

## 8 – тақырып

### Бағдарламалаудың объектілі-бейімделу стилі

- объектілі-бейімделген бағдарламалау, оның принциптері
- объектінің бейнеленуі
- мұралану
- инкапсуляция
- полиморфизм
- статикалық әдістер
- виртуалдық әдістер
- динамикалық объектілер
- әдіс түрін таңдау
- объектілердің ішкі көрінісі
- объектілерді экспорттайтын модульдер

ОБП үш принциптен құрылған — инкапсуляция, мұрагерлік и полиморфизм.

Мәліметтерді өндейтін мәліметтерді және алгоритмдарды бір бүтін топқа біріктіру. ОБП-да мәліметтер объектінің *өрісі*, ал алгоритм — объектінің *әдістері*, ал біріктіру — *инкапсуляция* деп аталады. Әдістер өрістермен және сыртқы мәліметтермен жұмыс істейді. Олар объектінің ойын іске асырады. Объектілер мұрагерлік қасиетіне ие болған. Ол арқылы өрістердің объект-ұрпақты және объект-арғы-атаны қолдана аламыз. Әр объекте саны кез келген ұрпақ және текқана бір арғы-атаны болуы мүмкін. Осы қасиет арқылы мұрагерлік иерархия талдарың құраламыз. Ұрпақ кейбір арғы-атанын әдістерін басып озады: әдіс бір атпен әр түрлі объектілерге әр түрлі орындалады. Ол әдістердің *полиморфизмі* деп аталады.

Мұрагерліе келесі түрде сипатталады:

```
Type <ұрпақтың аты>= object(<ата-ана түрінің аты>)
```

```
<объектінің қосылған өрістері>:<өрістердің түрлері>;
```

```
<қосылған және жаңадан анықталған әдістердің декларациясы>;
```

```
end;
```

Нүкте түрі, басқа объектілердің иерархия талдары - шеңбер, сызық, төртбұрыш түрлер арқылы мұрагер механизмын көрейік.

TRec



TLine

TCircle



TPoint

TPoint түрі ата-ана деп саналады, онда барлық графика объектілеріне жалпы қасиеттер келтірілген: координаталар, жану әдісі, сөну, жылжу, сосын олар басқа объектілермен мұраналады TCircle түрі TPoint түрінен ұрпағы ретінде радиус (R) өрісімен толтырлады, TLine түрі — кесіндінің екінші соңынын жылжу өрістерімен. TRect түрі TLine түріне тиісті төртбұрыштың қарама қарсы бұрыштарынын координаталары және олардың жылжыуы. Барлық мұрандалған объектілерде тек қана конструктор және Draw әдісі жабылып қалады.

TCircle түрі келесі түрде жұмыс істейді:

```
Type TCircle =object(TPoint)
```

```
Private
```

```
R:integer;      {радиус}
```

```
Public
```

```
Constructor Create(a,b,c:integer);
```

```
{Объектің өрістерін сипаттайды}
```

```
procedure Draw(color:word); virtual;
```

```
{Color түсімен шеңберді салады}
```

```
end;
```

```
constructor TCircle.Create(a,b,c:integer);
```

```
begin
```

```
inherited Create(a,b);
```

```
R:=c;
```

```
End;
```

```
Procedure TCircle.Draw(Color:word);
```

```
Begin
```

```
SetColor(Color);
```

```
Circle(X,Y,R);
```

```
End;
```

Қызметші сөз **inherited** Create ата-аналық конструкторды шақырады, мұрагерленген X және Y өрістерді сипаттайды.

TLine объекті түр келесі түрде сипатталады:

```
Type TLine=objec(TPoint)
```

```
Private
```

```
ShiftX, ShiftY:integer; {Екінше соңынын жылжыуы}
```

```
Public
```

```
Conctructor Create(X1, Y1,X2,Y2:integer);
```

```
{(X1,Y1),(X2,Y2) – кесіндінін соңдарынын  
координаталары}
```

```
procedure Draw(color:word); virtual;
```

```
{color түсімен кесіндіні сызады}
```

```
end;
```

Жаңа объекті құру жаңа жазуды құруға тиісті.

