Дәріс 9

Енгізу-шығару класстар жүйесі. Файлдарды енгізушығаруды ұйымдастыру. Объекттік типтер деректерін енгізу шығару.

Лекция жоспары:

1. Берілгендерді жеке диалогтық терезеде енгізу. InputBox функциясы
2. Стандартты типті- түрлендіру функциялары
3. Берілгендерді жеке диалогтық терезеде шығару. Showmessage процедурасы
4. Берілгендерді формада енгізу және шығару. TEdit, Tlabel, TButton класстары және олардың қасиеттері
5. Файл. Файл типтері
6. Типтік файл операторлары
7. Типсіз файлдар жөнінде

Лекция мазмұны

1. Берілгендерді жеке диалогтық терезеде енгізу. InputBox функциясы

Delphi –де айнымалы мәнін жадқа енгізу үшін Турбо Паскальда пайдалынылатын Read (оқу) операторы жоқ. Программада ол үшін басқа тәсілдердің қолданылуы мүмкін :

1) менщшіктеу;
2) енгізу терезесін пайдалану;
3) форманың Edit компоненті өрісін пайдалану.

Меншіктеу командасының программада жазылуы әдеттегідей , мысалы, x:=4.7;

Енгізу терезесі Delphi-дің стандартты **InputBox** функциясының терезесі. Программада InputBox (енгізу қорабы) функциясын пайдалану командасының жазылу үлгісі:

<айнымалы >:= **InputBox** (_<тақырып >‘,‘<түсініктеме >‘,‘<мән>‘) мұндағы,

айнымалы –мәні енгізу терезесіне енгізілетін жолдық типті айнымалы атауы (InputBox функциясының мәні әр кезде жолдық (**String**) типті. Мән меншіктелетін айнымалы (x) программада x: **String;** түрінде сипатталуы тиіс);

Тақырып-енгізу терезесінің тақырыбы ретінде жазылатын мәтін;

Түсініктеме- енгізу терезесінің ішінде жазылатын түсініктеме мәтін; **Мән**-енгізу терезесі көрінген кезде оның енгізу өрісінде көрінетін мәтін. Әдетте оны бос символдық етіп енгізеді. Мысалы ,x:=4.7 меншіктеу командасын **InputBox** функциясын пайдаланып, мынадай түрде беруге болады:

x:= InputBox (_Аргумент мәні ‘, ‘x:=‘ ‘’)

Команданың орындалу барысында көрінетін терезе 1.6- суретте көрсетілген .Терезенің енгізу өрісіне 4,7 мәнін енгізіп, ОК түймесін шерту керек (не КҚ клавишін бауғада болады).

Ескерту. Windows-тың күйге келтіріліп қойылуына байланысты нақты сан әдеттегідей үтір (,) арқылы бөлініп енгізілуі мүмкін. Ол меншіктелетін айнымалы (x) жолдық типті болғандықтан, қажет болса, оны сандық типті етіп түрлендіру керек.



Сурет 8 - Енгізу терезесі

2. Стандартты типті- түрлендіру функциялары

Функция	**Орындайтын іс -әрекетті**
StrToFloat(x)	Кескіні жолдық типті болатын x нақты санын қалпына келтіру
FloatStrTo (x)	x нақты санын жолдық типті ету
FloatStrToF (x,f,s,o)	x нақты санын форматты жолдық типке түрлендіруүМұндағы f-формат (кескіндеу форматы). Ол көбінесе **ffGeneral** не **ffFidex** түрінде жазылады; s-барлық цифрлар саны (дәлдік); o-ондық нүктеден соң жазылатын цифрлар саны (ондық дәлдік), мысалы, FloattostrF(x,ffFidex,7.3);
StrToInt(x)	Кескіні жолдық типті x бүтін санаын қалпына келтіру ;
IntToStr(x)	X бүтін санын жолдық типті ету (IntToStr-Integer to string, т.с.с.)

Ескерту. **1. FloatStrToF** функциясының форматын **(f) ffExponent** түрінде жазуғада болады. Ол санды экспонеттік етіп дөңгелектейді.

