

**O'zbekiston Respublikasi oliy va o'rta maxsus ta'lim
vazirligi**

Buxoro oziq-ovqat va engil sanoat texnologiyasi instituti

“KASBIY TA'LIM” FAKULTETI

“Informatika va axborot texnologiyalari” kafedrası

“Ob`ektga mo'ljallangan dasturlash” fanidan

MA`RUZALAR TO'PLAMI

Buxoro – 2010 y.

Ma`ruzalar matni muallifi: «Informatika va axborot texnologiyalari» kafedrası katta o`qituvchisi. Narziev U.Z.

Ma`ruzalar matni Buxoro oziq ovqat va engil sanoat texnologiyasi instituti «Kasbiy ta`lim» fakulteti 5140900 Kasb ta`limi "Informatika va axborot texnologiyalari" ta`lim yo`nalishi talabalariga «Ob`ektga mo`ljallangan dasturlash» fanidan ma`ruzalarni o`tkazish uchun mo`ljallangan.

Taqrizchilar: BuxOO va ESTI «Informatika va axborot texnologiyalari» kafedrası mudiri dots. Yodgorov O.O.

Mundarija

Kirish	4
1 – ma`ruza. Asosiy tushunchalar.....	5
1.1. Dasturiy zahiralarga talablar.....	5
1.2. Dasturlash muhiti tuzilishi.....	6
1.3. Dasturlash muhitining asosiy tashkil etuvchilari.....	6
1.4. Delphi dasturlash muhitining yordam tizimi	9
1.5. Komponentlar to'plami	12
2 – ma`ruza. Loihani boshqarish.	18
2.1. Delphi loihasi.....	18
2.2. Loihani boshqarish.....	20
2.3. Menyu bo'limlari sharhi.....	21
3-ma`ruza. Komponentlar to'plami sharhi.....	25
3.1. Standart komponentlar	25
3.2. Additional sahifasi	28
3.3. Dialogs sahifasi.....	29
3.4. System sahifasi	29
4-ma`ruza. Multimedia dasturlarini yaratish vositalari	32
4.1. Delphida multimedia	32
4.2. TMediaPlayer komponenti	33
4.3. Multimedia dasturining ikki ko'rinishi.....	34
5-Ma`ruza. Matnli va grafik chop qilish	37
5.1. Matnli rejimda chop qilish	37
5.2. Formannng ko'rinishini chop qilish.....	38
5.3. Grafik chop qilish (TPrinter ob`ekti)	38
5.4. Printer xususiyatlari.....	38
6-Ma`ruza. Delphida ob`ektlarning xususiyatlari.	40
6.1. Umumiy tushunchalar.	40
6.2. Vizual komponentlar xususiyatlarini dastur bajarilish vaqtida boshqarish	42
6.3. SHAPEDEM1 dasturi.....	42
6.4. SHAPEDEM2 dasturi.....	45
7-Ma`ruza. DDE	53
7.1. DDE asoslari	53
7.2. DDEning qo'llanilishi	54
7.3. DDE-serverlar	54
7.4. DDE-mijozlar.....	55
8-ma`ruza OLE	59
8.1. OLE asoslari	59
8.2. TOLEContainer ob`ekti.....	60
8.3. OLE dasturiga misol	61
8.4. OLE ob`ektini ma'lumotlar omborida saqlash.....	62
9-Ma`ruza. Ma'lumotlar ombori bilan ishlash asoslari.	64
9.1. Ma'lumotlar ombori nima?	64
9.2. Ma'lumotlar omboriga qo'yiladigan talablar.	64
9.3. Delphida ma'lumotlar ombori bilan ishlovchi komponentlar sharhi	65
9.4. TDBNavigator tugmalari va ular bajaradigan amallar.....	67
9.5. Ma'lumotlar omboriga murojaat qiluvchi dastur.....	68
Adabiyotlar	71

Kirish

Xozirgi vaqtda kompter texnologiyalari rivojlanib borayotgan vaqtda asosiy e`tibor foydalanuvchilarga emas, dasturchilarga qaratilmokda. Shuni hisobga olib, talabalarda dasturlar tuzish ko`nikmalarini hosil qilish va ularda kerakli amaliy dasturlarni yarata olish imkoniyati shakllantirishga katta e`tibor qaratilmoqda.

O`zbekiston Respublikasi Kadrlar Tayyorlash Milliy dasturida ham yosh avlodni yangi kompyuter texnologiyalarini egallashi, ularning bilimlarini chet el talabalari saviyasiga etkazish dolzarb masala qilib ko`yilgan.

“Informatika va axborot texnologiyalari” kafedrasida “Ob`ektga yo`naltirilgan dasturlash” fani kiritildi. Delphi amaliy dasturlar yaratishning kuchli muhiti hisoblanib, hozirgi vaqtda dunyodagi dasturchilarning 35%idan ko`prog`i shu dasturlash tilidan foydalanadi.

Ushbu ma`ruzalar matnida yuqori saviyali dasturlash tillaridan biri hisoblangan Delphi da amaliy dasturlarni yaratish usullari keltirilgan.

1 – ma`ruza. Asosiy tushunchalar.

Reja:

1. Dasturiy zahiralarga talablar.
2. Dasturlash muhiti tuzilishi.
3. Dasturlash muhitining asosiy tashkil etuvchilari.
4. Delphi dasturlash muhitining yordam tizimi.

Adabiyotlar ruyxati:

1. Bobrovskiy «Delphi 5», «Piter» Moskva 1997g.
2. Shumakov «Delphi 4 razrabotka baz dannых», «Piter» Moskva 1996g.
3. Pacheko, Teyksera «Delphi 5 posobie programmista», «Piter» Moskva 1999 g.



Dasturiy zahiralarga talablar.

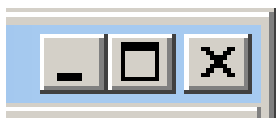
Delphi bu yuqori sifatga ega bo'lgan dasturlar yaratish vositasidir. Delphini yuklash uchun kamida 4MB tezkor xotiraga ega bo'lgan 386 kompyuter kerak bo'ladi. Lekin normal ishlash uchun kamida 8MB tezkor xotiraga ega bo'lgan 66MHz li 486DX kompyuteri to'g'ri keladi.. Delphida yaratilgan uncha katta bo'lmagan dasturlar ixtiyoriy kompyuterda ishlay oladi. Boshqacha qilib aytganda ularga Delphi muhiti uchun zarur bo'lgan tezkor xotira va protsessor tezligi talab qilinmaydi.

Dasturlash muhiti tuzilishi.

Delphi dasturlash muhitining tashqi ko'rinishi Windowsda ko'rish mumkin bo'lgan boshqa dasturlash muhitlari ko'rinishidan farq qiladi. Masalan, Borland Pascal for Windows 7.0, Borland C++ 4.0, Word for Windows, Program Manager – bularning barchasi MDI dasturlari hisoblanadi va Delphiga o'xshamaydi. MDI (Multiple Document Interface) – bitta asosiy oynaning ichida bir nechta quyi tipdagi oynalarni boshqarishni bildiradi.

Delphi muhiti esa, Single Document Interface (SDI) deb nomlanuvchi boshqa xususiyatga amal qiladi va alohida joylashgan bir nechta oynadan iborat.

Agar siz Delphiga o'xshash SDI dasturlardan foydalansangiz, boshqa dasturlar oynalari uni yopib qo'yimasligi uchun ularni pastga tushirib qo'yish kerak. Agar boshqa dasturga o'tish zaruriyati tug'ilib qoladigan bo'lsa, Delphi dasturi bosh oynasidagi kichraytirish tugmasini bosish kifoya.



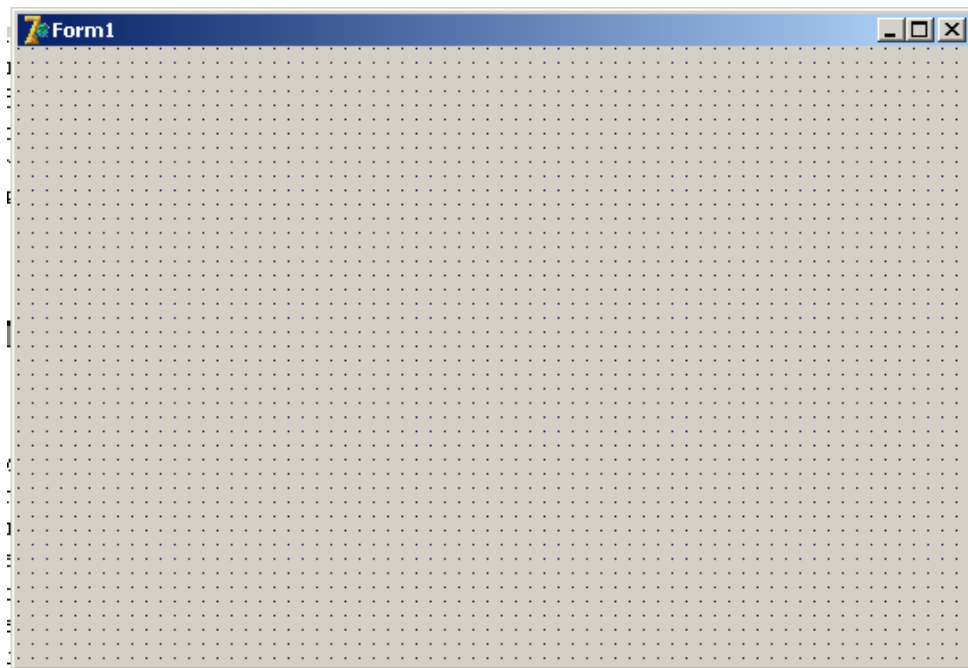
Dasturlash muhitining asosiy tashkil etuvchilari.

Quyida Delphining asosiy tashkil etuvchilari keltirilgan:

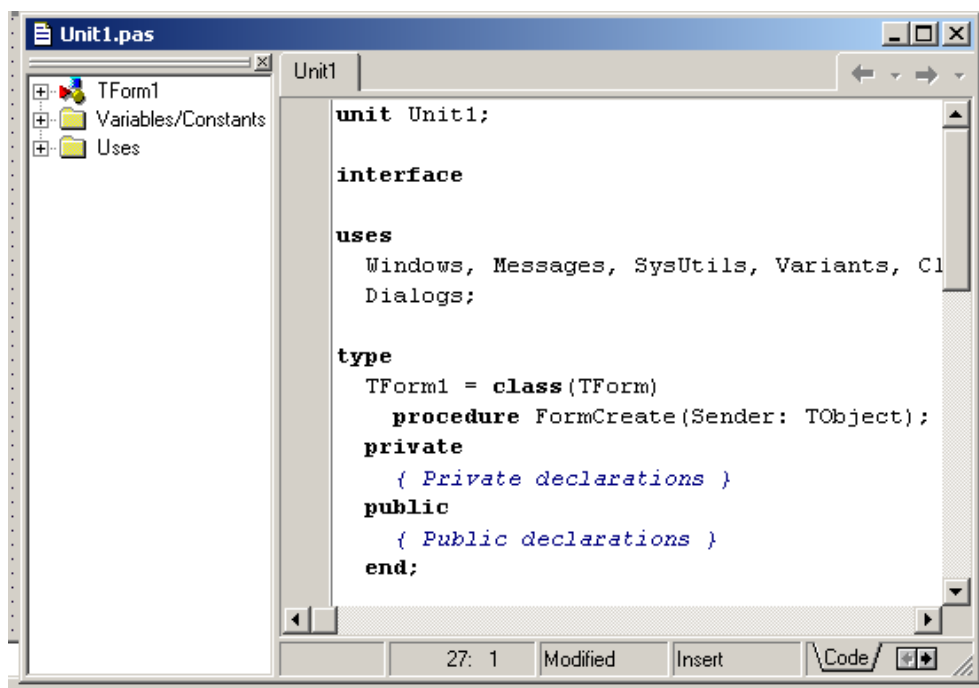
1. Formalar dizayneri (Form Designer)
2. Dastur matni muharriri (Editor Window)
3. Komponentlar to'plami (Component Palette)
4. Ob'ektlar inspektori (Object Inspector)
5. Yordam tizimi (On-line help)

Delphining bulardan tashqari sizga dasturlash muhiti va dasturni sozlash uchun xizmat qiluvchi tizim menyusi, uskunalar chizgichi kabi boshqa vositalari ham bor.

Delphida dasturchilar ko'p vaqtlarini Formalar dizayneri va Dastur matni muharriri orasida o'tish bilan o'tkazadilar. Siz dasturlashni boshlashdan oldin shu ikki muhim elementni ajrata olishingizga ishonch hosil qiling. Formalar dizayneri 1-rasmda, Dastur matni muharriri 2-rasmda keltirilgan.



1-rasm. Formalar dizayneri.

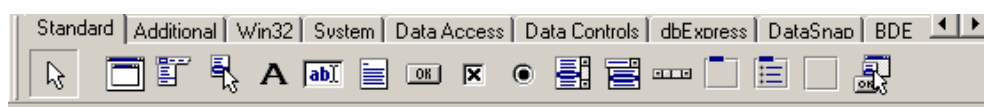


2-rasm. Dastur matni muharriri

Delphidagi Formalar dizaynerining tuzilishi shu qadar oddiyki, dasturning tashqi ko'rinishini yaratish bolalar o'yiniga o'xshab ketadi. Formalar dizayneri dastlab oddiy bo'sh oynadan iborat bo'ladi va siz uni keyinchalik komponentlar to'plamidan olingan ob'ektlar bilan to'ldirishingiz mumkin.

Formalar dizaynerining muhimligiga qaramasdan, dasturchilarning ko'p vaqtlari Dastur matni muharririda kechadi. Chunki bu erda dasturning asosiy kodi (ish bajaruvchi qismi) yoziladi.

Komponentlar to'plami sizga formalar dizayneriga o'rnatish va dasturning tashqi ko'rinishida ishtirok etuvchi ob'ektlarni tanlash uchun xizmat qiladi. Komponentlar to'plamidan foydalanish uchun avval kerakli ob'ekt ustiga sichqonchani bosing va so'ngra formalar dizayneriga sichqonchani bosish bilan tanlangan ob'ektni formaga qo'yishingiz mumkin.

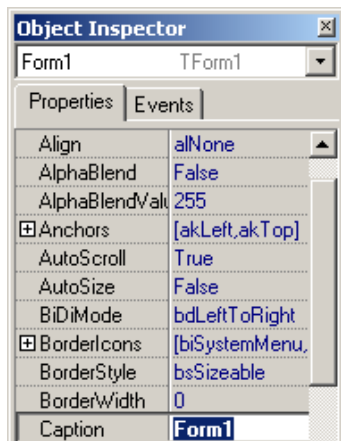


3- rasm. Komponentlar to'plami.

Komponentlar to'plamida ob'ektlar guruhlar buyicha alohida sahifalarga bo'lingan. To'plamda Standard, Additional, Dialogs va ho kazo sahifalar yig'ilgan. Agar siz bu nomlarning ixtiyoriysini bossangiz, shu guruh ob'ektlari joylashgan sahifaga o'tishingiz mumkin.

Masalan, siz formaga TEdit ob'ektini tashladingiz; siz uni bir joydan ikkinchi joyga ko'chirishingiz va uning o'lchamlarini o'zgartirishingiz mumkin. Boshqa ko'pchilik ob'ektlar bilan ham xuddi shunday ishlarni amalga oshirish mumkin. Lekin, dastur bajarilishida formada ko'rinmaydigan (TMenu yoki TdataBase kabi) ob'ektlar o'z ko'rinishini o'zgartirmaydi.

Formalar dizayneridan chap tomonda siz ob'ektlar inspektorini ko'rishingiz mumkin. Bu oynadagi ma'lumotlar formada tanlangan ob'ektning tipiga qarab, o'zgarib borishiga e'tibor bering. Har bir komponent bu ob'ektidir, va uning ko'rinishi hamda harakatini ob'ektlar inspektori yordamida o'zgartirish mumkinligini unutmang!



