

**Ўзбекистон Республикаси олий ва ўрта махсус таълим  
вазирлиги**

**Бухоро озиқ-овқат ва енгил саноат технологияси  
институти**

**“Ахборот технологиялари” кафедраси**

**Тузувчи: асс. Нарзиев У.З.**

**“Delphi да дастурлаш” фанидан**

**МАЪРУЗАЛАР МАТНИ**

**Бухоро – 2003й.**

Маърузалар матни муаллифи: «Ахборот технологиялари» кафедраси асс.  
Нарзиев У.З.

Маъруза матни 3-сон «Ахборот технологиялари» кафедраси мажлисида (7.11.03.) тасдиқланди.

Маърузалар матни Бухоро озик овкат ва енгил саноат технологияси институти «Қасбий таълим» факултети «МААТ» мутахассислигида «Delphiда дастурлаш» фанидан маърузаларни утказиш учун мулжалланган.

Такризчилар: БухОО ва ЕСТИ «Ахборот технологиялари» кафедраси мудири  
доц. Саидов Х.Қ.  
БухДУ «Информатика ва ҳисоблаш математикаси» кафедраси мудири проф.  
Жумаев З.Ш.

## Мундарижа

Кириш .....	4
1 – маъруза. Асосий тушунчалар .....	5
1.1. <i>Дастурий захираларга талаблар</i> .....	5
1.2. <i>Дастурлаш муҳити тузилиши</i> .....	6
1.3. <i>Дастурлаш муҳитининг асосий таъкил этувчилари</i> .....	6
1.4. <i>Delphi дастурлаш муҳитининг ёрдам тизими</i> .....	9
1.5. <i>Компонентлар тўплами</i> .....	12
2 – маъруза. Лоихани бошқариш .....	18
2.1. <i>Delphi лоихаси</i> .....	18
2.2. <i>Лоихани бошқариш</i> .....	20
2.3. <i>Меню бўлимлари шарҳи</i> .....	21
3-маъруза. Компонентлар тўплами шарҳи .....	25
3.1. <i>Стандарт компонентлар</i> .....	25
3.2. <i>Additional саҳифаси</i> .....	28
3.3. <i>Dialogs саҳифаси</i> .....	29
3.4. <i>System саҳифаси</i> .....	30
4-маъруза. Мултимедиа дастурларини яратиш воситалари .....	32
4.1. <i>Delphiда мултимедиа</i> .....	32
4.2. <i>TMediaPlayer компоненти</i> .....	33
4.3. <i>Мултимедиа дастурининг икки кўриниши</i> .....	34
5-Маъруза. Матнли ва график чоп қилиш .....	37
5.1. <i>Матнли режимда чоп қилиш</i> .....	37
5.2. <i>Форманнинг кўринишини чоп қилиш</i> .....	38
5.3. <i>График чоп қилиш (TPrinter объекти)</i> .....	38
5.4. <i>Printer хусусиятлари</i> .....	38
6-Маъруза. Delphiда объектларнинг хусусиятлари .....	40
6.1. <i>Умумий тушунчалар</i> .....	40
6.2. <i>Визуал компонентлар хусусиятларини дастур бажарилиши вақтида бошқариш</i> .....	42
6.3. <i>SHAPEDEMI дастури</i> .....	42
6.4. <i>SHAPEDEM2 дастури</i> .....	45
7-Маъруза. DDE .....	53
7.1. <i>DDE асослари</i> .....	53
7.2. <i>DDEнинг қўлланилиши</i> .....	54
7.3. <i>DDE-серверлар</i> .....	54
7.4. <i>DDE-мижозлар</i> .....	55
8-маъруза OLE .....	59
8.1. <i>OLE асослари</i> .....	59
8.2. <i>TOLEContainer объекти</i> .....	60
8.3. <i>OLE дастурига мисол</i> .....	62
8.4. <i>OLE объектини маълумотлар омборида сақлаш</i> .....	62
9-Маъруза. Маълумотлар омбори билан ишлаш асослари .....	64
9.1. <i>Маълумотлар омбори нима?</i> .....	64
9.2. <i>Маълумотлар омборига қўйиладиган талаблар</i> .....	64
9.3. <i>Delphiда маълумотлар омбори билан ишловчи компонентлар шарҳи</i> .....	65
9.4. <i>TDBNavigator тугмалари ва улар бажарадиган амаллар</i> .....	67
9.5. <i>Маълумотлар омборига мурожат қилувчи дастур</i> .....	68
Адабиётлар .....	71

## Кириш

Хозирги вақтда компьютер технологиялари ривожланиб бораётган вақтда асосий эътибор фойдаланувчиларга эмас, дастурчиларга қаратилмоқда. Шунини ҳисобга олиб, талабаларда дастурлар тузиш қуникмаларини ҳосил қилиш ва уларда керакли амалий дастурларни ярата олиш имконияти шакллантиришга катта эътибор қаратилмоқда.

Ўзбекистон Республикаси Кадрлар Тайёрлаш Миллий дастурида ҳам ёш авлодни янги компьютер технологияларини эгаллаши, уларнинг билимларини чет эл талабалари савиясига етказиш долзарб масала қилиб қуйилган.

“Ахборот технологиялари” кафедрасида 2003-2004 уқув йилида “Delphiда дастурлаш” фани киритилди. Delphi амалий дастурлар яратишнинг кучли муҳити ҳисобланиб, ҳозирги вақтда дунёдаги дастурчиларнинг 35%идан қупроғи шу дастурлаш тилидан фойдаланади.

Ушбу маърузалар матнида юқори савияли дастурлаш тилларидан бири ҳисобланган Delphi да амалий дастурларни яратиш усуллари келтирилган.

## 1 – маъруза. Асосий тушунчалар.

### Режа:

1. Дастурий захираларга талаблар.
2. Дастурлаш муҳити тузилиши.
3. Дастурлаш муҳитининг асосий ташкил этувчилари.
4. Delphi дастурлаш муҳитининг ёрдам тизими.

### Адабиётлар руйхати:

1. Бобровский «Delphi 5», «Питер» Москва 1997г.
2. Шумаков «Delphi 4 разработка баз данных», «Питер» Москва 1996г.
3. Пачеко, Тейксер «Delphi 5 пособие программиста», «Питер» Москва 1999 г.



### *Дастурий захираларга талаблар.*

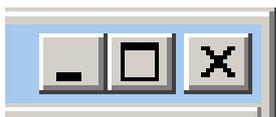
Delphi бу юқори сифатга эга бўлган дастурлар яратиш воситасидир. Delphi ни юклаш учун камида 4МВ тезкор хотирага эга бўлган 386 компьютер керак бўлади. Лекин нормал ишлаш учун камида 8МВ тезкор хотирага эга бўлган 66MHz ли 486DX компютери тўғри келади.. Delphi да яратилган унча катта бўлмаган дастурлар ихтиёрий компютерда ишлай олади. Бошқача қилиб айтганда уларга Delphi муҳити учун зарур бўлган тезкор хотира ва процессор тезлиги талаб қилинмайди.

### ***Дастурлаш муҳити тузилиши.***

Delphi дастурлаш муҳитининг ташқи кўриниши Windowsда кўриш мумкин бўлган бошқа дастурлаш муҳитлари кўринишидан фарқ қилади. Масалан, Borland Pascal for Windows 7.0, Borland C++ 4.0, Word for Windows, Program Manager – буларнинг барчаси MDI дастурлари ҳисобланади ва Delphiга ўхшамайди. MDI (Multiple Document Interface) – битта асосий ойнанинг ичида бир нечта қуйи типдаги ойналарни бошқаришни билдиради.

Delphi муҳити эса, Single Document Interface (SDI) деб номланувчи бошқа хусусиятга амал қилади ва алоҳида жойлашган бир нечта ойнадан иборат.

Агар сиз Delphiга ўхшаш SDI дастурлардан фойдалансангаз, бошқа дастурлар ойналари уни ёпиб қўймаслиги учун уларни пастга тушириб қўйиш керак. Агар бошқа дастурга ўтиш зарурияти туғилиб қоладиган бўлса, Delphi дастури бош ойнасидаги кичрайтириш тугмасини босиш кифоя.



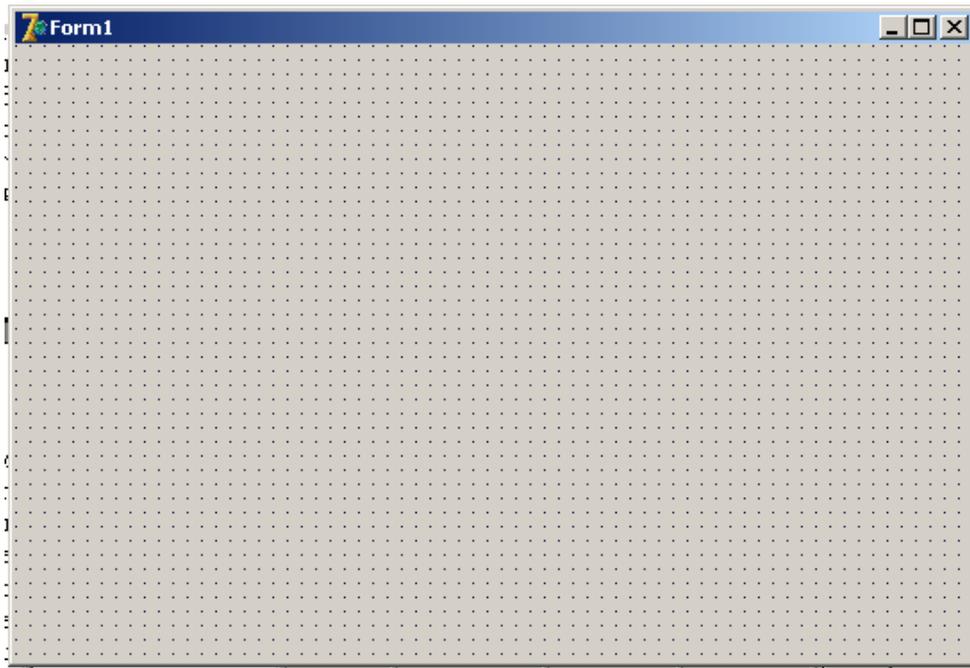
### ***Дастурлаш муҳитининг асосий ташкил этувчилари.***

Қуйида Delphiнинг асосий ташкил этувчилари келтирилган:

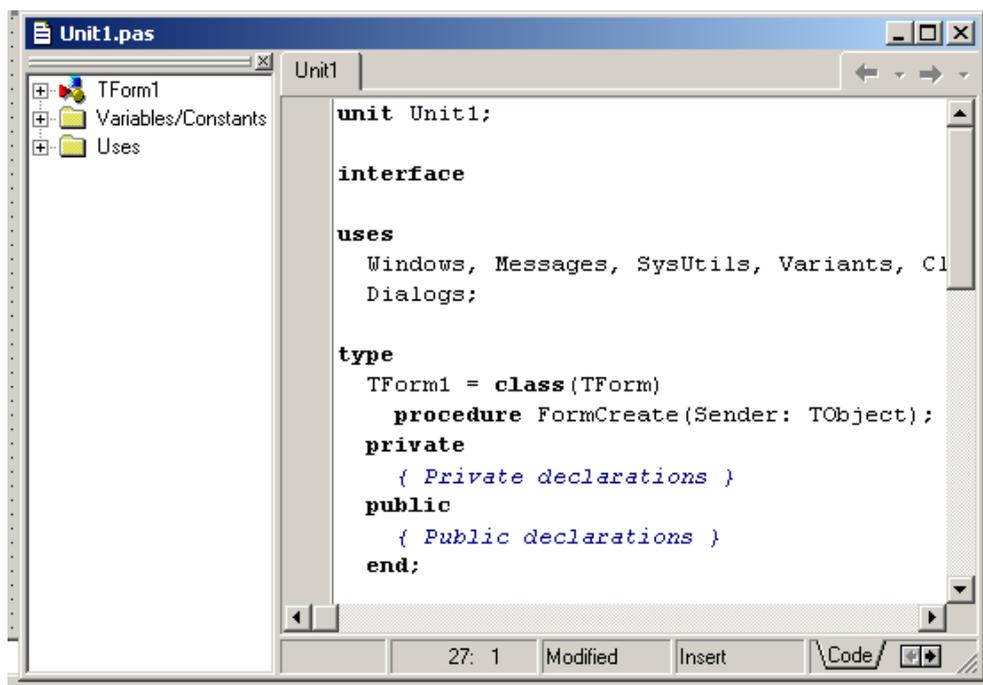
1. Формалар дизайнери (Form Designer)
2. Дастур матни муҳаррири (Editor Window)
3. Компонентлар тўплами (Component Palette)
4. Объектлар инспектори (Object Inspector)
5. Ёрдам тизими (On-line help)

Delphiнинг булардан ташқари сизга дастурлаш муҳити ва дастурни созлаш учун хизмат қилувчи тизим менюси, ускуналар чизгичи каби бошқа воситалари ҳам бор.

Delphiда дастурчилар кўп вақтларини Формалар дизайнери ва Дастур матни муҳаррири орасида ўтиш билан ўтказадилар. Сиз дастурлашни бошлашдан олдин шу икки муҳим элементни ажрата олишингизга ишонч ҳосил қилинг. Формалар дизайнери 1-расмда, Дастур матни муҳаррири 2-расмда келтирилган.



1-расм. Формалар дизайнери.



2-расм. Дастур матни муҳаррири

Delphiдаги Формалар дизайнерининг тузилиши шу қадар оддийки, дастурнинг ташқи кўринишини яратиш болалар ўйинига ўхшаб кетади. Формалар дизайнери дастлаб оддий бўш ойнадан иборат бўлади ва сиз уни кейинчалик компонентлар тўпламидан олинган объектлар билан тўлдиришингиз мумкин.

Формалар дизайнерининг муҳимлигига қарамасдан, дастурчиликнинг кўп вақтлари Дастур матни муҳарририда кечади. Чунки бу ерда дастурнинг асосий коди (иш бажарувчи қисми) ёзилади.

Компонентлар тўплами сизга формалар дизайнерига ўрнатиш ва дастурнинг ташқи кўринишида иштирок этувчи объектларни танлаш учун хизмат қилади. Компонентлар тўплamidан фойдаланиш учун аввал керакли объект устига сичқончани босинг ва сўнгра формалар дизайнерига сичқончани босиш билан танланган объектни формага қўйишингиз мумкин.

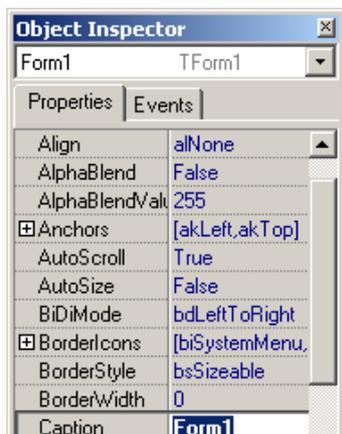


3- расм. Компонентлар тўплами.

Компонентлар тўпламида объектлар гуруҳлар буйича алоҳида саҳифаларга бўлинган. Тўплamidа Standard, Additional, Dialogs ва ҳо казо саҳифалар йиғилган. Агар сиз бу номларнинг ихтиёрийсини боссангиз, шу гуруҳ объектлари жойлашган саҳифага ўтишингиз мумкин.

Масалан, сиз формага TEdit объектini ташладингиз; сиз уни бир жойдан иккинчи жойга кўчиришингиз ва унинг ўлчамларини ўзгартиришингиз мумкин. Бошқа кўпчилик объектлар билан ҳам худди шундай ишларни амалга ошириш мумкин. Лекин, дастур бажарилишида формада кўринмайдиган (TMenu ёки TdataBase каби) объектлар ўз кўринишини ўзгартирмайди.

Формалар дизайнеридан чап томонда сиз объектлар инспекторини кўришингиз мумкин. Бу ойнадаги маълумотлар формада танланган объектнинг типига қараб, ўзгариб боришига эътибор беринг. Ҳар бир компонент бу объектдир, ва унинг кўриниши ҳамда ҳаракатини объектлар инспектори ёрдамида ўзгартириш мумкинлигини унутманг!



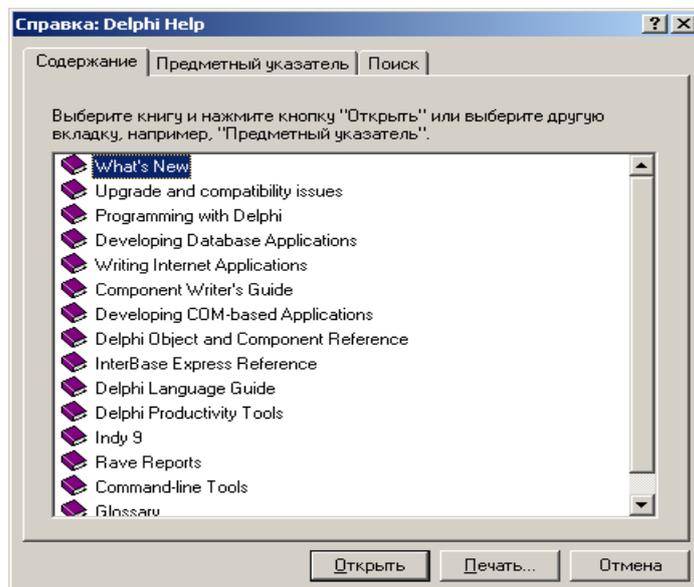
4-расм. Объектлар инспектори.