2. Delphi-де Турбо Паскальда қолданылатын **Str(x,s)** және оған кері **Val(s,x,code)** типті түрлендіру процедураларында пайдалану мүмкін.Мысалы, x=25 болса, **Str(x,s)** процедурасының орындалу нәтижесі:s=‘25‘;s=‘54‘ болса Val(s,x, code) процедурасының орындалу нәтижесі :x=54, code=0.

3. Берілгендерді жеке диалогтық терезеде шығару. Showmessage процедурасы

Delphi-де берілгендерді **Showmessage** (мәліметтерді көрсету) процедурасының терезесіне шығаруға болады. Процедураның (оператордың) жазылу түрі.

Showmessage(s);

Мұндағы s-жолдық типті өрнек .Егер ол сандық типті болса оны жолдық типке түрлендіру үшін s аргументінің орнына типті түрлендірудің сәйкес функциясын енгізу керек. Мсыалы, s нақты типті сан болса, оны шығару терезесінде көрсету операторының мынадай түрде жазылуы мүмкін:

Showmessag(FloatToStr(s)); не

Showmessag((FloatToStr(s,ffgeneral,7,3));

Жүйе соңғы операторды оқып шығып, жеті цифрдан тұратын нақты санды символдық типте шығару терезесінде көрсетеді.

4. Берілгендерді формада енгізу және шығару. TEdit, Tlabel, TButton класстары және олардың қасиеттері

Компоненттер понеліне формаға оқиғаларды өңдеуіш процедуралар тудыратын түрлі компоненттер (интерфейс элементтерін) орнатып, программа нәтижесін форма терезесінде көрсету – Delphi-де визуальды прогрммалаудың **ең түйінді негізі.** Әсіресе, ол үшін компоненттер понелінің **Standart** бетінен мәтіндермен жұмыс істейтін **TEdit,** мәтін енгізілетін **TLabel** және командалық **TButton** компоненттері жиі пайдаланады. Мысалы, формаға **Edit1** редакциялау өрісін орнатып, оның **Text** қасиеті арқылы не *іске қосу* командасын берген соң оған курсорды енгізіп, клавиатура арқылы бір жолдық мәтін енгізуге және оны редакциялауға болды. (Оны таңдап, буфер арқылы формада орнатылған екінші **Edit** өрісіне көшіруде мүмкін.)

Формада орнатылған **Label, Edit** компоненттерінің **Caption, Text** қасиеттеріне *праграмма арқылы* қол жеткізу – паскальдағы *жазу* өрісіне қол жеткізу сияқты. Алдымен обьект атауы жазылып, одан соң нүкте қойылады да, оның соңына қасиет атауы енгізіледі. Қасиетке мән меншіктеу тәсілі әдеттегідей. Мысалы:

Label1. Caption:=‘біз студенттерміз‘;

Edit1 өрісіне енгізілген қарапайым мәтінді (S) **Label1** –өрісіне шығару да мүмкін. Ол үшін процедура денесін мынадай етіп алу жеткілікті (Edit1.

Text- Edit1 обьектісінің Text қасиетін программа арқылы орындау әдісі):

```
Var s string;
    Begin
     S= Edit1. Text;
     Label1. Caption :=s;
End;
```

Формада TButton (түйме) компонентін орнатып, оны программаны іске қосу түймесі ретінде пайдалану қыйын емес.

Осы класстардың тағы да бірнеше қасиеттерін қарастырайық. **TLabel:**

- **AutoSize**: Boolean - компонент көлемі ішіне жазылған текстке қатысты өзгеру не өзгермейтіндігін белгілейді,
- **Layout**: TTextLayout; - компонент ішіндегі тексттің вертикаль бойынша реттелу түрін анықтайды.
- **Transparent**: Boolean; - компонент фоны мөлдір болуын не болмауын анықтайды. **TEdit:**
- **BorderStyle**: TSaticBorderStile; - компонент айналасында рамканың болуын (болмауын) анықтайды.
- **SelText:** string; - тексттің ерекшеленген фрагменті (TMemo классы үшін де тиісті).

тәсілдері (басқа да текстпен жұмыс істейтін компоненттерге де тиісті**):** - **Procedure Clear**; virtual – бүкіл текстті жояды.