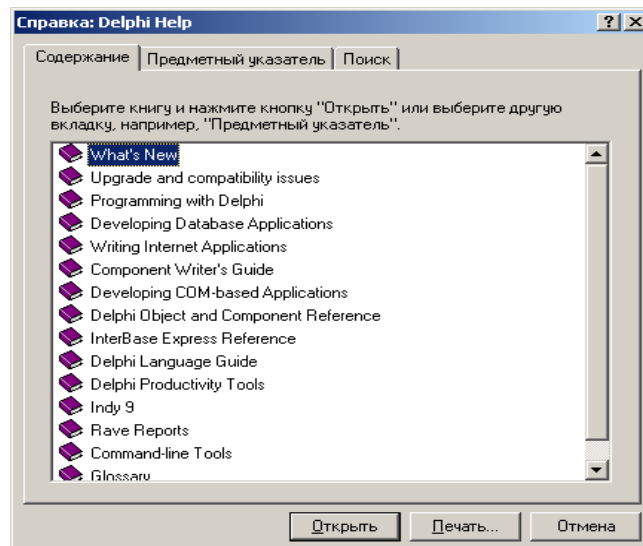
4-rasm. Ob`ektlar inspektori.

Ob`ektlar inspektori joriy ob`ektning harakatlarini aniqlash uchun ikki sahifadan tashkil topgan. Birinchi sahifa – xususiyatlar ro'yxati bo'lsa, ikkinchisi ob`ektning holatlari ro'yxatidir. Joriy ob`ektning biror tomonini o'zgartirish kerak bo'lsa, siz albatta Ob`ektlar inspektoriga murojaat qilishingiz kerak. Misol uchun siz TLabel elementining nomini o'zgartirishingiz, yoki uning Caption, Left, Top, Height, va Width kabi xususiyatlari yordamida joylashishini o'zgartirishingiz mumkin.

Ob`ektlar inspektori oynasining quyi qismidagi sahifa nomlaridan ob`ektlarning xususiyatlari va holatlari oynasiga o'tish mumkin. Holatlar sahifasi dastur matni muharriri bilan bevosita bog'liq bo'lib, undagi holatlarning ixtiyoriysiga sichqonchani ikki marta bossak, ob`ektning shu holatiga tegishli dastur sarlavhasi muharrirda paydo bo'ladi va sizga shu holatda bajariladigan ishlarni dastur ko'rinishida yozish imkoniyati beriladi. Delphi bu imkoniyatini keyinoq to'liq ko'rib o'tamiz.

Delphi dasturlash muhitining yordam tizimi

Delphi muhitining oxirgi muhim qismi bu yordam tizimidir (on-line help). Bu vositaga murojaat qilish uchun avval menyu satridan Help so'ngra Contents buyug'ini tanlash kerak. Ekranda 5-rasmda ko'rsatilgan - **ixtiyoriy ma'lumotni tez topish yordam tizimi** namoyon bo'ladi.



5-рasm. Delphi yordam tizimi

Yordam tizimi joriy holatga bog'liq hisoblanadi, ya'ni siz F1 tugmasini bosishingiz bilan joriy holatga mos ma'lumotni olishingiz mumkin. Masalan, Ob'ektlar inspektorida biror xususiyatni tanlab, F1 tugmasini bossak – joriy xususiyatning qo'llanilishi haqida ma'lumotga ega bo'lamiz

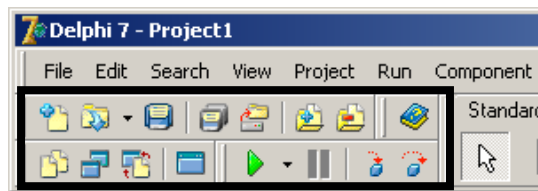
Qo'shimcha elementlar

Bu bo'limda biz asosan dasturlash uchun yordamchi hisoblangan uchta qo'shimcha elementlar haqida gapiramiz:

- Menyu (Menu System)
- Tezkor murojaat tugmalari (SpeedBar)
- Rasmlar muharriri (Image Editor)

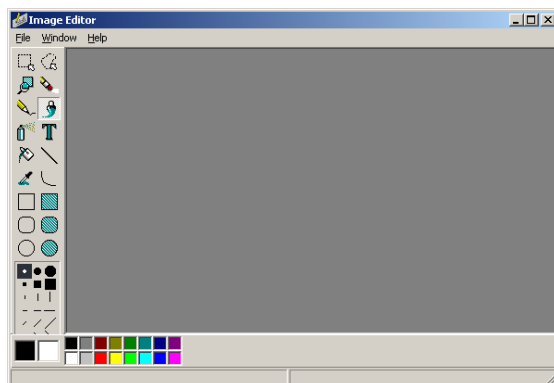
Menyu Delphining eng tezkor va moslashuvchan vositasidir, chunki, undagi buyruqlarni tezkor tugmalar yordamida ham bajarish mumkin. Menyuning qulayligi shundaki, bu erda tugmachalar va belgilardan farqli ravishda aniq so'zlar ishlatiladi.

SpeedBar bevosita menyuning quyi qismida, komponentlar to'plamidan chap tomonda joylashadi.



6-рasm: Tezkor murojaat tugmalari (SpeedBar)

7-rasmda keltirilgan Rasmlar muharriri Windowsdagi Paint dasturi kabi ishlaydi. Siz bu ob`ektga Tools menyusning Image Editor buyrug'ini tanlash bilan murojaat qilishingiz mumkin.



7-rasm. Rasmlar muharriri

Endi dasturchilar Delphida ko'p qo'llaydigan elementlar bilan tanishib o'tamiz.

Instrumental vositalar

Yuqorida ko'rib o'tilganlardan tashqari Delphi tarkibida yana beshta vosita mavjud.

Ular quyidagilar:

- Ichki dastur nazoratshisi
- Tashqi dastur nazoratshisi
- Buyruq satri kompilyatori
- WinSight
- WinSpector

Bu vositalarning alohida guruhga bulinganligi ularning boshqalaridan ko'ra kamroq amamiyatga molik ekanligida emas, balki, ular dasturlash jarayonida ancha mavhum texnik vazifalarni bajarishidadir.

Delphida kuchli dasturchi bo'lish uchun siz Delphi dastur nazoratshisi qanday tuzilganligini bilishingiz kerak. Nazoratshi sizga yozilgan dasturni qadamba-qadam tekshirish, bir martada dastur matnining bir qatorini bajarish va dasturdagi uzgaruvchilar qiymatlarini alohida ko'rish oynasi (Watsh)da kuzatish imkonini beradi.

Yuqorida sanab o'tilgan besh elementdan eng muhimi bo'lgan ichki dastur nazoratshisi dastur matni muharriri oynasida ishlaydi. Tashqi dastur nazoratshisining ichkidani farqi u

ancha tez va kuchliligidadir. Lekin uni ishlatish uchun avval Delphi muhitidan chiqish kerak.

Endi Kompilyatorlar haqida gaplashaylik. Tashqi kompilyator DCC.EXE deb nomlanadi, u asosan dasturni tashqi dastur nazoratshisi yordamida o'tqazishdan oldin ishlatiladi. Lekin ko'pchilik dasturchilar o'z dasturlarini Delphi muhitining o'zida nazoradan o'tkazadilar. WinSight va WinSpector Windows muhitida kuchli dasturchilar uchun qiziqarli hisoblanadi. Bu vositalar ikkilamchi hisoblanadi va faqat tor texnik maqsadlarda ishlatiladi.

Bu ikki vositadan WinSight ancha foydaliroq. Uning asosiy vazifasi Windows tomonidan yuborilayotgan xabarlardan sizni voqif qilib turishdir. Delphi malakasiz foydalanuvchilardan bu tizimning murakkab qismlarini yashirishga mo'ljallangan bo'lsada, Windows holatlarni boshqarish mumkin bo'lgan tizim sanaladi. Delphi sizga Windowsning hamma xabarlariga javob berish imkoniyatini beradi. Natijada, malakali foydalanuvchilarga WinSight zarur bo'lib qoladi.

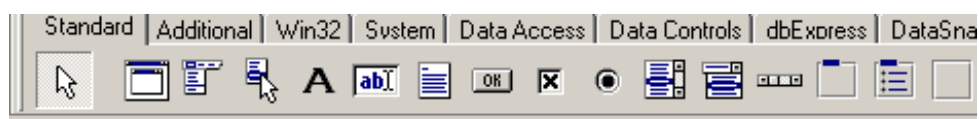
WinSpector EHMning joriy holatini matnli fayllarga saqlab qo'yish uchun xizmat qiladi; va siz bu fayllar yordamida dasturning qaysi qismi ishlayotganligini va nimalarni o'zgartirish kerakligini bilib olishingiz mumkin.

Standart komponentlar

Komponentlar to'plami

Delphi dasturlash muhiti bilan tanishishni davom ettirar ekanmiz, komponentlar to'plami haqida gapirishga to'g'ri keladi.

Komponentlar to'plamining birinchi sahifasida 14ta ob'ektlar joylashgan bo'lib, hamma dasturchilar albatta ushbu ob'ektlardan foydalanishgan. Chunki, juda ham kamchilik tugma, ro'yxat, kiritish oynalari va ho kazolardan foydalanmasdan dastur tuza oladi. Bu ob'ektlarning barchasi Windowsdagi sichqoncha va muloqot oynasi kabi muhimdir.



8-rasm. Standart Komponentlar to'plami

Har bir sahifadagi komponentlar to'plami va joylashishini o'zgartirish mumkin. Demak, siz mavjud komponentlarni o'rnini o'zgartirish bilan birgalikda ularning tarkibiga yangilarini qo'shishingiz mumkin.

Delphining standart komponentlari va ularni ishlatish haqidagi qisqacha izoh quyida keltirilgan. Bu komponentlarni o'rganayotgan vaqtda qo'l ostingizda kompyuter bo'lgani ma'qul, chunki birvarakayiga ularning ishlash printsiplari bilan ham tanishishingiz mumkin.

-  **TMainMenu** – o'z dasturingizga bosh menyu qo'shish imkonini beradi. **TmainMenu**ni formaga qo'ygan vaqtda u kichkina belgi ko'rinishini oladi; bunday belgilarni "ko'rinmas komponent" deb ataydilar, chunki, dastur bajarilish vaqtida ular ekranda ko'rinmaydi. Menyuni yaratish uch qadamdan iborat: (1) TMainMenu formaga o'rnatish, (2) ob'ektlar inspektorining Items xususiyati yordamida Menyu dizaynerini chaqirish, (3) Menyu Dizaynerida menyu qismlarini kiritish.

-  **TPopupMenu** yordamchi menyuni yaratish. Bu menyu sichqonchani o'ng tugmasi bosilganda ko'rinadi.

-  **TLabel** matnlarni ekranda namoyish qilish uchun qo'llanadi. Agar siz ob'ektlar inspektorining Font xususiyatiga sichkonchani ikki marta bossangiz, **Label**ning shrifti, rangi, harflar rangi va o'lchamlarini o'zgartirishingiz mumkin.

-  **TEdit** – Windowsdagi standart ma'lumot kiritish oynasi. U qisqa matnlarni namoyish qilishi va dastur bajarilish vaqtida foydalanuvchiga o'z ma'lumotlarini kiritish imkonini beradi.

-  **TMemo** – TEditning boshqacha ko'rinishi. Katta matnlar bilan ishlashni ko'zda tutadi. TMemo matn qismlarini qatordan qatorga ko'chirishi, buferda saqlashi, oxirgi amallardan voz kechishi, umuman olganda oddiy muxarrir amallarini bajara oladi. TMemo 32Kb matn hajmi bilan chegaralanadi, bu taxminan 10-20 sahifa matnga tengdir.

-  **TButton** dastur bajarilish vaqtida tugma bosilishi bilan biror amal bajarilishini ko'zda tutadi. Delphida hamma narsa oddiy. TButtonni formga ko'yib, unga

ikki marta sichqon tugmasini bossak, biz dastur matni muharririda tugma bosilish holatiga dastur tuzishimiz mumkin bo'lgan oynaga o'tamiz. Endi dastur matnini bajariladigan buyruqlar kodi bilan to'ldirish kerak. Masalan:

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

```
begin
```

```
MessageDlg('Ishlaringiz yaxshimi?',mtsonfirmation,mbYesNoCancel,0);
```

```
end;
```

-  **TCheckBox** chap tomonida kichkina darchasi bo'lgan matn satrini akslantiradi. Darchaga uning belgilanganligini bildiruvchi belgi quyish mumkin. Masalan, Worddagi Pechat muloqot oynasining Pechat v fayl bo'limida aynan shu ob'ekt ishlatilgan.

-  **TRadioButton** bir nechta holatlardan birini tanlash imkonini beradi. Masalan, Worddagi Pechat muloqot oynasining chop qilinishi kerak bo'lgan sahifalar oralag'ini tanlash bo'limida aynan shu ob'ekt ishlatilgan.

-  **TListBox** ro'yxatli ma'lumotlarni namoyish qilishga mo'ljallangan masalan Worddagi Otkryt buyrug'i yordamida ochiladigan muloqot oynasidagi fayllar va kataloglar ro'yxati aynan shu ob'ektga joylashtirilgan.

-  **TSomboBox** tashqi ko'rinishidan ListBoxga o'xshab ketadi, ular orasidagi farq ComboBoxning yuqorisida ma'lumot kiritish mumkin bo'lgan maydoni ham bor. ComboBoxning ko'plab turlari mavjud, ko'p ishlatiladigani esa, ro'yxati pastga ochiladigan (drop-down combo box)dir.

-  **TScrollbar** – o'tkazish yo'lakchasi. Ko'pincha tahrirlanadigan yoki ko'rinadigan ma'lumotlar ekran chegaralaridan chiqib ketganda paydo bo'ladi.

-  **TGroupBox** Windowsga formadagi ob'ektlarning qanday joylashganligini bildirish uchun qullash mumkin.



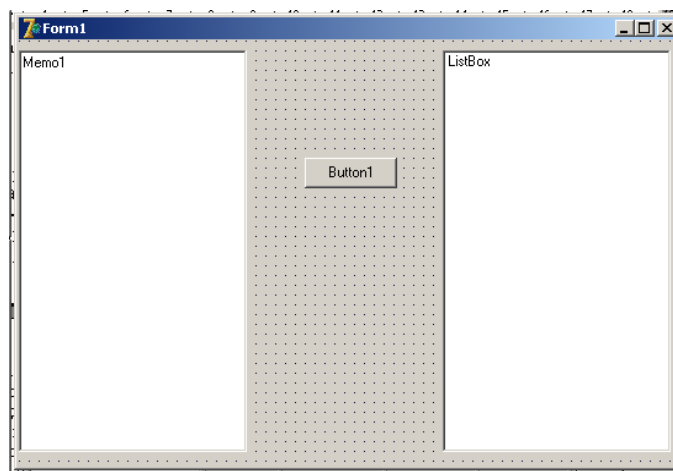
- **TPanel** – TGroupBoxga o'xshash ob'ekt bo'lib, bezash uchun ishlatiladi.

Agar sizga bundan ko'ra to'liqroq ma'lumotlar kerak bo'lsa, Delphining yordam tizimidan foydalaning

Ob'ektlar Inspektori haqida.

Oldinroq biz Ob'ektlar Inspektori (Object Inspector) bilan bir oz tanishib o'tdik. Endi u bilan yaqinroq tanishish vaqti etdi. Ob'ektlar Inspektorining asosiy vazifasi formaga tashlangan ixtiyoriy ob'ektning xususiyatlarini tahrirlash hisoblanadi. Bundan tashqari u yordamida formaning xususiyatlarini ham o'zgartirish mumkin.

Ob'ektlar Inspektorini o'rganishning eng qulay usuli – u bilan ishlashdir. Avval yangi loiha ochamiz File | New Project. So'ngra formaga TMemo, TButton va TListBox ob'ektlarini 9-rasmda ko'rsatilganidek joylashtiring.