Объектлар инспектори жорий объектнинг ҳаракатларини аниқлаш учун икки саҳифадан ташкил топган. Биринчи саҳифа – хусусиятлар рўйхати бўлса, иккинчиси объектнинг ҳолатлари рўйхатидир. Жорий объектнинг бирор томонини ўзгартириш керак бўлса, сиз албатта Объектлар инспекторига мурожаат қилишингиз керак. Мисол учун сиз TLabel элементининг номини ўзгартиришингиз, ёки унинг Caption, Left, Top, Height, ва Width каби хусусиятлари ёрдамида жойлашишини ўзгартиришингиз мумкин.

Объектлар инспектори ойнасининг қуйи қисмидаги саҳифа номларидан объектларнинг хусусиятлари ва ҳолатлари ойнасига ўтиш мумкин. Ҳолатлар саҳифаси дастур матни муҳаррири билан бевосита боғлиқ бўлиб, ундаги ҳолатларнинг ихтиёрийсига сичқончани икки марта боссак, объектнинг шу ҳолатига тегишли дастур сарлавҳаси муҳаррирда пайдо бўлади ва сизга шу ҳолатда бажариладиган ишларни дастур кўринишида ёзиш имконияти берилади. Delphi бу имкониятини кейинроқ тўлиқ кўриб ўтамыз.

### ***Delphi дастурлаш муҳитининг ёрдам тизими***

Delphi муҳитининг охириги муҳим қисми бу ёрдам тизимидир (on-line help). Бу воситага мурожаат қилиш учун аввал меню сатридан Help сўнгра Contents буйуғини танлаш керак. Экранда 5-расмда кўрсатилган - **ихтиёрий маълумотни тез топиш ёрдам тизими** намоён бўлади.



5-расм. Delphi ёрдам тизими

Ёрдам тизими жорий ҳолатга боғлиқ ҳисобланади, яъни сиз F1 тугмасини босишингиз билан жорий ҳолатга мос маълумотни олишингиз мумкин. Масалан, Объектлар инспекторида бирор хусусиятни танлаб, F1 тугмасини боссак – жорий хусусиятнинг қўлланилиши ҳақида маълумотга эга бўламиз

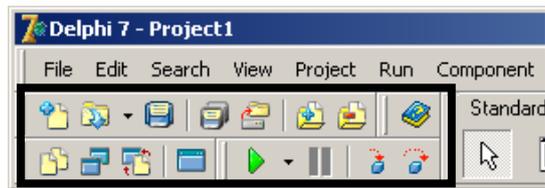
#### Қўшимча элементлар

Бу бўлимда биз асосан дастурлаш учун ёрдамчи ҳисобланган учта қўшимча элементлар ҳақида гапирамиз:

- Меню (Menu System)
- Тезкор мурожаат тугмалари (SpeedBar)
- Расмлар муҳаррири (Image Editor)

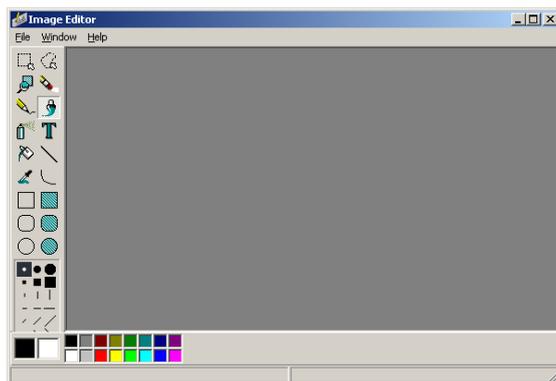
Меню Delphiнинг энг тезкор ва мослашувчан воситасидир, чунки, ундаги буйруқларни тезкор тугмалар ёрдамида ҳам бажариш мумкин. Менюнинг қулайлиги шундаки, бу ерда тугмачалар ва белгилардан фарқли равишда аниқ сўзлар ишлатилади.

SpeedBar бевосита менюниг қуйи қисмида, компонентлар тўпламидан чап томонда жойлашади.



6-расм: Тезкор муружаат тугмалари (SpeedBar)

7-расмда келтирилган Расмлар муҳаррири Windowsдаги Paint дастури каби ишлайди. Сиз бу объектга Tools менюснинг Image Editor буйруғини танлаш билан муружаат қилишингиз мумкин.



7-расм. Расмлар муҳаррири

Энди дастурчилар Delphiда кўп қўллайдиган элементлар билан танишиб ўтамиз.

### Инструментал воситалар

Юқорида кўриб ўтилганлардан ташқари Delphi таркибида яна бешта восита мавжуд. Улар қуйидагилар:

- Ички дастур назоратчиси
- Ташқи дастур назоратчиси
- Буйруқ сатри компилятори
- WinSight
- WinSpector

Бу воситаларнинг алоҳида гуруҳга булинганлиги уларнинг бошқаларидан кўра камроқ амамиятга молик эканлигида эмас, балки, улар дастурлаш жараёнида анча мавҳум техник вазифаларни бажаришидадир.

Delphiда кучли дастурчи бўлиш учун сиз Delphi дастур назоратчиси қандай тузилганлигини билишингиз керак. Назоратчи сизга ёзилган дастурни қадамба-

қадам текшириш, бир мартада дастур матнининг бир қаторини бажариш ва дастурдаги узгарувчилар қийматларини алоҳида кўриш ойнаси (Watch)да кузатиш имконини беради.

Юқорида санаб ўтилган беш элементдан энг муҳими бўлган ички дастур назоратчиси дастур матни муҳаррири ойнасида ишлайди. Ташқи дастур назоратчисининг ичкидан фарқи у анча тез ва кучлилигидадир. Лекин уни ишлатиш учун аввал Delphi муҳитидан чиқиш керак.

Энди Компиляторлар ҳақида гаплашайлик. Ташқи компилятор DCC.EXE деб номланади, у асосан дастурни ташқи дастур назоратчиси ёрдамида ўтказишдан олдин ишлатилади. Лекин кўпчилик дастурчилар ўз дастурларини Delphi муҳитининг ўзида назоратдан ўтказадилар.

WinSight ва WinSpector Windows муҳитида кучли дастурчилар учун қизиқарли ҳисобланади. Бу воситалар иккиламчи ҳисобланади ва фақат тор техник мақсадларда ишлатилади.

Бу икки воситадан WinSight анча фойдалироқ. Унинг асосий вазифаси Windows томонидан юборилаётган хабарлардан сизни воқиф қилиб туришдир. Delphi малакасиз фойдаланувчилардан бу тизимнинг мураккаб қисмларини яширишга мўлжалланган бўлсада, Windows ҳолатларни бошқариш мумкин бўлган тизим саналади. Delphi сизга Windowsнинг ҳамма хабарларига жавоб бериш имкониятини беради. Натижада, малакали фойдаланувчиларга WinSight зарур бўлиб қолади.

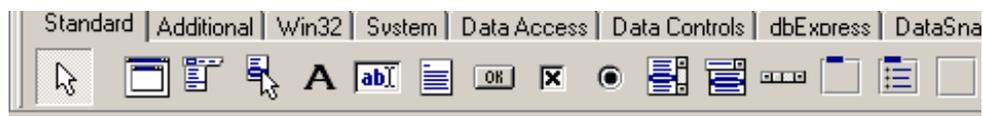
WinSpector ЭХМнинг жорий ҳолатини матнли файлларга сақлаб қўйиш учун хизмат қилади; ва сиз бу файллар ёрдамида дастурнинг қайси қисми ишламаётганлигини ва нималарни ўзгартириш кераклигини билиб олишингиз мумкин.

Стандарт компонентлар

### ***Компонентлар тўплами***

Delphi дастурлаш муҳити билан танишишни давом эттирар эканмиз, компонентлар тўплами ҳақида гапиришга тўғри келади.

Компонентлар тўпламининг биринчи саҳифасида 14та объектлар жойлашган бўлиб, ҳамма дастурчилар албатта ушбу объектлардан фойдаланишган. Чунки, жуда ҳам камчилик тугма, рўйхат, киритиш ойналари ва ҳо казолардан фойдаланмасдан дастур туза олади. Бу объектларнинг барчаси Windowsдаги сичқонча ва мулоқот ойнаси каби муҳимдир.



8-расм. Стандарт Компонентлар тўплами

Ҳар бир саҳифадаги компонентлар тўплами ва жойлашишини ўзгартириш мумкин. Демак, сиз мавжуд компонентларни ўрнини ўзгартириш билан биргаликда уларнинг таркибига янгиларини қўшишингиз мумкин.

Delphiнинг стандарт компонентлари ва уларни ишлатиш ҳақидаги қисқача изоҳ қуйида келтирилган. Бу компонентларни ўрганаётган вақтда қўл остингизда компьютер бўлгани маъқул, чунки бирваракайига уларнинг ишлаш принципи билан ҳам танишишингиз мумкин.

-  **TMainMenu** – ўз дастурингизга бош меню қўшиш имконини беради. *TmainMenu*ни формага қўйган вақтда у кичкина белги кўринишини олади; бундай белгиларни "кўринмас компонент" деб атайдилар, чунки, дастур бажарилар вақтида улар экранда кўринмайди. Менюни яратиш уч қадамдан иборат: (1) TMainMenu формага ўрнатиш, (2) объектлар инспекторининг Items хусусияти ёрдамида Меню дизайнерини чақириш, (3) Меню Дизайнерида меню қисмларини киритиш.

-  **TPopupMenu** ёрдамчи менюни яратиш. Бу меню сичқончанинг ўнг тугмаси босилганда кўринади.

-  **TLabel** матнларни экранда намоиш қилиш учун қўлланади. Агар сиз объектлар инспекторининг Font хусусиятига сичқончани икки марта боссангиз, *Label*нинг шрифти, ранги, ҳарфлар ранги ва ўлчамларини ўзгартишингиз мумкин.

-  **TEdit** – Windowsдаги стандарт маълумот киритиш ойнаси. У қисқа матнларни намоиш қилиши ва дастур бажарилиш вақтида фойдаланувчига ўз маълумотларини киритиш имконини беради.

-  **TMemo** – TEditнинг бошқача кўриниши. Катта мантлар билан ишлашни кўзда тутди. TMemo матн қисмларини қатордан қаторга кўчириши, буферда сақлаши, охирги амаллардан воз кечиши, умуман олганда оддий мухаррир амалларини бажара олади. TMemo 32Кб матн ҳажми билан чегараланади, бу тахминан 10-20 саҳифа матнга тенгдир.

-  **TButton** дастур бажарилаш вақтида тугма босилиши билан бирор амал бажарилишини кўзда тутди. Delphiда ҳамма нарса оддий. TButtonни формга кўйиб, унга икки марта сичқон тугмасини боссак, биз дастур матни мухарририда тугма босилиш ҳолатига дастур тузишимиз мумкин бўлган ойнага ўтамиз. Энди дастур матнини бажариладиган буйруқлар коди билан тўлдириш керак. Масалан:

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

```
begin
```

```
MessageDlg('Ишларингиз яхшими?',mtConfirmation,mbYesNoCancel,0);
```

```
end;
```

-  **TCheckBox** чап томонида кичкина дарчаси бўлган матн сатрини акслантиради. Дарчага унинг белгиланганлигини билдирувчи белги қуйиш мумкин. Масалан, Wordдаги Печать мулоқот ойнасининг Печать в файл бўлимида айнан шу объект ишлатилган.

-  **TRadioButton** бир нечта ҳолатлардан бирини танлаш имконини беради. Масалан, Wordдаги Печать мулоқот ойнасининг чоп қилиниши керак бўлган саҳифалар оралағини танлаш бўлимида айнан шу объект ишлатилган.

-  **TListBox** рўйхатли маълумотларни намойиш қилишга мўлжалланган масалан Wordдаги Открыт буйруғи ёрдамида очиладиган мулоқот ойнасидаги файллар ва каталоглар рўйхати айнан шу объектга жойлаштирилган.

-  **TComboBox** ташқи кўринишидан ListBoxга ўхшаб кетади, улар орасидаги фарқ ComboBoxнинг юқорисида маълумот киритиш мумкин бўлган майдони ҳам бор. ComboBoxнинг кўплаб турлари мавжуд, кўп ишлатиладигани эса, рўйхати пастга очиладиган (drop-down combo box)дир.

-  **TScrollbar** – ўтказиш йўлакчаси. Кўпинча таҳрирланадиган ёки кўринадиган маълумотлар экран чегараларидан чиқиб кетганда пайдо бўлади.

-  **TGroupBox** Windowsга формадаги объектларнинг қандай жойлашганлигини билдириш учун қуллаш мумкин.

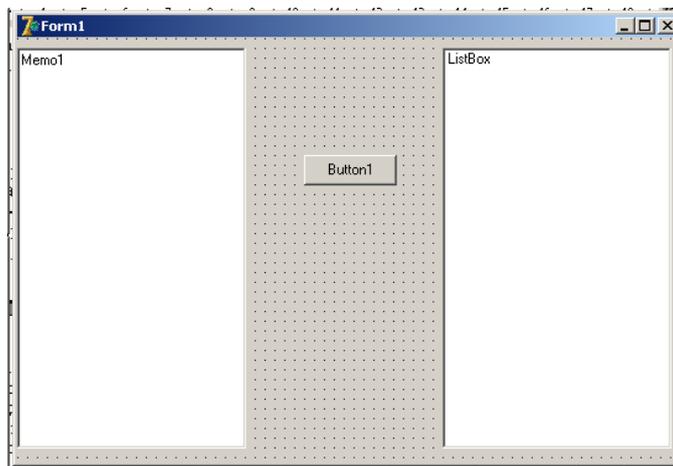
-  **TPanel** – TGroupBoxга ўхшаш объект бўлиб, безаш учун ишлатилади.

Агар сизга бундан кўра тўлиқроқ маълумотлар керак бўлса, Delphiнинг ёрдам тизимидан фойдаланинг

#### Объектлар Инспектори ҳақида.

Олдинроқ биз Объектлар Инспектори (Object Inspector) билан бир оз танишиб ўтдик. Энди у билан яқинроқ танишиш вақти етди. Объектлар Инспекторининг асосий вазифаси формага ташланган ихтиёрий объектнинг хусусиятларини таҳрирлаш ҳисобланади. Бундан ташқари у ёрдамида форманинг хусусиятларини ҳам ўзгартириш мумкин.

Объектлар Инспекторини ўрганишнинг энг қулай усули – у билан ишлашдир. Аввал янги лоиха очамиз File | New Project. Сўнгра формага TMemo, TButton ва TListBox объектларини 9-расмда кўрсатилганидек жойлаштиринг.



9-расм. Формага объектларни жойлаштириш.

Хусусиятлар билан ишлашни бошлашда `Ctrl3D` хусусиятидан бошлаймиз (жимлик қоидаси бўйича фаол). Формага сичқон тугмасини босинг, Объектлар Инспекторига ўтинг ва `Ctrl3D` хусусияти қийматини сичқонча ёрдамида бир неча марта ўзгартириб кўринг. Бу амал форманинг ташқи кўринишини ўзгартираётганлигига эътибор қаратинг. Форманинг `Ctrl3D` хусусияти қийматини ўзгартириш шу формага ташланган ҳамма объектларнинг шу хусусияти қийматини ўзгартиради.

Формага дастлабки ҳолатини қайтаринг ва ундаги `TMemo` объектини белгиланг, сўнгра `<Shift>` тугмасини босиб туриб, `TListBox` объектини белгиланг. Энди сиз формадаги бир нечта объектларни белгилай оласиз. Объектлар гуруҳини белгилаганда Объектлар Инспекторида фақат шу гуруҳ объектлари учун умумий бўлган хусусиятлар кўринади ва уларнинг қиймати ўзгартирилиши белгиланган объектларнинг барчаси учун амал қилади. Агар бу гуруҳда умумий хусусиятларга эга бўлган объектлар бўлмаса, Объектлар Инспекторида ҳеч қандай хусусият кўринмайди.

## Назорат саволлари

1. Delphi дастурлаш муҳитининг тизим заҳираларига талаблари.
2. Формалар дизайнерининг вазифалари.
3. Дастур коди муҳаррири вазифалари.
4. Объектлар инспектори нима?
5. Компонентлар тўпламидан нима мақсадда фойдаланиш мумкин?

