

с. 9 + 3. 26 - 018.1
X 17

К. З. Халықова, А. Р. Тұрғанбаева, Б. Ғ. Бостанов

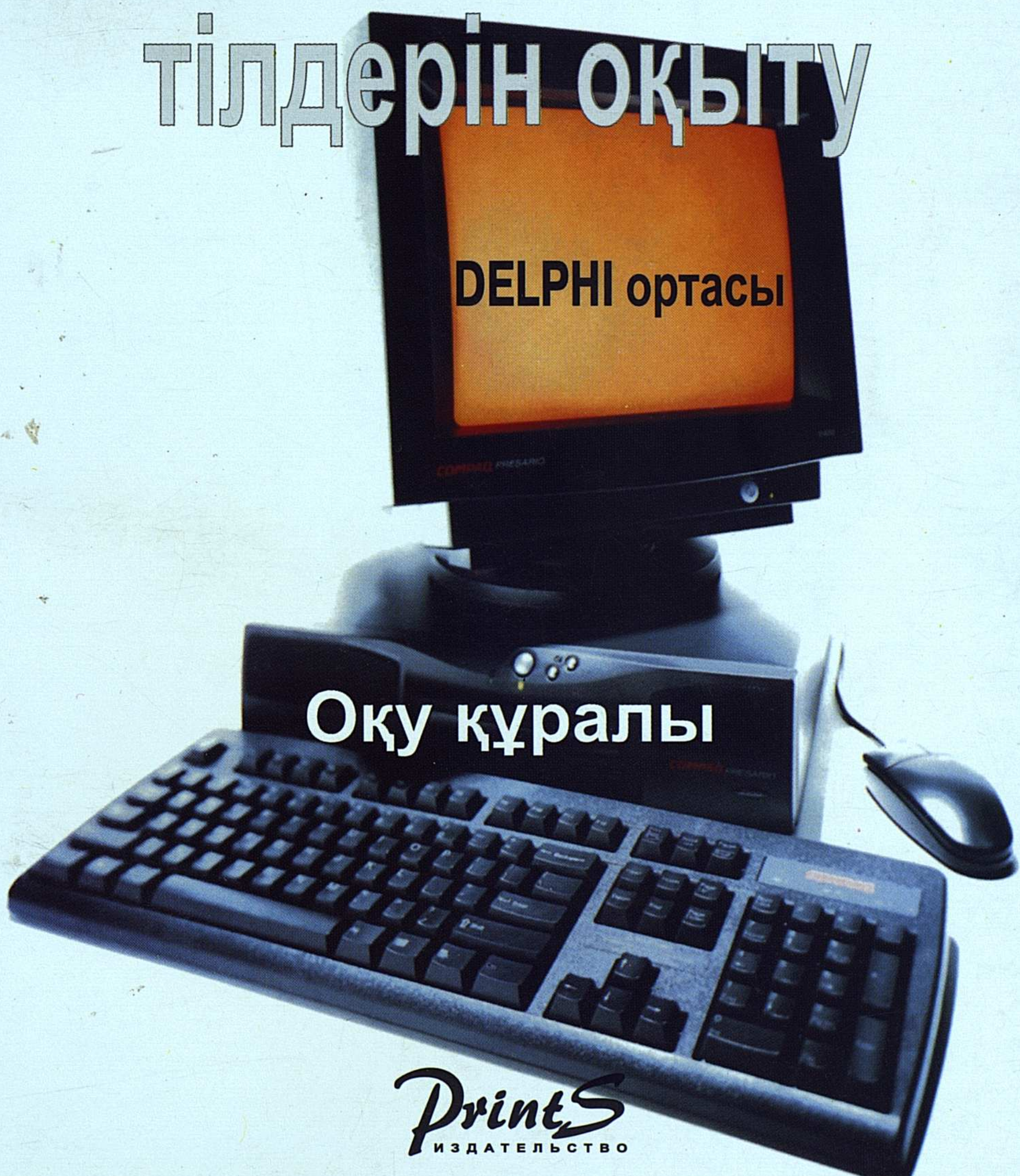
1 2009
16594к

ПРОГРАММАЛАУ

тілдерін оқыту

DELPHI ортасы

Оқу құралы



PrintS
ИЗДАТЕЛЬСТВО

А 2009/16594 к

32.973.26-018.1

X 17

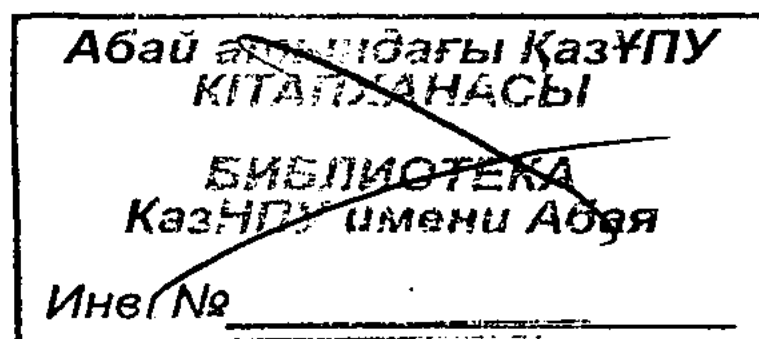
К. З. Халықова

А. Р. Тұрғанбаева

Б. Ғ. Бостанов

**ПРОГРАММАЛАУ ТІЛДЕРІН
ОҚЫТУ.
DELPHI ОРТАСЫ**

Оқу құралы



PrintS
ИЗДАТЕЛЬСТВО

АЛМАТЫ

004.43

ББК 32.973.26-018.1

X 17

*Оқу құралын жоғарғы оқу орындары студенттеріне
Қазақстан Республикасы Білім және ғылым министірлігі өткізген
оқулықтар мен оқу-әдістемелік әдебиеттер дайындау жөніндегі конкурс
комиссиясының “Білім” мамандығы бойынша тобы ұсынады.*

Пікір жазғандар:

С. М. Кеңесбаев, ф.-м.ғ.к., доцент; **Ш. И. Хамраев**, техн. ғ. к., доцент.

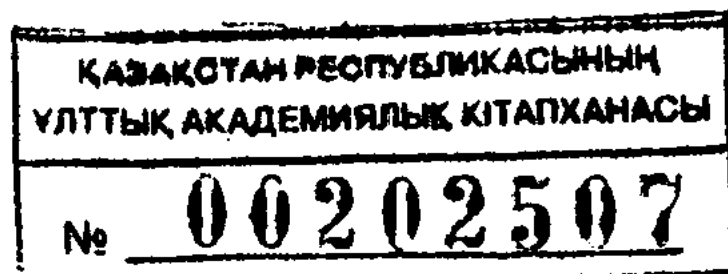
Халықова К. З., Тұрғанбаева А. Р., Бостанов Б. Ғ.

