енгізілетін мәтін жолының ұзындығын шектеуге болады. Мысалы, алаңға оқушының туылған жылы енгізілуі қажет болса, **MaxLength**-ті 4-ке тең етіп алу көп жағдайда оған адасып басқа деректі енгізіп жіберудің алдын алады.

Edit объектінің тағы бір пайдалы қасиеті **Font** болып, оның көмегімен алаңға енгізілетін мәтін үшін шрифтті, шрифт өлшемі, түсі және шрифттің тағы бірқатар қасиеттерін орнатуға болады. Бұл қасиет басқа объектілерде, мысалы, **Form** объектінде де бар. Егер біз шрифтті қосымша терезесі үшін орнатсақ, ол қосымша терезесіндегі барлық объектілерге де әсер етеді. Мұның алдын алу қажет болса, қосымша терезесіндегі объектінің **ParentFont** (“ата” шрифт) қасиетінің мәнін шыннан жалғанға өткізу қажет.

Edit-тің тағы бір пайдалы логикалық қасиеті **AutoSize** (өлшемді автоматты түрде таңдау) болып, алаңға енгізілетін мәтін көбейген сайын алаңның көлемі артып баруын қамтамасыз етеді. Егер мұнда алаңның жанындағы басқа объектілерді көлеңкілеп қоймауы керек болса, бұл қасиет мәнін шыннан жалғанға өткізу қажет болады.

Енгізу алаңы пайдаланушының паролін енгізуге арналған болса, парольді басқалардың көрмеуі үшін ол басқа белгімен, мысалы, “*” белгісімен алмастырылады. **Edit**-те бұл мүмкіндік те көзделген. Оның **PasswordChar** қасиеті алаңға енгізілген әріптерді алмастыратын белгінің реттік санын (32–127) білдіреді. Бастапқыда ол #0-ге тең және ол енгізілетін әріп басқасына алмаспайтынын білдіреді. Егер біз оны #42-мен алмастырсақ, алаңға енгізілген әріптердің орнына “x” әріпі бейнеленеді. #42-нің орнына тікелей экранға шығарылуы керек болған белгіні енгізуге де болады. Мысалы, “-“, “+“, “*”, “.”.

3. Пайдаланушыны құттықтау қосымшасы. Пайдаланушы қосымшаға өзі аты мен баршаға ортақ пароль: “12345-ті” енгізген болсын. Ол үшін **Delphi**-де жаңа қосымша жаратып, оның терезесіне екі **Label**, үш **Edit**, бір **Button** объектіні енгіземіз. Олардың қасиеттерін объект инспекторында (**Object Inspector**) төмендегідей өзгертеміз:

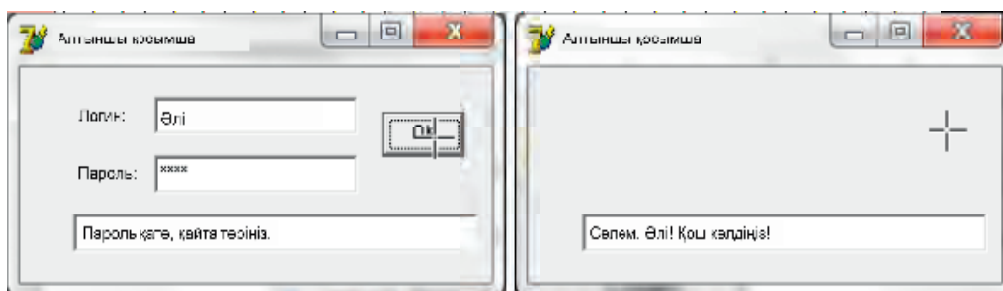
```
Form1.Caption:='Oltinchiilova';  
Label1.Caption:='';  
Label2.Caption:='';  
Button1.Caption:='OK';  
Edit1.Text:='';Edit2.Text:='';  
Edit3.Text:='';  
Edit2.Passwordchar:='#42';  
Edit3.ReadOnly:=True;
```

Осы жұмыстарды орындаған соң, **Button1**-ді екі рет басамыз және экранда пайда болған бағдарлама мәтіні терезесінде меңзер тұрған жерге төмендегі операторларды қатесіз енгіземіз:

```
if Edit2.Text='12345' then begin  
Edit3.Text:='Сәлем, '+Edit1.Text+'!Қош келдіңіз!';  
Edit1.Visible:=False;Edit2.Visible:=False;  
Label1.Visible:=False;Label2.Visible:=False;  
Button1.Visible:=False; end else  
Edit3.Text:='Пароль қате, қайта теріңіз.';
```

Бұл операторлар **Button1** түймешігін басқанда іске қосылады. Олардың біріншісі **Edit2** алаңына енгізілген парольдің дұрыстығын тексереді. Егер дұрыс болса, үшінші жолдағы оператор үшінші алаңға пайдаланушыны құттықтайтын мәтін жолын шығарады. Кейінгі үш жолдағы операторлар енді қажетсіз болған екі **Label1**, **Label2**, екі **Edit1**, **Edit2** мен **Button1** объектілерін терезеде көрінбейтін етеді. Кері жағдайда үшінші алаңдағы пароль қате терілгені туралы хабар шығады және қайта әрекет жасауға болады.

Төмендегі суреттердің біріншісінде пароль қате терілгендегі, екіншісінде пароль дұрыс терілгендегі жағдай бейнеленген.



ЕСІНДЕ САҚТА !

Edit объекті қосымшаға мәтін жолы көрінісіндегі деректі енгізу үшін қолданылады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. **Edit** басқару объекті қандай міндетті атқарады?
2. **Edit** объектінің **Text** қасиетінде не сақталады?
3. **Edit** объектінің **ReadOnly** қасиеті қандай міндетті атқарады?
4. Алтыншы қосымшаның паролін а) “I7:7200U”; ә) “123-12-20” етіп алмастыр.
5. Алтыншы қосымшадағы **Edit3**-тің орнына **ShowMessage** командасын пайдалан.



ҮЙГЕ ТАПСЫРМА

1. **Edit** объектінің **MaxLength** қасиеті қандай міндетті атқарады?
2. **Edit** объектінің **PasswordChar** қасиеті қандай міндетті атқарады?



48-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

1. Алтыншы қосымшада логин немесе пароль қате терілген болсын. **Edit3**-тегі жазу осы жайлы ескертіп тұр. Егер пайдаланушы тағы да қателік жасаса, **Edit3**-тегі жазу өзгермейді. Қосымшада ештеңе өзгермесе, пайдаланушы күманданып қалады. Мұның алдын алу үшін **Edit1** немесе **Edit2**-ге жаңа дерек енгізілсе **Edit3**-тегі жазу өшіп кетсін.

1.1. **Delphi**-ді іске қос және негізгі менюдің файл бөліміндегі **Open Project** тармағының көмегімен алтыншы қосымшаны жүктеп ал.

1.2. **Edit1**-ді тышқанмен екі рет бас. Экранда пайда болған **TForm1.Edit1Change** процедурасына **Edit3.Text:=**"; операторын енгіз.

1.3. **Edit2**-ні тышқанмен екі рет бас. Экранда пайда болған **TForm1.Edit2Change** процедурасына **Edit3.Text:=**"; операторын енгіз.

1.4. Қосымшаны іске қосып, өзгерістер болғанына көз жеткіз.

1.5. Қосымшаны сақта және оны жап.

2. Пайдаланушы ескі парольді жаңасына алмастыратын қосымша жарат.

2.1. Алтыншы қосымшаны қайта жүктеп, оны **05_01** папкаға сақта.

2.2. Қосымша терезесіне **Label** мен **Edit** объектілерін қос және олардың қасиеттерін төмендегі суретте көрсетілгеніндей етіп өзгерт:



2.3. **OK** түймешігінің процедурасын өзгерт.

2.4. Қосымшаны іске қосып, дұрыс жұмыс істеуін тексер.

2.5. Қосымшаны сақта және оны жап.

3. Алдыңғы сабақта берілген жаттығуларды компьютерде орында.



49-САБАҚ. ДЕРЕК ТҮРІН ӨЗГЕРТУ

1. Деректі мәтін жолын бүтін санға және керісінше өзгерту.

Есінде болса, **Pascal** бағдарламалау тілінде бір түрдегі айнымалыға екінші түрдегі өрнектің мәнін беруге болмайтын еді. **Delphi**-де де бұған тыйым салынған. Оны қалай айналып өтуге болады?

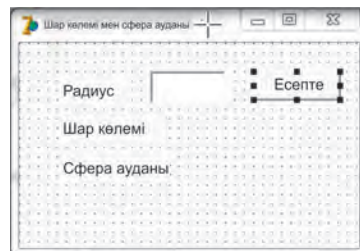
Мысалы, **Button1** объектінің ені (**Width** қасиеті) қанша болуы (бүтін сан түрі) **Edit**-тің **Text** қасиетінен (мәтін жолы түрі) алынуы керек болса, **Button1.Width:=Edit1.Text** операторынан пайдалану қате болады. Нәтижеде қосымша бағдарламасы жұмыс істеуден тоқтап, қателік туралы ескертеді.

Object Pascal-да айнымалыны бір түрден екіншісіне өткізу үшін бірқатар стандарт функциялар бар. Олар төмендегілер:

- 1) **StrToInt** (string to integer) – мәтін жолын бүтін санға өткізеді;
- 2) **IntToStr** (integer to string) – бүтін санды мәтін жолына өткізеді;
- 3) **StrToFloat** (string to float) – мәтін жолын шын санға өткізеді;
- 4) **FloatToStr** (float to string) – шын санды мәтін жолына өткізеді.

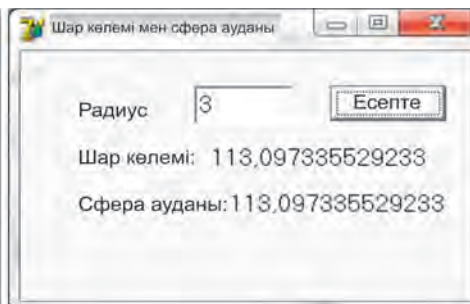
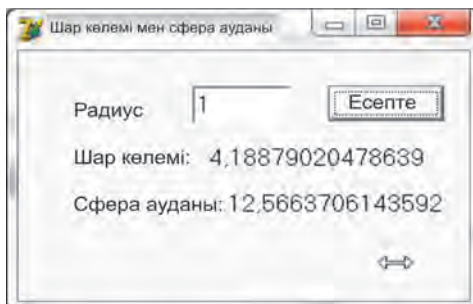
Жоғарыдағы операторды осы функцияларды пайдалана отырып, қатесіз жазамыз: **Button1.Width:=StrToInt(Edit1.Text);**

2. Шар көлемі мен сфера ауданы қосымшасы. Шар көлемі мен оған сәйкес сфераның ауданын есептейтін қосымша жаратамыз. Оған үш **Label**, бір **Edit** пен бір **Button** объектілерін орнатамыз. Қосымша терезесін және ондағы объектілерді мына суреттегідей етіп өзгертеміз:



Қосымша терезесіндегі **Button1**-ді екі рет басып, пайда болған бағдарлама коды терезесіне төмендегі екі жолды енгіземіз. Олардың әрбіреуі жеке бір жолға сыймаған болса, біреуден қатарға енгізілуі қажет. Екі жол да нүктелі үтірмен аяқталады.

```
Label2.Caption:='Шар_көлемі: '+FloatToStr(4/3*pi*sqr  
(StrToFloat(Edit1.Text))*StrToFloat(Edit1.Text));  
Label3.Caption:='Сфера_ауданы: '+FloatToStr(4*pi*sqr  
(StrToFloat(Edit1.Text)));
```



F9 түймешігін екі рет басып, қосымшаны іске қосамыз. Алдымен енгізу алаңына 1-ді енгізіп, **Есепте** түймешігін басамыз, кейін радиусы

3-ке тең шар көлемі мен сфера ауданын есептейміз. Қосымша жұмысының нәтижелері жоғарыдағы суреттерде көрсетілген.

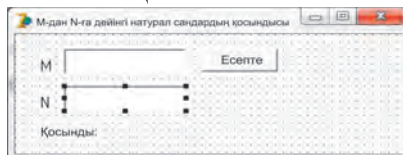
3. Қосымшаларда айнымалыларды пайдалану. Соңғы екі жол өте күрделі көрініске ие болып, оларды енгізуде қателік жасау ештеңе емес. Шардың көлемі мен сфераның ауданы $V = \frac{4}{3}\pi r^3$; $S = 4\pi r^2$

формулалармен есептелуін ескерсек, жоғарыдағы формулалардың шынымен де шала болғаны айқын көрінеді.

Осы кемшіліктерді жоюдың жолы бар ма? Әрине бар. Ол үшін қосымша айнымалыларды пайдалану қажет. Айнымалылар бағдарлама мәтініне қалай енгізілуін есіңе түсіру үшін бағдарлама коды терезесімен және ондағы бағдарлама мәтінімен жақсырақ танысамыз.

4. Бүтін сандар қосындысы қосымшасы. m және n бүтін сандары берілген болсын. Олар мен олардың арасындағы барлық бүтін санның қосындысын табатын қосымша жаратамыз. Оған үш **Label**, екі **Edit** пен бір **Button** объектіін орнатамыз. Олардың орналасуы мен қасиеттері төмендегі суретте берілген.

Сандардың қосындысын есептеу **Button1** түймешігін басқан кезде орындалады. Оны қосымшаға енгізу үшін осы түймешікті екі рет басамыз. Нәтижеде экранда бағдарлама мәтіні терезесі пайда болады. Ондағы жазулармен танысайық:



```
procedure TForm1.Button1Click(Sender: TObject);  
begin end;
```

Бұл **Pascal** тілінде жазылған қарапайым процедура, қазірше ол бос. Демек, біз қосымша терезесі **Form1**-ге орналасқан **Button1**-ді басқанда қосымшаны басқаруды **TForm1.Button1Click** атты процедураға өткізеді екен. Осы процедураға ол жасауы қажет боған істі орындайтын операторларды енгіземіз. Процедураның толық көрінісі төмендегідей:

```
procedure TForm1.Button1Click(Sender: TObject);  
var i,m,n,s:integer;  
begin
```

```

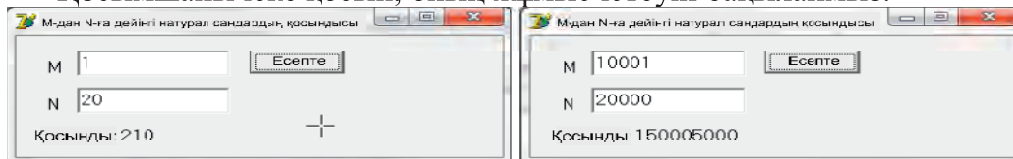
m:=Strtoint(Edit1.Text);
n:=Strtoint(Edit2.Text);
s:=0; for i:=m to n do s:=s+i;
Label3.Caption:='Қосынды: '+Inttostr(s); end;

```

Процедураның екінші жолында **var** операторының көмегімен төрт бүтін сан түріндегі айнымалы сипатталған. Үшінші жолдан процедураның денесі басталады. Оған орындалатын операторлар орналасқан болады. Төртінші жолда **Edit1** алаңына енгізілген сан бүтін сан түріне айналдырылып, *m* айнымалысына беріледі. Бесінші жолда **Edit2**-ге енгізілген сан *n* айнымалысына беріледі.

Алтыншы жолда қосынды сақталатын айнымалыға бастауыш мән: 0 беріледі. Жетінші жолдағы цикл қажетті қосындыны табады. Сегізінші жолда табылған қосындының мәнін **Label3** арқылы қосымша терезесіне шығарылады. Тоғызыншы жолдағы оператор процедураның аяқталғанын білдіреді.

Қосымшаны іске қосып, оның жұмыс істеуін бақылаймыз.



ЕСІНДЕ САҚТА !

Мән беру операторында айнымалы мен өрнек түрлерінің айырмашылығы болса, бағдарламада қателік пайда болады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Деректің қандай түрлерін білесің?
2. **StrToInt** деректі қандай түрден қандай түрге өткізеді?
3. **FloatToStr** деректі қандай түрден қандай түрге өткізеді?
4. **StrToFloat** деректі қандай түрден қандай түрге өткізеді?
5. а) квадрат; ә) тік төртбұрыш периметрі мен ауданын есептейтін қосымша жарат.
6. *m*-нен *n*-ға дейінгі барлық санның а) квадратының; ә) квадрат түбірінің қосындысын есептейтін қосымша жарат.



ҮЙГЕ ТАПСЫРМА

1. Деректің түрін өзгертетін функцияларды дәптеріңе жазып ал.