Таянч иборалар.

Формалар дизайнери

Объектлар инспектори

Компонентлар тўплами

TListBox

TMemo

TPanel

TButton

TEdit

TLabel.

## **2 – маъруза. Лоихани бошқариш.**

### **Режа**

1. Delphi лоихаси.
2. Лоихани бошқариш

### Адабиётлар руйхати:

1. Компьютердаги ёрдам файллари.
2. Бобровский «Delphi 5», «Питер» Москва 1997г.
3. Шумаков «Delphi 4 разработка баз данных», «Питер» Москва 1996г.
4. Пачеко, Тейксеря «Delphi 5 пособие программиста», «Питер» Москва 1999 г.
5. Фаронов «Delphi 4 учебное пособие», «Питер» Москва 1995 г.

### ***Delphi лоихаси.***

Delphiнинг ихтиёрий лоихаси камида олтига файл яратади. Улардан учтаси лоиха томонидан яратилади ва дастурчилар уларнинг таркибини кўпинча ўзгартирмайдилар. Улар қуйидагилар:

- Лоиханинг бош файли, дастлаб PROJECT1.DPR деб номланади.
- Дастурнинг дастлабки модули /unit/, у ишнинг бошида автоматик равишда яратилади. Файл UNIT1.PAS деб номланади ва унинг номини ихтиёрий ном билан алмаштириш мумкин. Масалан, MAIN.PAS.
- Бош форма файли жимлик қондасига биноан UNIT1.DFM деб номланади, у бош форманинг ташқи кўриниши ҳақидаги маълумотларни сақлайди.
- PROJECT1.RES файли автоматик ҳолда яратилади ва лоиха учун белгини сақлайди.
- PROJECT1.OPT файли матнли файл ҳисобланиб, сиз компиляторда қўллаган ўрнатмаларингизни сақлайди.
- PROJECT1.DSK файли ишчи соҳанинг ҳолати ҳақидаги маълумотларни сақлайди.

Албатта лоиха номини ўзгартириганда RES, OPT ва DSK кенгайтмали файлларнинг номлари ҳам ўзгаради.

Дастур компиляциясидан кейин қуйидаги кенгайтмали файллар ҳосил бўлади:  
DCU – тайёр модуллар.

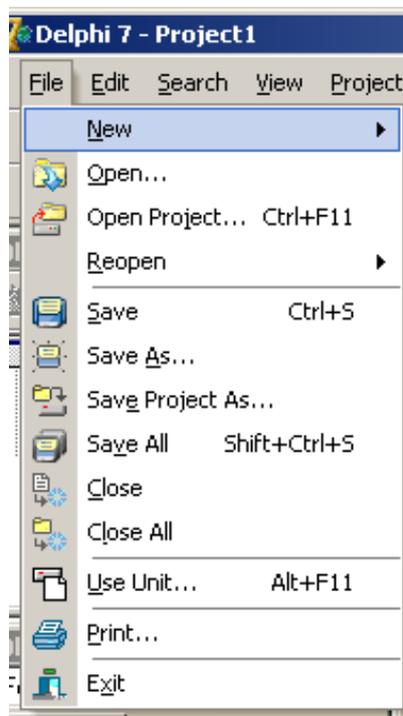
EXE – ишчи файл.

DSM – дастурнинг муҳитда ишга тушиши учун хизмат қилувчи файл, унинг ҳажми жуда катта ва иш яқунлангач уни ўчириш тавсия қилинади.

~РА, ~DP - backup. – муҳаррирнинг вақтинчалик файллари.

Менюнинг “File” бўлими

**Лоиханинг файллари билан ишлаш учун “File” менюсини танланг. “File” менюси қуйидагича тузилган:**



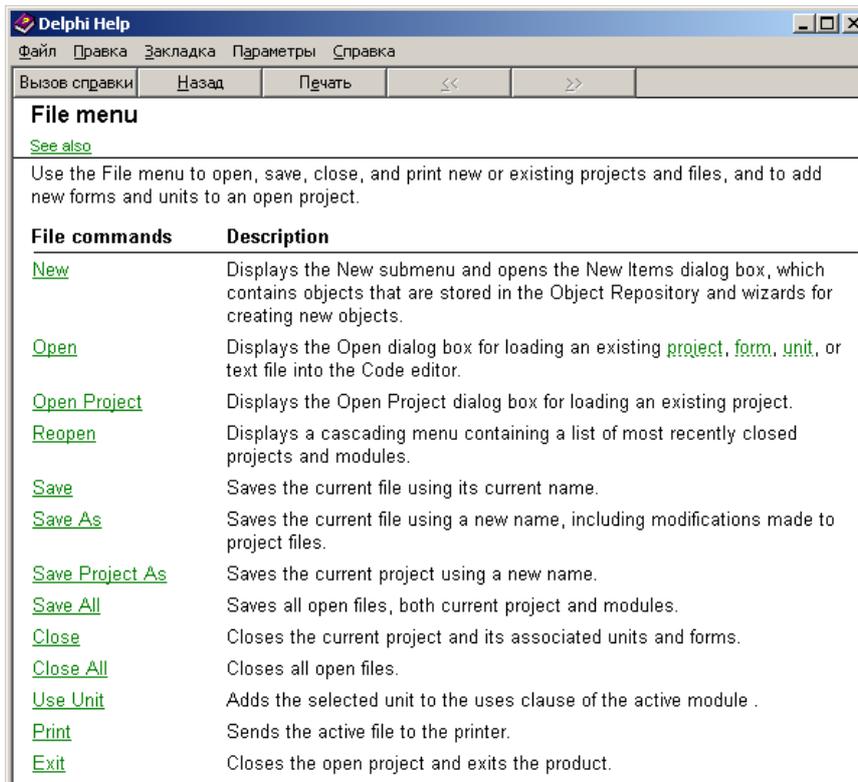
10-расм. Файл менюси кўриниши

Кўриб турганингиздай бу ерда бешта бўлим мавжуд:

- Биринчи бўлим лоиха билан тўлиқ ишлашга мўлжалланган.
- Иккинчи бўлим лоиханинг формалари, модуллари ва қисмлари устидан назоратга мўлжалланган.
- Учинчи бўлим лоихага файлларни қўшиш ва олиб ташлаш учун хизмат қилади.
- Тўртинчи бўлим чоп қилишни бошқаради.

- Бешинчи бўлим Delphідан чиқиш.

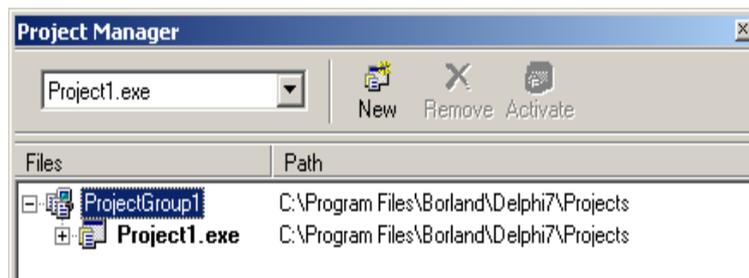
“File” менюсининг ҳар бир бўлими ёрдам тизимида изоҳланган бўлганлиги учун уларга тўхтаб ўтирмаймиз. Буларни куриш учун “File” менюсини танлаб, F1 тугмасини босинг.



11-расм Delphида “File” менюсидан фойдаланиш ҳақидаги ёрдам.

### *Лоиҳани бошқариш*

Энди сиз Файл менюси ёрдамида лоиҳа яратишни ўрганиб олдингиз ва лоҳани бошқарувчи Лоиҳалар Менежменти шарҳига ўтсак бўлади. Лоиҳалар Менеждери икки қисмдан иборат бўлиб, биринчи қисм бошқарувчи тугмалар ва ундан пастдаги иккинчи қисмда лоиҳа таркибига кирувчи модуллар номи ва уларнинг дискдаги ўрни кўрсатилган.



12-расм. Лоиҳалар менеждери.

Сиз бошқарувчи тугмалар ёрдамида лоихага янги файлларни қўшишингиз ёки олиб ташлашингиз мумкин. Бу ўзгаришлар дастлабки файлга ўзгартиришлар киритади, яъни лоихага янги модул қўшилса унга йўлланма DPR кенгайтмали файлда автоматик ҳолда ёзилади..

### **Меню бўлимлари шарҳи**

Менюнинг “File” бўлимини юқорида кўриб ўтдик. Қуйида менбнинг - “Edit”, “Search”, “View” ва “Compile” бўлимларини қисқача кўриб ўтамыз.

#### **Менюнинг “Edit” бўлими**

“Edit” менюси “Undo” ва “Redo” буйруқларини сақлайди, улар муҳаррирда ишлаётганда бехосдан қилинган хатолардан (масалан, матннинг керакли қисми ўчириб юборилди) воз кечишга мўлжалланган.

“Cut”, “Copy”, “Paste” ва “Delete” буйруқлари Windowsнинг бошқа дастурларидаги каби, лекин уларни фақат матнларга эмас, балки, объектларга ҳам қўллаш мумкин.

“Bring To Front”, “Send To Back”, “Align” ва “Size” объектларнинг формада туриши ва уларнинг ўлчамларини бошқаришга мўлжалланган.

#### **Менюнинг “Search” бўлими.**

“Search” менюсида “Find Error” (хатоликларни қидириш) буйруғи мавжуд бўлиб, у дастур бажарилиш вақтидаги хатоликларни топишга мўлжалланган. Хатолик ҳақидаги хабарда чиқарилган рақамни “Find Error” мулоқот ойасига киритинг ва муҳит дастурдаги хатолик содир бўлган қисмни кўрсатади.

#### **Менюнинг “View” бўлими.**

“View” менюсининг таркиби қуйидагича:

- Project Manager (Лоиҳа Менеджери).
- Project Source –муҳаррирга лоиханинг бош файли (DPR) ни юклайди.
- Экранда Object Inspectorни кўрсатиш керак ёки йўқлигини билдирувчи ўрнатма.
- Экранда Alignment Palette кўрсатиш керак ёки йўқлигини билдирувчи ўрнатма.

- Browser – дастурдаги объектлар иерархиясини намоиш қилувчи воситани чақириш.
- Watch, Breakpoint ва Call Stack – дастурни бошқариш билан боғлиқ ва уларга кейинроқ тўхталамиз.
- Component List – Компонентлар тўплами. Компонентларни номи бўйича қидиришда ёки сичқонча йўқлигида қўлланилади.
- Window List – Delphi муҳитида очилган ойналар рўйхати.
- Toggle Form/Unit, Units, Forms – форма ва мос модуллар орасида утиш, форма ва модуллар рўйхатдан танланади.
- New Edit Window – муҳаррирнинг қўшимчак ойнасини очади. Бу кўпинча бир файлнинг икки нусхасини билан ишлаганда қулайлик яратади.
- SpeedBar и Component Palette – экранда намоиш қилиш ўрнатмалари.

#### Менюнинг **“Compile”** бўлими.

Менюнинг “Compile” бўлимида лоиҳани ишга тайёрлаш (compile) ёки қайта ишлаш (build) мумкин. Агар Compile ёки Run танланса, Delphi фақат охириги компиляциядан кейин ўзгартирилган модулларни тайёрлайди. Build all эса, лоиҳа таркибидаги ҳамма модуллар ва дастур матнларини тайёрлайди. Syntax Check буйруғи эса, фақат дастур матни буйруқлари тўғри ёзилганлигини текширади.

Энг пастки – Information буйруғи дастур ҳақидаги умумий маълумотларни чиқаради.

#### Менюнинг **“Run”** бўлими.

“Run” бўлимидан дастурни ишга тайёрлаш ва дастур учун буйруқ сатри параметрларини бериш мақсадида фойдаланиш мумкин.

#### Менюнинг **Options** бўлими.

“Options” тизим енусининг анча мураккаб қисмидир. Сиз бу ердан лоиҳа ва Delphiнинг бутун ишчи муҳит учун ўрнатмаларни ўзгартиришингиз мумкин. В “Options”нинг етти буйруғи мавжуд:

Project

Environment

Tools

Gallery

Open Library

Install Components

Rebuild Library

Дастлабки тўрт буйруқ мулоқот ойналарини чақиради. Қуйида менюнинг “Options” бўлимига қисқача шарҳ келтириб ўтилган:

- Project – жорий лоихага тўғридан-тўғри таъсир қилувчи параметрларни ўрнатиш.

- Environment – дастурлаш муҳитининг тузилишини бошқариш (IDE). Масалан, бу ерда мухаррирда ишлатиладиган рангларни ўзгартириш мумкин.

- Tools – бош менюнинг Tools бўлимидаги ташқи дастурларни юклаш ва улардан воз кечиш. Масалан, сиз бирор модулдан кўп фойдалансангиз, уни рўйхатга кўшиб қўйиш мумкин.

- Gallery – Формалар ва лоихалар эксперти учун специфик ўрнатмаларни қўллайди.

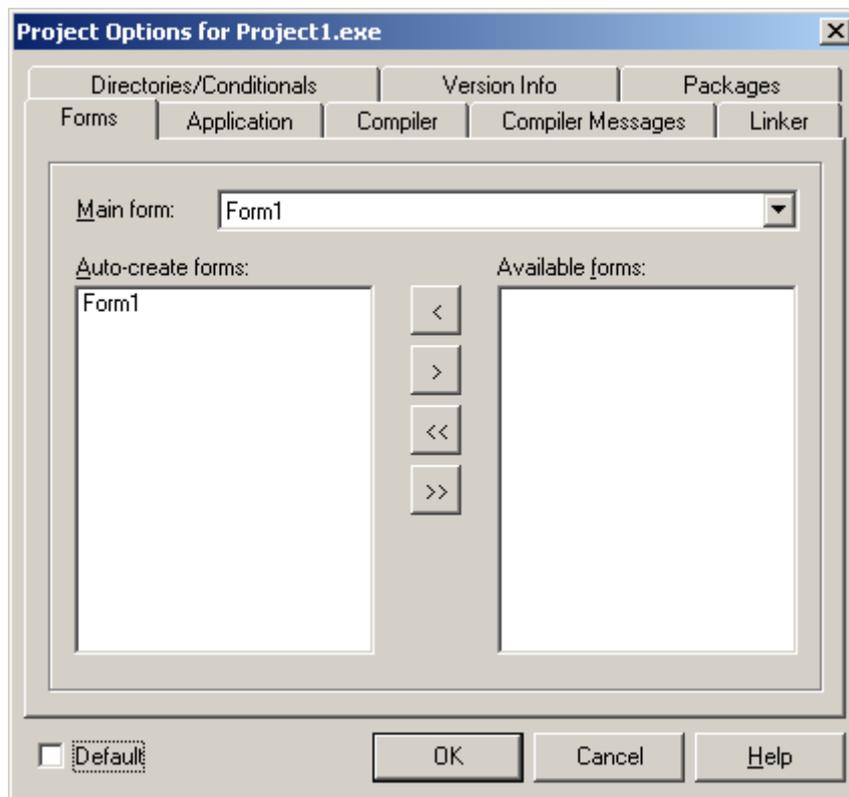
- Охирги учта буйруқ компонентлар тўпламини тузишга ёрдам беради.

Options | Project мулоқот ойнаси бешта асосий саҳифадан иборат:

- Forms саҳифасида лоихада ишлатиладиган барча формалар келтирилган;
- Application саҳифасида сиз дастур учун керак бўладиган сарлавҳа, ёрдам файли ва белгиларни аниқлашингиз мумкин.

- Compiler саҳифаси дастур матнидаги хатоликлар, дастур абажарилиш вақти хатоликлари, уни тайёрлаш каби вазифалар билан шуғулланади.

- Linker саҳифасида дастурни боғлаш шароитларини аниқлаш мумкин.



13-расм Options | Project мулоқот ойнаси.

- Directories/Conditionals саҳифаси – жорий лоиха учун фойдаланиладиган каталоглар кўрсатилади.

#### Назорат саволлари:

1. Дастур ишлаши учун яратиладиган файллар.
2. Дастур томонидан яратиладиган файлларнинг вазифалари нималардан иборат?
3. Delphi муҳити менюлари.
4. Ёрдам тизимидан фойдаланиш тартиби.
5. Лоихани бошқариш йўллари.

#### Таянч иборалар:

DCU; EXE; DSM; Лоиха;

Дастурни боғлаш;

Дастурларни юклаш;

Лоиха менежери.EXE

### 3-маъруза. Компонентлар тўплами шарҳи.

#### Режа:

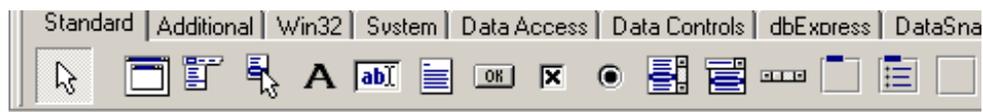
6. Стандарт компонентлар
7. Additional саҳифаси
8. Dialogs саҳифаси
9. System саҳифаси

#### Адабиётлар руйхати:

6. Компьютердаги ёрдам файллари.
7. Бобровский «Delphi 5», «Питер» Москва 1997г.
8. Шумаков «Delphi 4 разработка баз данных», «Питер» Москва 1996г.
9. Пачеко, Тейксер «Delphi 5 пособие программиста», «Питер» Москва 1999 г.
10. Фаронов «Delphi 4 учебное пособие», «Питер» Москва 1995 г.