X 17

Программалау тілдерін оқыту. Delphi ортасы: Оқу құралы. – Алматы:
Print-S, 2005. – 250 б.

ISBN 9965-482-11-X

Оқу құралында Object Pascal тілінде программа құру негіздері қарастырылады. Еңбек Абай атындағы ҚазҰПУ-дың “Информатика және қолданбалы математика” кафедрасында программалау тілдерін оқыту бағдарламасына сай жазылған. Жоғарғы оқу орындарының студенттеріне, оқытушыларына, орта мектептің информатика пәні мұғалімдері мен оқушыларына арналған.



X 4602000000
00 (05)-05

ББК 32.973.26-018.1

KX

ISBN 9965-482-11-X

© Халықова К. З., Тұрғанбаева А. Р., Бостанов Б. Ғ., 2005

КІРІСПЕ

Ақпараттық коммуникациялық технологиялардың қарқынмен дамып, күнделікті тұрмысқа енуіне байланысты соңғы кездері программалауға деген қызығушылықтың арта түскені белгілі. Қазіргі уақытта компьютерсіз қандай да бір жұмыс орындауды көз алдымызға елестету қиын. Бұл компьютермен жұмыс істейтін адамға қандай да бір ұмтылысты тудыруы мүмкін.

Есептеуіш техниканың қарқынмен дамуы тиімді программалық құралдарды жасау - объектілі бағдарланған программалау жүйелерінің жасалуына әкелді. Жылдам құру жүйесінің негізіне (RAD – Rapid Application Development – қосымшаларды жылдам құру ортасы) визуалды жобалау мен оқиғалы объектілі бағдарланған программалау технологиясы алынған. Мұндай программалаудың мәні – программа жасаушы құрал жұмыстың көп бөлігін өзі атқарып, программистке диалог терезелерін құрастыру мен оқиға өңдеушіні құру жұмыстарын қалдырды. RAD жүйелерінің ішінде Borland Delphi және Visual Basic орталары әртүрлі программалар құруға мүмкіндік береді: қарапайым бір терезелік қосымшалардан бастап, үлестірілген мәліметтер қорын басқару программаларына дейін. Borland Delphi ортасындағы программалау тілі Object Pascal тілі.

Программалау іскерлігін үйрену үшін нақты есептер шығарып, программасын жаза білу қажет. Ол үшін программалау тілі мен оны құру ортасын меңгеру керек. Объектілі бағдарланған программалау жүйелерінде программалаудағы негізгі ерекшелік алдымен программалау синтаксисі мен компоненттерді пайдалануды үйренуде болып табылады.

Бұл ұсынылып отырған оқу құралында объектілі бағдарланған программалау жүйелерінің негіздері берілген. Оқу құралы он тараудан тұрады. Оқу құралын Delphi ортасында программалауға қажетті теориялық материалдар мен мысалдар келтірілген. Әрбір тараудың соңында теориялық материалдарды бекітуге арналған бақылау сұрақтары мен жаттығулар берілген.

Авторлар тобы оқу әдістемелік құралын жазуға Абай атындағы Қазақ Ұлттық Педагогикалық университетінің физика-математика факультетіндегі информатика курсы, міндетті арнайы курстар мен мұғалімдер білімін жетілдіру институтында Delphi ортасы бойынша жүргізілген лекциялар мен практикалық сабақтарда жинақталған тәжірибелерді пайдаланды.

I тарау. Delphi ортасымен танысу

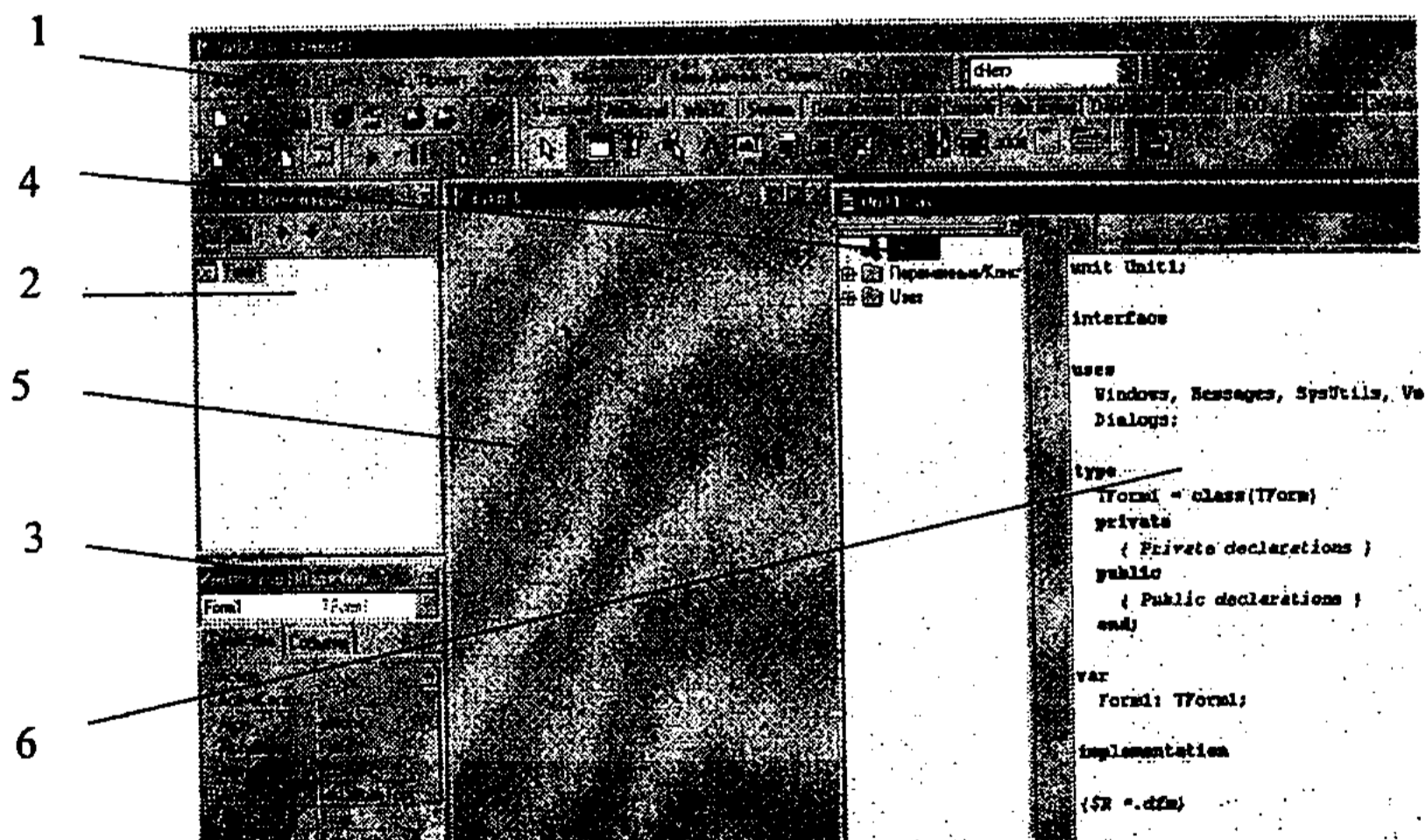
1.1. Delphi ортасының жұмысына жалпы сипаттама

Delphi ортасы - программист жұмысының өте жоғары тиімділігін қамтамасыз ететін орта.