50-САБАҚ. БАҚЫЛАУ ЖҰМЫСЫ

Төменде бақылау жұмысы вариантының үлгілері көрсетілген:

1-вариант

1. **Label** объектінің негізгі қасиеттері.
2. **FloatToStr** стандарт функциясы.
3. Шеңбердің ұзындығын есептейтін қосымша жарат.

2-вариант

1. **Button** объектінің негізгі қасиеттері.
2. **IntToStr** стандарт функциясы.
3. Шеңбердің ауданын есептейтін қосымша жарат.

3-вариант

1. **Edit** объектінің негізгі қасиеттері.
2. **StrToInt** стандарт функциясы.
3. Квадраттың ауданын есептейтін қосымша жарат.

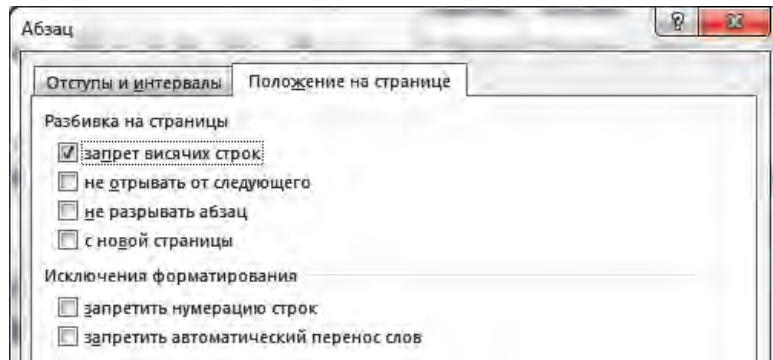
4-вариант

1. **Button** объектінің **OnClick** процедурасы.
2. **StrToFloat** стандарт функциясы.
3. Квадраттың периметрін есептейтін қосымша жарат.



51-САБАҚ. ҚОСЫМШАДА ЖАЛАУШАЛАРДЫ ПАЙДАЛАНУ

1. Жалаушаларды пайдалануға мысалдар. Қосымшаларда көп пайдаланылатын басқару объектілерінің бірі – жалауша. Жалауша орнатылған және орнатылмаған жағдайда болады. Егер жалауша орнатылған болса, қандай да бір жұмыс жасалады, орнатылмаған болса, ешнәрсе жасалмайды.



Жалаушаларды пайдалануға мысал етіп **Word** мәтін редакторының абзац тілдесу терезесіндегі тармақтардың бірі жоғарыдағы суретте көрсетілген. Онда алты жалауша болып, оның тек біреуі орнатылған.

Delphi-де жаратылған қосымшаларда оларды пайдалануға бола ма? Әрине болады, бұл объектілер қосымшаларды пайдалануды оңайлатып, олар интербелсенді қосымшаларда кеңінен қолданылады.

2. CheckBox басқару объекті және оның қасиеттері. Delphi-де жалауша объекті **CheckBox** (Тексеру қорабы) деп аталады. Бұл объект объектілер панелінің стандарт тармағында тоғызыншы болып орналасқан. Төмендегі суретте ол қызғыш жиекпен қоршалған:



CheckBox объектінің негізгі қасиеті **Checked** (орнатылған) болып, ол екі: шын және жалған мәндерді қабылдайды. Оның тағы бір пайдалы қасиеті бізге басқа объектілерде таныс болған **Caption** (тақырыпша). Оның көмегімен жалауша туралы оның жанында қысқа жазуларды жаратуға болады.

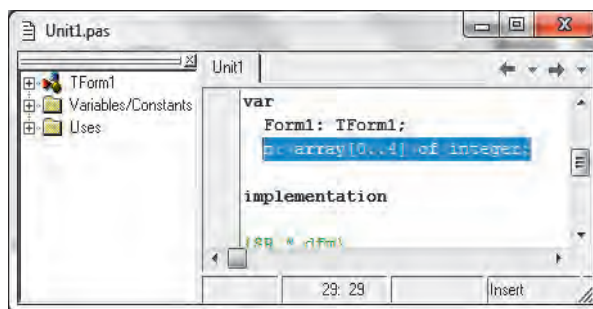
3. Қосымшада глобал айнымалыларды жарату. Қосымшада жалауша объекті болып, қосымша істеген кезде жалаушаның неше рет орнатылғаны маңызды болсын. Осы деректі қалай сақтауға болады.

Осы деректі сақтау үшін төмендегі жұмыс орындалады. Қосымшада бір артықша объект жаратып, оны көрінбейтін етіміз (**Visible:=0**). Алдымен оның енін 0-ге тең етіп (**Width:=0**), әр кез жалауша орнатылғанда енін біреуге арттырамыз (**Width:=Width+1**).

Бірақ бұл лажсыздық белгісі. Бұл тәсілмен айнымалылар массивін жаратуға болмайды. Қайтадан велосипед жаратуды қойып, соңғы қосымшамызды, бүтін сандар қосындысын есімізге түсірейік.

Ол жерде біз `var i,m,n,s:integer;` операторының көмегімен бірнеше айнымалыларды жаратқан едік. Тек бұл айнымалылар процедураның ішінде жаратылғаны үшін оларды процедураның сыртында пайдалануға болмайды. Мұндай айнымалылар **локал** (жергілікті) айнымалылар деп аталады. Олардан өзгеше болған **глобал** (жалпы) айнымалыларды ерікті түрде пайдалануға болады.

Глобал айнымалылар бағдарлама (қосымша) басында бейнеленеді. Осы орынды табу үшін жаратылған қосымшаның бағдарлама кодының терезесіне өтіп, сипаттау операторы `var`-ды табамыз.

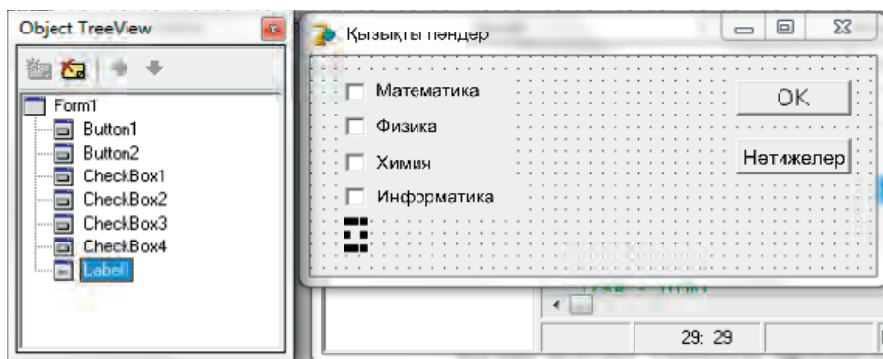


Оны тауып, одан кейін бізге қажетті болған сипаттау операторы `n: array[0..4] of integer;`-ді енгіземіз. Енді ол қосымша терезесі **Form1** айнымалысымен қатар тұр және **Form1** сияқты қосымшаның барлық жерінен пайдаланамыз.

Айнымалыларды сипаттаумен жұмыс бітпейді. Оларға тағы қосымша іске қосылуымен бастапқы мәндер: 0-дерді беру керек. Мұны қосымша терезесі жаратылғанда орындау қажет.

4. Қызықты пәндер қосымшасы. Жалаушаларды пайдалануды үйрену үшін қызықты пәндер сұрауы қосымшасын жаратамыз. Осы қосымша оқушылар арасында сұрау өткізіп, олардың қай пәнге қызығатынын үйренеді. Қосымшаны оңтайландыру үшін анық және жаратылыстану пәндерінің бірнешесімен шектелеміз.

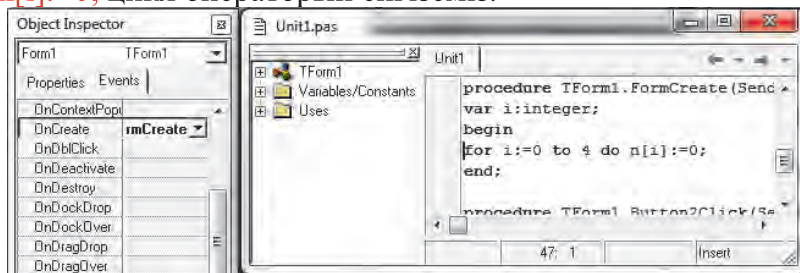
Delphi-де жаңа қосымша жаратамыз. Оған төрт **CheskBox**, екі **Button** мен бір **Label** объектілерін төмендегі суреттегідей етіп орнатамыз. Қосымша терезесі мен ондағы объектілердің тақырыпшасын **Caption** қасиеті көмегімен суреттегідей етіп орнатамыз:



Қосымша кодына `n: array[0..4] of integer;` операторын алдыңғы тармақта көрсетілген орынға енгіземіз (біреу алдыңғы суретке қара).

Осы массив элементтеріне бастапқы мәнді беру үшін қосымша терезесі **Form1**-ді таңдаймыз (қосымша терезесінің басқа объектілерден бос жеріне тышқанды басамыз). **Object inspector**-да **Events** (оқиғалар) тармағына өтіп, **OnCreate** (жаратылғанда) оқиғасын тауып, оның оң жағындағы бос ұяшыққа тышқанды екі рет басамыз.

Нәтижеде экранда бағдарлама қосымшасы терезесі мен ондағы **TForm1.FormCreate** процедурасы пайда болады. Осы процедурада **begin**-нен алдын `var i:integer;` сипаттау операторын, одан кейін `for i:=0 to 4 do n[i]:=0;` цикл операторын енгіземіз.



Енді қосымша терезесінде **Button1**-ді екі рет басып, экранның меңзер тұрған жеріне бағдарлама кодын енгіземіз.

```

n[0]:=n[0]+1;
if Checkbox1.Checked then n[1]:=n[1]+1; Checkbox1.Checked:=False;
if Checkbox2.Checked then n[2]:=n[2]+1; Checkbox2.Checked:=False;
if Checkbox3.Checked then n[3]:=n[3]+1; Checkbox3.Checked:=False;
if Checkbox4.Checked then n[4]:=n[4]+1; Checkbox4.Checked:=False;
  
```

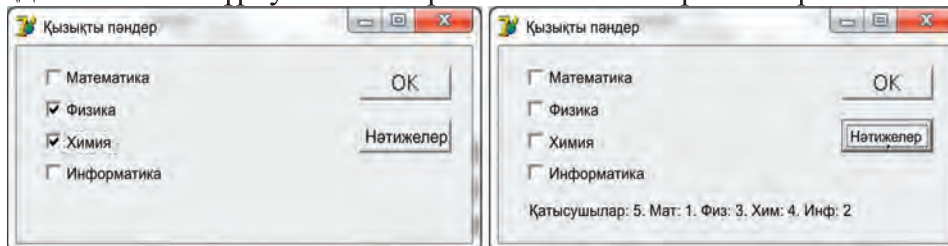

Осы операторлардың біріншісі әрдайым **OK** түймешігі басылғанда (келесі оқушы дауыс бергенде) дауыс берушілердің саны біреуге артады. Келесі төрт **if** операторының әрбірі **CheckBox**-тарға жалаушалар орнатылғанын тексереді және олар орнатылған (тиісті пәнге дауыс берілген) болса, дауыстардың саны сақталатын айнымалы мәні біреуге артады. **if** операторынан кейінгі операторлар дауыс беру анықталғаннан соң **CheckBox**-тардағы жалаушаларды алып тастайды және қосымша келесі дауыс беруге дайын болады.

Дауыс беру нәтижесін экранға шығару үшін қызмет ететін **Нәтижелер** түймешігін екі рет басып, оның процедурасына төмендегі операторларды енгіземіз:

```
Label1.Caption:='Қатысушылар: '+Inttostr(n[0]);  
Label1.Caption:=Label1.Caption+',Мат: '+Inttostr(n[1]);  
Label1.Caption:=Label1.Caption+',Физ: '+Inttostr(n[2]);  
Label1.Caption:=Label1.Caption+',Хим: '+Inttostr(n[3]);  
Label1.Caption:=Label1.Caption+',Инф: '+Inttostr(n[4]);
```

Осы операторлардың біріншісі дауыс беруге неше оқушы қатысқанын, екіншісі математикаға, үшіншісі физикаға, төртіншісі химияға, бесіншісі информатикаға неше дауыс бергенін **Label1** объектінің **Caption** қасиетінің көмегімен экранға шығарады.

Қосымшаны іске қосып, оның дұрыс жұмыс істеуін тексереміз. Төмендегі суреттерде қосымшаның оқушы қызыққан пәндерін таңдағаны және сұрау нәтижелері бейнеленген көрінісі берілген:



ЕСІНДЕ САҚТА !

Жалауша орнатылған болса, оған сәйкес амал орындалады, түсірілген болса орындалмайды.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Жалаушалар қандай міндетті атқарады?
2. Қосымшаға жалаушалар қайсы объектінің көмегімен енгізіледі?
3. Жалаушалардың орнатылғаны қайсы қасиеттің көмегімен тексеріледі?
4. Глобал айнымалылар қай жерде сипатталған?
5. Айнымалыларға бастапқы мәндерді қайда беруге болады?
6. Белсенді оқушылар сұрауын өткізетін қосымша жарат.
7. Шебер оқушылар сұрауын өткізетін қосымша жарат.



ҮЙГЕ ТАПСЫРМА

1. Глобал және локал айнымалылар несімен ерекшеленеді?



52-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

1. Енгізілген коэффициенттерге қарап квадрат тендеудің детерминанты мен түбірін табатын қосымша жарат.

2. Берілген аралықта $y=\sin x$ функциясының орташа мәнін есептейтін қосымша жарат.

3. Берілген санның квадратын, кубын, төртінші және алтыншы дәрежелі түбірлерін сәйкес жалаушалардың орнатылғанына қарай есептейтін қосымша жарат.

4. Алдыңғы сабақтардағы жаттығуларды компьютерде орында.

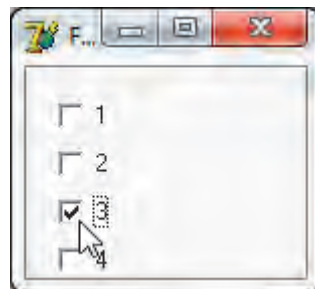
5. Қосымша терезесін жаратып, оған төрт жалауша орнат. Жалаушалардың қай бірі орнатылғанда қалғаны түсірілетін етіп жалаушаларды бағдарламала.

5.1. Жалаушалардың тақырыпшаларын тиісінше 1, 2, 3, 4-терге өзгерт.

5.2. Бірінші жалаушаны екі рет тышқанмен басып, оның `TForm1.CheckBox1Click` процедурасына

```
CheckBox2.Checked:=false;
CheckBox3.Checked:=false;
CheckBox4.Checked:=false;
```

операторларын енгіз.



5.3. Екінші жалаушаның осындай процедурасына бірінші, үшінші және төртінші жалаушаларды түсіру операторларын енгіз.

5.4. Үшінші және төртінші жалаушалар үшін де осы сияқты процедуралар жарат.

5.5. Қосымшаны іске қосып, оның дұрыс жұмыс істеуін тексер.



53-САБАҚ. ҚОСЫМШАДАҒЫ РАДИОТҮЙМЕШІКТЕР ТОБЫН ПАЙДАЛАНУ

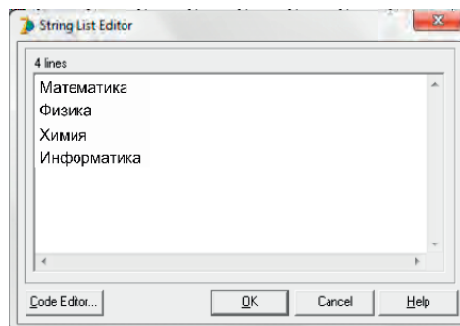
1. Радиотүймешіктерді пайдалануға мысалдар. Көбінесе бірнеше варианттың тек біреуін таңдауға тура келеді. Мысалы, жоғары оқу орнына кіру емтихандарында әрбір тест сұрағына төртеуден жауап болады, олардың тек біреуі дұрыс және тек бір жауапты таңдау керек.

Деректің **Word**-та абзацты форматтаудың төрт түрі: солдан, оңнан, ортадан және екі жақтан туралану тәсілдері бар болып, олардың тек біреуін абзацқа қолдануға болады. Олардың бірін таңдау басқасын жоюға әкеледі. Мұндай мысалды басқа салаларда да көрсетуге болады.