#### *Стандарт компонентлар*

Компонентлар тўпламининг биринчи саҳифасида 14та объектлар жойлашган бўлиб, ҳамма дастурчилар албатта ушбу объектлардан фойдаланишган. Чунки, жуда ҳам камчилик тугма, рўйхат, киритиш ойналари ва ҳо казолардан фойдаланмасдан дастур туза олади. Бу объектларнинг барчаси Windowsдаги сичқонча ва мулоқот ойнаси каби муҳимдир.



14-расм. Стандарт Компонентлар тўплами

Ҳар бир саҳифадаги компонентлар тўплами ва жойлашишини ўзгартириш мумкин. Демак, сиз мавжуд компонентларни ўрнини ўзгартириш билан биргаликда уларнинг таркибига янгиларини қўшишингиз мумкин.

Delphiнинг стандарт компонентлари ва уларни ишлатиш ҳақидаги қисқача изоҳ қуйида келтирилган. Бу компонентларни ўрганаётган вақтда қўл остингизда

компьютер бўлгани маъкул, чунки бирваракайига уларнинг ишлаш принципи билан ҳам танишишингиз мумкин.

Курсор – компонент ҳисобланмайди, балки, бирор объектни танлашдан воз кечиш воситасидир.

-  **TMainMenu** – ўз дастурингизга бош меню қўшиш имконини беради. *TMainMenu*ни формага қўйган вақтда у кичкина белги кўринишини олади; бундай белгиларни "кўринмас компонент" деб атайдилар, чунки, дастур бажарилар вақтида улар экранда кўринмайди. Менюни яратиш уч қадамдан иборат: (1) TMainMenu формага ўрнатиш, (2) объектлар инспекторининг Items хусусияти ёрдамида Меню дизайнерини чақириш, (3) Меню Дизайнерида меню қисмларини киритиш.

-  **TPopupMenu** ёрдамчи менюни яратиш. Бу меню сичқончанинг ўнг тугмаси босилганда кўринади.

-  **TLabel** матнларни экранда намойиш қилиш учун қўлланади. Агар сиз объектлар инспекторининг Font хусусиятига сичқончани икки марта боссангиз, *Label*нинг шрифти, ранги, ҳарфлар ранги ва ўлчамларини ўзгартишингиз мукин.

-  **TEdit** – Windowsдаги стандарт маълумот киритиш ойнаси. У қисқа матнларни намойиш қилиши ва дастур бажарилиш вақтида фойдаланувчига ўз матнини киритиш имконини беради.

-  **TMemo** – TEditнинг бошқача кўриниши. Катта матнлар билан ишлашни кўзда тутди. TMemo матн қисмларини қатордан қаторга кўчириши, буферда сақлаши, охириги амаллардан воз кечиши, умуман олганда оддий муҳаррир амалларини бажара олади. TMemo 32Кб матн ҳажми билан чегараланади, бу тахминан 10-20 саҳифа матнга тенгдир.

-  **TButton** дастур бажарилар вақтида тугма босилиши билан бирор амал бажарилишини кўзда тутди. Delphiда ҳамма нарса оддий. TButtonни формага қўйиб, унга икки марта сичқон тугмасини боссак, биз дастур матни муҳарририда тугма

босилиш ҳолатига дастур тузимиз мумкин бўлган ойнага ўтамиз. Энди дастур матнини бажариладиган буйруқлар коди билан тўлдириш керак. Масалан:

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

```
begin
```

```
MessageDlg('Ишларингиз яхшими?',mtConfirmation,mbYesNoCancel,0);
```

```
end;
```

-  **TCheckBox** чап томонида кичкина дарчаси бўлган матн сатрини акслантиради. Дарчага унинг белгиланганлигини билдирувчи белги қуйиш мумкин. Масалан, Wordдаги Печать мулоқот ойнасининг Печать в файл бўлимида айнан шу объект ишлатилган.

-  **TRadioButton** бир нечта ҳолатлардан бирини танлаш имконини беради. Масалан, Wordдаги Печать мулоқот ойнасининг чоп қилиниши керак бўлган саҳифалар оралағини танлаш бўлимида айнан шу объект ишлатилган.

-  **TListBox** рўйхатли маълумотларни намойиш қилишга мўлжалланган масалан Wordдаги Открыт буйруғи ёрдамида очиладиган мулоқот ойнасидаги файллар ва каталоглар рўйхати ойна шу объектга жойлаштирилган.

-  **TComboBox** ташқи кўринишидан ListBoxга ўхшаб кетади, улар орасидаги фарқ ComboBoxнинг юқорисида маълумот киритиш мумкин бўлган майдони ҳам бор. ComboBoxнинг кўплаб турлари мавжуд, кўп ишлатиладигани эса, рўйхати пастга очиладиган (drop-down combo box)дир.

-  **TScrollbar** – ўтказиш йўлакчаси. Кўпинча таҳрирланадиган ёки кўринадиган маълумотлар экрандан чиқиб кетганда пайдо бўлади.

-  **TGroupBox** Windowsга формадаги объектларнинг қандай жойлашганлигини билдириш учун қуллаш мумкин.



- **TPanel** – TGroupBoxга ўхшаш объект бўлиб, безаш учун ишлатилади.

Это полный список объектов на первой странице Палитры Компонент. Если Вам нужна дополнительная информация, то выберите на Палитре объект и нажмите клавишу F1 - появится Справочник с полным описанием данного объекта.

### *Additional саҳифаси*

Standard саҳифасида Windows 3.0 да пайдо бўлган бошқариш объектлари келтирилган, Additional саҳифаси ёрдамида эса, дастурнинг фойдаланувчи интерфейсини янада яхшилаш мумкин.

Компонентлар рўйхати:



**TBitBtn** – Tbuttonга ўхшаш тугма, лекин унда тасвир (glyph)ни ўрнатиш мумкин.).



**TSpeedButton** - буйруқларга тезкор мурожаат панелини яратиш тугмаси. Одатда бу тугмага фақат тасвир (glyph) жойлаштирилади.

**TTabSet** – Горизонтал сарлавҳалар. Одатда кўп саҳифали ойналарни яратиш учун TNoteBook билан биргаликда қўлланади. Саҳифалар номларини Tabs хусусиятида бериш мумкин.:

**TNoteBook** – кўп саҳифали мулоқотни яратиш учун мўлжалланганю бунда ҳар бир саҳифанинг ўз объектлар тўплами бўлади. Кўпинча TtabSet билан биргаликда ишлатилади.

**TTabbedNotebook** – сардавҳалари ўзида бўлган кўп саҳифали мулоқотлар яратиш.



**TMaskEdit** - Tedit га ўхшаш, лекин, киритишни форматлаш мумкин. Формат EditMask хусусиятида аниқланади.

**TOutline** – ўзаро боғланган маълумотларнинг иерархик муносабатини кўрсатишга мўлжалланган.

**TStringGrid**  - матнли маълумотларни жадвал кўринишида чиқаради.

**TDrawGrid**  - ихтиёрий типдаги маълумотларни жадвал кўринишида чиқаради.

**TImage**  - формада график тасвирларни намоиш қиладиотображаёт. Асосан BMP, ICO, WMF файлларни қабул қилади.

**TShape**  - формада оддий график объект (айлана, квадрат ...)ларни намоиш қилади.

**TBevel**  - интерфейсни рельефли безаш учун хизмат қилади

**THeader** – жадваллар учун ўзгарувчан кенгликка эга бўлган сарлавҳаларни сақлайди.

**TScrollBar**  - формада экранга сиғмайдиган объектларни кўрсатиш имконини берувчи йўлакчаларни ҳосил қилади.

### *Dialogs саҳифаси*

Dialogs саҳифасида Windowsнинг стандарт мулоқот ойналарини чақирувчи компонентлар жойлашган. Мулоқот ойналарининг ташқи кўриниши Windowsнинг версиясига боғлиқ бўлади. Бу саҳифадаги объектлар дастур бажарилиш вақтида кўринмайди ва мос мулоқойт ойналарини дастурий йўл билан чақириш керак, масалан:

```
if OpenFileDialog1.Execute then Image1.Picture.LoadFromFile(OpenDialog1.FileName);
```

Dialogs саҳифасида Windowsнинг мулоқот ойналари компонентлари қуйидаги тартибда жойлаштирилган: - OpenFileDialog; файлни танлаш - SaveDialog; файл сақлаш - FontDialog; шрифтни сошлаш - ColorDialog; ранг танлаш - PrintDialog; чоп қилиш - PrinterSetupDialog; принтерни сошлаш - FindDialog; сатрни қидириш- ReplaceDialog; алмашлаб қидириш

## *System саҳифаси*

**Timer** - таймер, OnTimer вазияти ҳар сафар Interval хусусиятида кўрсатилган вақтдан кейин бажарилади. Вақт оралиғи 1 дан 65535 мсгача бўлиши мумкин.

**TPaintBox** –чизиш учун жой. Ҳолатлар қайта ишловчисига TpaintBoxдаги сичқончанинг мос координаталари қайтарилади

**TFileListBox** - махсус ListBox унда (Directory хусусиятида) кўрсатилган каталогдаги файллар рўйхати чиқарилади.

**TDirectoryListBox** - махсус ListBox унда жорий дискдаги каталоглар тузилиши билан танишиш мумкин. FileList хусусиятида TFileListBox кўрсатилса, унда кейинги каталокка ўтилганда мос файллар чиқарилади.

**TDriveComboBox** – махсус ComboBox ундан жорий дискни танлаш мумкин. Агар DirList хусусиятида TDirectoryListBox кўрсатилса, диск алмаштирилиши билан мос каталоглар чиқарилади.

**TFilterComboBox** – файллар номи маскасини чиқариш учун махсус ComboBox. Маскалар рўйхати Filter хусусиятида аниқланади. FileList хусусиятида маска ўрнатилган TFileListBox кўрсатилади.

**TMediaPlayer** – мультимедиа воситалари билан ишлашни кўзда тутди.

**TOLEContainer** – ўзида OLE объектларни сақловчи контейнер.

Назорат саволлари:

1. Компонент деганда нимани тушунасиз?
2. Стандарт компонентлардан қайсиларини биласиз?
3. Dialogs саҳифасига кирувчи компонентлар қайсилар?
4. System саҳифасидаги компонентлар қандай амаллар учун мўлжалланган?

Таянч иборалар:

Timer;  
TPaintBox;  
THeader;  
TMediaPlayer;  
TOLEContainer;  
TBitBtn

## **4-маъруза. Мултимедиа дастурларини яратиш воситалари**

Режа:

5. Delphiда мултимедиа
6. TMediaPlayer компоненти
7. Мултимедиа дастурининг икки кўриниши.

Адабиётлар руйхати:

1. Компьютердаги ёрдам файллари.
2. Бобровский «Delphi 5», «Питер» Москва 1997г.
3. Шумаков «Delphi 4 разработка баз данных», «Питер» Москва 1996г.
4. Пачеко, Тейксер «Delphi 5 пособие программиста», «Питер» Москва 1999 г.
5. Фаронов «Delphi 4 учебное пособие», «Питер» Москва 1995 г.

### ***Мултимедиа нима?***

Бу саволнинг аниқ жавоби мавжуд эмас. Лекин бу ерда бир умумий қонидани келтириш мумкин: “мултимедиа” – компьютерда ишлатиладиган ҳаракат (анимация), овозлар, видеотасвирларнинг ҳамма кўриниши билан боғлиқ тушунчадир.

### ***Delphiда мултимедиа***

Delphiда TMediaPlayer компоненти мавжуд бўлиб, у сизга мултимедиа дастурларини тузишда ҳамма имкониятларни яратиб беради. Ушбу компонентдан фойдаланиш жуда оддий.

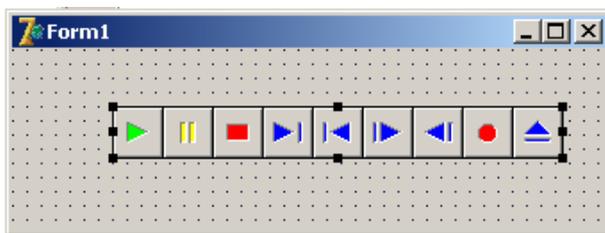
Бу оддийликни икки маънода тушуниш мумкин:

- Бир томондан – бу ихтиёрий дастурчига мултимедиа дастурини тузишга имкон яратилган.
- Иккинчи томондан, бу компонентда мултимедиа файллари билан бажариладиган ҳамма асосий амаллар киритилган, лекин сизга қуйи функциялар керак бўладиган бўлса, Delphi дастурлаш тили имкониятларидан фойдалинишингиз мумкин.

Ушбу маърузада компонент ишлаганда унинг ички функциялари қандай ишлатилишига тўхталмаймиз.

### ***TMediaPlayer компоненти***

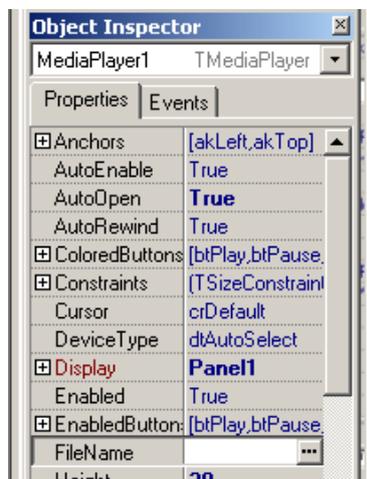
Дастлаб янги лоиха яратамиз ва формага TMediaPlayer (System саҳифаси) компонентини ташлаймиз.



15-расм: Формадаги TMediaPlayer компоненти.

TMediaPlayer компоненти қурилмани бошқариш панели каби тузилган. Магнитофонлардаги каби бу ерда ҳам “ишга тушириш”, “ўтказиш”, “ёзиш” ва бошқа тугмалари бор.

Компонентни формага ташлагач, сиз Объектлар Инспекторида "FileName" хусусиятини кўрасиз.



16- Расм. TMediaPlayerнинг Объектлар инспекторидаги хусусиятлари

Шу ерда сичқонча тугмасини босинг ва рўйхатдан AVI, WAV ёки MID кенгайтмали файлни танланг. Сўнгра AutoOpen хусусиятини True қийматга ўрнатиш керак.

Бу қадамлар бажарилгач, дастур юкланишга тайёр бўлади. Дастурни юклаб ишга тушириш (Play) тугмасини босинг ва сиз танлаган видеоклипингизни

кўришингиз ёки музикани тинглашингиз мумкин. Агар овоз ёки видеоклип ишга тушмасдан хатолик ҳақида хабар чиқарилса, икки ҳолат бўлиши мумкин:

1. Сиз файл номини ёки унгача бўлган йўлни нотуўғри киритгансиз.
2. Компьютерингиздаги мультимедиа қурилмалари Windowsга тўғри созланмаган. Бу эса, сизда керакли қурилманинг йўклигидан ёки керакли драйверлар ўрнатилмаганлигидан далолат беради.

TMediaPlayer компонентининг яна бир муҳим хусусияти - Display. Дастлабки мисолимизда бу хусусият бўш қолдирилган ва видеоклип алоҳида ойнада кўринган эди. Лекин, видеоклип намойишида экран сифатида масалан, **Панелдан** фойдаланиш мумкин. Формага TPanel компонентини ташланг ва Caption хусусиятидаги матнни ўчиринг. Сўнгра TMediaPlayer учун Display хусусиятидаги рўйхатдан Panel1ни танланг. Шундан сўнг дастурни юклаб, ишга тушириш тугмасини босиш керак.



17- расм. Панелда AVIни кўрсатиш.

### ***Мультимедиа дастурининг икки кўриниши.***

Мультимедиа дастурга мисол.