Delphi ортасын шақырғаннан кейін ортаның жұмысын басқаратын алты негізгі терезе шығады (1.1 – сурет):

- 1) негізгі терезе;
- 2) объектілер тармағының терезесі;
- 3) объектілер инспекторының терезесі;
- 4) браузер (көру) терезесі;
- 5) формалар терезесі;
- 6) программа кодының терезесі.

Программа кодының терезесін көру үшін F12 клавишын басу жеткілікті. Delphi-ді алғаш шақырғаннан кейін «Delphi» терезесі барлық терезенің үстіне шығады. Осы терезенің көмегімен Inprise коорпорациясының WEB-беттеріне кіруге мүмкіндік туады.



1.1-сурет. Delphi ортасының интерфейсі.

Негізгі терезе. Негізгі терезе құрылатын программаның жобаларын басқаратын негізгі қызметті атқарады. Бұл терезе экранда барлық уақытта болады және ең жоғарғы бөлігінде орналасады. Негізгі терезеде Delphi ортасының бас менюі, пиктографиялық командалық батырмалар жиынтығы, компоненттер палитрасы орналасқан.

Бас меню жобаны басқаруға арналған барлық қажетті командалардан тұрады. Бас менюдің барлық опциялары екінші деңгейде ашылатын опциялар тақырыбын қамтиды.

Бас терезенің барлық элементтері арнайы панельдерде орналасқан, оның сол жақ бөлігінде басқару батырмалары бар. Бас менюден басқа кез келген панельді терезеден алып тастауға болады.

Панельде көрінетін батырмалардың құрамын өзгерту үшін оған курсорды келтіріп, тышқанның оң жақ батырмасын басу жеткілікті. Бұдан кейін ашылған көмекші меню терезесінде барлық панельдердің аттары келтірілген және олардың статусы көрсетілген (жалаушалар). Олардың ішінен Customize-ды (настройка, баптау) таңдағаннан кейін баптау терезесі пайда болады. Енді қажет емес батырмаларды алып тастауға да болады, Commands терезесіндегі тізімнен қажетті батырмаларды таңдап, оны экранға тасып апаруға болады.

Негізгі терезеге жиі пайдаланылатын пиктограммаларды орналастырған ыңғайлы: Project\Syntax Check - программа командаларының синтаксисін тексеру; View/Debug Windows/Watches - отладкалық режимді бақылау терезесіне кіру және т.б.

1.2. Пиктографиялық батырмалар

Пиктографиялық батырмалар бас менюдің негізгі опцияларына жылдам кіруге мүмкіндік береді. Олар басқару белгілері бойынша 7 топқа бөлінген. Әрбір топ бір панельде орналасады:

1. Standard тобы;
2. View тобы;
3. Debug тобы;
4. Custome тобы;

5. Desktops тобы;

6. Internet тобы.

Галерея компоненттерінің тобы жеке қарастырылады.

1.3. Терезелерді түзету

Delphi ортасының 5-нұсқасынан бастап, үш жаңа интерфейстік элементтермен толықтырылған, олар Desktops тобында орналасқан. Осы инструменттердің көмегімен программист Delphi-дің қалған терезелерінің орналасуының бірнеше варианттарын дайындап, оны түзету файлында сақтап қоюына болады. Әдетте екі немесе үш терезенің негізгі конфигурациясы таңдалынады: форма жасау режимі, кодтау режимі және отладка. Форма жасау кезінде экранда форманың өзі, объектілер тармағы мен объектілер инспекторы терезелері көрініп тұруы тиіс.

Терезелердің сәйкес өлшемі мен оны орналастырудың күйін келтіргеннен кейін, оны мысалы, «Design desk» деген атпен сақтауға болады. Кодтау режимін түзеткеннен кейін «Code desk», отладкалық режимді «Debug desk» деген атпен сақтауға болады.

1.4. Delphi ортасының компоненттер палитрасы

Компоненттер палитрасы - Delphi ортасының ең басты байлығы болып табылады. Ол негізгі терезенің оң жақ бөлігінде орналасқан және онда қажетті компонентті жылдам іздеуді қамтамасыз ететін ашылатын беттері бар. Компонент деп белгілі бір қасиеті бар, программист формалар терезесіне орналастыра алатын қандай да бір басқару элементін түсінеміз. Компоненттердің көмегімен программаның сұлбесі (каркасы) жасалады, жалпы жағдайда экранда көрінетін терезелер, батырмалар, таңдаулар тізімі және т.б.

Батырмалар панелі тәрізді, компоненттер палитрасы да түзетіледі. Ол үшін арнайы редактор пайдаланылады, редактор терезесі компоненттер палитрасындағы кез келген пиктограммаға курсорды келтіріп, тышқанның оң жақ батырмасын басқанда және Properties опциясын таңдағанда пайда болады.

1.5. Формалар терезесі

Формалар терезесі Windows жобасы - болашақта программа жазатын терезе болып табылады. Басында бұл терезе бос болады. Дәлірек айтсақ, ол Windows-дің интерфейстік элементтері үшін стандартты жүйелік менюді шақыру батырмаларынан, терезені үлкейту, кішірейту, жабу батырмаларынан және тақырып жолынан тұрады. Әдетте, терезенің жұмыс алаңы нүктелік координаталық тормен толтырылған, ол формадағы компоненттердің орналасуын реттеу қызметін атқарады (бұл нүктелерді алып тастау үшін Tools/Environment Options меню командаларының көмегімен, Preferences қойымтасындағы Display Grid ауыстырып қосқышындағы жалаушаны алып тастау қажет).

1.6. Объектілер тармағының терезесі

Бұл терезе екпінді формада немесе мәліметтердің екпінді модулінде орналасқан жеке компоненттер арасындағы байланысты көрнекі бейнелеуге арналған. Терезедегі кез келген компонентті белгілеу формалар терезесіндегі сәйкес компонентті екпінді жасайды және объектілер инспекторы терезесінде осы компоненттің қасиеті бейнеленеді. Екі рет тышқанмен белгілеу Code Insight механизмінің жұмыс істеуіне әкеледі, ол OnClick оқиғасын өңдеу үшін дайындалғанды код терезесіне орналас-тырады.