- **Procedure ClearSelection**; - ерекшеленген текстті жояды.
- **Procedure CopytoClipboard**; - ерекшеленген текстті алмасу буферәне көшіреді.
- **Procedure PasteFromClipboard;**- алмасу буферіне көшірілген текстті курсор тұрған орынға қояды.

5. Файл. Файл типтері

Сыртқы жадтың (дискінің не дискеттің) атау берілген облысын файл не дискілік файл деп,дискіге жазылатын информацияны берілгендер файлы (БФ) деп атайды. БФ- ке мысал ретінде мынадай тізім алуға болады:

1. Шаймерденқызы Гулнар 1975 қазақ
2. Оралханұлы Ержан 1977 қазақ 3.

Нұрғалиқызы Жәмила 1978 қазақ

(тізім кез келген n- жолдан тұруы мүмкін).

Тізімнің әр жолында бір адам жөніндегі мәлімет берілген.

Ол сандық және мәтіндік болатын түрлі типті торт мәннен тұр. Мұндай мәліметті берілгендер файлының элементі деп атайды. Элемент жол не жазу деп те аталады; жол құрайтын мәндер өріс элементтері делінеді. Яғни,тізімнен тұратын берілгендер файлы - жолдар жиынтығы, әр жол- файлдың бір элементі. Берілген тізімде әр жол төрт өріс элементінен, ал файл үш элементтен және оларға сәйкес төрт бағаннан тұр. Өріс элементтері:

Нөмірі, Аты-жөні, Туған жылы, Ұлты (Nomer, Fam, Tjili, Ulti).

Мұндай құрылымға жаңа өрістерді қосу да мүмкін, мысалы, ―Үй адресі, телефон нөмірі‖, т.б.

БФ дискіге әріптен басталатын 1-8 символдан тұруы мүмкін жеке атау арқылы тиелуі тиіс, атаудың проект атынан бөлек болуы міндетті.

Ескерту.Кейде дискілік файл мен БФ- ті де файл деп атай береді.

Оларды сөйлем құрылымына қарап ажырату қиын емес. БФ-пен жүргізілуі мүмкін жұмыстар:

- файлды дискіге тиеп, сақтау (қажет болса, бір баған бойынша сорттап, тиеуге болады);
- оны экранға шығару не принтер арқылы қағазға басу;
- бір не бірнеше белгі бойынша нақты жолын не жолдарын заңды, экпанға шығарып көру.Оны редакциялап, дискіге қайта тиеу;
- файлдың бір не бірнеше жолын алып тастау не файлға жаңа жол қосу, т.б.

Delphi- де берілгендер файлымен тізбекті ену тәсілі бойынша жұмыс істеуге болады.

Файл үш түрде сипатталуы мүмкін: типтік, мәтіндік және типсіз форматта. Сипаттау тәсілдері:

<атау>=File of <тип>; // типтік

<атау>=Text File; // мәтіндік

<атау>=File; //типсіз

Мұндағы <атау> - тип атауы, <тип> - Object Pascal-дың кез келген сәйкес типі. Мысалы, жоғарыда жазылған тізім үшін берілгендер типін мынадай түрлерде сипаттауға болады:

```
1)  type  st= record
          nomer  :  char;
fam:string[25];  tjli:integer;
          ulti: string[6];
    end;
var f1: file of st; v:st;  //типтік  немесе 2)
var f2: Text File;  // мәтіндік  немесе 3)  var
f3:file;  // типсіз
```

Көбінесе файлды типтік, кей жағдайларда мәтіндік етіп сипаттайды.

Файлға жазу, файлдан оқу операторларымен таныстыру алдында мынадай белгілеулер енгізейік:

f- файлдық айнымалы (файлдық типті айнымалы ретінде программада қолданылатын идентификатор (атау)).

bf- берілгендер файлының атауы (атау меншіктелетін айнымалы. Мысалы, ―B2001‖ атаулы берілгендер файлы үшін bf=‘B2001‘. Әдетте файл атауы программа бойынша *Edit* өрісіне енгізіледі: bf:=Edit1.Text, не ол үшін *InputBox* функциясы пайдаланылуы мүмкін).

v – файлдық элемент(өріс элементтерінен тұратын жол).