2. RadioGroup басқару объекті және оның қасиеттері. Бірнеше варианттың бірін таңдау **Pascal**-да **Case** операторының көмегімен орындалатын еді. **Delphi**-де мұның бірнеше тәсілі бар болып, олардың біріншісі **RadioGroup** (радио түймешіктер тобы) деп аталады. Осы басқару объекті объектілер панельдерінің стандарт тармағында орналасқан болып, ол оңнан үшінші:

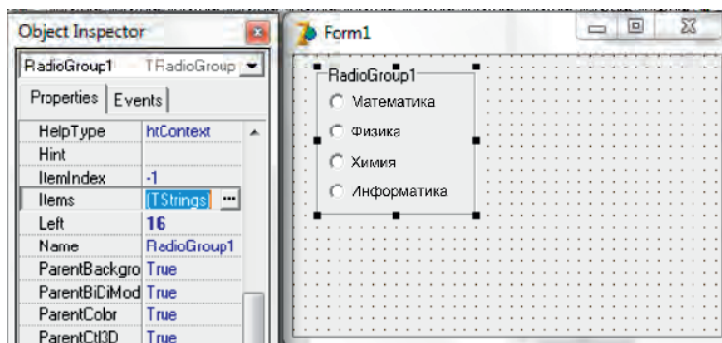


Бұл объектімен танысу үшін оны қосымша терезесіне орнатамыз. Оның негізгі қасиеттерінің бірі – осы кезге дейін кездеспеген **Items** (варианттар). **ObjectInspektorda** осы қасиетті екі рет басып таңдаймыз. Мұнда экранда оның тілдесу терезесі пайда болады. Оған радио түймешік үшін варианттардың аттары тізімін енгізуге болады. Оған төрт варианттың аттарын әрбірін жаңа жолдан енгіземіз Олар математика,



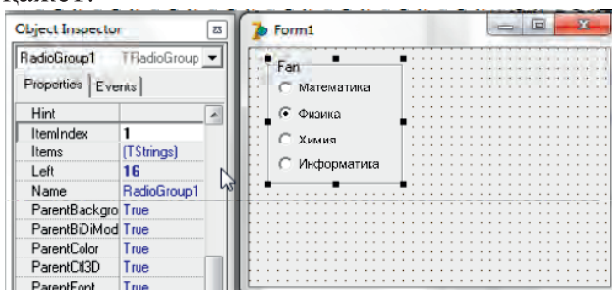
физика, химия және информатика болсын. Тілдесу терезесінің көрінісі төмендегідей болады.

Тілдесу терезесінде **OK** түймешігін басып, енгізілген варианттарды сақтап қоямыз немесе **Cancel** (жою) түймешігін басып, енгізілген варианттардан бас тартамыз. Енгізілген варианттарды қабылдасақ, олар дереу қосымша терезесінде пайда болады:



RadioGroup-тың тағы бір маңызды қасиеті **ItemIndex** (вариант индексі) болып, онда таңдалған варианттың нөмірі сақталады. Егер ешбір нұсқа таңдалмаған болса, бұл қасиеттің мәні -1 -ге тең болады. Объект инспекторында осы қасиетті 1 -ге тең етіп аламыз. Бұл объектінің да **Caption** қасиеті бар болып, оған объект инспекторында **Fun** деген жолды енгіземіз. Бұл өзгерістер қосымша терезесінде дереу бейнеленеді.

Өзгерістердің, бір қарағанда, ерекше қызық тұсы тізімдегі бірінші пән: математика емес, екінші тұрған физиканың таңдалғанында. Шындығында мұның ешқандай ерекше тұсы жоқ. Тек информатикадағы есеп көбінесе 1 -ден емес, 0 -ден басталатынын ұмытпау ғана қажет:



RadioGroup қасиеттерінің бірі **Columns** (бағандар) деп аталады. Бұл қасиет радио түймешіктер тобының варианттары неше баған болып орналасқанын көрсетеді.

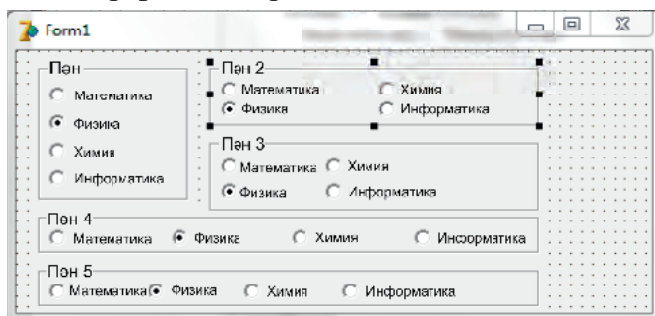
Төмендегі суретте төрт вариантты радио түймешіктер тобының біреуден бесеуге дейінгі бағандарда орналасқаны көрсетілген. Осылардан төмендегі қорытындылар келіп шығады:

1. Радио түймешіктер тобы үшін арналған орынның ені бағандар арасында тең бөлінеді.

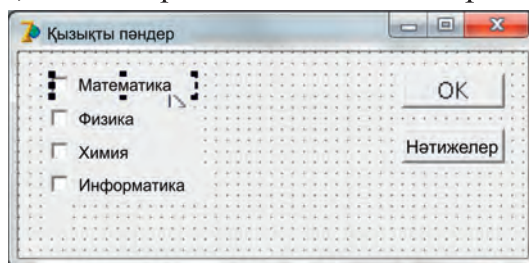
2. Түймешіктер санын бағандар санына бөліп, әрбір бағанда неше түймешік болатыны анықталады. Түймешіктер саны бөлшек сан болса, үлкен бүтін санға дейін жуықталады.

3. Әрбір баған түймешіктермен толықтырылады.

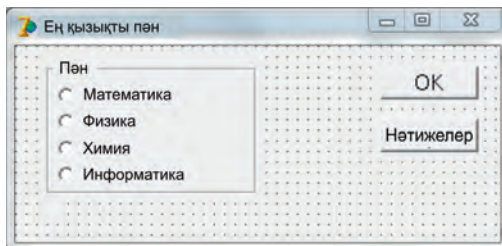
4. Соңғы бағандардың бірі немесе бірнешеуі бос болып қалуы мүмкін. Бос бағандар үшін де орын бөлінеді:



3. **Ең қызықты пән қосымшасы.** Қызықты пәндер қосымшасын бір шама өзгертеміз. Сұрауға әрбір оқушы тек бір ғана пәнді көрсете алатын болсын. Бұл қосымшада **Radiogroup** объектіні пайдаланамыз. Оның тақырыпшасын өзгертіп, ондағы төрт **Checkbox** объектіні алып тастаймыз. Ол үшін олардың үстіне кезек-кезегімен тышқанның сол түймешігін басып, сосын пернетақтаның **Delete** түймешігін басамыз:



Оның үстіне жоғарыда көрсетілген тәсілмен **Radiogroup** объектіін орналастырып, оған төрт радиотүймешікті төмендегідей етіп енгіземіз.

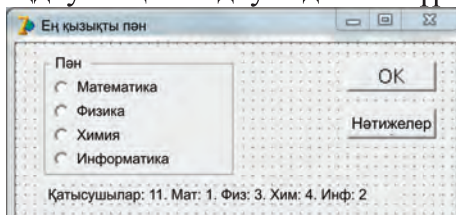


Енді **OK** түймешігін екі рет басып оның **TForm1.Button1Click** процедурасына төмендегі кодты енгіземіз:

```
procedure TForm1.Button1Click(Sender: TObject);  
var i:integer;begin  
n[0]:=n[0]+1;  
i:= Radiogroup1.Itemindex; if i>=0 then n[i+1]:=n[i+1]+1;  
Radiogroup1.Itemindex:=-1; end;
```

Осы процедура әр кез іске қосылғанда (қосымшадағы **OK** түймешігі басылғанда) екінші қатарда оператордың көмегімен жаңа айнымалы **i**-ді жаратады. Үшінші қатарда сұрауда қатысқандардың санын біреуге арттырады. Төртінші қатарда **Radiogroup1**-де қайсы вариант таңдалғанын **i** айнымалысына ұзатылады.

Егер осы мән 0-ден үлкен немесе оған тең болса, яғни қандай да бір вариант таңдалған болса, осы варианттың таңдалғандығының саны біреуге артады. Бесінші қатарда варианттар таңдалмаған түрде өткізіледі және процедура өз ісін аяқтайды. Қосымша кейінгі пайдаланушының таңдауын қабылдауға дайын тұрады:



ЕСІНДЕ САҚТА !

Радио түймешіктер тобы бірнеше варианттың біреуін таңдау үшін қызмет етеді.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Радио түймешіктер тобы объектінің **Items** қасиеті қандай мақсатта пайдаланылады?
2. Радио түймешіктер тобы объектінің **Columns** қасиеті қандай міндетті орындайды?
3. Қосымшаға сыныптастарыңның аттары жазылған радио түймешіктер тобын орнат. Олардың бірін таңдап, басқару түймешігі басылғанда таңдалған оқушыға оның атын айтып сәлем беретін қосымша жарат.
4. Алдыңғы қосымшаға тағы пәндердің атауы жазылған және бағалардың тізімі болған тағы да екі радио түймешіктер тобын енгіз. Таңдалған оқушының таңдалған пән бойынша таңдап алынған бағасын экранға шығаратын қосымша жарат.



ҮЙГЕ ТАПСЫРМА

1. Радио түймешіктер тобын қажет ететін жағдайларға мысал айт.
2. Радио түймешіктер тобы **RadioGroup** объекті объектілер панелінің қай жерінде орналасқан?



54-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

1. Жаңа қосымша жарат және оны **07_01** папкаға сақта.
 - 1.1. Оның терезесін **Тест жүйесі** деп ата.
 - 1.2. Радио түймешіктер тобын жаратып, оның тақырыпшасын **Тест сұрақтары** деп өзгерт. Оған 36 түйме қойып, олардың аттарын **1-ден 36-ға** дейін етіп, өзгертіп шық.
 - 1.3. Қосымша терезесіне жаңа радио түймешіктердің жаңа тобын қос. Оны **Жауаптар** деп ата. Оған төрт түймешік қосып, олардың аттарын **A, B, C, D** деп өзгерт.
 - 1.4. Қосымшаны іске қосып, оған тест сұрақтары мен жауаптарын таңдап көр.
 - 1.5. Қосымшаны сақта және оны жап.
2. Жаңа қосымша жарат және оны **07_02** папкаға сақта.
 - 2.1. Оның терезесін **Электронды сатып алу** деп ата.
 - 2.2. Қосымша терезесінде радио түймешіктер тобын жарат және оны **төлем түрі** деп ата.

- 2.3. Радио түймешіктер тізіміне төмендегі пункттерді енгіз:
PayNet; Click; UPay; Uzcard; Нақтылай; Ақша өткізу.
- 2.4. Қосымшаны іске қосып, төлем түрін өзгертіп көр.
- 2.5. Қосымшаны сақта және оны жап.
3. Жаңа қосымша жарат және оны **07_03** папкаға сақта.
- 3.1. Қосымша терезесін **Бітірушілер** деп ата.
- 3.2. Қосымша терезесіне радио түймешіктер тобы объектіін орнат және оны **Білім беру мекемесі** деп ата.
- 3.3. Радио түймешіктер тізіміне төмендегі пункттерді енгіз:
- **Жалпы білім беретін мектеп;**
 - **Академиялық лицей;**
 - **Кәсіптік-техникалық колледж.**
- 3.4. Қосымшаны іске қос және ондағы білім беру мекемесін таңда.
- 3.5. Қосымшаны сақта және оны жап.
4. Алдыңғы сабақта берілген жаттығуларды компьютерде орында.



55-САБАҚ. ListBox ЖӘНЕ ComboBox ОБЪЕКТІЛЕРІ

1. ListBox басқару объектісі және оның қасиеттері. **RadioGroup** объектінің бір ғана кемшілігі бар болып, ондағы варианттар тізімі ұзын болса, ол экранда көп орынды иелейді. Мұндай кемшілігі жоқ объектілер де бар. Олардың бірі – объектілер панелінің стандарт тармағында орналасқан **ListBox** (Тізім қорабы) басқару объекті:

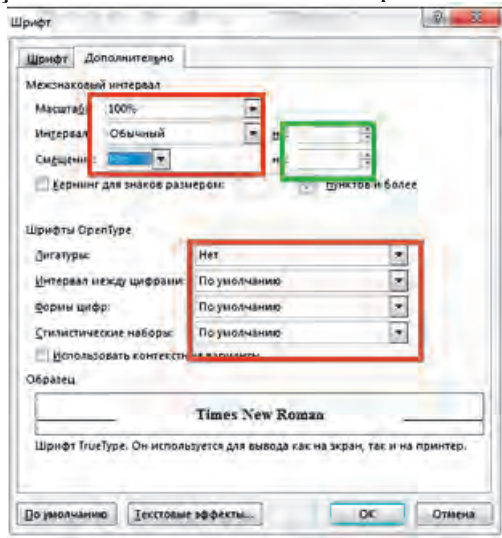


ListBox объекті экранда варианттар тізімін шығарады. Егер осы тізім ұзын болып, ол объектіге бөлінген орынға сыймаса, оның оң жағында вертикаль жылжығыш пайда болады. Оны жылжытып тізімнің қажетті жеріне өтуге болады.

ListBox объекті қасиетінің көбі **RadioGroup** объекті қасиеттерімен бірдей. Оның **Items** қасиетінің көмегімен тізім элементтері енгізіледі. **ItemIndex** қасиетінде тізімнің таңдалған элементінің реттік саны сақталады. Мұнда да реттік саны 0-ден басталады.

Columns қасиеті экранда неше баған болып шығуын көрсетеді. Егер оның мәні 0 болса, тізім бағандарға бөлінбейді. **ListBox**-тың

Columns қасиетінен тек өте қажет кезде ғана пайдалану керек. Өйткені бірнеше бағанға орналасқан тізім тізімге де ұқсамай қалады.



2. ComboBox басқару объекті және оның қасиеттері. **ComboBox** (жасырын тізім қорабы) объекті алдыңғылардан біршама қолайлы, оны көбірек пайдаланады. Оның қолайлылығы ол енгізу алаңы **Edit** сияқты аз орынды иелейді. Оның оң жағындағы үшбұрыш пішіндегі түймешікті басып, жасырын тізімді экранға шығаруға болады.

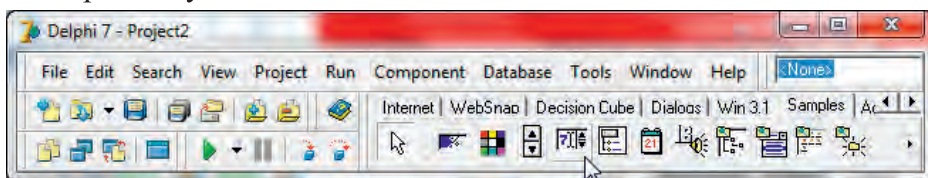
Жоғарыдағы суретте **Word** мәтін редакторының шрифт тілдесу терезесіндегі жасырын тізімдер қызыл жиекпен көрсетілген. Бұл объектінің де көп істетілетін қасиеттері **Items**, **ItemIndex** және **Column**. **ComboBox**-та тізім тармағын таңдау немесе жаңа тармақты енгізуге болады.

3. SpinEdit басқару объекті және оның қасиеттері. Алдыңғы қосымшаларда сандық шамаларды енгізу үшін **Edit** объектін пайдаланған едік. Оны пайдаланғанда енгізілген шаманы мәтін жолы түрінен бүтін сан жолы түріне өткізуге тура келген еді. Егер оған санды енгізуде қателікке жол берілсе, мысалы, 3-тің орнына “үшті” енгізсек, қосымша жұмысын тоқтатып, қате туралы хабарландырады.

SpinEdit (санды енгізу алаңы) объекті бүтін сандарды енгізу үшін арналған. Оған бұрыннан белгіленген аралықтағы бүтін санды енгізу

мүмкіндігі бар. Бұл қосымшалар үшін өте қолайлы. Жоғарыдағы суретте осы объектілердің екеуі жасыл жиекке алынған

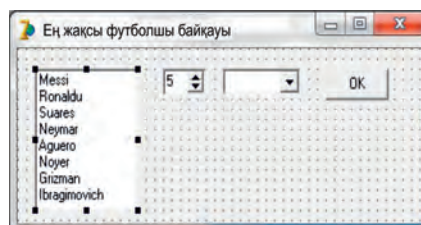
SpinEdit объекті объектілер панелінің **Samples** (Үлгілер) тармағына орналасқан. Бұл тармақ экранда көрінбейді. Оны көру үшін тармақтар тізімінің соңындағы үшбұрыш пішіндегі түймешіктердің екіншісін бірнеше рет басу қажет:



SpinEdit-те санды тікелей оның енгізу алаңына енгізуге болады. Ондағы санды енгізу алаңының оң жағындағы екі түймешіктің көмегімен бұрыннан белгіленген адыммен азайтуға немесе арттыруға болады.