Бу бўлимда биз мультимедиа дастурларга мисол кўриб ўтамыз. Янги лоиха яратинг (File | New Project). Формага TMediaPlayer, шунингдек файлларга мурожаат қилиш учун, TFileListBox, TDirectoryListBox, TDriveComboBox, TFilterComboBox компонентларини ташланг. DirectoryListBox1нинг ва FilterComboBox1нинг FileList хусусиятларида FileListBox1ни белгиланг. DriveComboBox1нинг DirList хусусиятига DirectoryListBox1ни ўрнатинг. FilterComboBox1нинг Filter хусусиятига керакли файллар кенгайтмаларини киритинг:

AVI File(\*.avi)|\*.avi

WAVE File(\*.wav)|\*.wav

MIDI file(\*.MID)|\*.mid

FileListBox1да Сичқончани икки марта босганда жорий файл ишга туширилишига эришишимиз керак. Ҳолатлар қайта ишловчисидаги FileListBox1нинг OnDbClick ҳолатига қуйидаги дастур матнини киритамиз:

```
Procedure TForm1.FileListBox1DbClick(Sender:TObject);
```

```
begin
```

```
  with MediaPlayer1 do
```

```
    begin
```

```
      Close;
```

```
      FileName:=FileListBox1.FileName;
```

```
      Open;
```

```
      Play;
```

```
    end;
```

```
end;
```

Дастурни юклаб натижани кўришингиз мумкин.

Дастур бажарилиш вақтида MediaPlayer объектининг ёки видеоклиппнинг жорий ҳолатини намойиш қилиш зарурияти туғилиб қолиши мумкин. Бунинг учун TMediaPlayer объектининг: Length, Position, OnNotify ва бошқа хусусиятлари мавжуд. Келинг лоихамизга вақт индикатори сифатида (TGauge) объектини ташлаймиз, у ўтган вақтни фоизлар бўйича кўрсатади. Унинг кўрсаткичини янгилаш учун таймердан фойдаланиш мумкин. Формага TTimer объектини ташланг, унинг учун Interval = 100 (100 миллисекунд) га ўрнатинг. OnTimer ҳолатига қуйидагини ёзиш керак:

```
procedure TForm1.Timer1Timer(Sender: TObject);
```

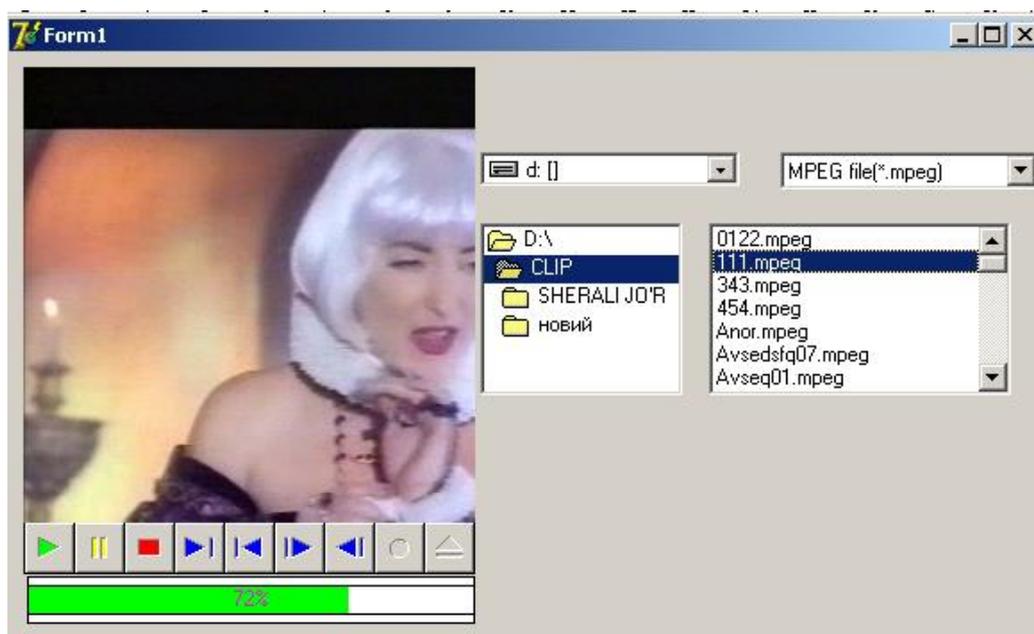
```
begin
```

```
  with MediaPlayer1 do
```

```
    if FileName<>" then
```

```
Gauge1.Progress:=Round(100*Position/Length);
```

```
end;
```



18 - Расм. Объектларнинг жорий ҳолатини намойиш қилиш

Дастурни юкланг, файл (AVI)ни танланг ва унга сичқон тугмасини икки марта босинг. Видеоклип намойиши вақтида биз қўйган вақт индикатори ўтган вақтга боғлиқ ҳолда фоизларни чиқариши керак.

Назорат саволлари:

1. Мультимедиа нима?
2. TMediaPlayer компоненти нима мақсадларда ишлатилади?
3. Мультимедиа файлларини бошқариш учун ишлатиладиган объектларни санаб ўтинг.
4. Мультимедиа файли лоихага қандай бириктириладти?

Таянч иборалар:

AVI; MediaPlayer; OnTimer; Хусусият; Length; Position; OnNotify

## 5-Маъруза. Матнли ва график чоп қилиш

### Режа:

5. Матнли режимда чоп қилиш
6. Форманнинг кўринишини чоп қилиш
7. График чоп қилиш (Tprinter объекти)

### Адабиётлар руйхати:

6. Компьютердаги ёрдам файллари.
7. Бобровский «Delphi 5», «Питер» Москва 1997г.
8. Шумаков «Delphi 4 разработка баз данных», «Питер» Москва 1996г.
9. Пачеко, Тейксер «Delphi 5 пособие программиста», «Питер» Москва 1999 г.
10. Фаронов «Delphi 4 учебное пособие», «Питер» Москва 1995 г.

### *Матнли режимда чоп қилиш*

Агар сиз матнли режимда ҳужжатни чоп қилмоқчи бўлсангиз бу иш қуйидагича амалаг оширилади. Принтерга сиз ддий матнли файл билан ишлагандай мурожаат қиласиз, ягона фарқ AssignFile процедураси ўрнига AssignPrn процедурасини ёзиш керак. Мисолда принтерга матннинг бирта сатри чиқарилади:

```
procedure TForm1.Button1Click(Sender: TObject);
var
  To_Prn : TextFile;
begin
  AssignPrn(To_Prn);
  Rewrite(To_Prn);
  Writeln(To_Prn, 'Printer in Text Mode');
  CloseFile(To_Prn);
end;
```

Delphi дастурлаш мухитида ҳам файллар билан ишлашда Pascal дагига ўхшаш процедуралар ишлатилади. Қуйида биз System модулидаги баъзи амалларни кўрсатиб ўтамыз:

- Assign ўрнига AssignFile
- Close ўрнига CloseFile
- Text ўрнига TextFile

### ***Форманинг кўринишини чоп қилиш***

Баъзида дастурчилар учун экрандаги тасвирнинг нусхаси керак бўлиб қолади. Delphi бу иш жуда оддий бажарилади - TForm объектида Print услуги бор бўлиб, уни керакли вақтда чақириш етарли.

### ***График чоп қилиш (TPrinter объекти)***

Қандай қилиб, Delphiда яратилган дастурдан график маълумотларни принтерга узатиш мумкин? Бунинг учун махсус (TPrinter синфидаги) Printer объекти мавжуд. У дастурга Printers модулини киритгандан сўнг фаоллашади. Бу объект ёрдамида график маълумотларни принтерга узатиш худди уларни экранга чиқаришдай оддийлашиб қолади. Printer объектининг хусусиятлари ва услубларини кўриб ўтамиз.

### ***Printer хусусиятлари***

Aborted – Булев (мантикий) тип; фойдаланувчи принтер ишини Abort.услуги ёрдамида тўхтатганлигини билдиради.

Canvas - Канва, график маълумотларни чиқариш учун жой.

Fonts – Ўрнатиш мумкин бўлган шрифтлар рўйхати.

Handle - Windows API тўғридан тўғри чақирилганда ишлатилади.

Orientation – Саҳифанинг мўлжали, вертикал ёки горизонтал.

PageWidth, PageHeight, PageNumber – мос ҳолда саҳифанинг кенглиги, баландлиги ва тартиб рақами.

Printers Тизимда ўрнатилган барча принтерларнинг номлари рўйхати

PrinterIndex Жорий принтернинг рақами. ЖҚБ чоп қилиш учун бу ерда -1 туриши керак.

Printing – Чоп қилиш (BeginDoc услугида) бошланганлигини билдирувчи мантикий тип.

Title – Print учун сарлавҳа.

Printer услублари:

Abort - BeginDoc режимида бошланган чоп қилишни тўхтатиш

BeginDoc - Канвада чизиш бошлангандан олдин чақирилади.

EndDoc – Канвада ҳамма нарса тайёр бўлгач принтер чоп қилишни бошалайди.

NewPage - Янги саҳифага ўтиш.

Қолган услублар билан оддий ҳолларда фойдаланиш керак эмас.

Шундай қилиб, график маълумотларни чоп қилиш тартиби қуйидагича:

- BeginDoc услубини бажарамиз
- канвада (Canvas) керакли нарсаларни чизамиз
- маълумотни бир нечта саҳифага жойлаш керак бўлса, NewPage услубини қўллаш
- чизилганларни принтерга йўллаб, EndDoc услубини бажаримиз.

Назорат саволлари:

1. Мультимедиа нима?
2. Delphiда мультимедиадан нима мақсадда фойдаланиш мумкин?
3. TMediaPlayer компонентининг вазифаси нималардан иборат?

Таянч иборалар:

BeginDoc

NewPage

EndDoc

AssignFile

CloseFile

TextFile

## **6-Маъруза. Delphiда объектларнинг хусусиятлари.**

### **Режа:**

1. Умумий тушунчалар.
2. Визуал компонентлар хусусиятларини дастур бажарилиш вақтида бошқариш.
3. SHAPEDEM1 дастури.
4. SHAPEDEM2 дастури.

### **Адабиётлар руйхати:**

1. Компьютердаги ёрдам файллари.
2. Бобровский «Delphi 5», «Питер» Москва 1997г.
3. Шумаков «Delphi 4 разработка баз данных», «Питер» Москва 1996г.
4. Пачеко, Тейксер «Delphi 5 пособие программиста», «Питер» Москва 1999 г.
5. Фаронов «Delphi 4 учебное пособие», «Питер» Москва 1995 г.

### ***Умумий тушунчалар.***

Хусусиятлар объектнинг муҳим атрибутаси эканлигига диққатингизни қаратгандирсиз. Фойдаланувчи (дастурчи) учун хусусият бу бирор тузилишнинг қиймат сақловчи майдонидир. Лекин, бу оддий майдон қиймати ўзгариши билан компонентнинг ташқи кўриниши бутунлай ўзгаради, чунки, хусусиятлар ўзида объектнинг шу майдонга ёзиш ва ўқиш ҳолатлари инкапсуляциясига боғлиқ бўлади. Хусусиятлар икки мақсадда хизмат қилади. Биринчидан, улар форма ёки компонентнинг ташқи кўринишини аниқлайди. Иккинчидан эса хусусиятлар форма ёки компонентнинг ҳаракатини белгилайди.

Хусусиятлар бир нечта типларга бўлиш мумкин.

- Оддий хусусиятлар – буларнинг қийматлари сонлар ёки сатрлар бўлади. Масалан, Left ва Top хусусиятлари форма ёки компонентнинг юқориги чапки бурчаги координаталарини аниқловчи бутун сонларни сақлайди. Caption ва Name

хусусиятлари форма ёки компонентнинг сарлавҳаси ва номини билдирувчи сатрларни сақлайди.

- Санаб ўтиладиган хусусиятлар – булар мавжуд рўйхатдаги қийматлардан бирини қабул қила оладиган хусусиятларир. Оддий мисол – **Boolean** типдаги хусусият, *True* ёки *False* қийматини қабул қила олади.

- Ички маълумотли хусусиятлар – булар ички қийматларга (ёки объектларга) эга булган хусусиятлардир. Бундай хусусиятлар Object Inspectorда чап томонидан “+” белгиси билан ажралиб туради. Бундай хусусиятлар ҳам икки хилда бўлади: *кўпликлар* ва *бирлашган қийматлар*. Object Inspectorда кўпликлар квадрат кавслар ёрдамида ифодаланади. Агар кўплик бўш бўлса, у [] сифатида тасвирланади. Кўплик кўринишидаги ички маълумотли хусусиятлар кўпинча мантиқий **Boolean** типда бўлади. Бирлашган қийматлар Объектлар Инспекторида баъзи катталиклар тўплами сифатида тасвирланади. Баъзи хусусиятлар масалан, Font ўз қийматини ўзгартириш учун мулоқот ойналарини чақира олади. Бунинг учун шу хусусият майдонининг ўнг томонидаги уч нуқтали кичик тугмачани босиш kifоя.

Delphiда компонентлар хусусиятларини лоиҳалаш вақтида (design time) ҳам, дастур бажарилаш вақтида (run time) ҳам ўзгартириш оддий.



Лоиҳалаш вақтида хусусиятларни ўзгартириш Формалар Дизайнери (Forms Designer) ёки, сизга маълум Объектлар Инспекторининг “Properties” саҳифаси

ёрдамида бажарилади. Масалан, тугманинг Height (баландлиги) ва Width (кенглиги) хусусиятларини ўзгартириш учун, унинг формадаги тасвирининг бир бурчагидан сичқонча ёрдамида тортиб, керакли ўлчамларга эришиш мумкин. Худди шу натижага Object Inspectorнинг Height ва Width хусусиятларни қийматларини ўзгартириб ҳам эришиш мумкин.

### ***Визуал компонентлар хусусиятларини дастур бажарилиш вақтида бошқариш***

Дастур бажарилиш вақтида компонентлар хусусиятлари қийматининг барча ўзгаришлари бажариладиган иш коди тўлиқ Паскаль тилида ёзилишини талаб қилади. Дастур бажарилиш вақтида Object Inspectorдан фойдаланиш мумкин эмас. Лекин, компонентлар хусусиятларига дастурий йўл билан муружаат қилиш анча қулай. Бирор хусусиятнинг қийматини ўзгартириш учун сиз қуйидагига ўхшаш код ёзишингиз кифоя:

```
MyComponent.Width := 35;
```

Бу сатр компонентнинг кенглиги Widthни 35 қийматга ўрнатади. Агар бундан олдин компонентнинг кенглиги 35 га тенг бўлмаса, сиз компонент қандай қилиб кенглигини ўзгартиришини кўришингиз мумкин.

Шундай қилиб, Object Inspector дастур бажарилиш вақтида қилиш мумкин бўлган ишни лоиҳалаш вақтида оддийроқ бажариш имкониятини беради холос. Бундан ташқари объектларнинг Object Inspectorда кўринмайдиган хусусиятлари ҳам бўлиши мумкин, уларни фақат дастурий йўл билан ўзгартириш мумкин.

### ***SHAPEDEM1 дастури***

SHAPEDEM1.DPR дастури , дастур бажарилиш вақтида фойдаланувчи интерфэйсини ўзгартиришнинг бир неча йўллари намоёни қилади. SHAPEDEM1 дастури атиги формага ўрнатилган TShape, ўтказиш йўлакчалари ва бир нечта тугмадан иборат.

**SHAPEDEM1.DPR** дастурининг коди:

```
program Shapedem;
```

```
uses
```

```
Forms,
```

```

Mina in 'MAIN.PAS' {Form1};
begin
Application.CreateForm(TForm1, Form1);
Application.Run;
end.

```

Бизнинг ScrollBarлар мисолимизда экраннинг ўртасидаги тасвирнинг ўлчамларини ўзгартириш учун ишлатилади. Тасвирнинг янги кўринишини танлаш учун ўзида тасвирлар номини сақловчи (ComboBox) дан, тасвир ёки форма рангини ўзгартириш учун стандарт ранг танлаш мулоқот ойнаси ва мос тугмалардан фойдаланамиз.

Форма рангини ўзгартириш учун Dialogs саҳифасидан ColorDialog объектини формага ўрнатинг. Шунингдек, формага оддий тугма Buttonни ташлаб қулайлик учун унинг Caption хусуиятидаги ёзувни “Форма ранги”га ва Name хусуиятидаги ёзувни “FormColor”га алмаштиринг. Шу тугмага сичқончани икки марта чиқиллатсак, қуйидаги кўринишга эга бўлган дастур матни чиқарилади:

```

procedure TForm1.FormColorClick(Sender: TObject);
begin
end;

```

Энди биз оддий кодларни киритамиз:

```

procedure TForm1.FormColorClick(Sender: TObject);
begin

```

```

if ColorDialog1.Execute then
Form1.Color := ColorDialog1.Color;
end;

```

Бу код дастур бажарилиш вақтида “Форма ранги” тугмаси босилгач, рангни танлаш мулоқот ойнасини чақиради. Агар сиз бу мулоқот ойнасида ОК тугмасини боссангиз қуйидаги сатр бажарилади:

```

Form1.Color:=ColorDialog1.Color;

```

Бу код форма(Form1)нинг ранги (Color)ни ColorDialog1 мулоқот ойнасида танлинган рангга ўрнатади.