Type <объектін аты> = object

<объектін өрістері>:<өрістердің түрлері>;

<объектілердің әдістерінің декларациясы>;

end;





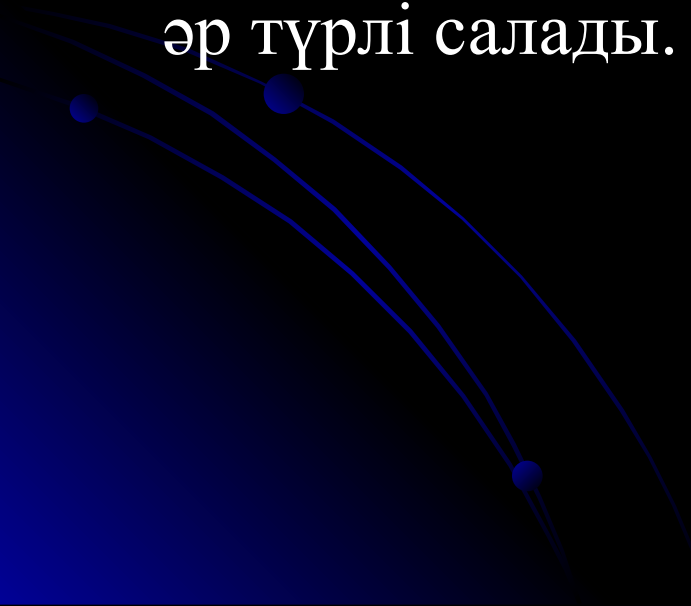
**Create** әдісі объекті құрады және оның өрістерін нақты мәндермен толтырады. Осындай процедураны конструктор деп атайды және **Procedure** сөзінің орнына **Constructor** сөзі қолданылады.

**Private** директивасы объектінің өрістерін және әдістерін программаның өзінің сыртқы процедураларына еркін қолдануға рұқсат береді. Объектінің өрістерінің мәнін әдістер арқылы өзгерту керек қате болмау үшін. Мысалы, нүктенің орнын Move әдісі арқылы ауыстыруға болады.

**Public** директивасы басқа модульдегі процедураларға объектінің элементтерін қолдануға рұқсат береді.

**Protected** ретінде сипатталған мәліметтер тек қана белгіленген объектінің ұрпақтары қолданады.

Әр түрлі объектілердің әдістеріне тең аттарды қолдану. Мысалы, айтылып кеткен объектілердің құру және сызу әдістері тең аттармен анықталған – Create, Draw, бірақ әр графикалық объект оларды әр түрлі салады.



## Қолданылған әдебиеттер тізімі

- 1) Марко Кэнту. Delphi 5 для профессионалов. –СПб.:Питер, 2001.
- 2) Бабушкина И. А., Окулов С.М. Практикум по объектно-ориентированному программированию. М.: БИНОМ, Лаборатория знаний, 2004. – 366 бет.: ил.
- 3) Хомоненко А.Д. и др. Delphi 7. – СПб.: БХВ-Петербург, 2004.- 1216 бет:ил.
- 4) Фаронов В.В. Delphi 5: Учебный курс.-М.: Нолидж, 2001.- 605 бет.: ил.
- 5) Фаронов А.В. TURBO PASCAL /учебник// Изд. «Питер» М.-2001.
- 6) Культин Н. Turbo Pascal в задачах и примерах. - СПб.: БХВ-Петербург, 2001.-256 бет: ил.
- 7) Матаев С. Delphi 7. Бағдарлама құру негіздері: Оқу құралы. Қарағанды, 2005. – 271 б.
- 8) Н.Культин. Основы программирования в Delphi7. – СПб.: БХВ-Петербург, 2003.