1.7. Объектілер инспекторының терезесі

Формаға орналасатын кез келген компонент бірқатар параметрлер жиынымен сипатталады: орналасуы, өлшемі, түсі және т.б. Мысалы, компоненттің орналасуы мен өлшемін программист формалар терезесінде компонентті басқара отырып, өзгерте алады. Ал, объектілер инспекторы терезесі қалған өзгертулерді орындауға арналған. Бұл терезеде екі бет бар: Properties (Қасиет) және Events (Оқиға). Properties беті – компоненттің керекті қасиеттерін тағайындау қызметін атқарады, Events беті – қандай да бір оқиғаға компоненттің жауабын анықтайды.

Объектілер инспекторы терезесінің кез келген беті екі бағанды кесте түрінде беріледі, сол бағанда қасиеттің немесе оқиғаның аты, ал оң жақта - қасиеттің нақты мәні немесе сәйкес оқиға өңдейтін қосалқы программаның аты болады.

Кесте жолының оң жақ бағанын тышқанмен белгілеу ондағы көрсетілген қасиеттің мәнін екпінді жасайды, бұдан жолдың оң жақ шетінде үш нүкте (...) немесе ↓ батырмаларының бірі пайда болады. Үш нүкте (...) батырмасын белгілеу күрделі қасиеттің мәнін тағайындайтын сұхбат терезесін экранға шығарады, ↓ батырмасын белгілеу қарапайым қасиеттің мүмкін болатын мәндерінің тізімін шығарады. Объектілер инспекторы терезесінің жоғарғы бөлігінде формада орналасқан барлық компоненттердің ашылатын тізімі орналасады. Форманың өзі де компонент болғандықтан, оның аты да осы тізімде болады.

Тышқанның оң жақ батырмасын басқанда ашылатын терезенің контекстік менюінде терезені түзетуге мүмкіндік беретін опциялар бар. Мысалы, Stay On Top командасын таңдағаннан кейін, Объектілер Инспекторы терезесі оның екпінділігіне тәуелсіз басқа терезелердің үстінде “қалықтап” жүреді.

1.8. Программалар коды терезесі

Код терезесі — программа мәтінін құрып, жөндеуге арналған. Бұл мәтін арнайы ереже бойынша құрылады және программаның жұмыс алгоритмін сипаттайды. Мәтінді жазу ережелерінің жиынтығы программалау тілі деп аталады. Delphi жүйесінде Объект Pascal программалау тілі пайдаланылады. Ол кеңінен таралған Pascal тілінің кеңейтілген, жетілдірілген нұсқасы, оны алғаш рет швейцариялық ғалым Н.Вирт 1970 жылы ұсынған және оны Borland коорпорациясының қызметкерлері жетілдірді (олардың құрған тілдері Turbo Pascal, Borland Pascal және Object Pascal деп аталады).

Ең алғашында Код терезесінде толыққанды Windows терезесі ретінде бос форманы басқаруды қамтамасыз ететін алғашқы шағын мәтін шығады. Жобамен жұмыс барысында программист оған программаны басқаруды жүзеге асыратын алғашқы толықтырулар жасайды.

Тіпті қарапайым программа құру үшін программа кодын құрып өзгертуге, жөндеуге тура келеді, төменде код терезесімен жұмыс істеудің негізгі тәсілдері келтіріледі. Жаңа жобаны ашқаннан кейін төмендегідей жолдар көрінеді:

```
Unit Unit1;  
  Interface  
  Uses Windows, Messages, SysUtils, Classes, Graphics,  
    Controls, Forms, Dialogs;  
  Type  
  Tform1=class(Tform)  
    Private {Private Declarations}  
    Public {Public Declarations}  
  End;  
  Var  
    Form1:Tform1;  
  Implementation  
    {SR*.DFM}  
End.
```

Delphi бұл жолдарды жаңа форма үшін код терезесіне автоматты түрде қояды. Код терезесі программаның жұмысы басталғаннан кейінгі жағын қарастырады. Ал форма терезесі оның сыртқы көрінісі. Екі терезе бір-бірімен тығыз байланысты, Delphi ортасы

```
Unit Unit1;
```

және

```
implementation
```

жолдарының арасына қажетті жолдарды қояды.

Әзір бұл мәтін бөлігіне өзгеріс енгізбейміз. Кейінірек программаның мәтін терезесіне

```
{SR*.DFM}
```

және

```
End.
```

жолдарының арасына қажетті мәтінді енгіземіз. Терезеге жаңа жол енгізу үшін қажетті орынға курсорды келтіріп, клавиатурадан мәтін енгіземіз. Программадағы кеткен қатені жөндеу BS және Delete клавиштарының көмегімен жүзеге асырылады. Егер толық

жолды өшіру қажет болса, онда CTRL+Y клавиштар комбинациясы басылады. Мәтіндегі соңғы өзгерісті алып тастау үшін (Отмена) CTRL+Z немесе менюден Edit/Undo командалары пайдаланылады. Код терезесімен бірге әдетте, Code Explorer броузер терезесі де екпінді болады, ол терезеде кодтың жолдары көп болған жағдайда керекті элементті іздеуді жеңілдету үшін пайдаланылады.

Броузер терезесіндегі элементті тышқанмен екі рет белгілеу сәйкес объектіні сипаттау үшін код терезесіне курсорды орналастырады немесе оны алғашқы сақтау кезінде орындалады. Аса күрделі емес оқу программаларын құруда броузер терезесін жауып қоюға болады. Броузер терезесін шығару үшін View Explorer терезесін тандаймыз.

6-нұсқада код терезесінің «Code» және «Diagram» деген беттері бар. Алғашында бұл бет бос болады, оны толтыру үшін Инспектор терезесінен компоненттік байланыстарымен тышқанмен тасымалдап әкелу қажет. Диаграммалар өрісіне компоненттерді тасымалдауда SHIFT клавишін басып ұстап тұрып, қажетті орынға қойғаннан кейін, тышқанның сол жақ клавишін басу жеткілікті. Диаграммалар бетін басып шығаруға болады.

Бақылау сұрақтары мен жаттығулар

1. Delphi ортасының жұмысы қалай басқарылады?
2. Негізгі терезенің атқаратын қызметі қандай және ол қандай бөлімдерден тұрады?
3. Пикторграфиялық батырмалардың қызметі қандай және олар қаншаға бөлінген? Қалай аталады?
4. Компонент дегеніміз не?
5. Delphi ортасының компоненттер палитрасының қызметі қандай?
6. Форма терезесінің міндеті не?
7. Объектілер тармағы терезесі қандай қызмет атқарады?
8. Объектілер инспекторы терезесі қандай қызмет атқарады?
9. Программалар коды терезесінің қызметі қандай және ол форма терезесімен қалай байланысады?

10. Программалар коды терезесі ашылғанда Delphi ортасы қандай жолдарды автоматты түрде шығарады?

11. Броузер терезесінің қызметі не және ол қалай шақырылады?