9-rasm. Formaga ob'ektlarni joylashtirish.

Xususiyatlar bilan ishlashni boshlashda Ctl3D xususiyatidan boshlaymiz (jimlik qoidasi bo'yicha faol). Formaga sichqon tugmasini bosing, Ob'ektlar Inspektoriga o'ting va Ctl3D xususiyati qiymatini sichqoncha yordamida bir necha marta o'zgartirib ko'ring. Bu amal formaning tashqi ko'rinishini o'zgartirayotganligiga e'tibor qarating. Formaning Ctl3D xususiyati qiymatini o'zgartirish shu formaga tashlangan hamma ob'ektlarning shu xususiyati qiymatini o'zgartiradi.

Formaga dastlabki holatini qaytaring va undagi TMemo ob'ektini belgilang, so'ngra <Shift> tugmasini bosib turib, TListBox ob'ektini belgilang. Endi siz formadagi bir nechta ob'ektlarni belgilay olasiz. Ob'ektlar guruhini belgilaganda Ob'ektlar Inspektorida faqat

shu guruh ob`ektlari uchun umumiy bo'lgan xususiyatlar ko'rinadi va ularning qiymati o'zgartirilishi belgilangan ob`ektlarning barchasi uchun amal qiladi. Agar bu guruhda umumiy xususiyatlarga ega bo'lgan ob`ektlar bo'lmasa, Ob`ektlar Inspektorida hech qanday xususiyat ko'rinmaydi.

Nazorat savollari

1. Delphi dasturlash muhitining tizim zahiralariga talablari.
2. Formalar dizaynerining vazifalari.
3. Dastur kodi muharriri vazifalari.
4. Ob`ektlar inspektori nima?
5. Komponentlar to'plamidan nima maqsadda foydalanish mumkin?

Tayanch iboralar.

Formalar dizayneri

Ob`ektlar inspektori

Komponentlar to'plami

TListBox

TMemo

TPanel

TButton

TEdit

TLabel.

2 – ma`ruza. Loihani boshqarish.

Reja

1. Delphi loihasi.
2. Loihani boshqarish

Adabiyotlar ruyxati:

1. Kompyuterdagi yordam fayllari.
2. Bobrovskiy «Delphi 5», «Piter» Moskva 1997g.
3. Shumakov «Delphi 4 razrabotka baz dannых», «Piter» Moskva 1996g.
4. Pacheko, Teyksera «Delphi 5 posobie programmista», «Piter» Moskva 1999 g.
5. Faronov «Delphi 4 uchebnoe posobie», «Piter» Moskva 1995 g.

Delphi loihasi.

Delphining ixtiyoriy loihasi kamida oltita fayl yaratadi. Ulardan uchta loiha tomonidan yaratiladi va dasturchilar ularning tarkibini ko'pincha o'zgartirmaydilar. Ular quyidagilar:

- Loihaning bosh fayli, dastlab PROJECT1.DPR deb nomlanadi.
- Dasturning dastlabki moduli /unit/, u ishning boshida avtomatik ravishda yaratiladi. Fayl UNIT1.PAS deb nomlanadi va uning nomini ixtiyoriy nom bilan almashtirish mumkin. Masalan, MAIN.PAS.
- Bosh forma fayli jimlik qoidasiga binoan UNIT1.DFM deb nomlanadi, u bosh formaning tashqi ko'rinishi haqidagi ma'lumotlarni saqlaydi.
- PROJECT1.RES fayli avtomatik holda yaratiladi va loiha uchun belgini saqlaydi.
- PROJECT1.OPT fayli matnli fayl hisoblanib, siz kompilyatorda qo'llagan o'rnatmalaringizni saqlaydi.
- PROJECT1.DSK fayli ishchi sohaning holati haqidagi ma'lumotlarni saqlaydi.

Albatta loiha nomini o'zgartiriganda RES, OPT va DSK kengaytmali fayllarning nomlari ham o'zgaradi.

Dastur kompilyatsiyasidan keyin quyidagi kengaytmali fayllar hosil bo'ladi:

DCU – tayyor modullar.

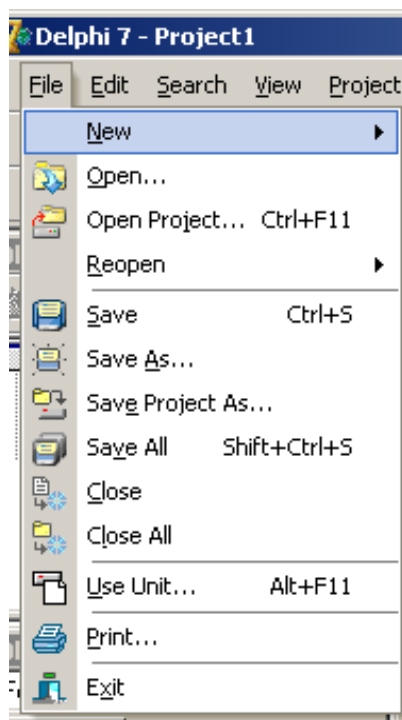
EXE – ishchi fayl.

DSM – dasturning muhitda ishga tushishi uchun xizmat qiluvchi fayl, uning hajmi juda katta va ish yakunlangach uni o'chirish tavsiya qilinadi.

~PA, ~DP - backup. – muharrirning vaqtinchalik fayllari.

Menyuning “File” bulimi

Loihaning fayllari bilan ishlash uchun “File” menyusini tanlang. “File” menyusini quyidagicha tuzilgan:

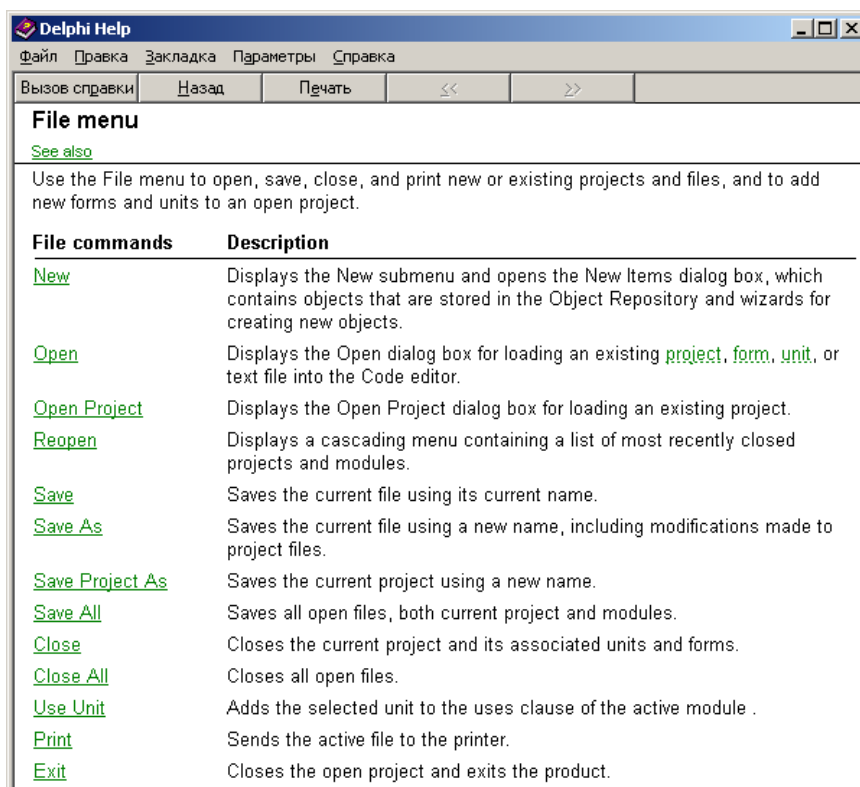


10-rasm. Fayl menyusini ko'rinishi

Ko'rib turganingizday bu erda beshta bo'lim mavjud:

- Birinchi bo'lim loiha bilan to'liq ishlashga mo'ljallangan.
- Ikkinchi bo'lim loihaning formalari, modullari va qismlari ustidan nazoratga mo'ljallangan.
- Uchinchi bo'lim loihaga fayllarni qo'shish va olib tashlash uchun xizmat qiladi.
- To'rtinchi bo'lim chop qilishni boshqaradi.
- Beshinchi bo'lim Delphidan chiqish.

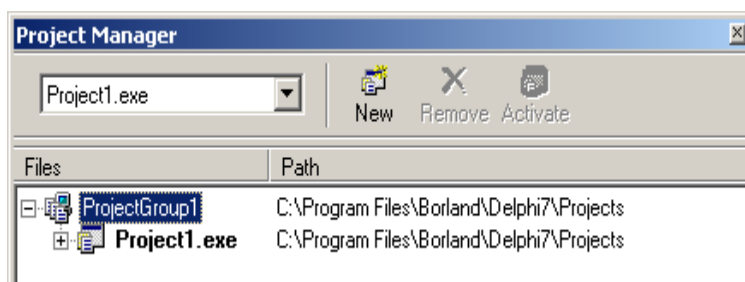
“File” menyusining har bir bo’limi yordam tizimida izohlangan bo’lganligi uchun ularga to’xtab o’tirmaymiz. Bularni kurish uchun “File” menyusini tanlab, F1 tugmasini bosong.



11-rasm Delphida “File” menyusidan foydalanish haqidagi yordam.

Loihani boshqarish

Endi siz Fayl menyusida loiha yaratishni o’rganib oldingiz va lohani boshqaruvchi Loihalar Menejmenti sharhiga o’tsak bo’ladi. Loihalar Menedjeri ikki qismdan iborat bo’lib, birinchi qism boshqaruvchi tugmalar va undan pastdagi ikkinchi qismda loiha tarkibiga kiruvchi modullar nomi va ularning diskdagi o’rni ko’rsatilgan.



12-rasm. Loihalar menedjeri.

Siz boshqaruvchi tugmalar yordamida loihaga yangi fayllarni qo’shishingiz yoki olib tashlashingiz mumkin. Bu o’zgarishlash dastlabki faylga o’zgartirishlar kiritadi, ya’ni

loihaga yangi modul qo'shilsa unga yo'llanma DPR kengaytmali faylda avtomatik holda yoziladi..

Menyu bo'limlari sharhi

Menyuning "File" bo'limini yuqorida ko'rib o'tdik. Quyida menbning - "Edit", "Search", "View" va "Compile" bo'limlarini qisqacha ko'rib o'tamiz.

Menyuning "*Edit*" bo'limi

"Edit" menyusi "Undo" va "Redo" buyruqlarini saqlaydi, ular muharrirda ishlayotganda bexosdan qilingan xatolardan (masalan, matnnig kerakli qismi o'chirib yuborildi) voz kechishga mo'ljallangan.

"Cut", "Copy", "Paste" va "Delete" buyruqlari Windowsning boshqa dasturlaridagi kabi, lekin ularni faqat matnlarga emas, balki, ob'ektlarga ham qullash mumkin.

"Bring To Front", "Send To Back", "Align" va "Size" ob'ektlarning formada turishi va ularning o'lchamlarini boshqarishga mo'ljallangan.

Menyuning "*Search*" bo'limi.

"Search" menyusida "Find Error" (xatoliklarni qidirish) buyrug'i mavjud bo'lib, u dastur bajarilish vaqtidagi xatoliklarni topishga mo'ljallangan. Xatolik haqidagi xabarda chiqarilgan raqamni "*Find Error*" muloqot o'rtasiga kiritish va muhit dasturdagi xatolik sodir bo'lgan qismni ko'rsatadi.

Menyuning "*View*" bo'limi.

"View" menyusining tarkibi quyidagicha:

- Project Manager (Loiha Menedjeri).
- Project Source –muharrirga loihaning bosh fayli (DPR) ni yuklaydi.
- Ekranda Object Inspectorni ko'rsatish kerak yoki yo'qlini bildiruvchi o'rnatma.
- Ekranda Alignment Palette ko'rsatish kerak yoki yo'qlini bildiruvchi o'rnatma.
- Browser – dasturdagi obektlar ierarxiasini namoyish qiluvchi vositani chaqirish.
- Watch, Breakpoint va Call Stack – dasturni boshqarish bilan bog'liq va ularga keyinroq to'xtalamiz.
- Component List – Komponentlar to'plami. Komponentlarni nomi bo'yicha qidirishda yoki sichqoncha yo'qligida qo'llaniladi.

- Window List – Delphi muhitida ochilgan oynalar ro'yxati.
- Toggle Form/Unit, Units, Forms – forma va mos modullar orasida utish, forma va modullar ro'yxatdan tanlanadi.
- New Edit Window – muharrirning qo'shimchak oynasini ochadi. Bu ko'pincha bir faylning ikki nusxasini bilan ishlaganda qulaylik yaratadi.
- SpeedBar i Component Palette –ekranda namoyish qilish o'rnatmalari.

Menyuning **“Compile”** bo'limi.

Menyuning “Compile” bo'limida loihani ishga tayyorlash (compile) yoki qayta ishlash (build) mumkin. Agar Compile yoki Run tanlansa, Delphi faqat oxirgi kompilyatsiyadan keyin o'zgartirilgan modullarni tayyorlaydi. Build all esa, loiha tarkibidagi hamma modullar va dastur matnlarini tayyorlaydi. Syntax Check buyrug'i esa, faqat dastur matni buyruqlari to'g'ri yozilganligini tekshiradi.

Eng pastki – Information buyug'i dastur haqidagi umumiy ma'lumotlarni chiqaradi.

Menyuning **“Run”** bo'limi.

“Run” bo'limidan dasturni ishga tayyorlash va dastur uchun buyruq satri parametrlarini berish maqsadida foydalanish mumkin.

Menyuning **Options** bo'limi.

“Options” tizim enyusining ancha murakkab qismidir. Siz bu erdan loiha va Delphining butun ishchi muhit uchun o'rnatmalarni o'zgartirishingiz mumkin. V “Options”ning etti buyrug'i mavjud:

Project

Environment

Tools

Gallery

Open Library

Install Components

Rebuild Library

Dastlabki to'rt buyruq muloqot oynalarini chaqiradi. Quyida menyuning “Options” bo'limiga qisqacha sharh keltirib o'tilgan:

- Project – joriy loihaga to'g'ridan-to'g'ri ta'sir qiluvchi parametrlarni o'rnatish.
- Environment – dasturlash muhitining tuzilishini boshqarish (IDE). Masalan, bu erda muxarrirda ishlatiladigan ranglarni o'zgartirish mumkin.

- Tools – bosh menyuning Tools bo'limidagi tashqi dasturlarni yuklash va ulardan voz kechish. Masalan, siz biror moduldan ko'p foydalansangiz, uni ro'yxatga qo'shib qo'yish mumkin.

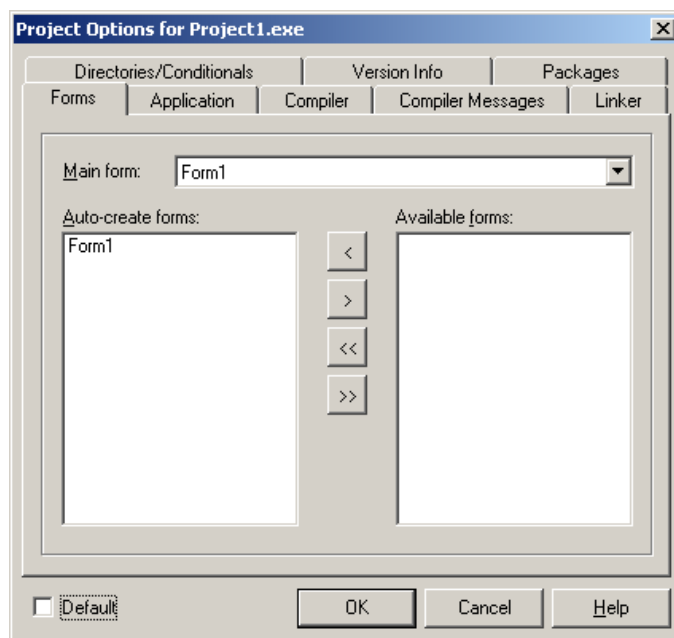
- Gallery – Formalar va loihalar eksperti uchun spetsifik o'rnatmalarni qo'llaydi.
- Oxirgi uchta buyruq komponentlar to'plamini tuzishga yordam beradi.