Тасвир рангини ўзгартириш учун ҳам худди шундай ишлар бажарилади. Энди сиз формага бошқа тугма ўрнатасиз ва унинг номини “ShapeColor” ва сарлавҳасини “Тасвир ранги” деб ўрнатасиз. Шу тугмага сичқон тугмасини икки марта босиб, қуйидагича код ёзасиз:

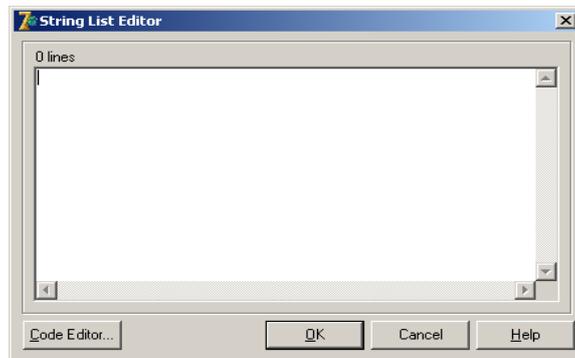
```
procedure TForm1.ShapeColorClick(Sender: TObject);  
begin  
    if ColorDialog1.Execute then                                     -  
Shape1.Brush.Color := ColorDialog1.Color;  
end;
```

Энди анча мураккаброқ бўлган иш ўтказиш йўлакчалари (**ScrollBar**) ёрдамида тасвир ўлчамларини ўзгартириш дастурини тузамиз.

Дастлаб, формага иккита ScrollBar компонентини ташлаймиз ва биринчи объект учун Kind хусусиятини **sbHorizontal**га, иккинчиси учун - **sbVertical**га ўрнатамиз. Объектлар Инспекторидан “Events” саҳифасига ўтамыз ва ҳар бир ўтказиш йўлакчалари учун OnChange ҳолатига дастур ёзиш учун сичқон тугмасини босамиз ва муҳаррирда қуйидагича код киритамиз:

```
procedure TForm1.ScrollBar1Change(Sender: TObject);  
begin  
    Shape1.Width := ScrollBar1.Position * 3;  
end;  
procedure TForm1.ScrollBar2Change(Sender: TObject);  
begin  
    Shape1.Height := ScrollBar2.Position * 2;  
end;
```

Энди, формадаги тасвирнинг кўринишини ўзгартириш масаласини кўриб ўтамыз. Формага ComboBox объектини ташлаймиз ва унинг Items хусусиятига сичқонни чиқиллатиб, String List Editor муҳарририни чақирамиз.



20-расм. Формадаги ComboBox объектининг Items хусусияти ёрдамида чақирилувчи ойна

Бу муҳарирда қуйидаги сатрларни киритамиз (*stCircle*, *stEllipse*, *stRectangle*, *stRoundRect*, *stRoundSquare*, *stSquare*). Сўнгра формада ComboBox1 объектини танлаб, Объектлар Инспекторининг “Events” саҳифасидаги OnClick хусусиятига сичқон тугмасини икки марта босиб, дастур матни муҳаррири ойнасига ўтамыз. У ерда қуйидаги ёзув бўлади:

```
procedure TForm1.ComboBox1Click(Sender: TObject);  
begin  
end;
```

Энди биз ўз кодимишни қўшиб, бажариладиган ишни кўрсатамыз:

```
procedure TForm1.ComboBox1Click(Sender: TObject);  
begin  
Shape1.Shape := TShapeType(ComboBox1.ItemIndex);  
end;
```

Бу сатрдаги код Shape1 компонентининг Shape хусусиятини фойдаланувчи рўйхатдан танлаган кўринишга ўрнатади. Энди сиз дастурни юклаб, унинг имкониятларидан фойдаланишингиз мумкин.

### ***SHAPEDEM2 дастури***

Энди фойдаланувчи форма ўлчамларини ўзгартириб ўтирмаслиги учун дастурнинг ўзи форма ўлчамларини бошқариши муаммосини ҳал қилайлик. Бунинг учун қуйидагиларни бажарамиз:

- Формадаги Тугмалар ва ComboBox объектларини олиб ташлаймиз. Уларнинг ўрнига MainMenu компонентини ўрнатамыз.

- Форма ўлчамлари ўзгариши билан ўтказиш йўлакчалари ўз ҳалатини ўзгартиришига эришамиз.
- Ўтказиш йўлакчаларининг Position хусусияти форма ўлчамларини тўғри акслантиришига эришимиз.

**SHAPDEM2** дастури. FormOnResize услуги қўшилган. Бош модуль кўриниши.

**unit Main;**

**interface**

**uses**

**WinTypes, WinProcs, Classes, Graphics, Forms, Controls, ColorDlg, StdCtrls, Menus, Dialogs, ExtCtrls;**

**type**

**TForm1 = class(TForm)**

**Shape1: TShape;**

**ColorDialog1: TColorDialog;**

**ScrollBar1: TScrollBar;**

**ScrollBar2: TScrollBar;**

**MainMenu1: TMainMenu;**

**Shapes1: TMenuItem;**

**ShapeColor1: TMenuItem;**

**FormColor1: TMenuItem;**

**Shapes2: TMenuItem;**

**Rectangle1: TMenuItem;**

**Square1: TMenuItem;**

**RoundRect1: TMenuItem;**

**RoundSquare1: TMenuItem;**

**Ellipes1: TMenuItem;**

**Circle1: TMenuItem;**

**Exit1: TMenuItem;**

**procedure NewShapeClick(Sender: TObject);**

```

procedure ShapeColorClick(Sender: TObject);
procedure FormColorClick(Sender: TObject);
procedure ScrollBar2Change(Sender: TObject);
procedure ScrollBar1Change(Sender: TObject);
procedure FormResize(Sender: TObject);
procedure Exit1Click(Sender: TObject);
private
{ Private declarations }
public
{ Public declarations }
end;
var
Form1: TForm1;
implementation
{$R *.DFM}
procedure TForm1.NewShapeClick(Sender: TObject);
begin
Shape1.Shape := TShapeType((Sender as TMenuItem).Tag);
end;
procedure TForm1.ShapeColorClick(Sender: TObject);
begin
if ColorDialog1.Execute then
Shape1.Brush.Color := ColorDialog1.Color;
end;
procedure TForm1.FormColorClick(Sender: TObject);
begin
if ColorDialog1.Execute then
Form1.Color := ColorDialog1.Color;
end;

```

```

procedure TForm1.ScrollBar2Change(Sender: TObject);
begin
Shape1.Height := ScrollBar2.Position;
end;
procedure TForm1.ScrollBar1Change(Sender: TObject);
begin
Shape1.Width := ScrollBar1.Position;
end;
procedure TForm1.FormResize(Sender: TObject);
var
Menu,
Caption,
Frame: Integer;
begin
Caption := GetSystemMetrics(sm_cyCaption);
Frame := GetSystemMetrics(sm_cxFrame) * 2;
Menu := GetSystemMetrics(sm_cyMenu);
Scrollbar1.Max := Width;
Scrollbar2.Max := Height;
Scrollbar2.Left := Width - Frame - Scrollbar2.Width;
Scrollbar1.Top := Height - ScrollBar2.Width - Frame - Caption - Menu;
Scrollbar1.Width := Width - Scrollbar2.Width - Frame;
Scrollbar2.Height := Height - Frame - Caption - Menu - Scrollbar1.Height;
end;
procedure TForm1.Exit1Click(Sender: TObject);
begin
Close;
end;
end.

```

Дастур учун бош меню MainMenu (компонентлар тўпламининг “Standard” саҳифасида) компоненти ёрдамида яратилади. Уни формага қўйиб, икки марта сичқонни чиқиллатинг – шунда керакли меню қисмларини киритиш имкони берилади. SHAPEDEM2 дастурининг менюсини учта асосий менюдан (“Ранг”, “Тасвир” ва “Чиқиш”) иборат қилиб яратамиз.

Биринчи бўлим учун қуйидаги қисмларни яратамиз:

- Тасвир ранги
- Форма ранги.

Иккинчиси учун:

- Тўртбурчак
- Квадрат
- Айлана Тўртбурчак.
- Айлана квадрат
- Эллипс
- Айлана

Менюнинг учинчи бўлимида қисм менюлар бўлмайди.

Сўнгра тасвир кўриниши билан боғлиқ қисмларга рақамларни бериб чиқамиз. Бунинг учун менюнинг ҳар бир қисмида мавжуд бўлган Tag хусусиятидан фойдаланамиз. Tag (Integer типиди) хусусиятидан фойдаланувчи ихтиёрий фойдаанишлари мумкин. Биз “Тўртбурчакка” Tag хусусиятининг 0 қийматини, “Квадрат”га 1 қийматини ... берамиз.

Энди дастур матнини келтириб ўтаемиз:

```
procedure TForm1.ShapeColorClick(Sender: TObject);
```

```
begin
```

```
if ColorDialog1.Execute then
```

```
Shape1.Brush.Color := ColorDialog1.Color;
```

```
end;
```

```
procedure TForm1.FormColorClick(Sender: TObject);
```

```
begin
```

```
if ColorDialog1.Execute then  
Form1.Color := ColorDialog1.Color;  
end;
```

Кўряписизки кодлар дастлабки дастурдан умуман ўзгаргани йўқ, фақат бу амаллар меню ёрдамида бажариляпти.

Худди шундай тасвир кўринишини ўзгартириш ҳам олдинги танловга ўхшаш:

```
procedure TForm1.NewShapeClick(Sender: TObject);  
begin  
Shape1.Shape := TShapeType((Sender as TMenuItem).Tag);  
end;
```

SHAPEDEM2 дастуридаги иккита ўтказиш йўлакчаси доимо форма ўлчамлари қандай бўлишиданқатъий назар ойна чегаралари ёнида жойлашади.

```
procedure TForm1.FormResize(Sender: TObject);  
var  
Menu, Caption, Frame: Integer;  
begin  
Caption := GetSystemMetrics(sm_cyCaption);  
Frame := GetSystemMetrics(sm_cxFrame) * 2;  
Menu := GetSystemMetrics(sm_cyMenu);  
Scrollbar1.Max := Width;  
Scrollbar2.Max := Height;  
Scrollbar2.Left := Width - Frame - Scrollbar2.Width;  
Scrollbar2.Height := Height - Frame - Caption - Menu;  
Scrollbar1.Top :=  
Height - Scrollbar2.Width - Frame - Caption - Menu;  
Scrollbar1.Width := Width - Scrollbar2.Width - Frame;  
end;
```

FormResize услубида дастур программа Scrollbarларнинг янги ўлчамларини ҳисоблайди:

```
Scrollbar1.Max := Width;  
Scrollbar2.Max := Height;  
Scrollbar2.Left := Width - Frame - Scrollbar2.Width;  
Scrollbar2.Height := Height - Frame - Caption - Menu;  
Scrollbar1.Top :=  
Height - Scrollbar2.Width - Frame - Caption - Menu;  
Scrollbar1.Width := Width - Scrollbar2.Width - Frame;
```

SHAPDEM2 дастурида сиз Min ва Max хусусиятларининг қийматларини Position қийматлари оралиғи форманинг жорий ўлчамларини кўрсатадиган қилиб ўрнатишингиз мумкин. Бу ерда FormResize дастуридан мос кодлар келтирилган.

```
procedure TForm1.FormResize(Sender: TObject);
```

```
begin
```

```
...
```

```
Scrollbar1.Max := Width;
```

```
Scrollbar2.Max := Height;
```

```
...
```

```
end;
```

Юқорида кўрсатилган икки қатор код иккала ўтказиш йўлакчалари энг катта қийматларини форманинг бўйи ва энига тенглаштиради, бу эса, тасвирни форма бўйлаб ёйишга ёрдам беради. Энди сизга Position қийматини бирор сонга кўпайтириш шарт эмас.

```
procedure TForm1.Scrollbar2Change (Sender: TObject);
```

```
begin
```

```
Shape1.Height := Scrollbar2.Position;
```

```
end;
```

Сиз шундан кейин SHAPDEM2 дастурини юкласангиз у форманинг ихтиёрий ўлчамларида ишлашига гувоҳ бўласиз. Бундан ташқари сиз тасвир кўринишини ва

тасвир рангини меню орқали бошқара оласиз, бу эса жиддий дастурларга хос хусусият ҳисобланади.

Назорат саволлари:

1. Хусусиятлар деганда нимани тушунасиш?
2. Хусусиятлар неса типга бўлинади?
3. Хусусиятларни бошқаришнинг қандай усуллари мавжуд?

Таянч иборалар:

Бирлашган қийматлар.

Кўпликалар.

Хусусиятларни дастур бажарилишида бошқариш.

Тасвир ранги.

Форма ранги.

## **7-Маъруза. DDE**

### **Режа:**

4. DDE асослари.
5. DDEнинг қўлланилиши
6. DDE-серверлар
7. DDE-мижозлар
8. DDE бўйича ReportSmithни бошқариш

### **Адабиётлар руйхати:**

1. Компьютердаги ёрдам файллари.
2. Бобровский «Delphi 5», «Питер» Москва 1997г.
3. Шумаков «Delphi 4 разработка баз данных», «Питер» Москва 1996г.
4. Пачеко, Тейксер «Delphi 5 пособие программиста», «Питер» Москва 1999 г.
5. Фаронов «Delphi 4 учебное пособие», «Питер» Москва 1995 г.

### ***DDE асослари***

DDEML қисқартмаси Dynamic Data Exchange Management Library (динамик маълумотлар алмашиш кутубхонаси) маъносини англатади. DDEML бу Dynamic Data Exchange (DDE) деб номланувчи мураккаб хабарлар тизими устидан назоратдир. DDE сақловчи кутубхона Windowsнинг дастлабки хабарлар тизими имкониятларини ошириш учун ишлаб чиқилган.

Dynamic Data Exchange бундай номланишига сабаб, у икки дастурни бажарилиш вақтида ўзаро динамик ҳолда маълумотлар алмашишига имкон беради. Икки дастур орасидаги алоқани шундай яратиш мумкинки, бирида қилинган ўзгариш иккинчисида аксланади. Бундан ташқари DDE ёрдамида ўз дастурингиздан бошқа дастурларни масалан, Word for Windows, Report Smith, Excel ва бошқаларни бошқариш мумкин.

Қуйида Delphi компонентларидан DDE қуриш учун қандай фойдаланиш тушунтирилган.

## ***DDEнинг қўлланилиши***

DDE бўйича бошқа дастурдан маълумотлар олувчи ёки бошқа дастурни DDE буйруқлари ёрдамида бошқарувчи дастур DDE-мижози ҳисобланади. Бу ҳолда иккинчи дастур DDE-сервери бўлади. Бир дастурнинг ўзи бир вақтда ҳам мижоз, ҳам сервер бўлиши мумкин (масалан, Microsoft Word).

Энди келинг Delphi қандай қилиб DDE-дастурларни тузиш мумкинлигини кўриб ўтайлик.

### ***DDE-серверлар***

Delphiда DDE-серверни қуриш учун System саҳифасидаги икки - TDdeServerConv ва TDdeServerItem компонентлари хизмат қилади. Одатда лоихада битта TDdeServerConv объекти ва бир ёки бир нечта TDdeServerItem объекти ишлатилади. DDE-сервернинг сервисларига мурожаат қилиш учун мижоз бир неча параметрларни билиши керак: сервис номи (Service Name) – бу дастур номи (одатда – EXE кенгайтмасиз бажарилувчи файл номи, тўлиқ йули билан); Topic Name - Delphiда бу ном TDdeServerConv компоненти; Item Name – Delphiда бу TDdeServerItemнинг керакли бўлими номи.