II тарау. Визуалды программалау негіздері

Delphi-де программалау екі процестің өзара тығыз әрекеттесуінен тұрады: программаның визуалдық көрінісін құру және программаға толық қажетті басқаруды беру үшін программа кодын жазу процесі. Код жазу үшін код терезесі пайдаланылады, программа құру үшін қалған терезелер, оның ішінде форма терезесі пайдаланылады.

Форма мен код терезелерінің мазмұндарының арасында үздіксіз байланыс болады. Формаға компонентті орналастыру программа кодының автоматты өзгеруіне әкеледі және керісінше автоматты немесе қолмен қойылған код фрагменттерін өшіру сәйкес компоненттерді өшіруге әкеледі. Сондықтан программистер алдымен форма құрады, қажет болған жағдайда код фрагментін жазуға көшеді. Сондықтан, біз Delphi-де программалауды формалар терезесінен бастаймыз.

2.1. Бос форма және оны түрлендіру

Біз формалар терезесі, программалар терезесі Windows жобасынан тұратынын айттық. Сөйтіп File/New/Application опциясын таңдағаннан кейін қарапайым программа дайын болады. Дегенмен, бұған дейін: өзіңіз жұмыс істейтін каталогты құрып, Delphi-дің күйін келтіру (баптау) қажет.

2.2. Delphi-ді баптау

Программалар жобасымен жұмыс істеу процесінде сізге формалар мен модульдер жиынтығын құру қажет. Бұл мәліметтерді жеке папкада файлдар түрінде сақтаған дұрыс. Оның ішінде өзіңіздің жұмыс папкаңызды құруға болады. “My_first_program”

деген атпен папка құрыңыз, оның ішіне өзіңіздің программаңызды сақтайтын папка құрыңыз.

Екіншіден, Delphi ортасының стандартты келтірілген күйіне өзгеріс енгізу қажет. Бұл сіздің құрған программаңыздың соңғы нұсқасының автоматты түрде сақталуы үшін керек.

Ол үшін Tools/Environment Options менюін таңдаңыз, пайда болған сұхбат терезесінде Preferences бетінің екпінді екеніне назар аударыңыз. Осы беттің сол жақ жоғарғы бұрышында Autosave Options тобында «Editor Files» және «Desktop» деген ауыстырып қосқыштары бар. Ауыстырып қосқыштарды екпінді жасау - программа коды терезесінің мәтінін және Delphi терезесінің жалпы орналасуын автоматты түрде сақтайды. Бұл программаны жоғалудан қорғайды. Тышқанмен белгілеу арқылы осы ауыстырып қосқыштарды екпінді жасаймыз. Компиляция барысын қадағалап отыру үшін Compiling және Running тобындағы Show Compiler Progress ауыстырып қосқышын екпінді жасаңыз. Сондай-ақ, программа кодында түсіндірмені - мәтін фрагменттерін сақтауға болады, ол программа жұмысына әсер етпейді, бірақ оның мәтінін түсінікті етіп көрсетеді. Мәтін шығару үшін шрифті тағайындау қажет:

Tools/Editor Options → Display бетінде Editor Font-қа Times New Roman шрифт түрін тағайындау қажет.

Енді алғашқы программаны жүргізуге бәрі дайын болды. Негізгі терезеден ► батырмасын немесе F9 басқару клавишын басыңыз. Сонымен, Delphi ортасындағы программа үш кезеңнен өтеді: компиляция—компоновка—орындау.

Компиляция кезеңінде код терезесінде дайындалған программа мәтіні Object Pascal тіліне ауыстырылады; компоновка кезеңінде қажетті қосалқы программалар қосылады, ал орындалу кезеңінде дайын программа жедел жадыға шақырылып, орындалуға жіберіледі.

F9 клавишін басқаннан кейін Save Unit1 As сұхбат терезесі пайда болады, яғни Delphi Unit1.pas модулі үшін файлдың аты мен ол орналасатын папканың атын сұрайды, сондай-ақ, модульдің де атын өзгерту қажет (немесе әрбір есепке жеке папка ашып, Save All командасы арқылы бәрін сақтаған дұрыс).

Delphi-дегі атаулар. Delphi-дегі атау латын әріптерінен басталатын әріптер, цифрлар және сызықша болуы тиіс, атау цифрдан басталмауы керек, ұзын атау бере беруге болады.

2.3. Форманың қасиеттерін өзгерту

Алдымен модуль ұғымына тоқталайық. Әрбір жаңа форма құрғанда модуль құрылады. Программаны компиляциялау кезінде Delphi заты .PAS, .DFM және .DCU болып келген файлдарды әрбір модуль үшін құрады: .PAS файл – программа коды терезесіндегі мәтін көшірмесінен тұрады; заты -.DFM файлында форма терезесінің мазмұны сақталады; ал .DCU файлында-екі файлдан алынған мәтінді машиналық нұсқауға түрлендіру нәтижесі сақталады. .DCU файлдары компилятормен құрылады және ол компоновщиктің жұмысы үшін қажетті базаны береді. Одан кейін ол заты .EXE болып келген біртұтас шақырылатын файлға түрленеді. Енді программаны түрлендіріп көрейік, алдымен терезе тақырыбын өзгертейік. Алғаш Delphi жүйесіне кіргеннен кейін, терезе тақырыбы форма тақырыбына сәйкес келеді. Тақырыпты өзгерту үшін Объектілер Инспекторы терезесіне бару қажет. Form1 программасының терезесін жабыңыз. Объектілер Инспекторы терезесінің Caption жолын тышқанмен белгілеңіз. Осы жолдың оң жақ бағаны мәтінімен ерекшеленеді және ерекшеленгеннің оң жағында жыпылықтаған курсор болады. Клавиатураны кириллицаға ауыстырып, “программа” деп жазыңыз, одан кейін F9-ды басыңыз. Программаның бұл орындалуы – тақырыбы “программа” деп аталатын терезе құру. Объектілер Инспекторы терезесі арқылы программалар терезесінің бір қасиетін – тақырыбын өзгерттік. Дәл осылай формалардың кез келген қасиетін өзгертуге болады.

2.4. Формаға жаңа компоненттер орналастыру


Формаға жаңа компонент орналастыруды қарастырайық. Ол үшін алдымен орналасатын компонентті таңдау қажет, одан кейін форма терезесіндегі компонент орналасатын орынды тышқанмен белгілеу жеткілікті.

Енді программалар терезесіне әртүрлі жазуларды, түсіндірмелерді орналастыруға арналған Label (Белгі) компонентін орналастырайық. Алдымен, компоненттер палитрасынан Standard бетінің ашылғанына көз жеткізу қажет. Курсорды әрбір компоненттің үстінен жүргізгенде оның аты жазылады. Компоненттер палитрасына курсорды келтіріп, тышқанмен белгілейміз, одан кейін форма терезесінде белгінің орнын көрсетеміз.