Options | Project muloqot oynasi beshta asosiy sahifadan iborat:

- Forms sahifasida loihada ishlatiladigan barcha formalar keltirilgan;
- Application sahifasida siz dastur uchun kerak bo'ladigan sarlavha, yordam fayli va belgilarni aniqlashingiz mumkin.

- Compiler sahifasi dastur matnidagi xatoliklar, dastur abajarilish vaqti xatoliklari, uni tayyorlash kabi vazifalar bilan shug'ullanadi.

- Linker sahifasida dasturni bog'lash sharoitlarini aniqlash mumkin.



13-rasm Options | Project muloqot oynasi.

- Directories/Conditionals sahifasi – joriy loiha uchun foydalaniladigan kataloglar ko'rsatiladi.

Nazorat savollari:

1. Dastur ishlashi uchun yaratiladigan fayllar.
2. Dastur tomonidan yaratiladigan fayllarning vazifalari nimalardan iborat?
3. Delphi muhiti menyulari.
4. Yordam tizimidan foydalanish tartibi.
5. Loihani boshqarish yo'llari.

Tayanch iboralar:

DCU; EXE; DSM; Loiha;

Dasturni bog'lash;

Dasturlarni yuklash;

Loiha menejeri.EXE

3-ma`ruza. Komponentlar to'plami sharhi.

Reja:

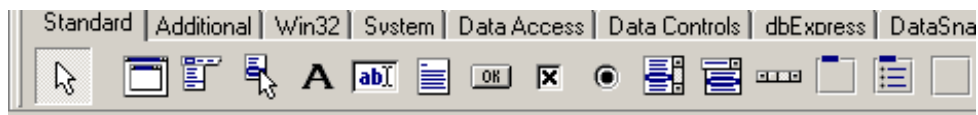
6. Standart komponentlar
7. Additional sahifasi
8. Dialogs sahifasi
9. System sahifasi

Adabiyotlar ruyxati:

6. Kompyuterdagi yordam fayllari.
7. Bobrovskiy «Delphi 5», «Piter» Moskva 1997g.
8. Shumakov «Delphi 4 razrabotka baz dannых», «Piter» Moskva 1996g.
9. Pacheko, Teyksera «Delphi 5 posobie programmista», «Piter» Moskva 1999 g.
10. Faronov «Delphi 4 uchebnoe posobie», «Piter» Moskva 1995 g.

Standart komponentlar

Komponentlar to'plamining birinchi sahifasida 14ta ob'ektlar joylashgan bo'lib, hamma dasturchilar albatta ushbu ob'ektlardan foydalanishgan. Chunki, juda ham kamchilik tugma, ro'yxat, kiritish oynalari va ho kazolardan foydalanmasdan dastur tuza oladi. Bu ob'ektlarning barchasi Windowsdagi sichqoncha va muloqot oynasi kabi muhimdir.



14-rasm. Standart Komponentlar to'plami

Har bir sahifadagi komponentlar to'plami va joylashishini o'zgartirish mumkin. Demak, siz mavjud komponentlarni o'rnini o'zgartirish bilan birgalikda ularning tarkibiga yangilarini qo'shishingiz mumkin.

Delphining standart komponentlari va ularni ishlatish haqidagi qisqacha izoh quyida keltirilgan. Bu komponentlarni o'rganayotgan vaqtda qo'l ostingizda kompyuter bo'lgani ma`qul, chunki birvarakayiga ularning ishlash printsiplari bilan ham tanishishingiz mumkin.

Kursor – komponent hisoblanmaydi, balki, biror ob`ektни tanlashdan voz kechish vositasidir.



- **TMainMenu** – o`z dasturingizga bosh menyu qo`shish imkonini beradi. **TmainMenu**ni formaga qo`yan vaqtda u kichkina belgi ko`rinishini oladi; bunday belgilarni "ko`rinmas komponent" deb ataydilar, chunki, dastur bajarilash vaqtida ular ekranda ko`rinmaydi. Menyuni yaratish uch qadamdan iborat: (1) TMainMenu formaga o`rnatish, (2) ob`ektlar inspektorining Items xususiyati yordamida Menyu dizaynerini chaqirish, (3) Menyu Dizaynerida menyu qismlarini kiritish.



- **TPopupMenu** yordamchi menyuni yaratish. Bu menyu sichqonchaning o`ng tugmasi bosilganda ko`rinadi.



- **TLabel** matnlarni ekranda namoyish qilish uchun qo`llanadi. Agar siz ob`ektlar inspektorining Font xususiyatiga sichkonchani ikki marta bossangiz, **Label**ning shrifti, rangi, harflar rangi va o`lchamlarini o`zgartishingiz mumkin.



- **TEdit** – Windowsdagi standart ma`lumot kiritish oynasi. U qisqa matnlarni namoyish qilishi va dastur bajarilish vaqtida foydalanuvchiga o`z matnini kiritish imkonini beradi.



- **TMemo** – TEditning boshqacha ko`rinishi. Katta matnlar bilan ishlashni ko`zda tutadi. TMemo matn qismlarini qatordan qatorga ko`chirishi, buferda saqlashi, oxirgi amallardan voz kechishi, umuman olganda oddiy muxarrir amallarini bajara oladi. TMemo 32Kb matn hajmi bilan chegaralanadi, bu taxminan 10-20 sahifa matnga tengdir.



- **TButton** dastur bajarilash vaqtida tugma bosilishi bilan biror amal bajarilishini ko`zda tutadi. Delphida hamma narsa oddiy. TButtonni formga ko`yib, unga ikki marta sichqon tugmasini bossak, biz dastur matni muharririda tugma bosilish holatiga dastur tuzimiz mumkin bo`lgan oynaga o`tamiz. Endi dastur matnini bajariladigan buyruqlar kodi bilan to`ldirish kerak. Masalan:

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

```
begin
```

```
MessageDlg('Ishlaringiz yaxshimi?',mtsonfirmation,mbYesNoCancel,0);
```

```
end;
```

-  **TCheckBox** chap tomonida kichkina darchasi bo'lgan matn satrini akslantiradi. Darchaga uning belgilanganligini bildiruvchi belgi quyish mumkin. Masalan, Worddagi Pechat muloqot oynasining Pechat v fayl bo'limida aynan shu ob'ekt ishlatilgan.

-  **TRadioButton** bir nechta holatlardan birini tanlash imkonini beradi. Masalan, Worddagi Pechat muloqot oynasining chop qilinishi kerak bo'lgan sahifalar oralag'ini tanlash bo'limida aynan shu ob'ekt ishlatilgan.

-  **TListBox** ro'yxatli ma'lumotlarni namoyish qilishga mo'ljallangan masalan Worddagi Otkryt buyrug'i yordamida ochiladigan muloqot oynasidagi fayllar va kataloglar ro'yxati oynan shu ob'ektga joylashtirilgan.

-  **TSomboBox** tashqi ko'rinishidan ListBoxga o'xshab ketadi, ular orasidagi farq ComboBoxning yuqorisida ma'lumot kiritish mumkin bo'lgan maydoni ham bor. ComboBoxning ko'plab turlari mavjud, ko'p ishlatiladigani esa, ro'yxati pastga ochiladigan (drop-down combo box)dir.

-  **TScrollbar** – o'tkazish yo'lakchasi. Ko'pincha tahrirlanadigan yoki ko'rinadigan ma'lumotlar ekrandan chiqib ketganda paydo bo'ladi.

-  **TGroupBox** Windowsga formadagi ob'ektlarning qanday joylashganligini bildirish uchun qullash mumkin.

-  **TPanel** – TGroupBoxga o'xshash ob'ekt bo'lib, bezash uchun ishlatiladi.

Eto polnyy spisok ob`ektov na pervoy stranitse Palitry Komponent. Esli Vam nujna dopolnitelnaya informatsiya, to vyberite na Palitre ob`ekt i najmite klavishu F1 - poyavitsya Spravochnik s polnym opisaniem dannogo ob`ekta.

Additional sahifasi

Standard sahifasida Windows 3.0 da paydo bo`lgan boshqarish ob`ektlari keltirilgan, Additional sahifasi yordamida esa, dasturning foydalanuvchi interfeysini yanada yaxshilash mumkin.

Komponentlar ro`yxati:

TBitBtn  – Tbuttonga o`xshash tugma, lekin unda tasvir (glyph)ni o`rnatish mumkin.).

TSpeedButton  - buyruqlarga tezkor murojaat panelini yaratish tugmasi. Odatda bu tugmaga faqat tasvir (glyph) joylashtiriladi.

TTabSet – Gorizontal sarlavhalar. Odatda ko`p sahifali oynalarni yaratish uchun TNoteBook bilan birgalikda qo`llanadi. Sahifalar nomlarini Tabs xususiyatida berish mumkin.:

TNoteBook – ko`p sahifali muloqotni yaratish uchun mo`ljallanganyu bunda har bir sahifaning o`z ob`ektlar to`plami bo`ladi. Ko`pincha TtabSet bilan birgalikda ishlatiladi.

TTabbedNotebook – sardavhalari o`zida bo`lgan ko`p sahifali muloqotlar yaratish.

TMaskEdit  - Tedit ga o`xshash, lekin, kiritishni formatlash mumkin. Format EditMask xususiyatida aniqlanadi.

TOutline – o`zaro bog`langan ma`lumotlarning ierarxik munosabatini ko`rsatishga mo`ljallangan.

TStringGrid  - matnli ma`lumotlarni jadval ko`rinishida chiqaradi.

TDrawGrid  - ixtiyoriy tipdagi ma`lumotlarni jadval ko`rinishida chiqaradi.

TImage  - formada grafik tasvirlarni namoyish qiladi tobira et. Asosan BMP, ICO, WMF fayllarni qabul qiladi.

TShape  - formada oddiy grafik ob`ekt (aylana, kvadrat ...)larni namoyish qiladi.

TBevel  - interfeysni relefli bezash uchun xizmat qiladi

THeader – jadvallar uchun o`zgaruvchan kenglikka ega bo`lgan sarlavhalarni saqlaydi.

TScrollBar  - formada ekranga sig`maydigan ob`ektlarni ko`rsatish imkonini beruvchi yo`lakchalarni hosil qiladi.

Dialogs sahifasi

Dialogs sahifasida Windowsning standart muloqot oynalarini chaqiruvchi komponentlar joylashgan. Muloqot oynalarining tashqi ko`rinishi Windowsning versiyasiga bog`liq bo`ladi. Bu sahifadagi ob`ektlar dastur bajarilish vaqtida ko`rinmaydi va mos muloqot oynalarini dasturiy yo`l bilan chaqirish kerak, masalan:

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

Dialogs sahifasida Windowsning muloqot oynalari komponentlari quyidagi tartibda joylashtirilgan: - OpenFileDialog; faylni tanlash - SaveDialog; fayl saqlash - FontDialog; shriftni sozlash - ColorDialog; rang tanlash - PrintDialog; chop qilish - PrinterSetupDialog; printerni sozlash - FindDialog; satrni qidirish- ReplaceDialog; almashlab qidirish

System sahifasi

Timer - taymer, OnTimer vaziyati har safar Interval xususiyatida ko`rsatilgan vaqtdan keyin bajariladi. Vaqt oralig`i 1 dan 65535 msgacha bo`lishi mumkin.

TPaintBox –chizish uchun joy. Holatlar qayta ishlovchisiga TpaintBoxdagi sichqonchanning mos koordinatalari qaytariladi

TFileListBox - maxsus ListBox unda (Directory xususiyatida) ko'rsatilgan katalogdagi fayllar ro'yxati chiqariladi.

TDirectoryListBox - maxsus ListBox unda joriy diskdagi kataloglar tuzilishi bilan tanishish mumkin. FileList xususiyatida TFileListBox ko'rsatilsa, unda keyingi katalogga o'tilganda mos fayllar chiqariladi.

TDriveComboBox – maxsus ComboBox undan joriy diskni tanlash mumkin. Agar DirList xususiyatida TDirectoryListBox ko'rsatilsa, disk almashtirilishi bilan mos kataloglar chiqariladi.

TFilterComboBox – fayllar nomi maskasini chiqarish uchun maxsus ComboBox. Maskalar ro'yxati Filter xususiyatida aniqlanadi. FileList xususiyatida maska o'rnatilayotgan TFileListBox ko'rsatiladi.

TMediaPlayer – multimedia vositalari bilan ishlashni ko'zda tutadi.

TOLEContainer – o'zida OLE ob'ektlarni saqlovchi konteyner.

Nazorat savollari:

1. Komponent deganda nimani tushunasiz?
2. Standart komponentlardan qaysilarini bilasiz?
3. Dialogs sahifasiga kiruvchi komponentlar qaysilar?
4. System sahifasidagi komponentlar qanday amallar uchun mo'ljallangan?

Tayanch iboralar:

Timer; TPaintBox; THeader; TMediaPlayer; TOLEContainer; TBitBtn

4-ma`ruza. Multimedia dasturlarini yaratish vositalari

Reja:

5. Delphida multimedia
6. TMediaPlayer komponenti
7. Multimedia dasturining ikki ko'rinishi.

Adabiyotlar ruyxati:

1. Kompyuterdagi yordam fayllari.
2. Bobrovskiy «Delphi 5», «Piter» Moskva 1997g.
3. Shumakov «Delphi 4 razrabotka baz dannyx», «Piter» Moskva 1996g.
4. Pacheko, Teyksera «Delphi 5 posobie programmista», «Piter» Moskva 1999 g.
5. Faronov «Delphi 4 uchebnoe posobie», «Piter» Moskva 1995 g.

Multimedia nima?

Bu savolning aniq javobi mavjud emas. Lekin bu erda bir umumiy qoidani keltirish mumkin: “multimedia” – kompyuterda ishlatiladigan harakat (animatsiya), ovozlari, videotasvirlarning hamma ko'rinishi bilan bog'liq tushunchadir.

Delphida multimedia

Delphida TMediaPlayer komponenti mavjud bo'lib, u sizga multimedia dasturlarini tuzishda hamma imkoniyatlarni yaratib beradi. Ushbu komponentdan foydalanish juda oddiy.

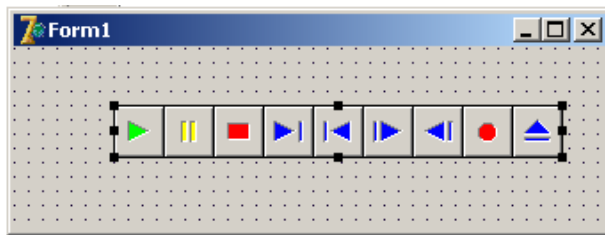
Bu oddiylikni ikki ma`noda tushunish mumkin:

- Bir tomondan – bu ixtiyoriy dasturchiga multimedia dasturini tuzishga imkon yaratilgan.
- Ikkinchi tomondan, bu komponentda multimedia fayllari bilan bajariladigan hamma asosiy amallar kiritilgan, lekin sizga quyi funktsiyalar kerak bo'ladigan bo'lsa, Delphi dasturlash tili imkoniyatlaridan foydalanishingiz mumkin.

Ushbu ma`ruzada komponent ishlaganda uning ichki funktsiyalari qanday ishlatilishiga to'xtalmaymiz.

TMediaPlayer komponenti

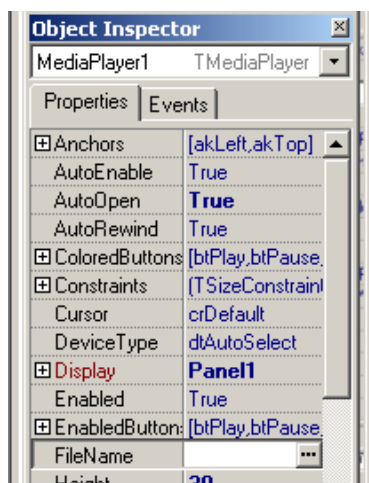
Dastlab yangi loiha yaratamiz va formaga TMediaPlayer (System sahifasi) komponentini tashlaymiz.