SpinEdit-ке енгізілген санның мәні оның **Value** (мән) қасиетінде сақталады. Санның қабылдайтын ең кіші мәні **MinValue**, ең үлкен мәні **MaxValue** қасиеттерінде сақталады. Егер осы қасиеттердің мәні 0-ге тең болса, санның мәніне шекара қойылмайды. **Increment** (адым) қасиетінде сан мәнінің өсу адымы көрсетеледі. Бастапқыда ол 1-ге тең.

4. Ең жақсы футболшы қосымшасы. Жаңа объектілерді пайдалануды үйрену үшін ең жақсы футболшы байқауының қосымшасын жаратамыз. Жаңа қосымшаны жаратып, оның терезесіне біреу-біреуден **ListBox**, **ComboBox**, **SpinEdit** және **Button** объектілерін суреттегідей орнатамыз.



ListBox-тың **Items** қасиетін екі рет басып, пайда болған тілдесу терезесінде жоғарыдағы суретте көрсетілген 8 футболшының тізімін енгіземіз. **SpinEdit** объектіні таңдап, **Object Inspector** терезесінде оның **MaxValue** қасиетін 5-ке, **MinValue** қасиетін 1-ге, **Value** қасиетін 5-ке тең етіп орнатамыз.

Button1 объектіні екі рет басып, бағдарлама коды терезесін шығарып, **Button1Click** процедурасына төмендегі кодты енгіземіз:

```

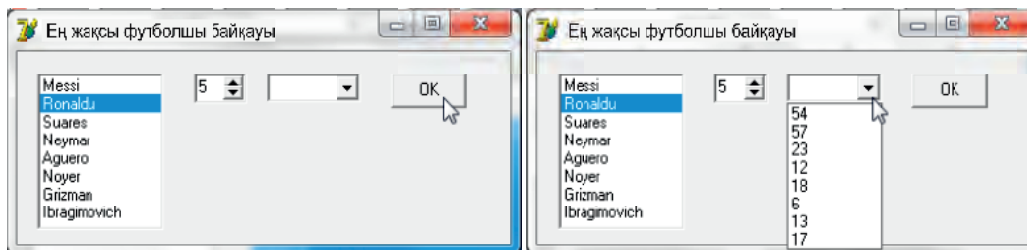
procedure TForm1.Button1Click(Sender: TObject);
var i,j:integer;
begin
i:=ListBox1.ItemIndex;    if i>=0 then begin
j:=StrToInt(ComboBox1.Items[i]);
ComboBox1.Items[i]:=IntToStr(j+SpinEdit1.Value);
end; end;

```

Процедураның бірінші, үшінші және жетінші жолдарының терезеде бар екенін ұмытпа. Екінші жолда екі жаңа айнымалы жаратылады. Төртінші жолда олардың біріншісі: **i**-ге таңдалған футболшының тізімдегі саны беріледі. Осы жолда осы санның теріс еместігі тексеріледі (кері жағдайда футболшы таңдалмаған болады). Алтыншы жолда екінші айнымалы: **j**-ға **ComboBox1** тізімінің **i**-жолындағы сан (бастапқыда ол 0-ге тең) беріледі. Алтыншы қатарда осы санға **SpinEdit1**-дегі сан (футболшыға берілген баға) қосылып, ол мәтін жолына айналады да, **ComboBox1** тізімінің **i**-жолына жазылады.

Нәтижеде **ComboBox1**-дегі тізімде әрбір футболшы топтаған балл жиналып барылады. Оны объектінің оң жағындағы үшбұрыш пішіндегі түймешікті басып, қалаған кезімізде көруімізге болады.

Қосымшаны **F9** түймешігін басып іске қосамыз. Ондағы бірінші тізімнен футболшыны таңдап, екінші алаңға оған берілген бағаны таңдаймыз және **OK** түймешікті басамыз (сол жақтағы сурет). Бірнеше рет футболшыларға дауыс беріп, нәтижені көру үшін үшінші алаңның оң жағындағы үшбұрыш түймешікті басамыз (оң жақтағы сурет).



ЕСІНДЕ САҚТА !

ComboBox объекті жасырын тізімді жаратуға қызмет етеді.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. **Listbox** объекті қандай міндетті атқарады?
2. **ComboBox** объектінің негізгі қасиеттерінің міндеттерін айт.
3. **SpinEdit** объектінің негізгі қасиеттерінің міндеттерін айт.
4. Қосымшаға **ListBox** объектіін орнат. Осы объектінің Асака зауытында жасалған автомашиналардың бірін тандайтын етіп қасиеттерін өзгерт.
5. Қосымшаға **ComboBox** объектіін орнат. Оның қасиеттерін өзгертіп, оған облыстардың тізімін енгіз.



ҮЙГЕ ТАПСЫРМА

1. **Listbox** объекті объектілер панелінің қай жерінде орналасқан?
2. **ComboBox** объекті қандай міндетті атқарады?



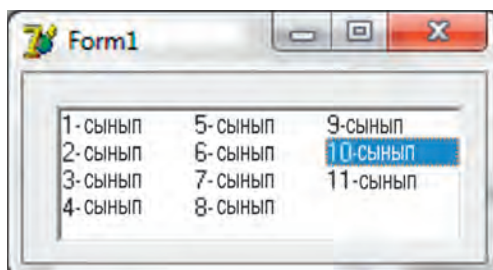
56-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

1. Алдыңғы практикалық жаттығулардағы **07_01**, **07_02** және **07_03** тапсырмаларын **ListBox** объектілерінің көмегімен жарат. Оларды сәйкесінше **08_01a**, **08_02a**, **08_03a** папкаларына сақта.
2. Алдыңғы практикалық жаттығулардағы **07_01**, **07_02** және **07_03** тапсырмаларын **ComboBox** объектілерінің көмегімен жарат. Оларды сәйкесінше **08_01b**, **08_02b** және **08_03b** папкаларына сақта.
3. Edit объектіне енгізілген сандарды **ListBox** объектіінде бейнелейтін қосымша жарат.
4. Edit объектіне енгізілген сандарды **ComboBox** объектіінде бейнелейтін қосымша жарат.
5. 1-ден 11-ге дейінгі сыныптарды **ListBox** объектіінде бейнелейтін қосымша жарат.
- 5.1. Жаңа қосымша жаратып, терезесіне **ListBox** объектіін орнат.
- 5.2. Қосымша терезесінің бос жерін екі рет басып, экранда пайда болатын процедураны төмендегідей етіп толықтыр:

```
procedure TForm1.FormCreate(Sender: TObject);
var
i:integer;
begin
listbox1.Columns:=3;
```

```
for i:=0 to 10 do
listBox1.Items[i]:=inttostr(i+1)+'-sinf';
end;
```

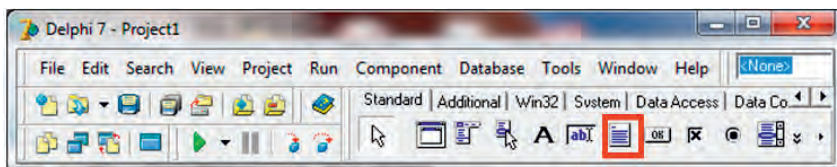
5.3. Қосымшаны іске қос және дұрыс жұмыс істеуін тексеріп көр:



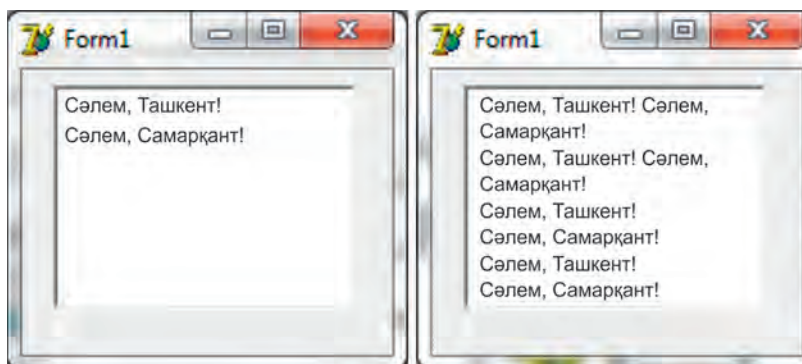
57-САБАҚ. Мемо БАСҚАРУ ОБЪЕКТИ

1. Мемо объекті және оның мүмкіндіктері. Кейбір қосымшаларда үлкен көлемдегі дерекпен жұмыс істеу мүмкіндігі де болуы кажет. **Delphi**-дің біз осы кезге дейін танысқан мүмкіндіктерінде ол жоқ. Енді біз мұндай мүмкіндікті жарататын басқару объектімен танысамыз.

Бұл объектінің аты **Мемо** болып, ол **Memorandum** сөзінің қысқартылған көрінісі және **қызмет үшін жазулар** деген мағынаны білдіреді. Ол объектілер панелінің стандарт тармағында орналасқан:



Delphi-де жаңа қосымша жаратып, оған осы объектіні орнатамыз. Қосымшаны іске қосып, тышқанды **Мемо**-ның үстіне бассaq, оның жұмыс алаңында меңзер пайда болады. Оған мәтін енгізуге болады:

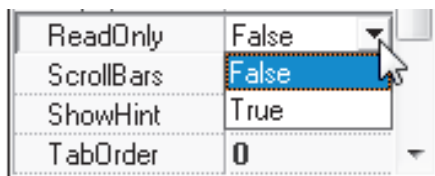


Оған мәтін енгізгенімізде жол толса, кейінгі жолдан мәтін енгізуді жалғастыруға болады. Терезе мәтінге толғанда мәтін жоғарыға қарай жылжып, жаңа қатар пайда болады. Мәтін енгізуді жалғастыру мүмкін.

Тышқанды жылжытып немесе меңзерді басқару түймешіктерінің көмегімен мәтіннің бөлігін белгілеп, оны **Ctrl+X**, **Ctrl+C**, **Ctrl+V** түймешіктерінің көмегімен көшірме жасауға, қиып алуға және мәтінге орналастыруға болады. Қысқаша айтқанда бұл объект қарапайым мәтін редакторы міндетін орындайды.

2. Memo объектінің негізгі қасиеттері. **Memo** объектіне қанша мәтін енгізуге болады? Оның шекарасы бар ма? Әрине, бар. Осы объектіге енгізілген мәтінде көп болса $2^{16}=65536$ әріп және басқа да белгілер болуы мүмкін. Бұл 1000-нан астам мәтін жолы дегені.

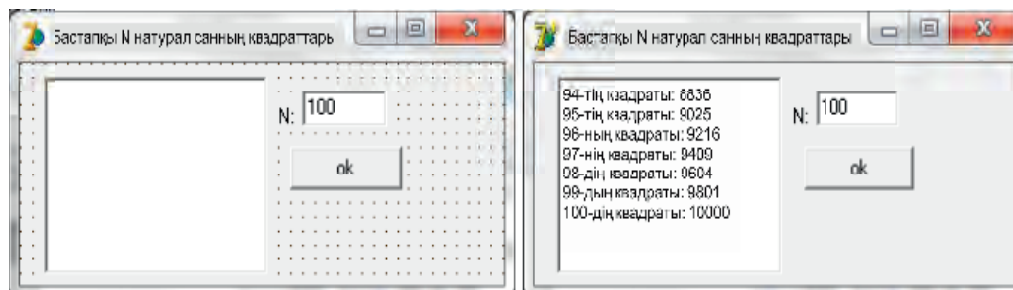
Бірақ, біз бұл объектіге мәтін енгізуді емес, оған деректерді шығаруды көздеген едік. Егер біз оған пайдаланушы дерек енгізе алмауын қалайтын болсақ, бұл объектінің **ReadOnly** (тек оқу үшін) атты қасиетін **False**-тен (жалған) **True**-ге (шын) өткізуіміз қажет:



Memo-ға енгізілген мәтін оның **Text** қасиетінде сақталады. Онда біреуден көп мәтін жолы болуы мүмкін. Мәтін жолдарын бірінен-бірін бөлу үшін олардың арасына жолдың аяғы мен басына өту белгілерін қою қажет болады. Мұны төмендегідей жүзеге асыруға болады:

```
Memol.Text:='1-satr'+#13+#10+'2-satr';
```


Оны пайдалануды үйрену үшін бастапқы n натурал санның квадраттарын табатын қосымша жаратамыз. Оған біреуден **Memo**, **Edit**, **Button** объектілерін орнатып, қасиеттерін төмендегі суретте көрсетілгендей етіп өзгертеміз:



Button түймешігін екі рет басып, экранда пайда болған процедураға төмедегілерді енгіземіз:

```
procedure TForm1.Button1Click(Sender: TObject);
var i,i2,n:integer;
begin
n:=StrToInt(Edit1.Text);
for i:=1 to n do begin i2:=i*i;
Memo1.Text:=Memo1.Text+IntToStr(i)+'ның_квадраты
: '+IntToStr(i2)+#13+#10;
end;
end;
```

Қосымшаны іске қосамыз және бастапқы 100, 1000 натурал санның квадраттарын табамыз.

Memo-ның **Text** қасиетімен істеу біршама қолайсыз, өйткені оның ұзындығы 65 мыңға дейін болуы мүмкін. Оның орнына **Lines** (жолдар) қасиетін пайдалануға болады. **Lines** мәтін жолдары массиві болып, осы массивтің қажетті элементін индексі арқылы пайдалануға болады. **Memo1.Lines[0]:='123'+Memo1.Lines[0]** мәтіні бірінші жолының (жолды нөмірлеу 0-мен басталады) басына 123-ті қосып қояды.

Lines негізі қасиет емес, объект болып, өзі қасиеті мен тәсілдеріне ие. Мысалы, **Memo1.Lines.Clear** барлық жолдарды өшіріп тастайды. **Memo1.Lines.Count** қасиеті арқылы мәтінде неше жол бар екенін біліп

алуға болады. `Memo1.Lines.Add('Соңғы жол')` мәтін соңына жаңа жол қосылады және оған **Соңғы жол** деген тіркес енгізіледі.

Жаңа жолды мәтіннің қалаған жеріне қосуға болады. Ол үшін `Memo1.Lines.Insert` тәсілі бар. Мысалы, `Memo1.Lines.Insert(4, 'жаңа 5-жол')`; операторы бесінші жолдан алдын тағы бір **жаңа 5-жол** деген жолды енгізеді. Келесі жолдар біреуге төменге жылжиды.

`Lines`-тің тағы екі пайдалы тәсілі бұл **SaveTo** және **LoadFrom** болып, олардың көмегімен объектіге мәтінді мәтін пішімінде файлға сақтап қоюға және сақталған мәтінді объектіге жүктеуге болады. Мысалы, `Memo1.Lines.SaveTo('d:\a1.txt')`; операторы **Memo**-ға енгізілген мәтінді **d:** дискідегі **a1.txt** атты файлға сақтап қояды.

`Memo1.Lines.LoadFrom('d:\a1.txt')`; операторы сақталған файлды объектіге жүктеп алады.

3. Жай сандар қосымшасы. 2017 жай сан ба? Егер ол жай сан болмаса, оған ең жақын жай санды табыңдар. Ескертіп өтеміз, жай сан деп, тек өзіне және 1-ге қалдықсыз бөлінетін натурал санға айтылады. Егер сан күрделі болса, оның бөлгіштерінен еш болмаса біреуі оның квадрат түбірінен кіші болады. Кері жағдайда барлық бөлгіштердің көбейтіндісі берілген сандардан үлкен болып кетеді. Демек, берілген сан n -ның жай сан екенін анықтау үшін оны $\sqrt{n}$ -ға болған сандарға бөлінбейтінін тексеру жеткілікті.