Жаңа компоненттің стандартты аты Label1, батырмадағы жазудың да аты осындай болады. Объектілер Инспекторы терезесіндегі Caption жолының көмегімен бұл жазуды оңай өзгертуге болады: “программа N1” деп енгіземіз. Бұл енгізілген жазу формадағы компоненттен көрінеді.

Енді жазудың түсі мен шрифті өзгертейік. Ол үшін Объектілер Инспекторы терезесіндегі Font қасиетін тышқанмен белгілейміз, батырманың көмегімен жолдың оң жақ бөлігінен шрифті түзететін сұхбат терезесі ашылады. Size тізімінен шрифтің өлшемін 24, Color тізімінен қажетті түсті таңдаймыз. ОК батырмасымен терезені жабамыз. Форма терезесіндегі компоненттегі жазу өзгертуге сәйкес қасиетін өзгертеді.

2.5. Оқиғаға жауапты қалыптастыру

1. **Формаларды түрлендіру.** Алғашқы программаға кезекті түрлендіруді жасайық: формаға тағы бір компонент орналастырайық. Мысалы, батырманы алайық, ол тышқанның оң жақ батырмасын басуға байланысты оқиғаға жауап беретін болсын. Батырма компоненті Standard бетінде компоненттер палитрасында  пиктограммасымен бейнеленеді. Бұл компонентті форма терезесіне белгінің төмен жағына орналастырайық.

2. **OnClick оқиғасын өңдеуші.** Тышқанның сол жақ батырмасын басқанда жұмыс істеп тұрған программада OnClick оқиғасы пайда болады. Әзір бұл оқиға программамен ешқандай өңделмейді, сондықтан батырманы басу ешқандай нәтиже бермейді. Батырма басылғанда жауап беруді жүзеге асыру үшін Object Pascal тілінде программа фрагментін жазу қажет, ол оқиғаны өңдеуші деп аталады. Бұл фрагментте программист батырма

басылғанда программаның не істейтінін көрсететін мәтіндік жолдар тізбегі жазылады. Фрагмент Object Pascal-дің арнайы қосалқы программасы - процедура түрінде өрнектеледі.

Delphi OnClick оқиғасын өңдеу процедурасына дайындықты өз бетімен жасау үшін компонентті тышқанмен жылдам екі рет белгілеу қажет. Нәтижесінде Код терезесінде төмендегідей мәтін фрагменті болады:

```
Procedure TForm1.Button1Click (Sender:Tobject);  
Begin
```

```
End;
```

Енді фрагментті талдайық. Procedure сөзі компиляторға қосалқы программа-процедураның басы жөнінде хабарлайды. Delphi ортасында сондай-ақ, қосалқы программа - функция пайдаланылуы мүмкін, бұл жағдайда Function сөзі пайдаланылады. Одан кейін: TForm1.Button1Click процедураның аты жазылған. Атау құрама сөзден тұрады: TForm1 - класс атауы; Button1Click процедураның атауы. Енді класс дегеніміз не? Delphi ортасында класс деп өзіне ұқсас экземпляр (көшірмесі) құрудың үлгісі қызметін атқаратын, аяқталған программа фрагментін айтады. Программист класс құрғаннан кейін оны бірнеше программаға немесе бір программаның бірнеше жеріне қоюға болады. Delphi-дің құрамына Borland коорпорациясының программистері құрған стандартты кластар деп аталатын бірнеше жүз кластар кіреді. Стандартты кластар жиынтығы осы программалау жүйесінің қуатты мүмкіндіктерін анықтайды. Әрбір компонент белгілі бір класқа жатады, ал формаға қойылатын компоненттердің барлық нақты экземплярлары сандық индексі бар класс атауын алады. Пайдаланылып отырған Delphi ортасында кластардың барлық атауларын T әрібінен бастау келісілген. Сөйтіп, TForm атауы TForm стандартты класс үлгісі бойынша құрылған кластың атын білдіреді. Код терезесіндегі мәтіннің басына назар аударсаңыздар, төмендегідей жолдар көрінеді:

```
Type TForm1=class(Tform)  
    Button1:Tbutton;  
    Label1:Tlabel;  
Procedure Button1Click (Sender:Tobject);
```



```

Private
  {Private declarations}
Public
  {public declarations}
End;
Var
  Form1:Tform1;

```

Tform1=class(form) жолы Tform стандартты класынан туған (үлгі бойынша құрылған) Tform1 жаңа класын анықтайды. Ал, Form1:TForm1 жолы осы кластың Form1 атты экземплярын (көшірмесін) құрады. Стандартты Tform класы бос Windows терезесін сипаттайды, сонымен бірге TForm1 – белгі және батырма компоненттері қойылған терезені сипаттайды. Бұл компоненттерді сипаттайтын жолдар:

```

Button1:TButton;
Label1:TLabel;

```

Мұндағы Button1 компоненті Tbutton стандартты класының экземпляры; ал Label1 компоненті Tlabel стандартты класының экземпляры.

TForm1.Button1Click процедура атауынан кейін дөңгелек жақшада параметрді шақыру сипатталған:

```
Sender:Tobject
```

“Sender” атты параметр Tobject класына жатады. Sender параметрі Delphi ортасына барлық жағдай үшін қойылған: оның көмегімен Button1Click қосалқы программасы OnClick оқиғасы қандай компонент құрғанын анықтап береді. Жол толығымен

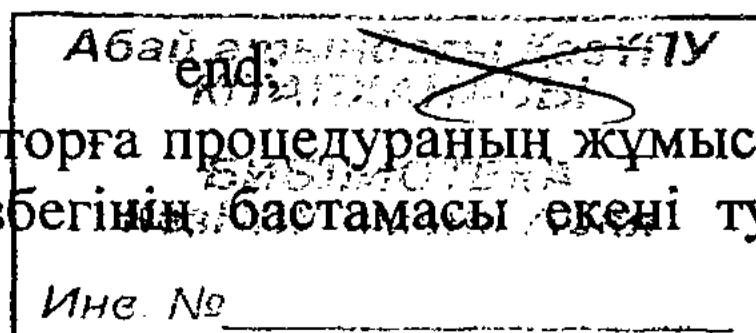
```
Procedure TForm1.Button1Click (Sender:Tobject);
```

процедураның тақырыбы деп аталады. Object Pascal тілінде “;” белгісі тілдің сөйлемінің аяқталғандығын компиляторға хабарлайды. Программа мәтіні жеке сөйлемдерден құралады. Әрбір сөйлемнің соңына нүктелі үтір қойылады. Төмендегі үш жол процедура денесін анықтайды:

```
begin
```

```
end;
```

Begin сөзі компиляторға процедураның жұмыс алгоритмін сипаттайтын сөйлемдер тізбегінің бастамасы екені туралы белгі



береді, ал `end` сөзі осы тізбектің соңын білдіреді. Біздің жағдайда процедура денесінде ешқандай сипаттама жоқ: Delphi процедура жазуға дайындықты құрып берді. Программист процедура денесін қажетті сөйлемдермен толтырады. `Button1` батырмасын әрбір басқан сайын басқару процедура денесіне беріледі, яғни `begin` және `end` сөздерінің арасына оқиғаға жауап ретінде орындалатын программа фрагменті жазылады. Біздің жағдайда батырма қандай да бір дыбыс шығару үшін `Begin...end` сөздерінің аралығына төмендегі сөйлемді жазайық:

```
MESSAGEBEEP(MB_OK);
```

Компьютердің дыбыстық жүйесін алдын-ала қосып, программасын орындайық. Сонымен, `Button1` батырмасын басқанда компьютер дыбыстық сигнал береді, яғни процедура денесіне қойылған жол стандартты процедураны орындайды.