TDdeServerConv объектининг вазифалари – DDEни умумий бошқариш ва макровларнинг бажарилиши учун мижозлардан сўровларни қабул қилиш. Охири амал OnExecuteMacro ҳолатлар қайта ишловчисида бажарилади, масалан, бизнинг ҳолдақуйидагича:

```
procedure TDdeSrvrForm.doMacro(Sender: TObject; Msg: TStrings);
var
  Text: string;
begin
  Text := '';
  if Msg.Count > 0 then Text := Msg.Strings[0];
  MessageDlg ('Executing Macro - ' + Text, mtInformation, [mbOK], 0);
end;
```

TDdeServerItem объекти TDdeServerConv билан боғланади ва DDE бўйича айнан нима юборилишини аниқланади. Бунинг учун унда Text ва Lines хусусиятлари мавжуд. (Text нинг қиймати доимо Lines[0] га тенг бўлади) бу қийматлар ўзгарганда сервер билан боғланган ҳамма мижозларга янгиланган маълумотлар юборилади. Бизнинг дастуримизда Lines хусусияти қийматининг ўзгариши Edit1 компонентининг OnChange ҳолатлари қайта ишловчисида амалга ошади:

```
procedure TDdeSrvrForm.doOnChange(Sender: TObject);
begin
if not FInPoke then
DdeTestItem.Lines := Edit1.Lines;
end;
```

Шу компонентнинг ўзи мижозлардан маълумот олишни аъминлайди, бизнинг мисолимизда бу ҳолат Poke Data тугмаси босилганда, OnPokeData ҳолати қайта ишловчисида бажарилади:

```
procedure TDdeSrvrForm.doOnPoke(Sender: TObject);
begin
  FInPoke := True;
  Edit1.Lines := DdeTestItem.Lines;
  FInPoke := False;
end;
```

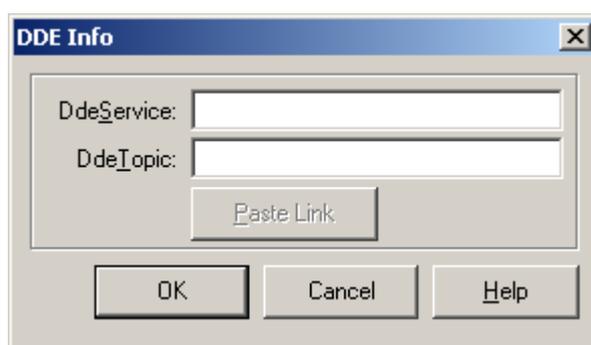
Шундай қилиб биз Delphi компонентлари ёрдамида қурилган тўлиқ ишловчи DDE-серверни кўриб ўтдик. Кўпинча мавжуд DDE-серверлар DDE имкониятларини тўлиқ намойиш қила олмайди ва сервис имкониятларнинг фақат бир қисми билан чегараланилади. Масалан, ReportSmith фақатгина DDE бўйича буйруқлар (макрослар)ни бажаради.

### ***DDE-мижозлар***

Delphiда DDE-мижозни яратиш учун иккита TDDEClientConv ва TDDEClientItem компонентлари ишлатилади. Сервердагига ўхшаш бу ерда ҳам

лоихада битта TDDEClientConv объекти ва бир ёки бир нечта TDDEClientItem объекти ишлатилади.

TDDEClientConv объекти сервер билан алоқа ўрнатиш ва умуман DDE-алоқани бошқариш учун хизмат қилади. DDE-сервер билан алоқа нафақат Формалар дизайнери режимида, балки, дастур бажарилиш вақтида ҳам ўрнатиш мумкин. Биринчи усулда компонентнинг керакли хусусиятлари қўлда тўлдириб чиқилади. Булар DdeService, DdeTopic ва ServiceApplicationлардир. Формалар дизайнери режимида Объектлар Инспекторидаги дастлабки икки хусусиятга сичқончани икки марта чиқиллатсак - DDE-алоқа мулоқот ойнасига мурожаат қилишимиз мумкин.



21-расм. DDE алоқани ўрнатиш ойнаси.

Мулоқот ойнасида DDE Service ва DDE Topic лар номини киритинг. Масалан, DDE-сервер сифатида DataBase Desktop келса, унга бирорта Paradox жадвалини юклаш, бирор майдонни белгилаш ва менюдан Edit|Copy бажариш керак. Шундаё сўнг мулоқот ойнасига кириб Paste Link тугмасини босиш керак. Мулоқот ойнаси майдонлари мос ҳолда тўлдирилади.

ServiceApplication хусусияти фақат дастур жорий каталогда турмаган вақтда ва DDEService майдонидаги ёзув дастур номидан фарқли бўлган ҳолларда тўлдирилади. Бу ерда дастур номи тўлиқ йўли билан кўрсатилиши керак. Report Smith билан ишлаганда бу ерда масалан: C:\RPTSMITH\RPTSMITH ёзувини кўрсатиш керак.

Бизнинг мисолимизда алоқа дастур бажарилиш вақтида менюнинг File|New Link ва Edit|Paste Link буйруқлари ёрдамида яратилади. Меню File|New Link буйруғида DDE бўйича алоқа объектнинг мос услуги ёрдамида боғланади

TDdeServerConv ва OpenLinkларни бажариш керак эмас, чунки ConnectMode хусусияти ddeAutomatic қийматига эга:

```
procedure TFormD.doNewLink(Sender: TObject);
begin
  DdeClient.SetLink(AppName.Text, TopicName.Text);
  DdeClientItem.DdeConv := DdeClient;
  DdeClientItem.DdeItem := ItemName.Text;
end;
```

бу ерда TDdeClientItem объекути хусусиятлари тўлдирилади.

Менюнинг Edit|Past Link буйруғида DDE бўйча алоқа Clipboardдаги маълумотларни қўллаш йўли билан боғланади:

```
procedure TFormD.doPasteLink(Sender: TObject);
var
  Service, Topic, Item : String;
begin
  if GetPasteLinkInfo (Service, Topic, Item) then
  begin
    AppName.Text := Service;
    TopicName.Text := Topic;
    ItemName.Text := Item;
    DdeClient.SetLink (Service, Topic);
    DdeClientItem.DdeConv := DdeClient;
    DdeClientItem.DdeItem := ItemName.Text;
  end;
end;
```

алоқа боғлангандан кейин DDE бўйича келувчи маълумотларни бошқариш керак, бу иш TDdeClientItem объектининг OnChange ҳолатлар қайта ишловчисида ажарилади.

```
procedure TFormD.DdeClientItemChange(Sender: TObject);
```

```
begin
```

```
DdeDat.Lines := DdeClientItem.Lines;
```

```
end;
```

Бу иш TDdeClientItem объектининг ягона вазифасидир.

TDdeClientConv объектига яна икки вазифа юкланади: маълумотларни серверга узатиш ва макросларни бажариш. Бунинг учун бу объектнинг мос услублари мавжуд. Куйида макросларнинг серверда бажарилишини кўриб ўтамиз:

```
procedure TFormD.doMacro(Sender: TObject);
```

```
begin
```

```
DdeClient.ExecuteMacroLines(XEdit.Lines, True);
```

```
end;
```

Пересылка данных на сервер:

```
procedure TFormD.doPoke (Sender: TObject);
```

```
begin
```

```
DdeClient.PokeDataLines(DdeClientItem.DdeItem,XEdit.Lines);
```

```
end;
```

Назорат саволлари:

1. DDE ни изоҳланг.
2. DDEнинг қўлланилиши
3. DDEлар орасида алоқа ўрнатиш усуллари.
4. Дастурлар орасида динамик боғланишнинг яна қандай усуллари биласиз?

Таянч иборалар:

DDE

DDE серверлар.

DDE мижозлар.

DDE алоқа

Динамик боғланиш.

## 8-маъруза. OLE

### Режа:

5. OLE асослари
6. TOLEContainer объекти
7. OLE дастурига мисол
8. OLE объектини маълумотлар омборида сақлаш

### Адабиётлар руйхати:

1. Компьютердаги ёрдам файллари.
2. Бобровский «Delphi 5», «Питер» Москва 1997г.
3. Шумаков «Delphi 4 разработка баз данных», «Питер» Москва 1996г.
4. Пачеко, Тейксер «Delphi 5 пособие программиста», «Питер» Москва 1999 г.
5. Фаронов «Delphi 4 учебное пособие», «Питер» Москва 1995 г.

### *OLE асослари*

OLE сўзи Objects Linked and Embedded (Боғланган ва Ички Объектлар – Б ва ИО ) маъносини билдиради. Дастурлар орасида бўлинадиган маълумотлар *OLE объект* дейилади. OLE объектларини сақловчи дастур, *OLE контейнер (OLE Container)* дейилади.

*Масалан*, MicroSoft Word хужжатга график объектларни, аудио ҳамда видеоклипларни ва бошқа кўплаб объектларни жамлаши мумкин (бундай хужжатни баъзан тузилган хужжат - compound document дейилади).

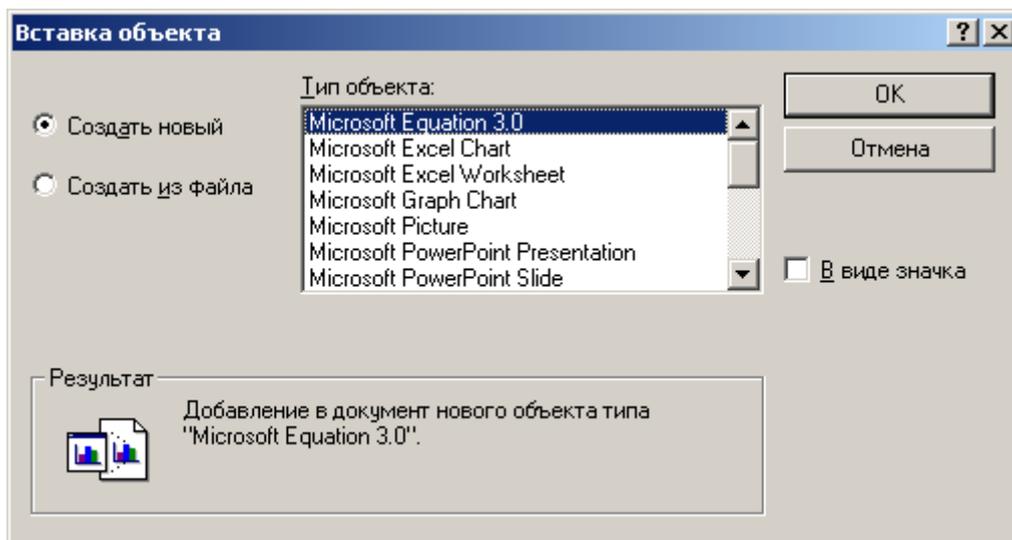
Номидан кўриниб турибдики, OLE объектларни OLE контейнерга кўшиш ёки унга ўрнатиш мумкин. Биринчи ҳолатда маълумотлар дискда файл кўринишида сақланади ва ихтиёрий дастур уларга мурожаат қилиши ҳамда ўзгартириши мумкин бўлади. Иккинчи ҳолатда маълумотлар OLE контейнерига ўрнатилади ва бу маълумотларни фақат у кўриши ва ўзгартириши мумкин.

Агар DDE ёрдамида матнлар билан ишлаш мумкин бўлса, OLE ёрдамида эса, дастурларга ихтиёрий типдаги маълумотларни киритиш мумкин. Худди DDEда

кўриб ўтганимиздек, мижоз дастур (OLE контейнер) ишлаши учун OLE сервер мавжуд бўлтиш талаб қилинади. Мижоз дастурда ишлаётган фойдаланувчи хар сафар OLE объектга кўриш ёки таҳрирлаш учун мурожаат қилганда сервер дастур ишга туширилади ва маълумотлар шу ерда таҳрирланади.

### ***TOLEContainer объекти***

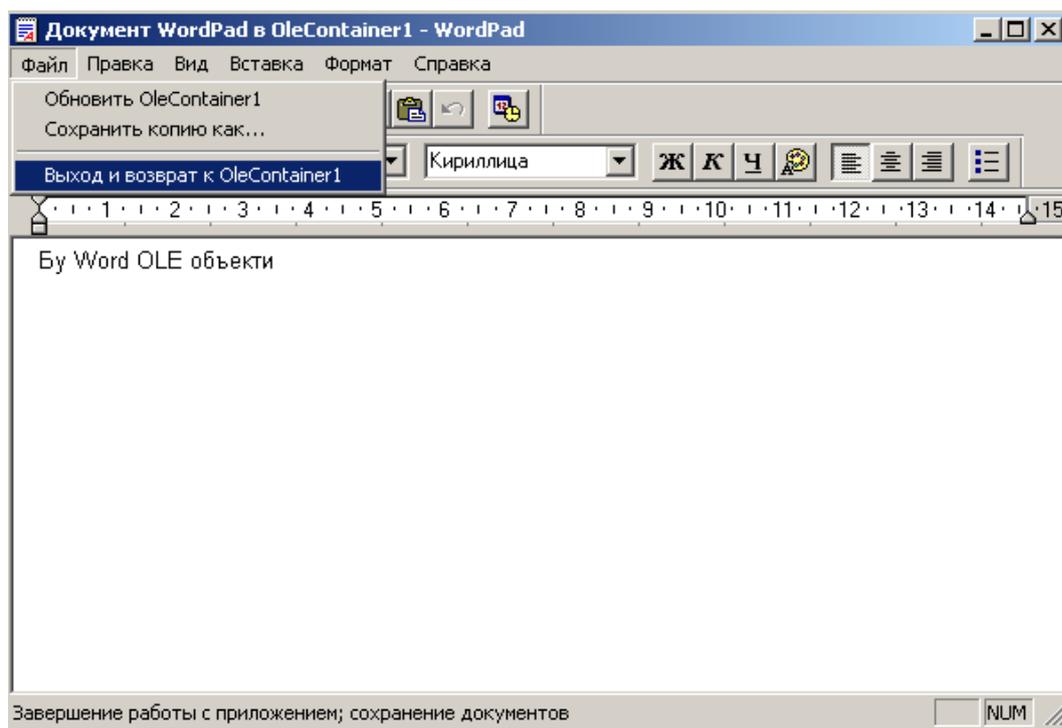
TOLEContainer объекти System саҳифасида жойлашган ва OLE-контейнер дастурларини тузиш учун ишлатилади. TOLEContainer OLEнинг ҳамма ички мураккабликларини яширади ва дастурчи учун анча оддий интерфейсни тақдим қилади. OLE объектни ишлатувчи оддий дастур мисолини кўриб ўтамиз. Янги лоиха яратинг ва формага TOLEContainer объектини ташланг, Объектлар Инспекторидаги ObjClass ёки ObjDoc хусусиятларига сичқонни икки марта босинг – шунда Windowsнинг стандарт “Insert Object” мулоқот ойнаси кўринади.



22-расм. “Insert Object” мулоқот ойнаси

Бу мулоқот ойнасида тизимда мавжуд барча OLE-серверлар рўйхати келтирилган. OLE-объект типи сиз танлаган сервер томонидан аникланади. Агар сиз янги объект яратсангиз (Create New), у ҳолда, ОК тугмаси босилгач, янги объект яратиладиган OLE-сервер ишга тушади. Срвер дастуридан чиқилгач, янги OLE объект дастурга ўрнатилади (embedded object). Янги OLE объектни серверлардан бирининг форматида сақланувчи мавжуд файл ёрдамида ҳам яратиш мумкин. Бунинг учун Create from File бўлимини танлаш керак.

Келинг мисол сифатида дастуримизга Microsoft Word Document ни танлаб янги объект яратамиз.



23-расм. OLE хужжати киритилган дастур

OK тугмасини боссангиз MS Word юкланади ва у ерда ихтиёрий матнни териш мумкин. Ишни якунлаш мақсадида “File|Close and Return to Form1” буйруғини танланг.

OLE-контейнерга сичқонни босинг - MS Word OLE-объектдаги хужжат билан пайдо бўлади ва уни таҳрирлаш мумкин, бу ҳолда ҳамма ўзгаришлар OLE-объектда сақланади.

TOLEContainer дастурга объектнинг ўз хусусиятлари билан (яъни уни турли даражада катталаштириш ёки кичрайтириш мумкин – Zoom хусусияти) ёки пиктограмма, кўринишида бирга ўрнатади.

OLE-объект танлови фақат формалар дизайнери режимида эмас, балки дастур бажарилиш вақтида ҳам бажарилиши мумкин. Иш натижаларини файлларда сақлаш ва керак бўлганда уни файлдан қайта тиклаш мумкин, бунинг учун TOLEContainer иккита SaveToFile ва LoadFromFile услубларига эга.

### ***OLE дастурига мисол***

Қуйидаги дастур TOLEContainerнинг барча асосий имкониятларини намоёниш қилади ва қуйидагилар имконини беради:

- дастур бажарилиш вақтида янги OLE контейнер яратиш;

- OLE объектни Windowsнинг стандарт “Insert Object” ойнаси ёрдамида, ёки Clipboard ёрдамида, ёки “олиб ташлаш” (drag-and-drop) технологияси ёрдамида ўрнатиш;

- OLE объектни файлда сақлаш ва уни қайта тиклаш;

OLE объектларни drag-and-drop технологияси ёрдамида ўрнатиш анча қизиқарли. MS Wordни юкланг (унинг ойнаси ва дастур ойнасини экранда ёнма-ён ўрнатиш), ихтиёрий матнни теринг, уни белгиланг ва сичқон ёрдамида тортиб дастурнинг бош оёғасига ташланг. Энди дастурда шу матнни сақловчи янги ички OLE контейнер ҳосил бўлади. Бу имкониятни дастурий йўл билан бажариш анча мушкул.