Осы мысалды шешу үшін n және одан кейінгі k натурал сандардың арасындағы жай сандардың барлығын табатын қосымша жаратамыз. Берілген параметрлерді енгізу үшін екі **Label**, екі **Edit**, бір **Button** және бір **Memo** объектілерін жаратылатын қосымша терезесіне жайғастырып, оның қасиеттерін бізге қажетті көріністе орнатамыз:



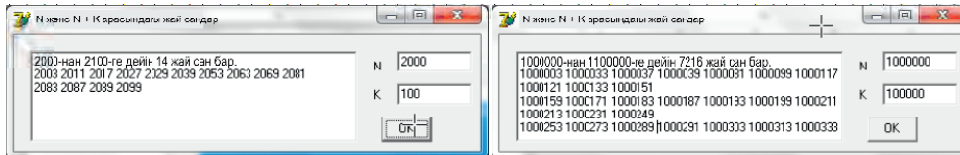
OK түймешігін екі рет басып, оған төмендегі кодты енгіземіз:

```
procedure TForm1.Button1Click(Sender: TObject);  
var i,j,n,k,a:integer;
```

```

b:boolean;
begin
n:=StrToInt(Edit1.Text);
k:=StrToInt(Edit2.Text);
for i:=n to n+k do begin
b:=true; a:=round(sqrt(i));j:=2;
while b and (j<=a) do
if i mod j = 0 then
b:=false else j:=j+1;
if b then Memo1.Lines.Add(IntToStr(i));
end; end;

```



ЕСІНДЕ САҚТА!

Memo объектінде бірнеше мың жол мәтін орналасуы мүмкін.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. **Memo** объекті объектілер панелінің қай жерінде орналасқан?
2. **Edit** алаңына енгізілген мәтін жолын **Memo**-ға әрбір әріпі жеке қатарға орналасатын етіп шығар.
3. **Memo**-ға 50 жолдан құралған мәтін енгіз. Олардың жұп нөмірлі жолдарын мәтіннен алып таста.



ҮЙГЕ ТАПСЫРМА

1. **Memo** объектінің **Text** қасиеті қандай міндетті атқарады?
2. **Memo** объектінің **Lines** элементі мәтінмен қалай жұмыс істейді?



58-САБАҚ. БАҚЫЛАУ ЖҰМЫСЫ

Төменде бақылау жұмысы варианттарының үлгілері келтірілген:

1-вариант

1. **CheckBox** объекті мен оның негізгі қасиеттері.

2. **Memo** объектіне мәтін жолдарын енгізу.
3. Бастапқы 10 тақ санды экранға шығаратын қосымша жарат.

2-вариант

1. **RadioGroup** объекті мен оның негізгі қасиеттері.
2. **Memo** объектінің **ReadOnly** қасиеті.
3. Бастапқы 15 жұп санды экранға шығаратын қосымша жарат.

3-вариант

1. **ListBox** объекті мен оның негізгі қасиеттері.
2. **Memo** объектінің **Text** қасиеті.
3. Бастапқы 8 7-ге еселі санды экранға шығаратын қосымша жарат.

4-вариант

1. **ComboBox** объекті мен оның негізгі қасиеттері.
2. **Memo** объектінің **Lines** қасиеті.
3. Бастапқы 12 5-ке еселі санды экранға шығаратын қосымша жарат.



59-САБАҚ. DELPHI-ДЕ ГРАФИКАМЕН ЖҰМЫС ІСТЕУ

1. **Delphi-дің графиктік мүмкіндіктері.** **Delphi**-де графикамен жұмыс істеу үшін кең мүмкіндіктер жаратылған. Бірқатар объектілердің **Canvas** (полотно, сырт) деген қасиеті бар болып, оның өзі объект болып саналады. Оның қасиеті мен тәсілдерін пайдаланып, осы объект иелеген жерде түрлі пішіндерді сызуға болады.

Мысалы, қосымша терезесіне тік төртбұрыш сызу үшін `Form1.Canvas.Rectangle(10, 20, 200, 100);` операторынан пайдаланамыз. `Listbox1.Canvas.Ellips(0, 10, 60, 50);` болса **ListBox1**-де жарты осьтері 30 бен 20-ға тең болған және ұштары (0, 10) мен (60, 50) нүктелерінде орналасқан тік төртбұрышқа іштей орналасқан эллипс сызылады.

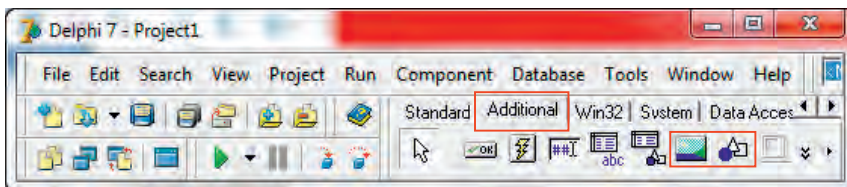
Осы пішіндерді экранда көру үшін жаңа қосымша жаратамыз. Оған **Button** және **ListBox** объектіін орнатамыз. **Button1** түймешігін екі рет басып, меңзер тұрған жерге жоғарыдағы екі операторды енгіземіз:

```
Form1.Canvas.Rectangle(10, 20, 200, 100);
ListBox1.Canvas.Ellips(0, 10, 60, 50);
```

Қосымшаны іске қосып, ондағы түймешікті бассaq, экранда төртбұрыш және эллипс пайда болады. Төмендегі суретте санақ жүйесінің басы ретінде объектінің жоғарғы сол бұрышы алынған. Алдымен горизонталь, соң вертикаль координаталар көрсетіледі:



2. Image және Shape объектілері. Delphi-де графикамен жұмыс істеу үшін арнаулы басқару объектілері бар. Олар **Image** (сурет, бейне) және **Shape** (пішін) болып, объектілер панелінің **Additional** (қосымша) тармағында орналасқан:



Осы объектілердің **Canvas** (полотно) объектілерінде қасиет, тәсіл мен “ата” объектілердің көмегімен графика жаратылады. **Canvas**-тың негізгі жұмыс құралдары **Pen** (қалам) мен **Brush** (қылқалам) болып, олардың біріншісі сызықтарды сызады. Екіншісі салаларды бояйды.

Pen-нің (қалам) қасиеттерінің негізгілері **Width** (ені), **Style** (стилі) және **Color** (түсі). `Image1.Canvas.Pen.Width:=5;` операторы қаламда сызылатын сызықтың кеңдігін 5 пикселге тең етіп орнатады.

Style қасиеті сызылатын сызықтардың көрінісін өзгертеді. Ол төмендегі мәні алдын ала анықталған өзгермейтіндердің біріне тең болуы мүмкін:

1. **psSolid** – үздіксіз (алдын ала орнатылған).
2. **psDash** – штрих (кіші кесінділер).
3. **psDot** – нүктелер.
4. **psDashDot** – кесінді және нүкте.
5. **psDashDotDot** – кесінді, нүкте және нүкте.
6. **psClear** – көрінбейтін.

Қаламның сыртын өзгертетін оператор

`Image1.Canvas.Pen.Style:=psDash;` сияқты болады.

Pen-нің (қалам) **Color** (түс) қасиеті сызылатын сызықтың түсін белгілеп береді. Осы қасиетте де алдын ала берілген өзгермейтіндерді пайдалануға болады. Мысалы, **clBlack** (қара, алдын ала орнатылған), **clWhite** (ақ), **clBlue** (көк), **clGreen** (жасыл), **clRed** (қызыл), **clYellow** (сары), **clGrey** (сұр), **clSilver** (күміс рең), **clAqua** (көгілдір) және тағы басқа оннан астам өзгермейтіндер бар.

`Image1.Canvas.Pen.color:=clRed;` операторы сызылатын сызықтың түсін қызылға өзгертеді. Delphi-де 16 миллионнан астам түстерді пайдалануға болады. Осы түстердің барлығы үш негізгі түс: **қызыл**, **жасыл** және **көк** түстерді қосу арқылы жаратылады.

Негізгі түстердің әрбіріне бір байт бөлінген. Осы үш байтта $2^{24}=16777216$ түрдегі түсті сақтауға болады. Түстердің реттік санын 16-лық санақ жүйесінде жазу қолайлы. Осы санақ жүйесінде үш байтқа 6 бөлмелі сан сәйкес келеді. Олардан ең кіші екі бөлме қызыл, ортадағы екі бөлме жасыл және ең үлкен екі бөлме көк түске сәйкес келеді: **\$BBGRR** (мұнда **BB** көк, **GG** жасыл, **RR** қызыл түске сәйкес сандар).

Мысалы, көк **\$FF0000**, жасыл **\$00FF00**, қызыл **\$0000FF**, сары **\$00FFFF**, қызғылт **\$FF00FF** он алтылық санмен өрнектеледі. Егер он алтылық санақ жүйесінде жұмыс істеу қолайсыз болса, онда **RGB(RR, GG, BB)** стандарт функцияны пайдалануға болады.

Бұл функцияның үш аргументі болып, олардың біріншісі **қызыл**, екіншісі **жасыл**, үшіншісі **көк** түстерге сәйкес келеді. Осы аргументтердің мәндері 0-ден 255-ке дейін болуы мүмкін. Мысалы, **RGB(0, 0, 0)** қара түсті, **RGB(255, 255, 255)** ақ түсті, **RGB(0, 255, 255)** **көгілдір** түсті, **RGB(127, 0, 0)** **тоқ қызыл** түсті, **RGB(0, 127, 0)** **тоқ жасыл** түсті, **RGB(200, 200, 0)** **тоқ сары** түсті береді.

Brush-тың (қылқалам) түсі де осылай анықталады. Оның стилін төмендегі өзгермейтіндер арқылы өзгертуге болады:

1. **bsSolid** – үздіксіз (бір тегіс).
2. **bsClear** – көрінбейтін.
3. **bsBDiagonal** – жанама диагональ бойынша.
4. **bsFDiagonal** – негізгі диагональ бойынша.
5. **bsCross** – тор көз сияқты.

6. **bsDiagCross** – диагональ бойынша тор көз.

7. **bsHorizontal** – горизонталь.

8. **bsVertical** – вертикаль.

3. Pixels (пикселдер) қасиеті. **Canvas**-тың **Pixels** қасиетінің көмегімен сурет сызылатын аумақтағы әрбір нүктені басқаруға болады. Бұл қасиетте пикселдер екі өлшемді массивті құрайды. 50-жолдың 100-бағанындағы пикселді пайдалану үшін **Pixels[100,50]** деп оның екі индексін көрсету жеткілікті. Егер біз осы пикселдің түсін қызылға өзгертуді қаласақ, **Image1.Canvas.Pixels[100,50] := RGB(255, 0, 0);** операторын пайдаланамыз.

Ал **PC:=Image1.Canvas.Pixels[100,50];** операторы пикселдің түсін **PC** өзгерушісіне сақтау мүмкіндігін тудырады.

4. MoveTo және LineTo тәсілдері. Сызба сызуда тік сызық кесіндісін сызу көп кездеседі. Осы ретте **Delphi** қандай мүмкіндікке ие? **Canvas** объектінің **MoveTo** (...-ға өт) және **LineTo** (...-ға дейін сыз) тәсілдері кесіндіні сызуға қызмет етеді. Олардың екеуі де екеуден бүтін мәнді аргументке ие. Осы аргументтердің көмегімен суреттегі нүктелердің координаталары көрсетіледі.

Ұштары (x1, y1) мен (x2, y2) нүктелердегі кесіндіні сызуға алдын **MoveTo(x1,y1)** тәсілімен кесіндінің бірінші ұшына өтіліп, **Lineto(x2,y2)** тәсілімен екінші ұшына дейінгі кесінді сызылады. Мұнда графикалық меңзер (x2, y2) нүктеге өтеді. Егер тағы бір рет **Lineto(x3,y3)** тәсілі қолданылса, ұштары (x2, y2) мен (x3, y3) нүктелеріндегі кесінді сызылып, графикалық меңзер (x3, y3) нүктеге өтеді.

Бұл тәсілдердің қолданылуын **Тұрақты N бұрыш диагональдары** деп аталған қосымшада қарастырамыз. **Delphi**-де жаңа қосымша жаратып, оған біреуден **Image**, **SpinEdit** және **Button** объектілерін орнатамыз.

Button-ның тақырыпшасын **chiz**-ге, **SpinEdit** тақырыпшасын **N**-ге өзгертеміз. **SpinEdit**-тің ең үлкен мәнін (**MaxValue**) 40-қа, ең кіші мәнін (**MinValue**) 3-ке, ағымдағы мәнін (**Value**) 7-ге тең етіп аламыз. **Image**-дің биіктігі (**Height**) мен енін (**Width**) 500-ге тең етіп аламыз.

Button түймешігін екі рет басып, оған диагональдар командасын енгіземіз. Мұндай команда қазірше жоқ, сондықтан осы аттағы процедураны **TForm1.Button1Click** процедурасынан бұрын енгіземіз:


```

procedure diagonallar;
var i,j,n,r:integer;
x,y: array [0..40] of integer; begin
r:=Form1.Imagel.Height div 2;
Form1.Imagel.Canvas.Pen.Color:=clRed;
Form1.Imagel.Canvas.Rectangle(-1,-1,2*r+2,2*r+2);
n:=form1.SpinEdit1.Value;
for i:=1 to n do begin
y[i]:=round(r-r*cos(2*pi/n*i));
x[i]:=round(r+r*sin(2*pi/n*i)); end;
for i:=1 to n-1 do for j:=i+1 to n do begin
Form1.Imagel.Canvas.MoveTo(x[i],y[i]);
Form1.Imagel.Canvas.LineTo(x[j],y[j]); end; end;

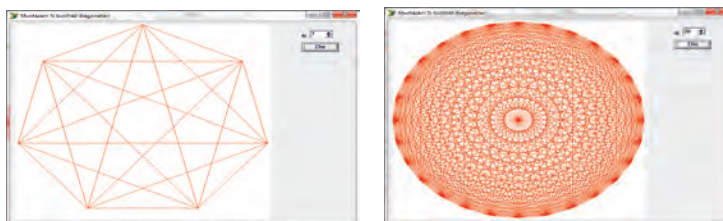
```

Процедураға алты айнымалыны енгіземіз. i мен j айнымалылар циклін басқаруға пайдаланылады. n айнымалыда көпбұрыштың бұрыштарының саны, r айнымалысында тұрақты көпбұрышқа сырттай сызылған шеңбер радиусы сақталады. x пен y массивтерінде көпбұрыш ұштарының координаталары сақталады.

Процедураның денесіндегі операторларды үш бөлікке бөлуге болады. Бірінші бөлікте төрт оператор болып, олардың біріншісі шеңбер радиусын сурет биіктігінің жартысына тең етіп алады. Екінші оператор сызықтардың қызыл түсте сызылуын қамтамасыз етеді. Үшінші оператор сызудан алдын суретті тазалайды. Төртінші оператор көпбұрыштың неше бұрышы болуын анықтайды.

Екінші бөлікте тек бір ғана оператор бар. Онда көпбұрыш ұштарының координаталары: $x[i]$ және $y[i]$ анықталады.

Үшінші бөлікте де бір іштей орналасқан цикл операторы болып, ол көпбұрыштың жақтары мен диагональдарын сызады. Қосымшаны іске қосамыз. Алдымен тұрақты жеті бұрышты, соң тұрақты 28 бұрышты сызамыз. Олар төмендегі суретте көрсетілген:





ЕСІНДЕ САҚТА !

Canvas элементі көптеген объектілерде бар және онда сурет сызуға болады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Графикамен жұмыс істеу қалай орындалады?
2. **Image** объекті объектілер панелінің қай жерінде орналасқан?
3. **RGB** функциясы қандай міндетті атқарады?
4. Экранға қызыл, жасыл және қара түсті үш тік төртбұрыш сызылған қосымша жарат.
5. Экранда көк, күмісreң және сары түсті үш эллипс сызатын қосымша жарат.



ҮЙГЕ ТАПСЫРМА

1. Қалам объектінің негізгі қасиеттерін айт.
2. Қылқалам объектінің қандай қасиеттері бар?



60-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУ

1. Төмендегі графиктік міндетті атқаратын қосымшаларды жарат.
 - 1.1. Экранға іштей орналасқан 10 тік төртбұрышты шығаратын қосымша жарат.
 - 1.2. Экранға іштей орналасқан және төменгі сол бұрышы өзгермейтін 12 квадрат шығаратын қосымша жарат.
 - 1.3. Экранға іштей орналасқан және 10 тік бұрышты үшбұрышты шығаратын қосымша жарат.
 - 1.4. Экранға іштей орналасқан 15 және орталықтары бір нүктеде болған 15 шеңбер шығаратын қосымша жарат.
 - 1.5. Экранға іштей орналасқан және орталықтары бір нүктеде болған 8 эллипс шығаратын қосымша жарат.
 - 1.6. Экранға іштей орналасқан және сан алуан түстегі 5 тік төртбұрышты шығаратын қосымша жарат.
 - 1.7. Экранға іштей орналасқан және түрлі қалыңдықтағы сызықтармен сызылған 5 шеңбер шығаратын қосымша жарат.

1.8. Экранға түрлі тәсілдегі сызықтармен сызылған 5 эллипс сызатын қосымша жарат.