15-rasm: Formadagi TMediaPlayer komponenti.

TMediaPlayer komponenti qurilmani boshqarish paneli kabi tuzilgan. Magnitofonlardagi kabi bu erda ham “ishga tushirish”, “o’tkazish”, “yozish” va boshqa tugmalari bor.

Komponentni formaga tashlagach, siz Ob`ektlar Inspektorida "FileName" xususiyatini ko’rasiz.



16- Rasm. TMediaPlayerning Ob`ektlar inspektoridagi xususiyatlari

Shu erda sichqoncha tugmasini bosing va ro’yxatdan AVI, WAV yoki MID kengaytmali faylni tanlang. So’ngra AutoOpen xususiyatini True qiymatga o’rnatish kerak.

Bu qadamlar bajarilgach, dastur yuklanishga tayyor bo’ladi. Dasturni yuklab ishga tushirish (Play) tugmasini bosing va siz tanlagan videoklipingizni ko’rishingiz yoki musiqani tinglashingiz mumkin. Agar ovoz yoki videoklip ishga tushmasdan xatolik haqida xabar chiqarilsa, ikki holat bo’lishi mumkin:

1. Siz fayl nomini yoki ungacha bo’lgan yo’lni noto’g’ri kiritgansiz.

2. Kompyuteringizdagi multimedia qurilmalari Windowsga to'g'ri sozlanmagan. Bu esa, sizda kerakli qurilmaning yo'kligidan yoki kerakli drayverlar o'rnatilmaganligidan dalolat beradi.

TMediaPlayer komponentining yana bir muhim xususiyati - Display. Dastlabki misolimizda bu xususiyat bo'sh qoldirilgan va videoklip aloxida oynada ko'ringan edi. Lekin, videoklip namoyishida ekran sifatida masalan, **Panel**dan foydalanish mumkin. Formaga Tpanel komponentini tashlang va Caption xususiyatidagi matnni o'chiring. So'ngra TMediaPlayer uchun Display xususiyatidagi ro'yxatdan Panel1ni tanlang. Shundan so'ng dasturni yuklab, ishga tushirish tugmasini bosish kerak.



17- rasm. Panelda AVIni ko'rsatish.

Multimedia dasturining ikki ko'rinishi.

Multimediali dasturga misol.

Bu bo'limda biz multimediali dasturlarga misol ko'rib o'tamiz. Yangi loixa yarating (File | New Project). Formaga TMediaPlayer, shuningdek fayllarga murojaat qilish uchun, TFileListBox, TDirectoryListBox, TDriveComboBox, TFilterComboBox komponentlarini tashlang. DirectoryListBox1ning va FilterComboBox1ning FileList xususiyatlarida FileListBox1ni belgilang. DriveComboBox1ning DirList xususiyatiga DirectoryListBox1ni o'rnatang. FilterComboBox1ning Filter xususiyatiga kerakli fayllar kengaytmalarini kiriting:

AVI File(*.avi)|*.avi

WAVE File(*.wav)|*.wav

MIDI file(*.MID)|*.mid

FileListBox1da Sichqonchani ikki marta bosganda joriy fayl ishga tushirilishiga erishishimiz kerak. Holatlar qayta ishlovchisidagi FileListBox1ning OnDblClick holatiga quyidagi dastur matnini kiritamiz:

```
Procedure TForm1.FileListBox1DblClick(Sender:TObject);
```

```
begin
```

```
  with MediaPlayer1 do
```

```
    begin
```

```
      Close;
```

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

```
      Open;
```

```
      Play;
```

```
    end;
```

```
end;
```

Dasturni yuklab natijani ko'rishingiz mumkin.

Dastur bajarilish vaqtida MediaPlayer ob`ektining yoki videoklippning joriy holatini namoyish qilish zaruriyati tug'ilib qolishi mumkin. Buning uchun TMediaPlayer ob`ektining: Length, Position, OnNotify va boshqa xususiyatlari mavjud. Keling loihamizga vaqt indikatorini sifatida (TGauge) ob`ektini tashlaymiz, u o'tgan vaqtni foizlar bo'yicha ko'rsatadi. Uning ko'rsatkichini yangilash uchun taymerdan foydalanish mumkin. Formaga TTimer ob`ektini tashlang, uning uchun Interval = 100 (100 millisekund) ga o'rnatish. OnTimer holatiga quyidagini yozish kerak:

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

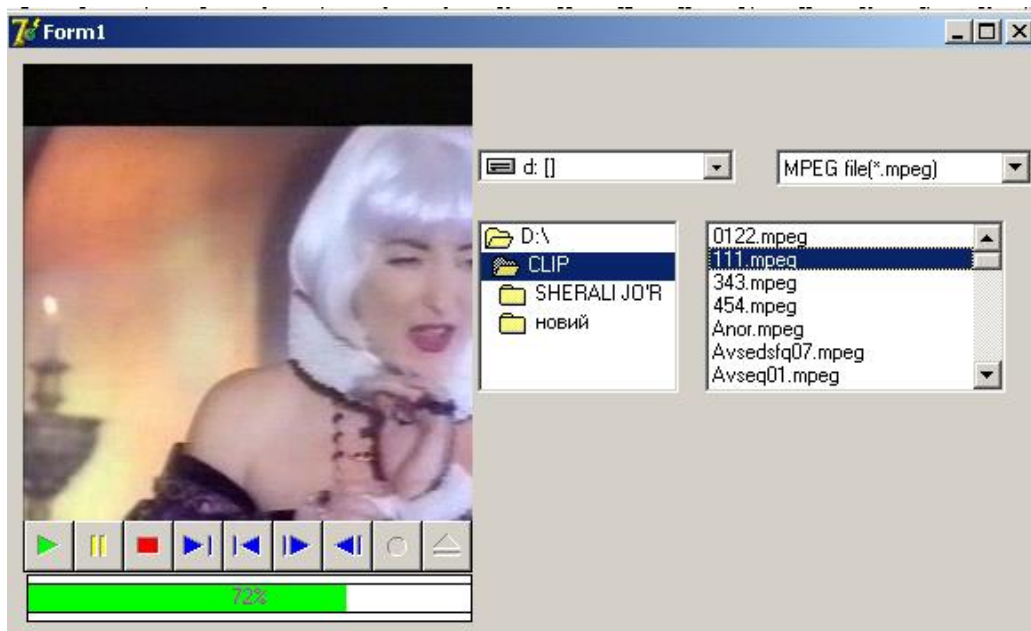
```
begin
```

```
  with MediaPlayer1 do
```

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

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

```
    end;
```



18 - Rasm. Ob`ektlarning joriy holatini namoyish qilish

Dasturni yuklang, fayl (AVI)ni tanlang va unga sichqon tugmasini ikki marta bosing. Videoklip namoyishi vaqtida biz qo'ygan vaqt indikatori o'tgan vaqtga bog'liq holda foizlarni chiqarishi kerak.

Nazorat savollari:

1. Multimedia nima?
2. TMediaPlayer komponenti nima maksadlarda ishlatiladi?
3. Multimedia fayllarini boshkarish uchun ishlatiladigan ob`ektlarni sanab o'ting.
4. Multimedia fayli loihaga qanday biriktiriladi?

Tayanch iboralar:

AVI; MediaPlayer; OnTimer; Xususiyat; Length; Position; OnNotify

5-Ma`ruza. Matnli va grafik chop qilish

Reja:

5. Matnli rejimda chop qilish
6. Formannng ko'rinishini chop qilish
7. Grafik chop qilish (Tprinter ob`ekti)

Adabiyotlar ruyxati:

6. Kompyuterdagi yordam fayllari.
7. Bobrovskiy «Delphi 5», «Piter» Moskva 1997g.
8. Shumakov «Delphi 4 razrabotka baz dannых», «Piter» Moskva 1996g.
9. Pacheko, Teyksera «Delphi 5 posobie programmista», «Piter» Moskva 1999 g.
10. Faronov «Delphi 4 uchebnoe posobie», «Piter» Moskva 1995 g.

Matnli rejimda chop qilish

Agar siz matnli rejimda hujjatni chop qilmoqchi bo'lsangiz bu ish quyidagicha amalag oshiriladi. Printerга siz ddiy matnli fayl bilan ishlaganday murojaat qilasiz, yagona farq AssignFile protsedurasi o'rniga AssignPrn protsedurasini yozish kerak. Misolda printerга matnning birta satri chiqariladi:

```
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 dasturlash muxitida ham fayllar bilan ishlashda Pascal dagiga o'xshash protseduralar ishlatiladi. Quyida biz System modulidagi ba`zi amallarni ko'rsatib o'tamiz:

- Assign o'rniga AssignFile

- Close o'rniga CloseFile
- Text o'rniga TextFile

Formannng ko'inishini chop qilish

Ba`zida dasturchilar uchun ekrandagi tasvirning nusxasi kerak bo'lib qoladi. Delphi bu ish juda oddiy bajariladi - TForm ob`ektida Print uslubi bor bo'lib, uni kerakli vaqtda chaqirish etarli.

Grafik chop qilish (TPrinter ob`ekti)

Qanday qilib, Delphida yaratilgan dasturdan grafik ma`lumotlarni printerga uzatish mumkin? Buning uchun maxsus (TPrinter sinfidagi) Printer ob`ekti mavjud. U dasturga Printers modulini kiritgandan so'ng faollashadi. Bu ob`ekt yordamida grafik ma`lumotlarni printerga uzatish xuddi ularni ekranga chiqarishday oddiylashib qoladi. Printer ob`ektining xususiyatlari va uslublarini ko'rib o'tamiz.

Printer xususiyatlari

Aborted – Bulev (mantiqiy) tip; foydalanuvchi printer ishini Abort.uslubi yordamida to'xtatganligini bildiradi.

Canvas - Kanva, grafik ma`lumotlarni chiqarish uchun joy.

Fonts – O'rnatish mumkin bo'lgan shriftlar ro'yxati.

Handle - Windows API to'g'ridan to'g'ri chaqirilganda ishlatiladi.

Orientation – Sahifaning mo'ljali, vertikal yoki gorizontal.

PageWidth, PageHeight, PageNumber – mos holda sahifaning kengligi, balandligi va tartib raqami.

Printers Tizimda o'rnatilgan barcha printerlarning nomlari ro'yxati

PrinterIndex Joriy printerning raqami. JQB chop qilish uchun bu erda -1 turishi kerak.

Printing – Chop qilish (BeginDoc uslubida) boshlanganligini bildiruvchi mantiqiy tip.

Title – Print uchun sarlavha.

Printer uslublari:

Abort - BeginDoc rejimida boshlangan chop qilishni to'xtatish

BeginDoc - Kanvada chizish boshlangandan oldin chaqiriladi.

EndDoc – Kanvada hamma narsa tayyor bo'lgach printer chop qilishni boshalaydi.

NewPage - Yangi sahifaga o'tish.

Qolgan uslublar bilan oddiy hollarda foydalanish kerak emas.

Shunday qilib, grafik ma'lumotlarni chop qilish tartibi quyidagicha:

- BeginDoc uslubini bajaramiz
- kanvada (Canvas) kerakli narsalarni chizamiz
- ma'lumotni bir nechta sahifaga joylash kerak bo'lsa, NewPage uslubini qo'llash
- chizilganlarni printeriga yo'llab, EndDoc uslubini bajarimiz.

Nazorat savollari:

1. Multimedia nima?
2. Delphida multimediadan nima maqsadda foydalanish mumkin?
3. TMediaPlayer komponentining vazifasi nimalardan iborat?

Tayanch iboralar:

BeginDoc; NewPage; EndDoc; AssignFile; CloseFile; TextFile

6-Ma`ruza. Delphida ob`ektlarning xususiyatlari.

Reja:

1. Umumiy tushunchalar.
2. Vizual komponentlar xususiyatlarini dastur bajarilish vaqtida boshqarish.
3. SHAPEDEM1 dasturi.
4. SHAPEDEM2 dasturi.

Adabiyotlar ruyxati:

1. Kompyuterdagi yordam fayllari.
2. Bobrovskiy «Delphi 5», «Piter» Moskva 1997g.
3. Shumakov «Delphi 4 razrabotka baz dannых», «Piter» Moskva 1996g.
4. Pacheko, Teyksera «Delphi 5 posobie programmista», «Piter» Moskva 1999 g.
5. Faronov «Delphi 4 uchebnoe posobie», «Piter» Moskva 1995 g.

Umumiy tushunchalar.

Xususiyatlar ob`ektning muhim atributasi ekanligiga diqqatingizni qaratgandirsiz. Foydalanuvchi (dasturchi) uchun xususiyat bu biror tuzilishning qiymat saqlovchi maydonidir. Lekin, bu oddiy maydon qiymati o'zgarishi bilan komponentning tashqi ko'rinishi butunlay o'zgaradi, chunki, xususiyatlar o'zida ob`ektning shu maydonga yozish va o'qish holatlari inkapsulyatsiyasiga bog'liq bo'ladi. Xususiyatlar ikki maqsadda xizmat qiladi. Birinchidan, ular forma yoki komponentning tashqi ko'rinishini aniqlaydi. Ikkinchidan esa xususiyatlar forma yoki komponentning harakatini belgilaydi.

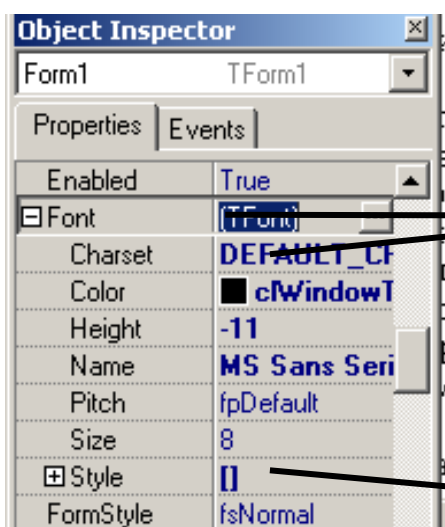
Xususiyatlar bir nechta tiplarga bo'lish mumkin.

- Oddiy xususiyatlar – bularnig qiymatlari sonlar yoki satlar bo'ladi. Masalan, Left va Top xususiyatlari forma yoki komponentning yuqorigi chapki burchagi koordinatalarini aniqlovchi butun sonlarni saqlaydi. Caption va Name xususiyatlari forma yoki komponentning sarlavhasi va nomini bildiruvchi satrlarni saqlaydi.

- Sanab o'tiladigan xususiyatlar – bular mavjud ro'yxatdagi qiymatlardan birini qabul qila oladigan xususiyatlarir. Oddiy misol – **Boolean** tipidagi xususiyat, *True* yoki *False* qiymatini qabul qila oladi.

- Ichki ma'lumotli xususiyatlar – bular ichki qiymatlarga (yoki ob'ektlarga) ega bulgan xususiyatlardir. Bunday xususiyatlar Object Inspektorda chap tomonidan “+” belgisi bilan ajralib turadi. Bunday xususiyatlar ham ikki xilda bo'ladi: *ko'pliklar* va *birlashgan qiymatlar*. Object Inspektorda ko'pliklar kvadrat qavslar yordamida ifodalanadi. Agar ko'plik bo'sh bo'lsa, u [] sifatida tasvirlanadi. Ko'plik ko'rinishidagi ichki ma'lumotli xususiyatlar ko'pincha mantiqiy **Boolean** tipida bo'ladi. Birlashgan qiymatlar Ob'ektlar Inspektorida ba'zi kattaliklar to'plami sifatida tasvirlanadi. Ba'zi xususiyatlar masalan, Font o'z qiymatini o'zgartirish uchun muloqot oynalarini chaqira oladi. Buning uchun shu xususiyat maydonining o'ng tomonidagi uch nuqtali kichik tugmachani bosish kifoya.

Delphida komponentlar xususiyatlarini loihalash vaqtida (design time) ham, dastur bajarilash vaqtida (run time) ham o'zgartirish oddiy.



19-расм ички маълумотли хусусиятларнинг
Объектлар Инспекторида кўриниши
а) Бирлашган қийматлар.

б) Кўпликлар

Loihalash vaqtida xususiyatlarni o'zgartirish Formalar Dizayneri (Forms Designer) yoki, sizga ma'lum Ob'ektlar Inspektorining “Properties” sahifasi yordamida bajariladi. Masalan, tugmaning Height (balandligi) va Width (kengligi) xususiyatlarini o'zgartirish uchun, uning formadagi tasvirining bir burchagidan sichqoncha yordamida tortib, kerakli

o'lchamlarga erishish mumkin. Xuddi shu natijaga Object Inspectorning Height va Width xususiyatlarni qiymatlarini o'zgartirib ham erishish mumkin.

Vizual komponentlar xususiyatlarini dastur bajarilish vaqtida boshqarish

Dastur bajarilish vaqtida komponentlar xususiyatlari qiymatining barcha o'zgarishlari bajariladigan ish kodi to'liq Paskal tilida yozilishini talab qiladi. Dastur bajarilish vaqtida Object Inspektordan foydalanish mumkin emas. Lekin, komponentlar xususiyatlariga dasturiy yo'l bilan murojaat qilish ancha qulay. Biror xususiyatning qiymatini o'zgartirish uchun siz quyidagiga o'xshash kod yozishingiz kifoya:

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

Bu satr komponentning kengligi Widthni 35 qiymatga o'rnatadi. Agar bundan oldin komponentning kengligi 35 ga teng bo'lmasa, siz komponent qanday qilib kengligini o'zgartirishini ko'rishingiz mumkin.

Shunday qilib, Object Inspector dastur bajarilish vaqtida qilish mumkin bo'lgan ishni loihalash vaqtida oddiyroq bajarish imkoniyatini beradi xolos. Bundan tashqari ob'ektlarning Object Inspektorda ko'rinmaydigan xususiyatlari ham bo'lishi mumkin, ularni faqat dasturiy yo'l bilan o'zgartirish mumkin.

SHAPEDEM1 dasturi

SHAPEDEM1.DPR dasturi , dastur bajarilish vaqtida foydalanuvchi interfnysini o'zgartirishning bir necha yo'llarini namoyish qiladi. SHAPEDEM1 dasturi atigi formaga o'rnatilgan TShape, o'tkazish yo'lakchalari va bir nechta tugmadan iborat.

SHAPEDEM1.DPR dasturining kodi:

```
program Shapedem;  
  
uses  
Forms,  
Mina in 'MAIN.PAS' {Form1};  
  
begin  
Application.CreateForm(TForm1, Form1);  
Application.Run;  
end.
```

Bizning ScrollBarlar misolimizda ekranning o'rtasidagi tasvirning o'lchamlarini o'zgartirish uchun ishlatiladi. Tasvirning yangi ko'rinishini tanlash uchun o'zida tasvirlar nomini saqllovchi (ComboBox) dan, tasvir yoki forma rangini o'zgartirish uchun standart rang tanlash muloqot oynasi va mos tugmalardan foydalanamiz.

Forma rangini o'zgartirish uchun Dialogs sahifasidan ColorDialog ob'ektini formaga o'rnatish. Shuningdek, formaga oddiy tugma Buttonni tashlab qulaylik uchun uning Caption xususiyatidagi yozuvni "Forma rangi"ga va Name xususiyatidagi yozuvni "FormColor"ga almashtirish. Shu tugmaga sichqonchani ikki marta chiqillatsak, quyidagi ko'rinishga ega bo'lgan dastur matni chiqariladi:

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

```
begin
```

```
end;
```

Endi biz oddiy kodlarni kiritamiz:

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

```
begin
```

```
if ColorDialog1.Execute then
```

```
Form1.Color := ColorDialog1.Color;
```

```
end;
```

Bu kod dastur bajarilish vaqtida "Forma rangi" tugmasi bosilgach, rangni tanlash muloqot oynasini chaqiradi. Agar siz bu muloqot oynasida OK tugmasini bossangiz quyidagi satr bajariladi:

```
Form1.Color:=ColorDialog1.Color;
```

Bu kod forma(Form1)ning rangi (Color)ni ColorDialog1 muloqot oynasida tanlingan rangga o'rnatadi.

Tasvir rangini o'zgartirish uchun ham xuddi shunday ishlar bajariladi. Endi siz formaga boshqa tugma o'rnatasiz va uning nomini "ShapeColor" va sarlavxasini "Tasvir rangi" deb o'rnatasiz. Shu tugmaga sichqon tugmasini ikki marta bosib, quyidagicha kod yozasiz:

```

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

```

Endi ancha murakkabroq bo'lgan ish o'tkazish yo'lakchalari (**ScrollBar**) yordamida tasvir o'lchamlarini o'zgartirish dasturini tuzamiz.

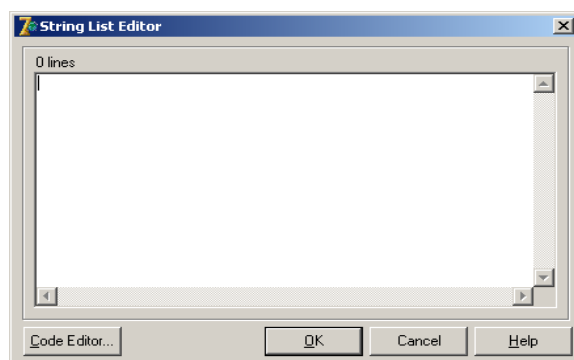
Dastlab, formaga ikkita ScrollBar komponentini tashlaymiz va birinchi ob'ekt uchun Kind xususiyatini **sbHorizontal**ga, ikkinchisi uchun - **sbVertical**ga o'rnatamiz. Ob'ektlar Inspektoridan "Events" sahifasiga o'tamiz va har bir o'tkazish yo'lakchalari uchun OnChange holatiga dastur yozish uchun sichqon tugmasini bosamiz va muharrirda quyidagicha kod kiritamiz:

```

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

```

Endi, formadagi tasvirning ko'rinishini o'zgartirish masalasini ko'rib o'tamiz. Formaga ComboBox ob'ektini tashlaymiz va uning Items xususiyatiga sichqonni chiqillatib, String List Editor muharririni chaqiramiz.



20-rasm. Formadagi ComboBox ob`ektining Items xususiyati yordamida chaqiriluvchi oyna

Bu muharirda quyidagi satrlarni kiritamiz (*stsircle*, *stEllipse*, *stRectangle*, *stRoundRect*, *stRoundSquare*, *stSquare*). So`ngra formada ComboBox1 ob`ektini tanlab, Ob`ektlar Inspektorining “Events” sahifasidagi OnClick xususiyatiga sichqon tugmasini ikki marta bosib, dastur matni muharriri oynasiga o`tamiz. U erda quyidagi yozuv bo`ladi:

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

```
begin
```

```
end;
```

Endi biz o`z kodimizni qo`shib, bajariladigan ishni ko`rsatamiz:

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

```
begin
```

```
Shape1.Shape := TShapeType(ComboBox1.ItemIndex);
```

```
end;
```

Bu satrdagi kod Shape1 komponentining Shape xususiyatini foydalanuvchi ro`yxatdan tanlagan ko`rinishga o`rnatadi. Endi siz dasturni yuklab, uning imkoniyatlaridan foydalanishingiz mumkin.

SHAPEDEM2 dasturi

Endi foydalanuvchi forma o`lchamlarini o`zgartirib o`tirmasligi uchun dasturning o`zi forma o`lchamlarini boshqarishi muammosini hal qilaylik. Buning uchun quyidagilarni bajaramiz:

- Formadagi Tugmalar va ComboBox ob`ektlarini olib tashlaymiz. Ularning o`rniga MainMenu komponentini o`rnatamiz.
- Forma o`lchamlari o`zgarishi bilan o`tkazish yo`lakchalari o`z halatini o`zgartirishiga erishamiz.
- O`tkazish yo`lakchalarining Position xususiyati forma o`lchamlarini to`gri akslantirishiga erishimiz.

SHAPDEM2 dasturi. FormOnResize uslubi qo`shilgan. Bosh modul ko`rinishi.

```

    unit Main;
interface
uses
WinTypes, WinProcs, Classes, Graphics, Forms, Controls, ColorDlg, StdCtrls,
Menus, Dialogs, ExtCtrls;
type
TForm1 = class(TForm)
Shape1: TShape;
ColorDialog1: TSolorDialog;
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.

```

Dastur uchun bosh menyu MainMenu (komponentlar to'plamining "Standard" sahifasida) komponenti yordamida yaratiladi. Uni formaga qo'yib, ikki marta sichqonni chiqillating – shunda kerakli menyu qismlarini kiritish imkoni beriladi. SHAPEDEM2 dasturining menyusini uchta asosiy menyudan ("Rang", "Tasvir" va "Chiqish") iborat qilib yaratamiz.

Birinchi bo'lim uchun quyidagi qismlarni yaratamiz:

- Tasvir rangi
- Forma rangi.

Ikkinchisi uchun:

- To'rtburchak
- Kvadrat
- Aylana To'rtburchak.
- Aylana kvadrat
- Ellips
- Aylana

Menyuning uchinchi bo'limida qism menyular bo'lmaydi.

So'ngra tasvir ko'rinishi bilan bog'liq qismlarga raqamlarni berib chiqamiz. Buning uchun menyuning har bir qismida mavjud bo'lgan Tag xususiyatidan foydalanamiz. Tag (Integer tipida) xususiyatidan foydalanuvchi ixtiyoriy foydaanishlari mumkin. Biz "Turtburchakka" Tag xususiyatining 0 qiymatini, "Kvadrat"ga 1 qiymatini ... beramiz.

Endi dastur matnini keltirib o'tamiz:

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;

Ko'ryapsizki kodlar dastlabki dasturdan umuman o'zgargani yo'q, faqat bu amallar menyu yordamida bajarilyapti.

Xuddi shunday tasvir ko'rinishini o'zgartirish ham oldingi tanlovga o'xshash:

```

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

```

SHAPEDEM2 dasturidagi ikkita o'tkazish yo'lakchasi doimo forma o'lchamlari qanday bo'lishidanqat'iy nazar oyna chegaralari yonida joylashadi.

```

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 uslubida dastur programma Scrollbarlarning yangi o'lchamlarini hisoblaydi:

```

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;

SHAPEDEM2 dasturida siz Min va Max xususiyatlarining qiymatlarini Position qiymatlari oralig'i formaning joriy o'lchamlarini ko'rsatadigan qilib o'rnatishingiz mumkin. Bu erda FormResize dasturidan mos kodlar keltirilgan.

procedure TForm1.FormResize(Sender: TObject);

begin

...

Scrollbar1.Max := Width;

Scrollbar2.Max := Height;

...

end;

Yuqorida ko'rsatilgan ikki qator kod ikkala o'tkazish yo'lakchalari eng katta qiymatlarini formaning bo'yi va eniga tenglashtiradi, bu esa, tasvirni forma bo'ylab yoyishga yordam beradi. Endi sizga Position qiymatini biror songa ko'paytirish shart emas.

procedure TForm1.Scrollbar2Change (Sender: TObject);

begin

Shape1.Height := Scrollbar2.Position;

end;

Siz shundan keyin SHAPDEM2 dasturini yuklasangiz u formaning ixtiyoriy o'lchamlarida ishlashiga guvoh bo'lasiz. Bundan tashqari siz tasvir ko'rinishini va tasvir rangini menyu orqali boshqara olasiz, bu esa jiddiy dasturlarga xos xususiyat hisoblanadi.

Nazorat savollari:

1. Xususiyatlar deganda nimani tushunasiz?
2. Xususiyatlar nesta tipga bo'linadi?
3. Xususiyatlarni boshqarishning qanday usullari mavjud?

Tayanch iboralar:

Birlashgan qiymatlar. Ko'pliklar. Xususiyatlarni dastur bajarilishida boshqarish.
Tasvir rangi. Forma rangi.

7-Ma`ruza. DDE

Reja:

4. DDE asoslari.
5. DDEning qo'llanilishi
6. DDE-serverlar
7. DDE-mijozlar
8. DDE bo'yicha ReportSmithni boshqarish

Adabiyotlar ruyxati:

1. Kompyuterdagi yordam fayllari.
2. Bobrovskiy «Delphi 5», «Piter» Moskva 1997g.
3. Shumakov «Delphi 4 razrabotka baz dannyx», «Piter» Moskva 1996g.
4. Pacheko, Teyksera «Delphi 5 posobie programmista», «Piter» Moskva 1999 g.
5. Faronov «Delphi 4 uchebnoe posobie», «Piter» Moskva 1995 g.

DDE asoslari

DDEML qisqartmasi Dynamic Data Exchange Management Library (dinamik ma'lumotlar almashish kutubxonasi) ma'nosini anglatadi. DDEML bu Dynamic Data Exchange (DDE) deb nomlanuvchi murakkab xabarlar tizimi ustidan nazoratdir. DDE saqlovchi kutubxona Windowsning dastlabki xabarlar tizimi imkoniyatlarini oshirish uchun ishlab chiqilgan.

Dynamic Data Exchange bunday nomlanishiga sabab, u ikki dasturni bajarilish vaqtida o'zaro dinamik holda ma'lumotlar almashishiga imkon beradi. Ikki dastur orasidagi aloqani shunday yaratish mumkinki, birida qilingan o'zgarish ikkinchisida akslanadi. Bundan tashqari DDE yordamida o'z dasturingizdan boshqa dasturlarni masalan, Word for Windows, Report Smith, Excel va boshqalarni boshqarish mumkin.

Quyida Delphi komponentlaridan DDE qurish uchun qanday foydalanish tushuntirilgan.

DDEning qo'llanilishi

DDE bo'yicha boshqa dasturdan ma'lumotlar oluvchi yoki boshqa dasturni DDE buyruqlari yordamida boshqaruvchi dastur DDE-mijozi hisoblanadi. Bu holda ikkinchi dastur DDE-serveri bo'ladi. Bir dasturning o'zi bir vaqtda ham mijoz, ham server bo'lishi mumkin (masalan, MicroSoft Word).

Endi keling Delphi qanday qilib DDE-dasturlarni tuzish mumkinligini ko'rib o'taylik.

DDE-serverlar

Delphida DDE-serverni qurish uchun System sahifasidagi ikki - TDdeServerConv va TDdeServerItem komponentlari xizmat qiladi. Odatda loihada bitta TDdeServerConv ob'ekti va bir yoki bir nechta TDdeServerItem ob'ekti ishlatidi. DDE-serverning servislariga murojaat qilish uchun mijoz bir necha parametrlarni bilishi keark: servis nomi (Service Name) – bu dastur nomi (odatda – EXE kengaytmasiz bajariluvchi fayl nomi, to'liq yuli bilan); Topic Name - Delphida bu nom TDdeServerConv komponenti; Item Name – Delphida bu TDdeServerItemning kerakli bo'limi nomi.

TDdeServerConv ob'ektining vazifalari – DDEni umumiy boshqarish va makrovlarning bajarilishi uchun mijozlardan so'rovlarni qabul qilish. Oxirgi amal OnExecuteMacro holatlar qayta ishlovchisida bajariladi, masalan, bizning holdaquyidagicha:

```
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 ob`ekti TDdeServerConv bilan bog`lanadi va DDE bo`yicha aynan nima yuborilishini aniqlanadi. Buning uchun unda Text va Lines xususiyatlari mavjud. (Text ning qiymati doimo Lines[0] ga teng bo`ladi) bu qiymatlar o`zgarganda server bilan bog`langan hamma mijozlarga yangilangan ma`lumotlar yuboriladi. Bizning dasturimizda Lines xususiyati qiymatining o`zgarishi Edit1 komponentining OnChange holatlari qayta ishlovchisida amalga oshadi:

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

Shu komponentning o`zi mijozlardan ma`lumot olishni a`minlaydi, bizning misolimizda bu holat Poke Data tugmasi bosilganda, OnPokeData holati qayta ishlovchisida bajariladi:

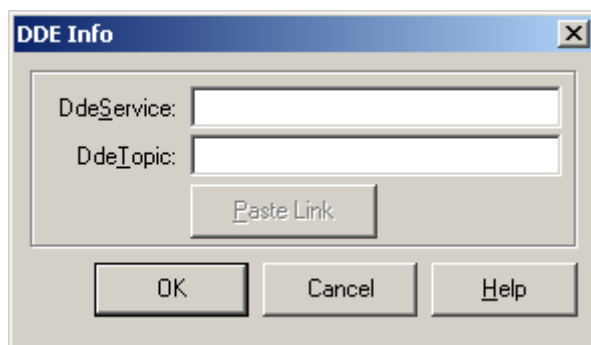
```
procedure TDdeSrvrForm.doOnPoke(Sender: TObject);  
begin  
FInPoke := True;  
Edit1.Lines := DdeTestItem.Lines;  
FInPoke := False;  
end;
```

Shunday qilib biz Delphi komponentlari yordamida qurilgan to`liq ishlovchi DDE-serverni ko`rib o`tdik. Ko`pincha mavjud DDE-serverlar DDE imkoniyatlarini to`liq namoyish qila olmaydi va servis imkoniyatlarning faqat bir qismi bilan chegaralaniladi. Masalan, ReportSmith faqatgina DDE bo`yicha buyruqlar (makroslar)ni bajaradi.