### ***OLE объектини маълумотлар омборида сақлаш***

Баъзан OLE объектини файлларда эмас, маълумотлар омборларида сақлаш зарурати туғилади (жадвалнинг BLOB типли майдонида). Бу ҳолда OLE объект албатта ўрнатилган бўлиши керак. Афсуски, Delphiнинг стандарт тўпламида TDBOLEContainer типдаги объект мавжуд эмас, бунинг учун биз OLE объектни SaveToStream ва LoadFromStream усуллари билан маълумотлар омборига жойлаштирамиз ва унга мурожаат қиламиз. Масалан:

```
procedure TOLEForm.SaveOLE(Sender: TObject);
var
    B1St : TBlobStream;
begin
    With Table1 do B1St:=TBlobStream.Create(BlobField(FieldByName('OLE')),
    bmReadWrite);
    OLEContainer.SaveToStream(B1St as TStream);
    B1St.Free; end;
```

### Назорат саволлари:

1. OLE тушунчасига изоҳ беринг.
2. TOLEContainer компонентидан нима мақсадда фойдаланилади?
3. OLE объектларга мисоллар келтиринг.
4. OLE объектларнинг типлари қанақа бўлади?

### Таянч иборалар:

OLEContainer

OLE объекти

OLE серверлар

OLE ҳужжат

SaveToStream

LoadFromStream

## **9-Маъруза. Маълумотлар омбори билан ишлаш асослари.**

### **Режа:**

1. Маълумотлар омбори нима?
2. Маълумотлар омборига қўйиладиган талаблар.
3. Delphiда маълумотлар омбори билан ишловчи компонентлар шарҳи
4. Маълумотлар омборига мурожаат қилувчи дастур

### **Адабиётлар руйхати:**

1. Компьютердаги ёрдам файллари.
2. Бобровский «Delphi 5», «Питер» Москва 1997г.
3. Шумаков «Delphi 4 разработка баз данных», «Питер» Москва 1996г.
4. Пачеко, Тейксер «Delphi 5 пособие программиста», «Питер» Москва 1999 г.
5. Фаронов «Delphi 4 учебное пособие», «Питер» Москва 1995 г.

### ***Маълумотлар омбори нима?***

Бу дарсада биз маълумотлар омбори билан ишлаш асосларини кўриб ўтамиз. Маълумотлар омборининг мақсади маънига эга бўлган маълумотларни бир ёки бир нечта жойда ортиқча сарфларсиз (хотира ва захираларни иқтисод қилиб) сақлашдир. Шундай қилиб маълумотлар омбори иккита асосий мақсадларни кўзда тутади: маълумотларни ёўқолишдан сақлаш ва уларнинг ишончилигини ошириш.

### ***Маълумотлар омборига қўйиладиган талаблар.***

Демак, яхши лоихаланган маълумотлар омбори:

- фойдаланувчиларнинг маълумотлар омборига бўлган барча талабларини қондиради. Шунинг учун маълумотлар омборини лоихалашдан олдин фойдаланувчиларнинг маълумотлар омборига бўлган талабларини кенг миқёсда ўрганиб чиқиш зарур.

- Маълумотларнинг мослиги ва ишончилигини кафолатлайди. Жадвалларни тузишда фойдаланувчи томонидан мумкин бўлмаган маълумотлар киритиланинг олдини олиш учун маълум шартларни киритиш керак.

## ***Delphiда маълумотлар омбори билан ишловчи компонентлар шарҳи***

Delphiда маълумотлар омбори билан ишловчи жуда кўп компонентлар бўлиб, биз улардан асосан TTable, TDbgrids, TDataSource, TDbNavigator компонентлари билан танишамиз.



TTable объекти маълумотлар омборидаги мавжуд жадвал билан мулоқот ўрнатиш учун хизмат қилади. TTable ихтиёрий типдаги (FoxPro, ODBC, SQL ...) маълумотлар омборининг ҳар бир ёзувига ва майдонига тўғридан тўғри мурожаат қила олади. Бу компонент шунингдек, алоҳида ҳисоботлар билан ҳам мулоқот ўрната олади.

TTABLE объектидан фойдаланишдан олдин унга маълумотлар омбори алясини улаш керак, яъни шу компонентнинг DatabaseName хусусиятида чиқадиган рўйхатдан керакли алясни танлаш ва TableName хусусиятидаги рўйхатдан керакли жадвал номини танлаш керак. TTABLE объектини фаоллаштириш учун Active хусусияти қийматини **true** га ўтказиш керак.



TADOTABLE объекти ҳам худди TTABLE объекти каби маълумотлар омборидаги бирор жадвалга боғланиш ва унга мурожаат қилиш учун хизмат қилади. Бу объектдан асосан MSAccess маълумотлар омборини бошқариш тизимида яратилган омборлар билан ишлашга мўлжалланган. Бу объект асосан TADOconnection объекти билан бирга қўлланилиб, TADOconnection маълумотлар омборига уланади. Шундан сўнг бир ёки бир нечта TAdotable объектлари Connection хусусияти ёрдамида TADOconnectionга уланади ва TableName хусусияти ёрдамида керакли жадвалга уланади. Объектни фаоллаштириш учун Active хусусияти қийматини **true** га ўтказиш керак. Бу объект ёрдамида маълумотлар омборидан маълумотларни бирор филтр ёрдамида ажратиб олиш мумкин.



TDATASOURCE объекти бевосита TTable ёки TAdoTableга боғланиб, маълумотлар омборидаги ёзувларни таҳрирлаш, уларга мурожаат қилиш имконини беради. Бунинг учун компонентнинг DataSet хусусиятидаги рўйхатдан керакли Table

элементи танланади ва шу орқали икки объект бирбирига боғланади. Ҳар бир алоҳида . TDATASOURCE объекти битта маълумотлар омборидаги битта жадвалга улана олади.

Юқоридаги учала объект ҳам дастур бажарилиш вақтида кўринмайдиган объект бўлиб, Формалар Дизайнери кўринишида уларни формага ташлаганда ўлчамларни ўзгартириб бўлмайди. Уларни маълумотлар омборига Формалар Дизайнери режимида ҳам, дастурий йўл билан дастур бажарилиш вақтида ҳам боғлаш мумкин.

Бунинг учун қуйидагича кодлар ёзилади:

```
begin
```

```
Table1.DatabaseName:='DBDEMOS';
```

```
Table1.TableName:='animals.dbf';
```

```
Table1.Active:=True;
```

```
DataSource1.DataSet:=Table1;
```

```
DBGrid1.DataSource:=DataSource1;
```

```
end;
```



TDBGRID объекти маълумотлар омборидаги ҳисоботлар, жадволлар ва сўровлардаги маълумотларни жадвал кўринишида намойиш этиш учун қўлланади. Бу объект ёрдамида маълумотлар омборидаги ёзувларни намойиш қилиш, таҳрирлаш ва ўзгартириш мумкин. Киритилган ўзгартиришлар жорий ёзув устида боради ва бу ўзгаришлар фақат сиз бошқа ёзувга ўтганингизда, ёки дастурни ёпганингизда сақлаб қолинади. TDBGRID объекти бевосита Datasource хусусияти ёрдамида TDataSource объектга боғланади ва шу орқали маълумотларни намойиш этади.



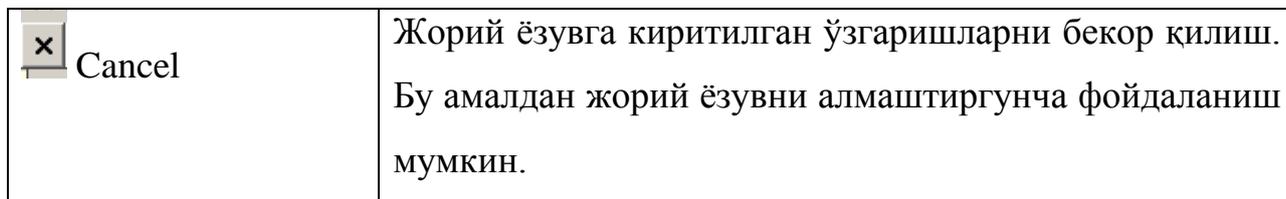
TDBNavigator (QDBCtrls) объекти дастурда TDBGRID ёки TDBEDIT компонентлаи орқали маълумотлар омбори ёзувларига мурожаат қилинаётган вақтда қўлланилади. TDBNavigator фойдаланувчига маълумотлар омборидаги ёзувларни таҳрирлаш ёки кўриб чиқишда қўл келади. Фойдаланувчи TDBNavigator

тугмалардан бирини босганда ша тугма билан боғланган амал дастурда бажарилади. Масалан, фойдаланувчи Insert тугмасини босганда маълумотлар жадвалида битта бўш қатор ҳосил бўлади.

### *TDBNavigator тугмалари ва улар бажарадиган амаллар*

Қуйида TDBNavigator тугмалари ва улар бажарадиган амалларни кўриб ўтамиз:

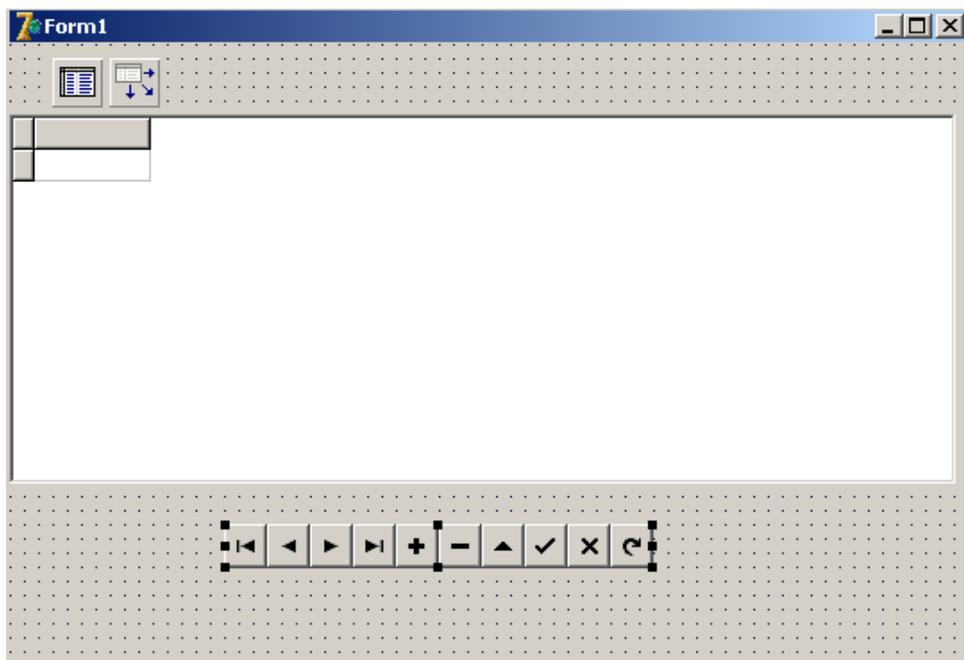
Тугма	Амал
 First	Маълумотлар омборидаги дастлабки ёзувни фаоллаштириш. У фақат жорий ёзув дастлабки ёзув бўлмагандагина фаол бўлади.
 Prior	Маълумотлар омборида жорий ёзувдан олдинги ёзувни фаоллаштириш. У фақат жорий ёзув дастлабки ёзув бўлмагандагина фаол бўлади.
 Next	Маълумотлар омборида жорий ёзувдан кейинги ёзувни фаоллаштириш. У фақат жорий ёзув охириги ёзув бўлмагандагина фаол бўлади.
 Last	Маълумотлар омборидаги охириги ёзувни фаоллаштириш. У фақат жорий ёзув охириги ёзув бўлмагандагина фаол бўлади.
 Insert	Жадвалга маълумотларни киритиш учун янги сатр қўшиш. Бунда сатрнинг ихтиёрий майдонига маълумот киритилганда ўзгаришлар сақланади
 Delete	Жорий ёзувни ўчириш. Бунда ёзувни ўчириш ҳақида сўров чиқарилади ва ўчирилган ёзув қайта тикланмайди.
 Edit	Жорий ёзувни ўзгартириш, таҳрирлаш мумкин бўлган ҳолатга ўтказиш.
 Post	Киритилган ўзгаришларни хотирада сақлаш. Бунда жорий майдонинг олдинги маълумотлари ўрнига киритилган ўзгаришлар сақланади.



### ***Маълумотлар омборига мурожаат қилувчи дастур***

Энди юқорида айтилганларни умумлаштириб, Delphiда маълумотлар омбори билан ишловчи дастурга мисол кўриб ўтамиз. Ушбу дастурда биз стандарт маълумотлар омбори бўлган DBDemos дан фойдаланамиз (Сиз ўзингиз тузган маълумотлар омборини ҳам аляс сифатида дастурларга киритишингиз мумкин).

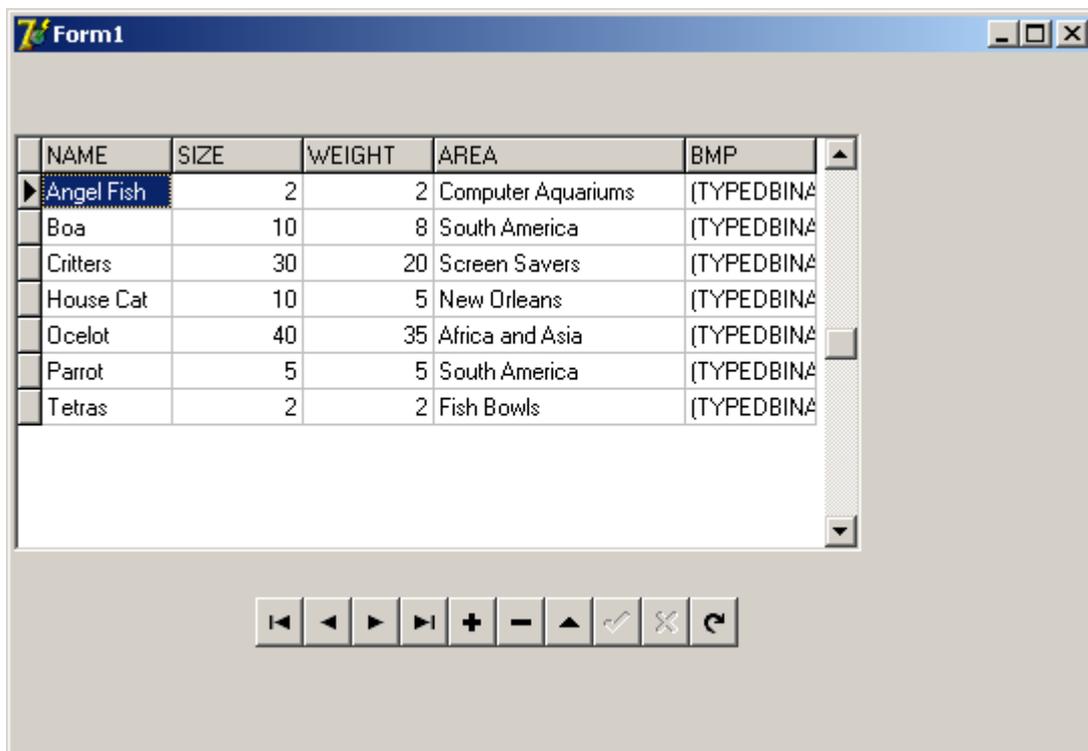
Демак, янги лоиҳа ҳосил қиламиз ва Формалар Дизайнери режимида формага BDE саҳифасидан TTable, DataAcces саҳифасидан TDataSours, Data Controls саҳифасидан TDBGrid ва TDBNavigator компонентларини ташлаймиз



24-Расм. Формалар дизайнери кўриниши.

Демак ҳосил бўлган тасвир тахминан расмдагидек бўлишига эришамиз. Сўнгра, Table1 (TTable) объектининг DatabaseName хусусиятидаги рўйхатдан DBDemosни танлаймиз, TableName хусусиятидаги рўйхатдан Animals.dbf жадвалини танлаб, Activ хусусияти қийматини Trueга ўрнатамиз. Энди DataSours компонентининг DataSet хусусиятидан Table1ни танлаймиз. Охири ишимиз DBGrid1 ва DBNavigator1ларнинг DataSours хусусиятидан DataSours1 ни танлаб,

дастурни ишга туширамиз. Бунинг учун биз ҳеч қандай дастур ёзмадик ва бу компонентларнинг ўзи маълумотлар билан оддий амалларни бажариш имконини беради.



25-расм. Дастурнинг бажарилаш вақтидаги кўриниши.

### Назорат саволлари:

1. Маълумотлар омбори нима?
2. Маълумотлар омборига қандай талаблар қўйилади?
3. Delphiда маълумотлар омбори билан ишлашга мўлжалланган компонентлар.
4. TDBNavigator тугмалари ва уларнинг вазифаси.

### Таянч иборалар:

Маълумотлар омбори

Жадвал.

Сўров.

Ҳисоботлар.

TTable

TADOTable

TDataSource

TDBGrid

TDBNavigator

## Адабиётлар

1. Компьютердаги ёрдам файллари.
2. Бобровский «Delphi 5», «Питер» Москва 1997г.
3. Шумаков «Delphi 4 разработка баз данных», «Питер» Москва 1996г.
4. Пачеко, Тейксер «Delphi 5 пособие программиста», «Питер» Москва 1999 г.
5. Фаронов «Delphi 4 учебное пособие», «Питер» Москва 1995 г.