1.9. Экранға іштей орналасқан және жоғарғы оң бұрышы өзгермейтін 11 тік төртбұрыш шығаратын қосымша жарат.

1.10. Экранға іштей орналасқан және ең жоғарғы нүктесі ортақ болған 9 шеңбер шығаратын қосымша жарат.

1.11. Экранға іштей орналасқан және ең сол жақтағы нүктесі ортақ болған 8 эллипс шығаратын қосымша жарат.

1.12. Экранға іштей орналасқан және жоғарғы сол бұрышы өзгермейтін 13 тік төртбұрыш шығаратын қосымша жарат.

1.13. Экранға іштей орналасқан және ең төменгі нүктесі ортақ болған 7 шеңбер шығаратын қосымша жарат.

1.14. Экранға іштей орналасқан және ең оң жақтағы нүктесі ортақ болған 14 эллипс шығаратын қосымша жарат.

1.15. Экранға іштей орналасқан және төменгі оң бұрышы өзгермейтін 16 квадрат шығаратын қосымша жарат.

2. Жаңа қосымша жарат және оған графиктік пішіндер деп ат қой.

2.1. Оған бір басқару түймешігі мен үш радиотүймешіктер тобын орнат.

2.2. Бірінші радио түймешіктер тобы экранға қандай пішін сызылуын анықтау үшін қызмет етсін. Мүмкін болған варианттар: квадрат, тік төртбұрыш, шеңбер және эллипс.

2.3. Екінші радиотүймешіктер тобы пішіндер қай жақтан тегістелуін белгілесін. Мүмкін болған варианттар: ортадан, оңнан, жоғарыдан, төменнен, жоғарғы сол, төменгі сол, жоғарғы оң, төменгі оң бұрыштар.

2.4. Үшінші радиотүймешіктер тобы сызықтардың түсін анықтасын. Мүмкін болған варианттар: қызыл, жасыл, сары, көгілдір және қара.

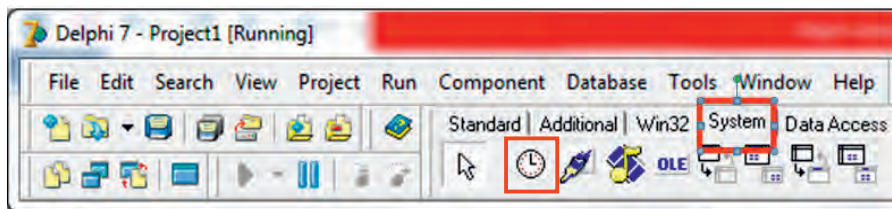
2.5. Басқару түймешігі басылғанда таңдалған геометриялық пішін, таңдалған түсте, таңдалған тегістеу тәсілімен іштей орналасқан түрде экранға шығарылсын.



61-САБАҚ. Timer ОБЪЕКТІ ЖӘНЕ ОНЫ ПАЙДАЛАНУ

1. Timer объектінің қасиеттері. Кейде қосымшада уақытқа байланысты амалдарды орындауға тура келеді. Мысалы, қосымшада уақытты көрсететін сағат болса, әр секундта көрсетіліп жатқан уақытты жаңалап тұру қажет. Кері жағдайда сағаттағы уақыт артта қалып кетеді. Тест бағдарламасында тапсырманы орындау үшін белгілі бір уақыт бөлінеді. Осы уақыт аяқталса, тест тапсыру да аяқталады. Презентацияда әр он (немесе жиырма) секундтан соң жаңа парақ экранға шығады. Анимациялық қосымшаларда қозғалып жатқан пішіннің жағдайы секундына бірнеше рет жаңаланып тұрады.

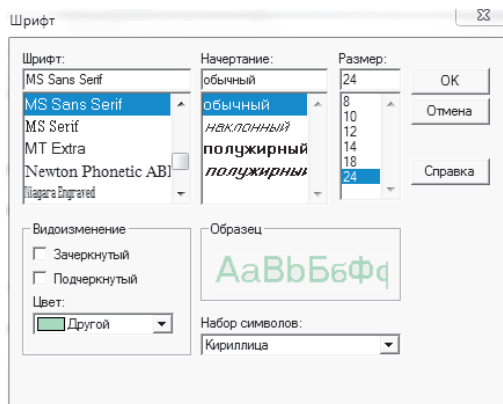
Осы мысалдың барлығында **Timer** объекті пайдаланылады. **Timer** объектінің ерекшелігі оның экранда көрінбеуі. Соған қарамастан, егер **Timer** белсенділендірілген болса, ол берілген уақыт аралығының өтуін күтеді. Осы уақыт аралығы өткен соң, қосымша өз жұмысын уақытша тоқтатып, **Timer**-дің арнаулы процедурасын орындайды. Одан соң, қосымша тағы өз жұмысын жалғастырады. **Timer** объекті объектілер панелінің **System** (Жүйе) тармағында орналасқан:



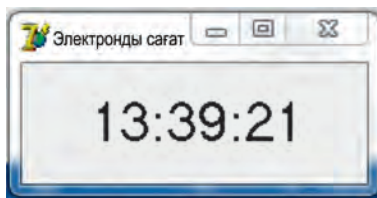
Timer объектінің тек екі қасиеті бар. Олардың біріншісі бізге таныс болған **Enabled** (белсенді). Бұл қасиет шын немесе жалған мәнді қабылдайды. Оның мәні **True** (шын) болуымен **Timer** іске қосылады. Мәні **False** болуымен **Timer** өз жұмысын тоқтатады. Бастапқыда бұл қасиеттің мәні **True**-ге тең болады.

Timer-дің екінші қасиеті **Interval** болып, онда **Timer** келесі ретте қанша уақыттан соң іске түсуін көрсетеді. Осы уақыт миллисекундтарда өлшенеді. Егер **Timer** әр екі секундте бір реттен іске түсуі керек болса, интервал қасиеті 2000-ға тең етіп орнатылады. Бастапқыда оның мәні 1000-ға тең болады.

2. Электронды сағат және Timer қосымшасы. Электронды сағат қосымшасын жаратамыз. Жаңа қосымша жаратып, оның тақырыпшасын электронды сағатқа өзгертеміз. Қосымша терезесіне екі объект: **Label** мен **Timer**-ді орнатамыз. **Label**-дің **Font** (шрифт) қасиетін екі рет басып, шрифт тілдесу терезесіне шрифт өлшемін 24-ке тең етіп орнатамыз:

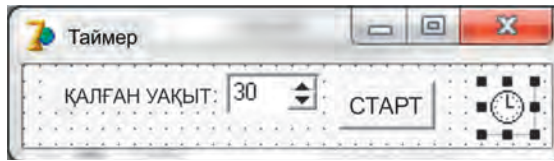


Timer объектін екі рет басып, оның процедурасына **Label1.Caption:=TimeToStr(Now);** операторын енгіземіз. Осы операторда екі стандарт функция пайдаланылады. Олардың біріншісі **Now**-дің (қазір) мәні компьютердің сағаты осы функция іске қосылған кездегі уақытына тең:



Екінші функция **TimeToStr** (**Time to String** уақыт түрінен мәтін жолы түріне деген мағынаны білдіреді) болып, оның мәні мәтін жолы түрінде болады. Осы мәнді **Label1** объектiнің **Caption** қасиетіне беруге болады.

3. Timer қосымшасы. Жаңа қосымша жаратып, оның терезесіне **Label**, **Button**, **SpinEdit** және **Timer** объектілерін орнатамыз. **Label** мен **Button** тақырыпшаларын **ҚАЛҒАН УАҚЫТ:** пен **СТАРТ-қа** өзгертеміз.



SpinEdit-тің қасиеттерін төмендегідей етіп орнатамыз:

```
SpinEdit1.MaxValue:=100;
SpinEdit1.Value:=30;
SpinEdit1.Enabled:=False;
```

Button түймешігін екі рет басып, оның процедурасына **Timer1.Enabled:=True**; операторын енгіземіз. **Timer**-ді екі рет басып, **TForm1.Timer1.Timer** процедурасына бағдарлама кодын енгіземіз:

```
SpinEdit1.Value:=SpinEdit1.Value-1;
if SpinEdit1.Value=0 then
begin
Timer1.Enabled:=False;
ShowMessage('УАҚЫТ БИТТИ. ');
Form1.Close;
end;
```

Бұл процедура іске қосылса, бірінші жолдағы оператор **SpinEdit**-тің мәнін біреуге азайтады. Екінші жолдағы оператор **SpinEdit**-тің мәні 0-ге тең болғанын тексереді. Егер 0 болса, төртінші жолдағы оператор **Timer**-дің жұмысын тоқтатады, бесінші жолда экранға тілдесу терезесі шығарылады. Осы терезе жабылуымен алтыншы жолдағы оператор қосымша терезесін жабады және қосымша өз жұмысын аяқтайды.



Қосымша іске қосылғанда **Timer**-дің уақытын **SpinEdit**-те орнатамыз және **Start** түймешігін басып, **Timer**-ді де іске қосамыз. **Timer**-дегі уақыт азайып барады және аяқталуымен экранға тілдесу терезесі шығады. Оны жапқанда қосымша да өз жұмысын аяқтайды.

4. Көпбұрыш диагональдары қосымшасын автоматтандыру.

Алдыңғы сабақта жаратылған тұрақты N бұрышының диагональдары

атты қосымшаның жұмысын автоматтандырамыз. Ондағы түймешікті басқанда қосымша автоматты түрде жұмыс істеу тәртібіне өтсін және әр секундта көпбұрыштың бұрыштары санын біреуге арттырып, оның барлық диагональдарын сызып шықсын. Түймешік тағы бір басылса автоматты жұмыс істеу тәртібінен шығып, күту тәртібіне өтсін. Түймешік тағы бір рет басылса қосымша өз жұмысын жалғастырсын.

Қосымшаны **Delphi**-ге қайта жүктеп, оған **Timer** объектіін орнатамыз. Оның **Enabled** қасиетінің мәнін жалғанға өткіземіз. **Button1** түймешігінің тақырыпшасын **Start**-қа өзгертеміз. Осы түймешікті екі рет басып, пайда болған процедураға мына екі операторды енгіземіз:

```
Form1.Timer1.Enabled:=not Form1.Timer1.Enabled;  
if Form1.Timer1.Enabled=True then  
Form1.Button1.Caption:='Stop' else  
Form1.Button1.Caption:='Start'
```

Бірінші жолдағы оператор **Timer1**-дің белсенділігін **Enabled** қасиетінің көмегімен керісінше өзгертсе, екінші оператор түймешіктегі жазуды **Enabled** мәніне қарап **Stop** немесе **Start** етіп өзгертеді.

Timer1-ді екі рет басып, процедурасына үш операторды енгіземіз:

```
Form1.SpinEdit1.Value:=form1.SpinEdit1.Value+1;  
if Form1.SpinEdit1.Value>40  
then Form1.SpinEdit1.Value:=3; diagonallar;
```

Осы операторлардың біріншісі **SpinEdit1**-дің мәнін біреуге арттырады. Келесі оператор осы мән 40-тан артып кетсе, оны 3-ке тең етіп қайта анықтайды. Үшінші оператор берілген көпбұрыштың жақтары мен диагональдарын сызып шығады.

Қосымшаны іске қосып, ондағы түймешікті басамыз. Экрандағы көпбұрыштардың диагональдарын сызылуын көреміз. Қажет кезде түймешікті тағы бір рет басып, қосымшаның жұмысын тоқтатып тұрамыз және экрандағы көпбұрыш диагональдарын толыққанды көріп шығамыз. Соң түймешікті тағы да басып, келесі көпбұрыштардың сызылуын тіктейміз.



ЕСІНДЕ САҚТА !

Анимацияларды жарату үшін **Timer** объектіі пайдаланылады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. **Timer** пайдаланылатын жағдайларға мысалдар айт.
2. **Timer** объекті объектілер панелінің қай тармағында орналасқан?
3. **Timer** объектінің интервал қасиеті қандай міндетті атқарады?
4. **Timer** объектінің **Enabled** қасиеті қандай мәнді қабылдайды және осы мән өзгерсе қандай оқиға болады?
5. **Timer** қосымшасында **ShowMessage** процедурасының орнына **Label** объекті пайдаланып, қайта жарат.
6. Электронды сағат қосымшасын әр секундта қысқа дыбыстық сигнал беретін етіп өзгерт.



ҮЙГЕ ТАПСЫРМА

1. **Timer** объектінің неше қасиеті бар?



62-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

1. Алдыңғы практикалық жаттығулардың тапсырмаларындағы қосымшаларға **Timer** объекті орнат және геометриялық пішіндерді 2 секунд интервалмен экранға шығар.
2. **Button** мен **Label**-ді пайдаланып сағат бағдарламасын жаса.
3. Қосымша терезесінің ені 100-ден 800-ге дейін секундiне 20 рет 1 адыммен өзгертiн қосымша жарат.
4. Бағдаршам қосымшасын жарат. Қызыл және жасыл шамдар 5, сары шам 1 секунд жанып тұрсын. Шамдар өшкенде күлреңге өтсiн.
 - 4.1. Қосымша терезесiн жарат және оған **Timer** объекті орнат.
 - 4.2. Қосымша коды терезесіне өтіп, **var Form1:TForm1;** операторын тап және одан соң **var i:integer;** операторын енгіз;
 - 4.3. **Timer** объекті екі рет басып оның процедурасына төмендегі операторларды енгіз:

```
procedure TForm1.Timer1Timer(Sender: TObject);
begin
  i:=i+1;
  if i mod 12 =1 then
  begin light_off;
  canvas.brush.Color:=clRed;
  canvas.Ellipse(20,20,70,70);
```

```

end;
if i mod 6 =0 then
begin light_off;
canvas.brush.Color:=clYellow;
canvas.Ellipse(80,20,130,70);end;
if i mod 12 =7 then
begin light_off;
canvas.brush.Color:=clGreen;
canvas.Ellipse(140,20,190,70);
end; end;

```

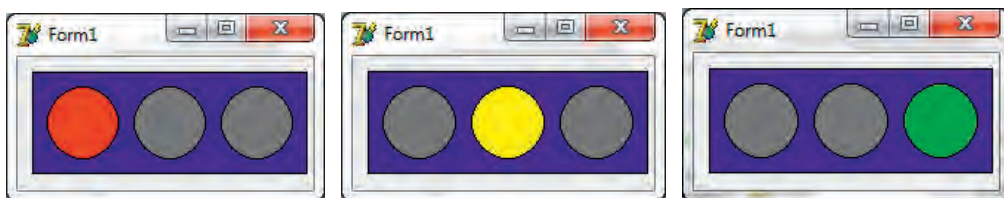
4.4. Осы процедурадан алдын `light_off` процедурасын енгіз:

```

procedure light_off;
begin
form1.canvas.brush.Color:=clBlue;
form1.canvas.Rectangle(10,10,200,80);
form1.canvas.brush.Color:=clGray;
form1.canvas.Ellipse(20,20,70,70);
form1.canvas.Ellipse(80,20,130,70);
form1.canvas.Ellipse(140,20,190,70);
end;

```

4.5. Қосымшаны іске қосып, оның жұмысын бақыла:



63-САБАҚ. СУРЕТКЕ БАСҚА ОБЪЕКТІЛЕРДІ ОРНАЛАСТЫРУ

1. Суретке мәтін орналастыру. Суреттермен жұмыс істегенде көп кездесетін амалдардың бірі – осы суретке мәтін енгізу. Осы амалды орындау үшін **Canvas**-тың тәсілі болған **TextOut** (мәтін шығару) пайдаланылады. Оның үш аргументі болып, олардың бастапқы екісі мәтін шығарылатын аумақтың жоғарғы сол бұрышының координаталары, үшіншісі экранға шығарылатын мәтін жолы.

Мысалы, `Image1.Canvas.TextOut(10,20, 'Сәлем!');` операторы сурет аумағының (10, 20) нүктесінен бастап, 'Сәлем!' деген жазу шығарады.

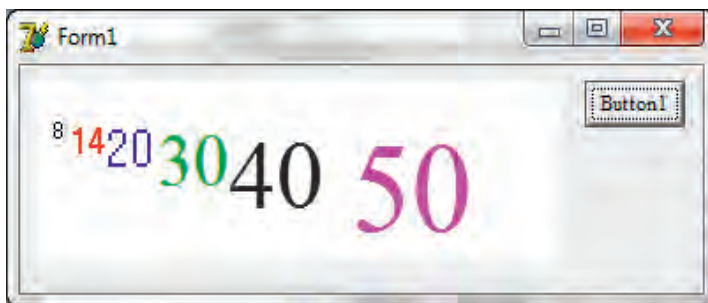
Суретке қойылатын жазудың қандай шрифтте шығуын `Canvas`-тың `Font` (шрифт) қасиеті белгілейді. Мысалы, шрифттің өлшемін 14 ету үшін `Image1.Canvas.Font.Size:=14;`, шрифт түсін қызыл ету үшін `Image1.Canvas.Font.Color:=clRed;` операторы пайдаланылады.