6. Типтік файл операторлары

AssignFile(f, bf). Ол **bf** файлын **f** айнымалысымен байланыстырады да (assign - тағайындау), одан әрі **f**-пен жұмыс **bf** мәнімен анықталған файл жұмысына баламалы болады. **bf** орнына файлдың толық атын жазу да мүмкін. Компьютер **f**-ті саны белгісіз элементтерден тұратын **bf**-тің атауы ретінде қабылдайды. Ол машинаның элементтер

санын алды-ала біліп алуының қажетсіздігіне байланысты. Оған қоса, бұл тәсіл бір программа бойынша түрлі БФ-тердіөңдеуге мүмкіндік береді.

Rewrite (f) – атауы *AssignFile* операторында анықталған берілгендер файлын дискіге тиеу үшін файлды ашу операторы (rewrite – жаңадан жазу). Ол **f** үшін дискіден арнайы орын ашып, *көрсеткіші (бастиекті*) осы орынның басына орналастырады.

Ескерту. Егер таңдап алынған БФ атауы бұрыннан сыртқы құрылғыда жазулы БФ атауымен бірдей болса, *rewrite(f*) операторын оқыған соң компьютер адымен оны өшіріп тастайды. Сондықтан дискіде жазулы **файлды қорғау** үшін файлға берілетін атауды мұқияттықпен таңдап алу керек.

Write (f, v) – атауы *AssignFile* операторында **f** арқылы белгіленген дискілік файлға БФ-тің кезекті элементін (**v**) тиеу операторы (**v** орнына мәндері файлға жазылатын үтірлер арқылы бөлінген айнымалылар тізімін енгізуге де болады).

CloseFile (f) – файлды жабу (оператор БФ-ті дискіге тиеу не дискіден оқу аяқталған кезде енгізіледі).

Reset (f) – атауы f арқылы белгіленген файлды ашып, бастиекті осы файлдың басына орналастырады (reset – қайта орнату).

Read (f, n: longint) – бастиекті файлдың n+1 нөмірлі жолының басына орналастырады. Ол файлдың (n+1) – жолын оқу үшін қажет.

Дискіде *файл жолдары* 0-ден бастап автоматты түрде ретімен нөмірленеді. Жоғарғы тізімде 1-ден бастап нөмірленген баған *өріс элементі* ретінде қарастырылған.

7. Типсіз файлдар жөнінде

Типсіз файлдар берілгендер байттар тізбегі түрінде тиеледі. Оларды блок бойынша жүтеу, шығару үшін пайдаланылатын операторлар - **BlockRead** және **BlockWrite** (үндемеген кезде блок ұзындығы128 байтқа тең.

Блоктар ретімен тиеліп оқылады. Файлдың кеңейтілуі - **dat**). Типсіз файлды тиеу программасына мысал:

```
Var f: file;
      B: byte;
      K: Integer;
Begin
      Assign File(f, ‘C:\B5.dat‘);
      Rewrite(f,k); b:=$41;
      For k:=1 to 120 do
            BlokWrite(f,b,k);
      CloseFile(f);
```

Программаның осы бөлігінің орындалу нәтижесінде C: нәтижесіне 120 _A‘ символынан тұратын B5.dat файлы жазылып шығады (мұндағы **$41** 16лық санау жүйесіндегі _A‘ символының коды)

Бақылау сұрақтары

1. Енгізу-шығару класстар жүйесі?
2. Файлдарды енгізу-шығаруды ұйымдастыру?
3. Объекттік типтер деректерін енгізу шығару?

Ұсынылатын әдебиеттер

1. Гамма Э. Хелм Р., Джонсон Р., Влиссидес Дж. Приемы объектно-ориентированного проектирование. Паттерны проектирования. – СПб: Питер, 2001.
2. Ишкова Э.А. С++ начала программирования. – М.:Бином, 2001.

Кетков Ю., Кетков А. Практика программирования: Visual Basic, C++ Builder, Delphi.