DDE-mijozlar

Delphida DDE-mijozni yaratish uchun ikkita TDDEClientsonv va TDDEClientItem komponentlari ishlatiladi. Serverdagiga o`xshash bu erda ham loihada bitta TDDEClientsonv ob`ekti va bir yoki bir nechta TDDEClientItem ob`ekti ishlatiladi.

TDDEClientsonv ob`ekti server bilan aloqa o`rnatish va umuman DDE-alokasi boshqarish uchun xizmat qiladi. DDE-server bilan aloqa nafaqat Formalar dizayneri rejimida, balki, dastur bajarilish vaqtida ham o`rnatish mumkin. Birinchi usulda komponentning kerakli xususiyatlari qo`lda to`ldirib chiqiladi. Bular DdeService, DdeTopic va ServiceApplicationlardir. Formalar dizayneri rejimida Ob`ektlar Inspektoridagi dastlabki ikki xususiyatga sichqonchani ikki marta chiqillatsak - DDE-aloka muloqot oynasiga murojaat qilishiimiz mumkin.



21-rasm. DDE aloqani o`rnatish oynasi.

Muloqot oynasida DDE Service va DDE Topic lar nomini kiriting. Masalan, DDE-server sifatida DataBase Desktop kelsa, unga birorta Paradox jadvalini yuklash, biror maydonni belgilash va menyudan Edit|Copy bajarish kerak. Shundae so`ng muloqot oynasiga kirib Paste Link tugmasini bosish kerak. Muloqot oynasi maydonlari mos holda to`ldiriladi.

ServiceApplication xususiyati faqat dastur joriy katalogda turmagan vaqtda va DDEService maydonidagi yozuv dastur nomidan farqli bo`lgan hollarda to`ldiriladi. Bu erda dastur nomi to`liq yo`li bilan ko`rsatilishi kerak. Report Smith bilan ishlaganda bu erda masalan: C:\RPTSMITH\RPTSMITH yozuvini ko`rsatish kerak.

Bizning misolimizda aloqa dastur bajarilish vaqtida menyuning File|New Link va Edit|Paste Link buyruqlari yordamida yaratiladi. Menyu File|New Link buyrug`ida DDE bo`yicha aloqa ob`ektning mos uslubi yordamida bog`lanadi TDdeServerConv va OpenLinklarni bajarish kerak emas, chunki ConnectMode xususiyati ddeAutomatic qiymatiga ega:

```
procedure TFormD.doNewLink(Sender: TObject);  
begin
```



```

DdeClient.SetLink(AppName.Text, TopicName.Text);
DdeClientItem.DdeConv := DdeClient;
DdeClientItem.DdeItem := ItemName.Text;
end;

```

bu erda TDdeClientItem ob`ekti xususiyatlari to`ldiriladi.

Menyuning Edit|Past Link buyrug`ida DDE bo`ycha aloqa Clipboarddagi ma`lumotlarni qo`llash yo`li bilan bog`lanadi:

```

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;

```

aloqa bog`langandan keyin DDE bo`yicha keluvchi ma`lumotlarni boshqarish kerak, bu ish TDdeClientItem ob`ektining OnChange holatlar qayta ishlovchisida bajariladi.

```

procedure TFormD.DdeClientItemChange(Sender: TObject);
begin
DdeDat.Lines := DdeClientItem.Lines;
end;

```

Bu ish TDdeClientItem ob`ektining yagona vazifasidir.

TDdeClientsonv ob`ektiga yana ikki vazifa yuklanadi: ma`lumotlarni serverga uzatish va makroslarni bajarish. Buning uchun bu ob`ektning mos uslublari mavjud. Quyida makroslarning serverda bajariishini ko`rib o`tamiz:

```
procedure TFormD.doMacro(Sender: TObject);  
begin  
DdeClient.ExecuteMacroLines(XEdit.Lines, True);  
end;
```

Пересылка данных на server:

```
procedure TFormD.doPoke (Sender: TObject);  
begin  
DdeClient.PokeDataLines(DdeClientItem.DdeItem,XEdit.Lines);  
end;
```

Nazorat savollari:

1. DDE ni izohlang.
2. DDEning qo`llanilishi
3. DDElar orasida aloqa o`rnatish usullari.
4. Dasturlar orasida dinamik bog`lanishning yana qanday usullarini bilasiz?

Tayanch iboralar:

DDE. DDE serverlar. DDE mijozlar. DDE aloqa. Dinamik bog`lanish.

8-ma`ruza. OLE

Reja:

5. OLE asoslari
6. TOLEContainer ob`ekti
7. OLE dasturiga misol
8. OLE ob`ektini ma`lumotlar omborida saqlash

Adabiyotlar ruyxati:

1. Kompyuterdagi yordam fayllari.
2. Bobrovskiy «Delphi 5», «Piter» Moskva 1997g.
3. Shumakov «Delphi 4 razrabotka baz dannyx», «Piter» Moskva 1996g.
4. Pacheko, Teyksera «Delphi 5 posobie programmista», «Piter» Moskva 1999 g.
5. Faronov «Delphi 4 uchebnoe posobie», «Piter» Moskva 1995 g.

OLE asoslari

OLE so'zi Objects Linked and Embedded (Bog'langan va Ichki Ob`ektlar – B va IO ) ma`nosini bildiradi. Dasturlar orasida bo'linadigan ma`lumotlar *OLE ob`ekt* deyiladi. OLE ob`ektlarini saqlovchi dastur, *OLE konteyner (OLE Container)* deyiladi.

Masalan, MicroSoft Word hujjatga grafik ob`ektlarni, audio xamda videokliplarni va boshqa ko'plab obe`ektlarni jamlashi mumkin (bunday hujjatni ba`zan tuzilgan hujjat - compound document deyiladi).

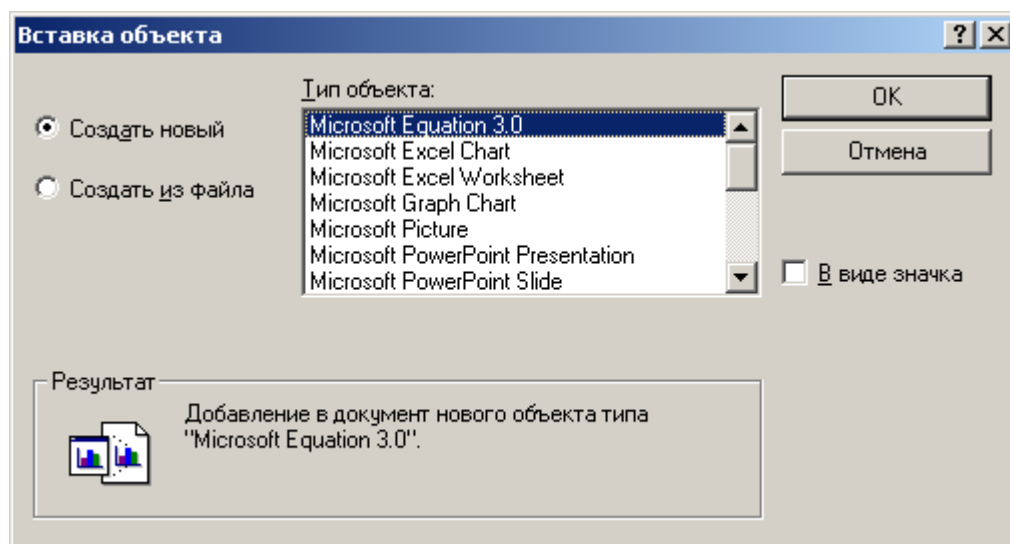
Nomidan ko'rinib turibdiki, OLE ob`ektlarni OLE konteynerga qo'shish yoki unga o'rnatish mumkin. Birinchi holatda ma`lumotlar diskda fayl ko'rinishida saqlanadi va ixtiyoriy dastur ularga murojaat qilishi hamda o'zgartirishi mumkin bo'ladi. Ikkinchi holatda ma`lumotlar OLE konteyneriga o'rnatiladi va bu ma`lumotlarni faqat u ko'rishi va o'zgartirishi mumkin.

Agar DDE yordamida matnlar bilan ishlash mumkin bo'lsa, OLE yordamida esa, dasturlarga ixtiyoriy tipdagi ma`lumotlarni kiritish mumkin. Xuddi DDEda ko'rib o'tganimizdek, mijoz dastur (OLE konteyner) ishlashi uchun OLE server mavjud bo'lsht

talab qilinadi. Mijoz dasturda ishlayotgan foydalanuvchi har safar OLE ob`ektga ko`rish yoki tahrirlash uchun murojaat qilganda server dastur ishga tushiriladi va ma`lumotlar shu erda tahrirlanadi.

TOLEContainer ob`ekti

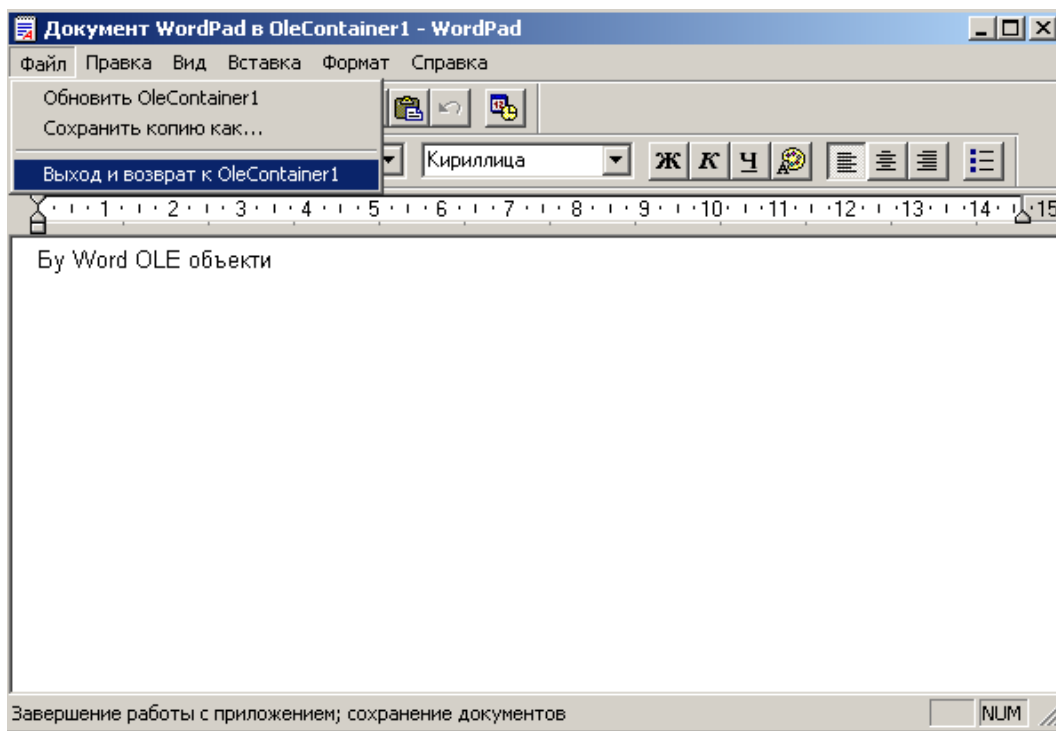
TOLEContainer ob`ekti System sahifasida joylashgan va OLE-konteyner dasturlarini tuzish uchun ishlatiladi. TOLEContainer OLEning hamma ichki murakkabliklarini yashiradi va dasturchi uchun ancha oddiy interfeysni taqdim qiladi. OLE ob`ektni ishlatuvchi oddiy dastur misolini ko`rib o`tamiz. Yangi loiha yarating va formaga TOLEContainer ob`ektini tashlang, Ob`ektlar Inspektoridagi ObjClass yoki ObjDoc xususiyatlariga sichqonni ikki marta bosing – shunda Windowsning standart “Insert Object” muloqot oynasi ko`rinadi.



22-rasm. “Insert Object” muloqot oynasi

Bu muloqot oynasida tizimda mavjud barcha OLE-serverlar ro`yxati keltirilgan. OLE-ob`ekt tipi siz tanlagan server tomonidan aniklanadi. Agar siz yangi ob`ekt yaratsangiz (Create New), u holda, OK tugmasi bosilgach, yangi ob`ekt yaratiladigan OLE-server ishga tushadi. Srver dasturidan chiqilgach, yangi OLE ob`ekt dasturga o`rnatiladi (embedded object). Yangi OLE ob`ektni serverlardan birining formatida saqlanuvchi mavjud fayl yordamida ham yaratish mumkin. Buning uchun Create from File bo`limini tanlash kerak.

Keling misol sifatida dasturimizga Microsoft Word Document ni tanlab yangi ob`ekt yaratamiz.



23-rasm. OLE hujjati kiritilgan dastur

OK tugmasini bossangiz MS Word yuklanadi va u erda ixtiyoriy matnni terish mumkin. Ishni yakunlash maqsadida “File|Close and Return to Form1” buyrug’ini tanlang.

OLE-konteynerga sichqonni bosing - MS Word OLE-ob`ektdagi hujjat bilan paydo bo`ladi va uni tahrirlash mumkin, bu holda hamma o`zgarishlar OLE-ob`ektda saqlanadi.

TOLEContainer dasturga ob`ektning o`z xususiyatlari bilan (ya`ni uni turli darajada kattalashtirish yoki kichraytirish mumkin – Zoom xususiyati) yoki piktogramma, ko`rinishida birga o`rnatadi.

OLE-ob`ekt tanlovi faqat formalar dizayneri rejimida emas, balki dastur bajarilish vaqtida ham bajarilishi mumkin. Ish natijalarini fayllarda saqlash va kerak bo`lganda uni fayldan qayta tiklash mumkin, buning uchun TOLEContainer ikkita SaveToFile va LoadFromFile uslublariga ega.