Осы операторларды қолданумен танысу үшін төмендегі қосымшаны жаратамыз. Оған `Image` және `Button` түймешіктерін орнатып, түймешікті екі рет басып, мына операторларды енгіземіз:

```
Image1.Canvas.Font.Size:=8;  
Image1.Canvas.TextOut(10,20, '8');  
Image1.Canvas.Font.Size:=14;  
Image1.Canvas.Font.Color:=clRed;  
Image1.Canvas.TextOut(20,20, '14');  
Image1.Canvas.Font.Size:=20;  
Image1.Canvas.Font.Color:=clBlue;  
Image1.Canvas.TextOut(40,20, '20');  
Image1.Canvas.Font.Name:='Times New Roman';  
Image1.Canvas.Font.Size:=30;  
Image1.Canvas.Font.Color:=clGreen;  
Image1.Canvas.TextOut(70,20, '30');  
Image1.Canvas.Font.Size:=40;  
Image1.Canvas.Font.Color:=clBlack;  
Image1.Canvas.TextOut(110,20, '40');  
Image1.Canvas.Font.Size:=50;  
Image1.Canvas.Font.Color:=RGB(255,0,255);  
Image1.Canvas.TextOut(180,20, '50');
```

Қосымшаны іске қосып, оның түймешігін басамыз.

Шрифтті өзіміз білген `Times New Roman`-ға ауыстырғанымыздан соң экрандағы мәтін сапасының жақсаруына назар аудар.

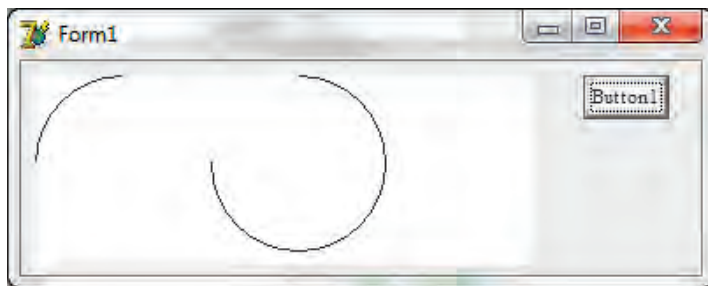


2. Суретке пішіндерді орналастыру. Суретке бірқатар геометриялық пішіндерді орналастыруға болады. Олардан тік төртбұрыш, эллипс, кесінділерді қалай орналастыруды алдыңғы сабақтарда қарастырған едік. Енді оларға басқа пішіндерді де қосамыз.

Эллипс доғасын суретке қою үшін `arc(x1,y1,x2,y2, x3,y3,x4,y4)` тәсілі пайдаланылады. Эллипстің ұштары $(x1, y1)$, $(x2, y2)$ нүктелерде болған тік төртбұрышқа іштей сызылған және оның орталығын $(x3, y3)$, $(x4, y4)$ нүктелермен біріктіретін тура сызықтармен қиылысатын бөлігі алынады. Мұнда үшінші нүктеден төртінші нүктеге сағат тіліне кері бағытта қозғалады.

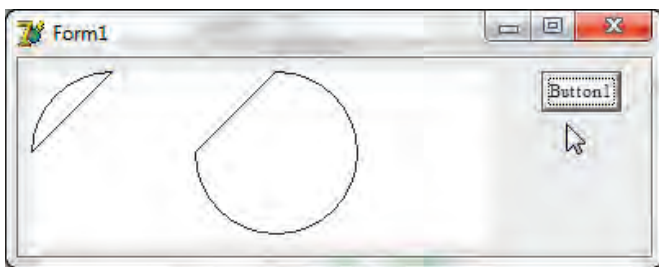
Эллипс доғасын сызатын операторлар және олар сызған доғалар төменде көрсетілген.

```
Image1.Canvas.Arc(0,0,100,100, 50,0,0,50);
Image1.Canvas.Arc(100,0,200,100, 0,50,150,0);
```

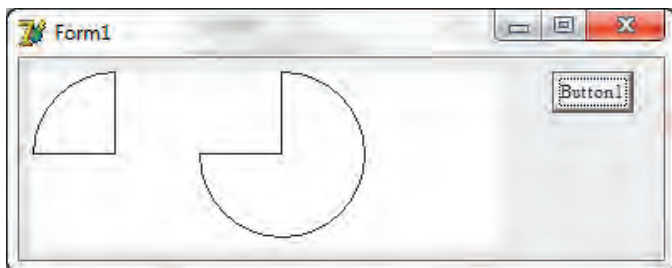


Эллипс сегментін суретке қосу үшін `Cord(x1,y1,x2,y2, x3,y3,x4,y4)` тәсілі пайдаланылады. Оған мысал төменде көрсетілген:

```
Image1.Canvas.Chord(0,0,100,100, 50,0,0,50);
Image1.Canvas.Chord(100,0,200,100, 0,50,150,0);
```



Эллипс секторын суретке қосу үшін $\text{Pie}(x1,y1,x2,y2, x3,y3,x4,y4)$ тәсілі пайдаланылады. Оған мысал төменде көрсетілген:



```
Image1.Canvas.pie(0,0,100,100, 50,0,0,50);
Image1.Canvas.pie(100,0,200,100, 0,50,150,0);
```



ЕСІНДЕ САҚТА !

Суретке мәтін мен геометриялық пішін орналастыруға болады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Суретке эллипс сегменті қалай орналастырылады?
2. Суретке эллипс секторы қалай орналастырылады?
3. Суретке бірнеше мәтін жолын енгізетін қосымша жарат.
4. Суретке бірнеше геометриялық пішін енгізетін қосымша жарат.



ҮЙГЕ ТАПСЫРМА

1. Суретке мәтін қалай орналастырылады?
2. Суретке эллипс доғасы қалай орналастырылады?



64-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

1. Жаңа қосымша жарат. Онда координата жүйесін сыз. Координата осьтерінің аттарын және ондағы сандарды суретке енгіз.

2. Жаңа қосымша жарат. Оған компьютер файлына орналасқан суретті жүкте. Сурет файлының атын суреттің оң бұрышына орнат.

3. Жаңа қосымша жарат. Оған бірнеше түстегі эллипс доғаларын орналастыр.

4. Жаңа қосымша жарат. Оған бірнеше түрдегі шеңбер доғаларын орналастыр.

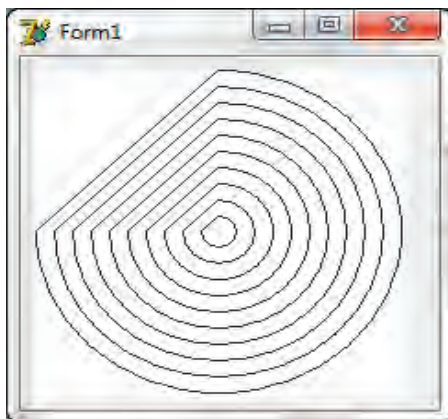
5. Жаңа қосымша жарат. Оған іштей орналасқан 10 эллипс сегментін орналастыр.

5.1. Қосымша терезесіне **Image** объектіні орналастыр.

5.2. Қосымша терезесінің бос жеріне тышқанды екі рет басып, төмендегі процедураны енгіз.

```
procedure TForm1.FormCreate(Sender: TObject);  
var i:integer;  
begin  
  for i:=10 downto 1 do  
    image1.Canvas.Chord(100-10*i,100-  
      10*i,100+10*i,100+10*i, 0,100,100,0);  
  end;
```

5.3. Қосымшаны іске қос.



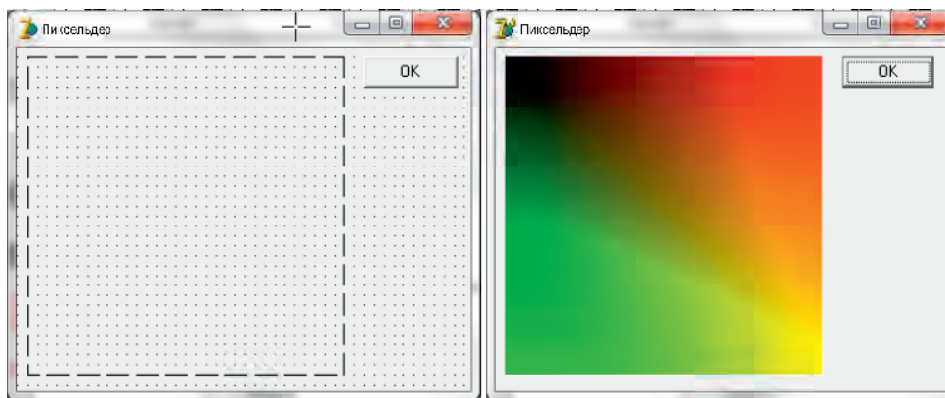


65-САБАҚ. ГРАФИКАЛЫҚ ЖӘНЕ АНИМАЦИЯЛЫҚ ҚОСЫМШАЛАРҒА МЫСАЛДАР

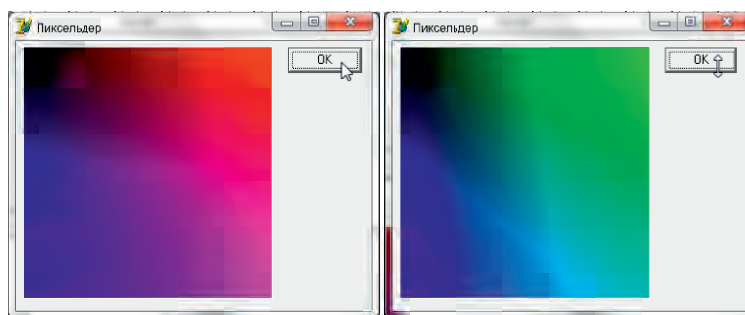
1. Рендер палитрасы. Пикселдер қасиетінің көмегімен қызыл және жасыл түстерді түрлі мөлшерде араластырумен пайда болған 65536 түсті пикселдерді экранда бейнелейтін қосымша жаратамыз. Қосымшаға **Image** (256×256) және **Button** объектілерін төмендегі суретте көрсетілгеніндей орналастырамыз. **Button** түймешігін екі рет басып, бағдарлама кодына төмендегі процедураны енгіземіз:

```
procedure TForm1.Button1Click(Sender: TObject);  
var i,j,түсі:integer;  
begin  
  for i:=0 to 255 do  
    for j:=0 to 255 do begin  
      түсі:=RGB(i,j,0);  
      Image1.Canvas.Pixels[i,j]:=түсі;  
    end;  
  end;
```

Қосымшаны іске қосып, **қызыл** және **жасыл** түстердің араласып, **сары** түстің пайда болғанын көреміз:



Қосымшадағы `түсі:=RGB(i,j,0);` операторын `түсі:=RGB(i,0,j);`-ға алмастырып, **қызыл** және **көк** түстердің араласып **қызғылт** түс, `түсі:=RGB(0,i,j);`-ға ауыстырып, **жасыл** және **көк** түстердің араласып **көгілдір** түс пайда болуына көз жеткіземіз:



2. Доп қосымшасы. **Timer** объектінен бірінші кезекте анимациялық қозғалыс жарату үшін пайдаланылады. Бұл қандай жүзеге асырылуын көру үшін жаңа қосымша жаратамыз. Оған жеткенінше үлкен **Image** объектіні, біреуден **Timer**, **CheckBox** және **SpinEdit** орнатамыз.

Қосымша терезесінің **Image** иелеген бөлігінде доп қозғалсын. Доп қабырғаға соғылғанда артқа қайтсын. Оның қозғалыс жылдамдығын **SpinEdit**-тегі мән белгілесін және оның өзгеруімен доптың жылдамдығы да өзгерсін. **CheckBox**-та жалауша орнатылған болса, доптың қозғалған кезіндегі ізі де көрінсін, кері жағдайда доптың ағымдағы жағдайы экранға шығарылсын. Доптың жағдайы секундына 20 рет (50 миллисекунд интервалмен) жаңаланып тұрсын.

CheckBox-тың тақырыпшасын **ізімен** деп өзгертеміз, **Checked** қасиетін шын деп өзгертеміз. **SpinEdit**-тің максимальды мәнін 10-ға тең етіп аламыз. **Timer**-дің **Interval** қасиетін 50-ге теңестіреміз.

Қосымшада доптың ағымдағы жағдайын сақтау үшін екі: x , y ; доп қозғалысының ағымдағы бағытын сақтау үшін тағы да екі: dx , dy айнымалылар қажет болады. Оларды сипаттайтын **var x,y,dx,dy:integer;** операторын бағдарлама коды терезесіне **var Form1: TForm1;**-ден соң орналастырамыз. Осы айнымалыларға бастапқы мәндерді беру үшін. **Form1**-дің басқа объектілердің бас бөлігін екі рет басамыз. Бағдарлама терезесінде пайда болған **TForm1.FormCreate** процедурасына **x:=0;y:=0;dx:=2;dy:=1;** операторларын енгіземіз.

Timer объектіні екі рет басып оның процедурасын төмендегідей толтырамыз:

```
procedure TForm1.Timer1Timer(Sender: TObject);
var n:integer;
```

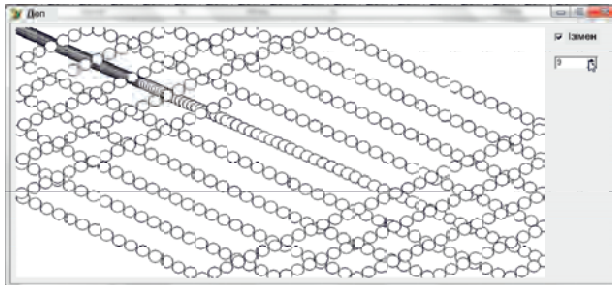
```

begin
n:=SpinEdit1.Value;
if not CheckBox1.Checked then
Image1.Canvas.Rectangle(-3,-
3,Image1.Width+3,Image1.Height+3); x:=x+dx*n;
if x>Image1.Width then
begin x:=2*Image1.Width-x; dx:=-dx;end;
if x<0 then begin x:=-x; dx:=-dx;end;
y:=y+dy*n;
if y>Image1.Height then
begin y:=2*Image1.Height-y; dy:=-dy;end;
if y<0 then begin y:=-y; dy:=-dy;end;
Image1.Canvas.Ellipse(x-10,y-10,x+10,y+10);
end;

```

Екінші жолда процедурада пайдалану үшін жаңа n атты айнымалы сипатталады. Төртінші жолда осы айнымалыға **SpinEdit1**-де берілген қозғалыс жылдамдығы мәні енгізіледі.

Бесінші жолда **CheckBox1**-де жалауша орнатылғаны тексеріледі. Егер осы жалауша орнатылмаған болса, алтыншы жолда доптың алдыңғы жағдайы бүкіл суретті өшіру арқылы жойылады:



Жетінші жолда доптың жаңа жағдайының x координатасы табылады. Сегізінші жолда осы жаңа жағдайдың суреттің оң жақ шекарасынан өтуі тексеріледі. Егер өтіп кетсе, x -тің жаңа мәні есептеп табылады және горизонталь қозғалыс бағыты dx -ке керісінше өзгертіледі. Тоғызыншы жолда доптың жаңа жағдайы суреттің сол жақ шекарасынан өтуі тексеріледі және осы жағдай болса, оның x координатасы қайта есептеліп, қозғалыс бағыты керісінше өзгертіледі.

Оныншы жолдан он екінші жолға дейін доптың **y** координатасы осылайша қайта есептеледі. Сонымен он үшінші жолда доптың жаңа жағдайы экранға сызылады.

Қосымшаны іске қосып, оның жұмыс істеуін бақылаймыз.



ЕСІНДЕ САҚТА !

Delphi-де 16 миллионнан астам түс бар, олар негізгі үш түс: қызыл, жасыл және көк түсті түрлі мөлшерде қосумен жаратылады.