2.6. Компонент қасиеттерінің динамикалық өзгерісі

Біздің программадағы `Button1` батырмасы дыбыс шығаратын болғандықтан, `Button1`-ге “дыбыс” сөзін меншіктедік. Мұны формалар терезесі мен объектілер инспекторы терезесінің көмегімен орындауға болады, яғни форма құру кезеңінде (Объектілер Инспекторы терезесінде `Button1` компонентінің `Caption` қасиетін өзгерту жеткілікті) жүзеге асыруға болады. Бірақ Delphi ортасымен толығырақ танысу үшін басқа тәсілді – программаны жүргізу кезеңінде жазуды динамикалық түрде өзгертуді қарастырайық. Ол үшін формаға арналған `OnCreate` оқиғаны өңдеушіні құрамыз да, ондағы қасиетті өзгертеміз.

OnCreate оқиғасы `Windows` терезесін құрғаннан кейін пайда болады. Оқиға өңдеушіні құру үшін Объектілер Инспекторы терезесінің жоғарғы бөлігіндегі компоненттер тізімін ашыңыз, `Form1` компонентін таңдап, `Events` бетінде осы компоненттің `OnCreate` қасиетін жылдам екі рет тышқанмен белгілеу қажет (`OnCreate` жолының оң жақ бөлігін белгілеу қажет). Delphi бұған жауап ретінде код терезесін екпінді жасап, `Tform1.FormCreate` процедурасына арналған дайындығын ұсынады. Процедураға төмендегідей өзгеріс енгізейік:

```
Procedure Tform1.FormCreate(Sender:Tobject);
```

Begin

```
Button1.Caption:='дыбыс';
```

End;

Бұл программа фрагментіне енгізген жолымыз Object Pascal тілінің меншіктеу операторы. Оператордың сол жақ бөлігінде Button1.Caption қасиеті көрсетілген, ал оң жақ бөлігінде – “дыбыс” мәні көрсетілген. Оператордың екі бөлігін меншіктеу белгісі “:=” байланыстырып тұр.

Button1.Caption құрама атауы қандай қасиет туралы айтылып жатқанын компиляторға дәл көрсету үшін қажет: біздің программада үш компонент пайдаланылады, олардың әрқайсысының Caption қасиеті бар. Button1 префиксі батырманың қасиетін өзгертеді. Қасиетке меншіктелетін мән мәтіндік жол болып табылады. Object Pascal тілінің ережесі бойынша мәтіндік жол апострофқа алынып жазылады. Апостроф ішіне кез келген символдар тізбегін жазуға болады, ол батырмадағы жаңа жазуды анықтайды. Программаның кезекті орындалуынан кейін батырмадағы жазудың өзгергенін көреміз. Бұдан шығатын қорытынды: кез келген компоненттің кез келген қасиетін программаның орындалуы барысында динамикалық өзгертуге болады.

Қорыта келгенде, Delphi-де программа құру процесі екі кезеңге бөлінеді: форма құру және кодтау.

Форма құру палитрадан компоненттерді таңдап, оны формаға орналастырудың көмегімен жүзеге асады.

Программист формаға орналасқан кез келген компоненттің орнын және тышқанның көмегімен өлшемін өзгертуіне болады.

Компонентке қажетті қасиетті беру үшін Объектілер Инспекторы терезесінің Properties беті пайдаланылады. Компонент қандай да бір оқиғаға жауап беру үшін программист оқиғаны өңдеушіні құруы қажет және оның аты Объектілер Инспекторы терезесінде Events бетінде көрсетілуі тиіс.

Оқиғаны өңдеуші құрама атауы бар процедура түрінде бейнеленеді. Атаудың бірінші бөлігі формаға арналған класс атын білдірсе, екінші бөлігі атаудың бірінші бөлігінен нүкте арқылы бөлінеді және ол еркін болуы мүмкін.

Процедура денесі begin...end сөздерінің арасында орналасады және Object Pascal тілінің жеке сөйлемдерінен (операторларынан) тұрады. Әрбір оператор соңына нүктелі үтір қойылады.

Компоненттің қасиеттері программаны жүргізу кезеңінде де өзгеруі мүмкін.

Бақылау сұрақтары мен жаттығулар

1. Delphi ортасында программалау қандай негізгі процестерді қамтиды?

2. Delphi ортасында программалау неден басталады және форма қалай түрлендіріледі?

3. Delphi ортасында программалау қанша кезеңнен тұрады және әрбір кезеңнің қызметі қандай?

4. Delphi ортасында пайдаланылатын атауларға қандай талаптар қойылады?

5. Программаны компиляциялау кезінде құрылатын файлдардың затын атап, әрқайсысына түсініктеме беріндер?

6. Модуль дегеніміз не?

7. Формаға компоненттер қалай орналастырылады?

8. OnClick оқиғаны өңдеушінің қызметі қандай?

9. OnClick оқиға өңдеушісіне байланысты код терезесіндегі шығатын мәтінге сипаттама беріңіз.

10. Класс дегеніміз не және ол программалар коды терезесінде қай бөлімде сипатталады?

11. Компоненттер қасиеттерін қандай жолдармен өзгертуге болады және OnCreate оқиғасының қызметі қандай?

12. Object Pascal тілінде бүтін сандарға қандай операциялар қолданылады?

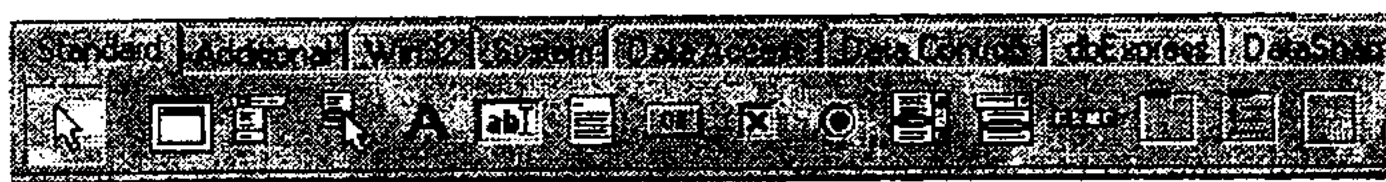
13. Программада әрбір операндаға өзгеріс енгізу қалай жүзеге асырылады?