OLE dasturiga misol

Quyidagi dastur TOLEContainerning barcha asosiy imkoniyatlarini namoyish qiladi va quyidagilar imkonini beradi:

- dastur bajarilish vaqtida yangi OLE konteyner yaratish;

- OLE ob`ektni Windowsning standart "Insert Object" oynasi yordamida, yoki Clipboard yordamida, yoki "olib tashlash" (drag-and-drop) texnologiyasi yordamida o'rnatish;

- OLE ob`ektni faylda saqlash va uni qayta tiklash;

OLE ob`ektlarni drag-and-drop texnologiyasi yordamida o'rnatish ancha qiziqarli. MS Wordni yuklang (uning oynasi va dastur oynasini ekranda yonma-yon o'rnatib), ixtiyoriy matnni tering, uni belgilang va sichqon yordamida tortib dasturning bosh oyeasiga tashlang. Endi dasturda shu matnni saqlovchi yangi ichki OLE konteyner hosil bo'ladi. Bu imkoniyatni dasturiy yo'l bilan bajarish ancha mushkul.

OLE ob`ektini ma'lumotlar omborida saqlash

Ba`zan OLE ob`ektini fayllarda emas, ma'lumotlar omborlarida saqlash zarurati tug'iladi (jadvalning BLOB tipli maydonida). Bu holda OLE ob`ekt albatta o'rnatilgan bo'lishi kerak. Afsuski, Delphining standart to'plamida TDBOLEContainer tipidagi ob`ekt mavjud emas, buning uchun biz OLE ob`ektni SaveToStream va LoadFromStream usullari bilan ma'lumotlar omboriga joylashtiramiz va unga murojaat qilamiz. Masalan:

```
procedure TOLEForm.SaveOLE(Sender: TObject);
```

```
var
```

```
    BlSt : TBlobStream;
```

```
begin
```

```
    With Table1 do BlSt:=TBlobStream.Create(BlobField(FieldByName('OLE')),  
    bmReadWrite);
```

```
    OLEContainer.SaveToStream(BlSt as TStream);
```

```
    BlSt.Free; end;
```

Nazorat savollari:

1. OLE tushunchasiga izoh bering.
2. TOLEContainer komponentidan nima maqsadda foydalaniladi?
3. OLE ob`ektlarga misollar keltiring.
4. OLE ob`ektlarning tiplari qanaqa bo'ladi?

Tayanch iboralar:

OLEContainer. OLE ob`ekti. OLE serverlar. OLE hujjat. SaveToStream.
LoadFromStream

9-Ma`ruza. Ma`lumotlar ombori bilan ishlash asoslari.

Reja:

1. Ma`lumotlar ombori nima?
2. Ma`lumotlar omboriga qo'yiladigan talablar.
3. Delphida ma`lumotlar ombori bilan ishlovchi komponentlar sharhi
4. Ma`lumotlar omboriga murojaat qiluvchi dastur

Adabiyotlar ruyxati:

1. Kompyuterdagi yordam fayllari.
2. Bobrovskiy «Delphi 5», «Piter» Moskva 1997g.
3. Shumakov «Delphi 4 razrabotka baz dannyx», «Piter» Moskva 1996g.
4. Pacheko, Teyksera «Delphi 5 posobie programmista», «Piter» Moskva 1999 g.
5. Faronov «Delphi 4 uchebnoe posobie», «Piter» Moskva 1995 g.

Ma`lumotlar ombori nima?

Bu darsda biz ma`lumotlar ombori bilan ishlash asoslarini ko'rib o'tamiz. Ma`lumotlar omborining maqsadi ma'niga ega bo'lgan ma`lumotlarni bir yoki bir nechta joyda ortiqcha sarflarsiz (xotira va zaxiralarni iqtisod qilib) saqlashdir. Shunday qilib ma`lumotlar ombori ikkita asosiy maqsadlarni ko'zda tutadi: ma`lumotlarni yoo'qolishdan saqlash va ularning ishonchliligini oshirish.

Ma`lumotlar omboriga qo'yiladigan talablar.

Demak, yaxshi loihalangan ma`lumotlar ombori:

- foydalanuvchilarning ma`lumotlar omboriga bo'lgan barcha talablarini qondiradi. Shuning uchun ma`lumotlar omborini loihalashdan oldin foydalanuvchilarning ma`lumotlar omboriga bo'lgan talablarini keng miqyosda o'rganib chiqish zarur.
- Ma`lumotlarning mosligi va ishonchliligini kafolatlaydi. Jadvallarni tuzishda foydalanuvchi tomonidan mumkin bo'lmagan ma`lumotlar kiritilashining oldini olish uchun ma`lum shartlarni kiritish kerak.

Delphida ma'lumotlar ombori bilan ishlovchi komponentlar sharhi

Delphida ma'lumotlar ombori bilan ishlovchi juda ko'p komponentlar bo'lib, biz ulardan asosan TTable, TDbgrids, TDataSource, TDbNavigator komponentlari bilan tanishamiz.



- TTable ob'ekti ma'lumotlar omboridagi mavjud jadval bilan mulokat o'rnatish uchun xizmat qiladi. TTable ixtiyoriy tipdagi (FoxPro, ODBC, SQL ...) ma'lumotlar omborining har bir yozuviga va maydoniga to'g'ridan to'g'ri murojaat qila oladi. Bu komponent shuningdek, alohida hisobotlar bilan ham muloqot o'rnatadi.

TTable ob'ektdan foydalanishdan oldin unga ma'lumotlar ombori alyasini ulash kerak, ya'ni shu komponentning DatabaseName xususiyatida chiqadigan ro'yxatdan kerakli alyasni tanlash va TableName xususiyatidagi ro'yxatdan kerakli jadval nomini tanlash kerak. TTable ob'ektini faollashtirish uchun Active xususiyati qiymatini **true** ga o'tkazish kerak.



TADOtable ob'ekti ham xuddi TTable ob'ekti kabi ma'lumotlar omboridagi biror jadvalga bog'lanish va unga murojaat qilish uchun xizmat qiladi. Bu ob'ektdan asosan MSAccess ma'lumotlar omborini boshqarish tizimida yaratilgan omborlar bilan ishlashga mo'ljallangan. Bu ob'ekt asosan TADOconnection ob'ekti bilan birga qo'llanilib, TADOconnection ma'lumotlar omboriga ulanadi. Shundan so'ng bir yoki bir nechta TAdotable ob'ektlari Connection xususiyati yordamida TADOconnectionga ulanadi va TableName xususiyati yordamida kerakli jadvalga ulanadi. Ob'ektni faollashtirish uchun Active xususiyati qiymatini **true** ga o'tkazish kerak. Bu ob'ekt yordamida ma'lumotlar omboridan ma'lumotlarni biror filtr yordamida ajratib olish mumkin.



TDataSource ob'ekti bevosita TTable yoki TAdoTablega bog'lanib, ma'lumotlar omboridagi yozuvlarni tahrirlash, ularga murojaat qilish imkonini beradi. Buning uchun komponentning DataSet xususiyatidagi ro'yxatdan kerakli Table elementi tanlanadi va shu orqali ikki ob'ekt birbiriga bog'lanadi. Har bir alohida TDataSource ob'ekti bitta ma'lumotlar omboridagi bitta jadvalga ulana oladi.

Yuqoridagi uchala ob`ekt ham dastur bajarilish vaqtida ko`rinmaydigan ob`ekt bo`lib, Formalar Dizayneri ko`rinishida ularni formaga tashlaganda o`lchamlarni o`zgartirib bo`lmaydi. Ularni ma`lumotlar omboriga Formalar Dizayneri rejimida ham, dasturiy yo`l bilan dastur bajarilish vaqtida ham bog`lash mumkin.

Buning uchun quyidagicha kodlar yoziladi:

begin

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

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

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

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

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

end;



TDBGRID ob`ekti ma`lumotlar omboridagi hisobotlar, jadvollar va so`rovlardagi ma`lumotlarni jadval ko`rinishida namoyish etish uchun qo`llanadi. Bu ob`ekt yordamida ma`lumotlar omboridagi yozuvlarni namoyish qilish, tahrirlash va o`zgartirish mumkin. Kiritilgan o`zgartirishlar joriy yozuv ustida boradi va bu o`zgarishlar faqat siz boshqa yozuvga o`tganingizda, yoki dasturni yopganingizda saqlab qolinadi. TDBGRID ob`ekti bevosita Datasource xususiyati yordamida TDataSource ob`ektga bog`lanadi va shu orqali ma`lumotlarni namoyish etadi.



TDBNavigator (QDBCtrls) ob`ekti dasturda TDBGRID yoki TDBEDIT komponentlari orqali ma`lumotlar ombori yozuvlariga murojaat qilinayotgan vaqtda qo`llaniladi. TDBNavigator foydalanuvchiga ma`lumotlar omboridagi yozuvlarni tahrirlash yoki ko`rib chiqishda qo`l keladi. Foydalanuvchi TDBNavigator tugmalardan birini bosganda sha tugma bilan bog`langan amal dasturda bajariladi. Masalan, foydalanuvchi Insert tugmasini bosganda ma`lumotlar jadvalida bitta bo`sh qator hosil bo`ladi.

TDBNavigator tugmalari va ular bajaradigan amallar

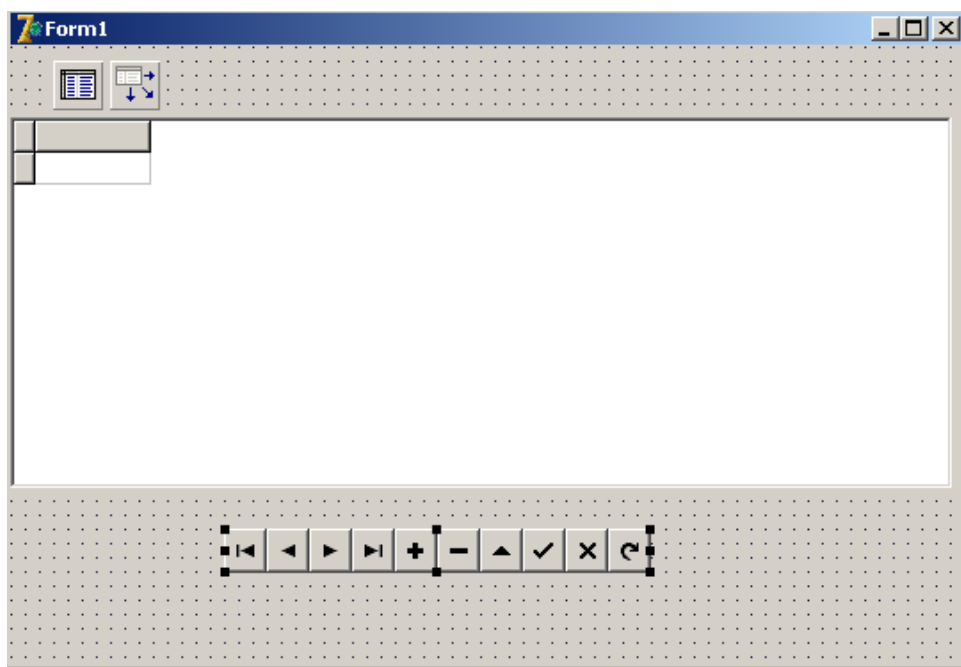
Quyida TDBNavigator tugmalari va ular bajaradigan amallarni ko'rib o'tamiz:

Tugma	Amal
 First	Ma'lumotlar omboridagi dastlabki yozuvni faollashtirish. U faqat joriy yozuv dastlabki yozuv bo'lmagandagina faol bo'ladi.
 Prior	Ma'lumotlar omborida joriy yozuvdan oldingi yozuvni faollashtirish. U faqat joriy yozuv dastlabki yozuv bo'lmagandagina faol bo'ladi.
 Next	Ma'lumotlar omborida joriy yozuvdan keyingi yozuvni faollashtirish. U faqat joriy yozuv oxirgi yozuv bo'lmagandagina faol bo'ladi.
 Last	Ma'lumotlar omboridagi oxirgi yozuvni faollashtirish. U faqat joriy yozuv oxirgi yozuv bo'lmagandagina faol bo'ladi.
 Insert	Jadvalga ma'lumotlarni kiritish uchun yangi satr qo'shish. Bunda satrning ixtiyoriy maydoniga ma'lumot kiritilganda o'zgarishlar saqlanadi
 Delete	Joriy yozuvni o'chirish. Bunda yozuvni o'chirish haqida so'rov chiqariladi va o'chirilgan yozuv qayta tiklanmaydi.
 Edit	Joriy yozuvni o'zgartirish, tahrirlash mumkin bo'lgan holatga o'tkazish.
 Post	Kiritilgan o'zgaishlarni xotirada saqlash. Bunda joriy maydoning oldingi ma'lumotlari o'rniga kiritilgan o'zgarishlar saqlanadi.
 Cancel	Joriy yozuvga kiritilgan o'zgarishlarni bekor qilish. Bu amaldan joriy yozuvni almashtirguncha foydalanish mumkin.

Ma'lumotlar omboriga murojaat qiluvchi dastur

Endi yuqorida aytilganlarni umumlashtirib, Delphida ma'lumotlar ombori bilan ishlovchi dasturga misol ko'rib o'tamiz. Ushbu dasturda biz standart ma'lumotlar ombori bo'lgan DBDemos dan foydalanamiz (Siz o'zingiz tuzgan ma'lumotlar omborini ham alyas sifatida dasturlarga kiritishingiz mumkin).

Demak, yangi loiha hosil qilamiz va Formalar Dizayneri rejimida formaga BDE sahifasidan TTable, DataAcces sahifasidan TDataSours, Data Controls sahifasidan TDBGrid va TDBNavigator komponentlarini tashlaymiz



24-Rasm. Formalar dizayneri ko'rinishi.

Demak hosil bo'lgan tasvir taxminan rasmdagidek bo'lishiga erishamiz. So'ngra, Table1 (TTable) ob'ektinig DatabaseName xususiyatidagi ro'yxatdan DBDemosni tanlaymiz, TableName xususiyatidagi ro'yxatdan Animals.dbf jadvalini tanlab, Activ xususiyati qiymatini Truega o'rnatamiz. Endi DataSours komponentining DataSet xususiyatidan Table1ni tanlaymiz. Oxirgi ishimiz DBGrid1 va DBNavigator1larning DataSours xususiyatidan DataSours1 ni tanlab, dasturni ishga tushiramiz. Buning uchun biz hech qanday dastur yozmadik va bu komponentlarning o'zi ma'lumotlar bilan oddiy amallarni bajarish imkonini beradi.

The screenshot shows a software window titled "Form1" with a standard Windows-style title bar (minimize, maximize, close buttons). Inside the window is a table with the following data:

	NAME	SIZE	WEIGHT	AREA	BMP
▶	Angel Fish	2	2	Computer Aquariums	(TYPEDBINA
	Boa	10	8	South America	(TYPEDBINA
	Critters	30	20	Screen Savers	(TYPEDBINA
	House Cat	10	5	New Orleans	(TYPEDBINA
	Ocelot	40	35	Africa and Asia	(TYPEDBINA
	Parrot	5	5	South America	(TYPEDBINA
	Tetras	2	2	Fish Bowls	(TYPEDBINA

Below the table is a set of navigation buttons: a left arrow, a double left arrow, a right arrow, a double right arrow, a plus sign, a minus sign, an up arrow, a checkmark, an 'X', and a circular arrow (refresh).

25-rasm. Dasturning bajarilash vaqtidagi ko'rinishi.

Nazorat savollari:

1. Ma`lumotlar ombori nima?
2. Ma`lumotlar omboriga qanday talablar qo'yiladi?
3. Delphida ma`lumotlar ombori bilan ishlashga mo'ljallangan komponentlar.
4. TDBNavigator tugmalari va ularning vazifasi.

Tayanch iboralar:

Ma`lumotlar ombori. Jadval. So'rov. Hisobotlar. TTable. TADOTable. TDataSource. TDBGrid. TDBNavigator

Adabiyotlar

1. Kompyuterdagi yordam fayllari.
2. Bobrovskiy «Delphi 5», «Piter» Moskva 1997g.
3. Shumakov «Delphi 4 razrabotka baz dannых», «Piter» Moskva 1996g.
4. Pacheko, Teyksera «Delphi 5 posobie programmista», «Piter» Moskva 1999 g.
5. Faronov «Delphi 4 uchebnoe posobie», «Piter» Moskva 1995 g.