СҰРАҚТАР МЕН ТАПСЫРМАЛАР

1. Пикселдер қосымшасында неше пикселдің түсі өзгереді?
2. Доп қосымшасында доптың жағдайы қайсы айнымалыларға қарап анықталады.
3. Доп қосымшасында доптың қозғалыс жылдамдығы қалай өзгертіледі?
4. Доп қосымшасында доп қозғалысының бағыты қайсы айнымалыда сақталады?
5. Пикселдер қосымшасында түстерді қалыптастыруда үшінші негізгі түстің де қатысуын қамтамасыз етіндер.
6. Доп қосымшасында доп бағытын **dx** және **dy** айнымалыларының бастыпқы мәндерін ауыстырумен өзгертіндер:
 - а) **dx:=1; dy:=1;**
 - ә) **dx:=1; dy:=2;**
 - б) **dx:=3; dy:=2;**
 - в) **dx:=3; dy:=1.**



ҮЙГЕ ТАПСЫРМА

1. Пикселдер қосымшасында квадраттың ұштары қандай түсте және олар қайсы түстердің араласуынан пайда болады?



66-САБАҚ. ПРАКТИКАЛЫҚ ЖАТТЫҒУЛАР

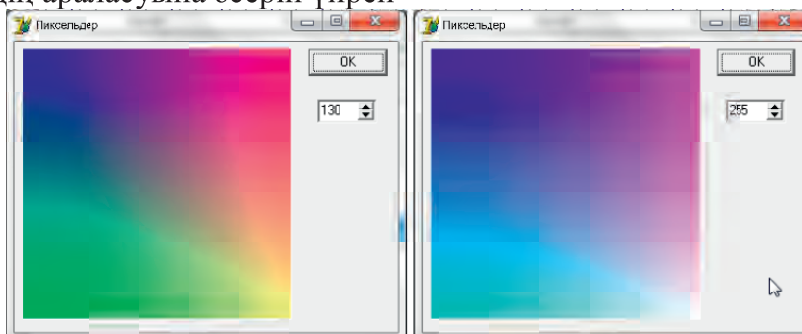
1. Пикселдер қосымшасында екі түстің араласуына үшінші түсті де қос. Ол үшін пикселдер қосымшасының қызыл және жасыл түстердің араластырылатын вариантын жүктеп ал.

1.1. Оған бір **SpinEdit** объектіні орнат. Оның қасиеттерінен төмендегілерді өзгерт.

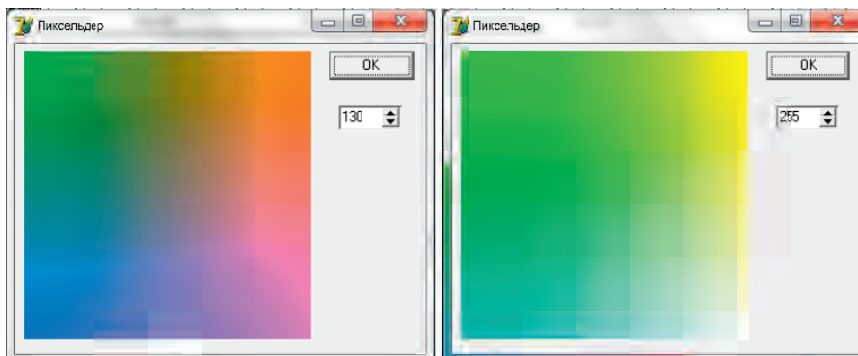
MaxValue:=255; Value:=130; Increment:=5;

1.2. Қосымшаның бағдарлама кодына **түсі:=RGB(i,j,0);** операторын **түсі:=RGB(i,j,SpinEdit1.Value);**-мен ауыстыр.

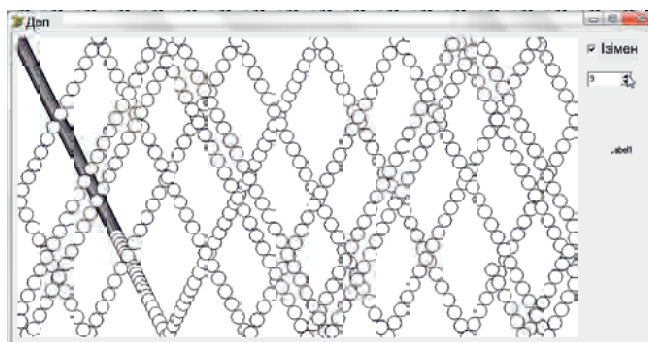
1.3. Қосымшаны іске қосып **көк** түстің **қызыл** және **жасыл** түстердің араласуына әсерін үйрен



2. Пикселдер қосымшасының басқа варианттарында да үшінші негізгі түсті қосып, өзгерістерді бақыла. Төмендегі суретте **жасыл** түстің **қызыл** және **көк** түстердің араласуына әсері көрсетілген.



3. Доп қосымшасын доптың қозғалысы төмендегі суретте көрсетілгендей болатын етіп өзгерт.



67-САБАҚ. БАҚЫЛАУ ЖҰМЫСЫ

Төменде бақылау жұмысының варианттары көрсетілген:

1-вариант

1. **Canvas**-тың **Pen** объекті.
2. Суретке эллипс орналастыру.
3. Қаламды қара түске өткізудің екі тәсілі.

2-вариант

1. **Canvas**-тың **Brush** объекті.
2. Суретке тік төртбұрышты орналастыру.
3. Қаламды сары түске өткізудің екі тәсілі.

3-вариант

1. **Canvas**-тың **Pixels** объекті.
2. Суретке кесіндіні сызу.
3. Қылқаламды ақ түске өткізудің екі жолы.

4-вариант

1. **Canvas** түстермен жұмыс істеу.
2. Суретке мәтін енгізу.
3. Қылқаламды жасыл түске өткізудің екі тәсілі.



68-САБАҚ. ҚАЙТАЛАУ ҮШІН СҰРАҚТАР

1. Бағдарламалау тәсілдері туралы нені білесің?
2. **Delphi** бағдарламалау ортасы интерфейсінің ерекшеліктерін санап өт.

3. Қосымша терезесі: **Form**-ның қасиеттерін және олардың міндеттерін айтып бер.
4. Басқару түймешігі **Button**-ның негізгі қасиеттері туралы не білесің?
5. **Button** түймешігіне тышқан басылғанда не жүзеге асады?
6. **ShowMessage** терезесі туралы не білесің?
7. **Label** объектінің міндетін айтып өт.
8. **Edit** объектінің қасиеттері мен міндеті неден құралған?
9. Объектілердің белсенділігі мен көрінбейтіндігі олардың қайсы қасиеттері арқылы орнатылады?
10. Айнымалылардың мәнін бір түрден екінші түрге өткізетін функцияларды айтып бер.
11. Қосымшаға жалаушалар қайсы объект арқылы енгізіледі?
12. Радиотүймешіктер тобы **RadioGroup**-тың негізгі қасиеттері қайсы?
13. **ListBox** объекті туралы не білесің?
14. **ComboBox** объектінің қолайлылығы неде?
15. **Memo** объектінің міндеті және оның қасиеттерін, объектілерін айтып бер.
16. **Delphi**-де графикамен жұмыс істеу қалай жүзеге асырылады?
17. Қаламмен жұмыс істеу туралы нені білесің?
18. Қылқаламның негізгі қасиеттерін айтып бер.
19. Суретке мәтін қалай енгізіледі?
20. Тура сызық кесіндісі қалай сызылады?
21. Тік төртбұрыш сызу қалай жүзеге асырылады?
22. Эллипс сызу үшін не істеу қажет?
23. Суретте түстермен жұмыс істеу туралы айтып бер.
24. Пикселдермен жұмыс істеу туралы нені білесің?
25. **Timer** объектінің міндеті мен қасиеттерін айтып бер.

ПАЙДАЛАНЫЛГАН ӘДЕБИЕТТЕР

1. Ўзбекистан Республикасы Министрлер Кабинетинин 2017 жылғы 6 сәуірдегі “Жалпы орта білім мен орта арнаулы, кәсіптік-техникалық білімнің мемлекеттік білім стандарттарын бекіту туралы” 187-санды қаулысы.
2. Ahmedov A., Taylaqov N. Informatika. AL va KHK uchun darslik. – T.: O‘zbekiston, 2002. 2-nashri. – 272 b.
3. Nazirov Sh.A., Musayev M.M., Ne‘matov A., Qobulov R.V. Delphi tilida dasturlash asoslari. – Toshkent, 2007. – 324 b.
4. Крыгин С.В. Microsoft Excel 2010. Учебное-практическое пособие. – Н. Новгород, 2011. – 72 с.
5. Карчевский Е.М., Филиппов И.Е., Филиппова И.А. Access 2010 в примерах. Учебное пособие. – Казан, 2012. – 140 с.
6. Taylaqov N.I. EHM ishlari va hisob usuli. KHK uchun darslik. – T.: “Ijod duyosi” nashryot uyi, 2003. – 224 b.
7. Холматов Т.Х., Тайлақов Н.И. Амалий математика, дастурлаш ва компьютернинг дастурий таъминоти. Олий ўқув юртлари учун ўқув қўлланма. – Т.: Мехнат, 2000. – 304 б.
8. Холматов Т.Х., Тайлақов Н.И., Назаров У.А. Информатика. Олий ўқув юртлари учун дарслик. – Т.: Ўзбекистон миллий энциклопедияси, 2003. – 254 б.
9. Taylaqov N.I. Informatika va hisoblash texnikasi asoslari. 9-sinf uchun darslik. – T.: “Ijod duyosi” nashryot uyi, 2002. – 152 b.

МАЗМҰНЫ

I ТАРАУ. MS EXCEL 2010-ДА АМАЛДАР МЕН ФОРМУЛАЛАР

1-сабақ. Қарапайым өрнектерді есептеу	5
2-сабақ. Ұяшыққа сілтеме: салыстырмалы, абсолют және аралас	8
3-сабақ. Сілтемені пайдаланып амал орындауда көшірудің артықшылығы	10
4-сабақ. Қарапайым және күрделі функциялардың графиктері	12
5-сабақ. Басқа параққа немесе кітапқа сілтеме жасау	14
6-сабақ. MS Excel-дегі функциялар кітапханасы	18
7-сабақ. Функция аргументі терезесі	21
8-сабақ. Формулалар жолын пайдалану.....	23
9-сабақ. Мәтіндік функциялар	26
10-сабақ. Логикалық функциялар	27
11-сабақ. Пысықтауға арналған практикалық жаттығулар	29
12-сабақ. Бақылау жұмысы	30
13-сабақ. Математикалық функциялар	33
14-сабақ. Көбейтіндіні есептеуге қатысты функциялар	36
15-сабақ. Статистикалық функциялар	38
16-сабақ. Пысықтауға арналған практикалық жаттығулар	40
17-сабақ. MS Excel-дің көмегімен кейбір мысалдарды шешу	40
18-сабақ. Практикалық жаттығулар	42
19-сабақ. Бақылау жұмысы	43

II ТАРАУ. ДЕРЕКҚОР

20-сабақ. Дерекқор туралы ұғым	45
21-сабақ. Дерекқорды басқару жүйелері	46
22-сабақ. Практикалық жаттығулар	48
23-сабақ. MS Access 2010-ның негізгі элементтері мен алаңдарының ерекшеліктері	49
24-сабақ. Практикалық жаттығулар	52
25-сабақ. MS Access 2010-да дерекқор құру	53
26-сабақ. Практикалық жаттығулар	57
27-сабақ. MS Access 2010-да кестелерді өзара байланыстыру	57
28-сабақ. Практикалық жаттығулар	63
29-сабақ. MS Access 2010-да деректерді берілген шаблон бойынша іздеу және қайта өңдеу	63
30-сабақ. MS Access 2010-да дерекқорды жаратуда және редакциялауда формаларды пайдалану	68
31-сабақ. Практикалық жаттығулар	72
32-сабақ. MS Access 2010-да математикалық амалдарды орындау	72
33-сабақ. Қайталауға қатысты тапсырмалар	76
34-сабақ. Бақылау жұмысы	77

ІІІ ТАРАУ. DELPHI ОРТАСЫНДА ҚОСЫМШАЛАР ЖАРАТУ

35-сабақ. Қосымшалар жаратудың заманалық тәсілдері.....	80
36-сабақ. Delphi бағдарламалау ортасы	84
37-сабақ. Қосымша және оның терезесі.....	87
38-сабақ. Практикалық жаттығулар	90
39-сабақ. Басқару түймешігі	91
40-сабақ. Практикалық жаттығулар	93
41-сабақ. Showmessage терезесі	94
42-сабақ. Бақылау жұмысы	96
43-сабақ. Қосымша терезесіне деректі орналастыру	96
44-сабақ. Практикалық жаттығулар	99
45-сабақ. Басқару объектілерінің белсенділігі мен көрінбеуі	99
46-сабақ. Практикалық жаттығулар	102
47-сабақ. Қосымшаға деректі енгізу	103
48-сабақ. Практикалық жаттығулар	106
49-сабақ. Дерек түрін өзгерту.....	107
50-сабақ. Бақылау жұмысы	111
51-сабақ. Қосымшада жалаушаларды пайдалану	111
52-сабақ. Практикалық жаттығулар	116
53-сабақ. Қосымшадағы радиотүймешіктер тобын пайдалану	117
54-сабақ. Практикалық жаттығулар	121
55-сабақ. ListBox және ComboBox объектілері	122
56-сабақ. Практикалық жаттығулар	126
57-сабақ. Мемо басқару объекті	127
58-сабақ. Бақылау жұмысы	131
59-сабақ. Delphi-де графикамен жұмыс істеу	132
60-сабақ. Практикалық жаттығулар	137
61-сабақ. Timer объекті және оны пайдалану	139
62-сабақ. Практикалық жаттығулар	143
63-сабақ. Суретке басқа объектілерді орналастыру	144
64-сабақ. Практикалық жаттығулар	148
65-сабақ. Графикалық және анимациялық қосымшаларға мысалдар	149
66-сабақ. Практикалық жаттығулар	153
67-сабақ. Бақылау жұмысы	154
68-сабақ. Қайталау үшін сұрақтар.....	154

Орта білім беретін мекемелердің 10-сыныбы
оқушыларына арналған оқулық

INFORMATIKA VA AXBOROT TEXNOLOGIYALARI

Taylaqov Norbek Isaqulovich
Axmedov Akrom Burxonovich
Pardayeva Mehriniso Daniyarovna
Abdugʻaniyev Abduvali Abdulhayevich
Mirsanov Uralboy Muxammadiyevich

Редакторы: Т. Ахмедов
Көркемдеуші редакторы: А. Абдусаломов
Техникалық редакторы: Қ. Мадияров
Қазақшаға аударған: Т. Сапарәліұлы
Корректоры: З. Серикбаева
Компьютерде терген: А. Ахмедов

Лицензия нөмірі АІ № 296 22.05.2017. Басуға рұқсат етілді
27.10.2017. Пішімі 70x90 1/16. Times New Roman гарнитурасы.

Шартты б.т. 9,0. Есептік б.т. 8,56. Таралымы 5321 дана.

Тапсырыс № 413.

Келісімшарт № 11-2017.

Өзбекстан Баспасөз және ақпарат агенттігінің Ғафур Ғұлам
атындағы баспа-полиграфия шығармашылық үйінде басылды.
100128, Ташкент, Лабзак көшесі, 86-үй.

Жалға берілген оқулықтың жағдайын көрсететін кесте

№	Оқушының аты-жөні	Оқу жылы	Оқулықты алған кездегі жағдайы	Сынып жетекшісінің қолы	Оқулықты тапсырған кездегі жағдайы	Сынып жетекшісінің қолы
1						
2						
3						
4						
5						
6						

Оқулық жалға беріліп, оқу жылының соңында қайтарып алынғанда жоғарыдағы кесте сынып жетекшісі тарапынан төмендегі бағалау өлшемдеріне орай толтырылады:

Жаңа	Оқулықтың бірінші рет пайдалануға берілген кездегі жағдайы.
Жақсы	Мұқабасы бүтін, оқулықтың негізгі бөлігінен ажырамаған. Барлық парақтары бар, жыртылмаған, көшпеген, беттерінде жазу мен сызықтар жоқ
Қанағаттанарлық	Мұқабасы езілген, біршама сызылып, шеттері бүлінген, оқулықтың негізгі бөлігінен ажыраған жерлері бар, пайдаланушы тарапынан қанағаттанарлық түрде жөнделген. Көшіп түскен парақтары қайта жөнделген, кейбір беттеріне сызылған.
Қанағаттанарсыз	Мұқабасы сызылған, жыртылған, негізгі бөлігінен ажыраған немесе түбегейлі жоқ, қанағаттанарсыз жөнделген. Беттері жыртылған, парақтары жеткіліксіз, сызылған, боялған. Оқулықты тіктеп болмайды.