III тарау. Компоненттермен танысу

Компоненттер дегеніміз жұмыс істейтін программада құрылатын, көрінетін бейнелер тұрғызылатын элементтер болып табылады. Компоненттер палитрасының ішінде бейне ретінде көрінбейтін де компоненттер жеткілікті. Сонымен, бұл тарауда біз Delphi ортасының негізгі компоненттеріне тоқталамыз.

3.1. STANDARD беті

STANDARD бетінде программа құруға ең қажетті Windows-ге арналған стандартты интерфейстік элементтерден тұратын компоненттер палитрасы орналасқан.



3.1-сурет. Standard беті.

Frame – басқа компоненттерді орналастыруға арналған контейнер қызметін атқарады (қызметі форма терезесімен бірдей). Формадан айырмашылығы компоненттерге дайындық құра отырып, компоненттер палитрасында орналасады.

MainMenu – Программаның бас менюі. Бұл компонент күрделі иерархиялық меню құрып жұмыс істеуге мүмкіндік береді.

Label – форманың бетіне мәтін шығаруға арналған. Компоненттің қасиеті мәтіннің түрі мен орналасуын анықтайды.

Edit – енгізу өрісі – символдар жолын жөндеуге арналған.

Button – командалық батырма.

Memo – бірнеше жолдан тұратын мәтіндік редактор құру элементі болып табылады.

RadioButton - қызметі басқа батырмалардың қызметімен байланысты болатын тәуелді батырма болып табылады. Егер сұхбат терезесінде бірнеше ауыстырып қосқыш пайдаланылатын болса, онда әрбір топты RadioGroup компоненті арқылы беруге болады.

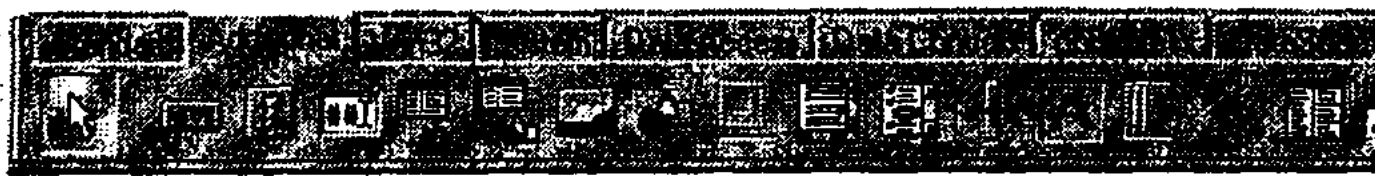
CheckBox – басқаларға тәуелсіз ауыстырып қосқыш батырманы сипаттайды.

ListBox - қажетті элементті таңдауға болатын тізімді береді.

ComboBox – енгізу өрісіне мәліметтерді клавиатурадан теріп енгізуге немесе тізімнен таңдауға мүмкіндік береді.

3.2. ADDITIONAL беті

Additional бетінде сұхбат терзелерінің түрін түрлендіруге мүмкіндік беретін 18 қосымша компоненттер орналасқан.



3.2-сурет. Additional беті.

Bitbtn – жазуы бар, пиктограммалы командалық батырма.

SpeedButton – пиктографиялық батырма. Әдетте бас менюдің опцияларына жылдам кіру үшін пайдаланылады.

MaskEdit – арнайы мәтіндік редактор. Енгізілген мәтінді сүзгілеуге мүмкіндік береді, мысалы, датаны дұрыс енгізу үшін қажет.

StringGrid – жолдар кестесі. Бұл компонент мәтіндік ақпаратты кесте түрінде шығаруға мүмкіндік береді.

DrawGrid – еркін кесте. **StringGrid** компонентінен айырмашылығы бұл компоненттің ұяшықтарында кез келген ақпарат, оның ішінде сурет те сақталады.

Image – сурет. Бұл компонент суреттерді, оның ішінде пиктограммалар мен метафайлдарды бейнелеуге арналған.

Shape – фигура. Бұл компоненттің көмегімен терезеге дұрыс геометриялық фигураларды - тіктөртбұрыш, шеңбер, эллипс және т.б. қоюға болады.

Bevel – жиек. Терезенің жеке бөліктерін үш өлшемді қоршаумен немесе жолақпен ерекшелеуге арналған.

ScrollBox – орағыш жолақтар панелі. **Panel** компонентінен айырмашылығы – егер орналасқан компоненттер оның

шекараларымен қиылыспаса, онда орағыш жолақтарды автоматты түрде қояды.

CheckBox – жиындық таңдаулар тізімі. Стандартты **ListBox** компонентінен айырмашылығы әрбір опциясының қасында бірнеше опцияны бірден таңдауды жеңілдететін **CheckBox** тәрізді тәуелсіз ауыстырып қосқышы бар.

Splitter – шекара. Бұл компонент формада көрінетін екі компоненттің арасына орналасады да, пайдаланушыға программаны жүргізу кезінде компоненттерді бір-бірінен бөліп тұрған шекараларын алмастыруға мүмкіндік береді.

StaticText – статикалық мәтін. Стандартты **Label** компонентінен айырмашылығы өзінің жеке **Windows** терезесінің болуымен сипатталады. Ол мәтінге қоршау жүргізуге мүмкіндік береді.

ControlBar – басқару жолағы. **DRAG&DOG** технологиясында “тақалып” тұратын компоненттерді орналастыратын контейнер қызметін атқарады.

ApplicationEvents – оқиғаны қабылдаушы. Егер бұл компонент формаға орналасқан болса, онда программаға арналған барлық **Windows** хабарламаларын алып отырады.

ValueListEditor – атау = мән жұбынан тұратын жолдар редакторы. Мұндай жұптар **Windows**-де жиі пайдаланылады.

LabeledEdit – бір жолдық редактор мен белгінің комбинациясы.

ColorBox – жүйелі түстердің бірін таңдауға арналған арнайы **ComboBox** варианты.

Chart – диаграмма. Бұл компонент мәліметтерді графикалық жолмен беруге арналған арнайы панельдер құруды жеңілдетеді.

ActionManager – әрекеттер менеджері. Төмендегі үш компонентпен бірге қосымшалар, интерфейс құруды қамтамасыз етеді.

ActionMainMenuBar – меню жолағы, ол оның опциялары **ActionManager** көмегімен құрылады.

ActionToolBar – **ActionManager** компонентінің көмегімен құрылатын пиктографиялық батырмаларды орналастыруға арналған жолақ.