

Објектно орјентисано програмирање

Савремен стил програмирања окренут је према подацима и назива се **објектно орјентисано програмирање**. По том моделу, програм се састоји од **објеката** који имају нека могућа стања и понашања. Стања представљају вредности објеката, које временом могу да се мењају. Понашања престављају правила мењања стања, реакцију на утицаје околине и начине утицања на околину. Целокупна обрада остварује се у облику међусобних деловања објеката у програму.

Сви објекти у програму групишу се по својим особинама. Групе објеката са сличним особинама чине **класе**. Објекти исте класе имају иста могућа стања и иста понашања. Објекти дате класе представљају појединачне промерке својих класа.

Појам класе и објекта најлакше се схвата кроз примере из свакодневног живота. Пас (класа) је животиња са свима познатим општим особинама и понашањем. Леси (објект) је један тачно одређени пас с конкретним особинама (боја, тежина, старост) и понашањем које се уклапа у понашање свих паса.

Стања објеката представљају се подацима унутар објеката који се називају **поља**. Поља могу да буду подаци простих и сложених типова. У просте типове спадају цели бројеви, реални пројеви итд. а у сложене типове низови и класе.

Понашање објеката остварује се поступцима унутар класе који се називају **методе**. Методе одговарају функцијама и процедурама у класичним програмским језицима.

Поља и методе класа, заједничким именом се називају **чланови** класа.

Чланови класе могу бити **јавни** или **приватни**. Приватним члановима може да се приступа само из унутрашњости посматране класе. То значи да приватна поља могу да користе или мењају само методе дате класе, приватне методе могу да позивају само друге методе исте класе. Јавним члановима може да се приступа без ограничења, како из унутрашњости класе, тако и из делова програма изван посматране класе. Каже се да ове особине означавају **права приступања** члановима.

У већини случаја поља су приватна, док су од метода неке приватне а неке јавне. Јавне методе су тада једина могућност да се дође до података садржаних у класним објектима. Управо је била и намера приликом увођења класа да се то и постигне.

Класе су прави типови јер:

- Одређују могуће вредности објеката
- Одређују могуће операције над објектима
- Спречавају извршавање било које друге операције над објектима
- Обезбеђују обавезну иницијализацију стварних објеката
- Обезбеђују уништавање објеката када више нију потребни.

Дефинисање и декларисање класа

Дефиниција класе представља навођење свих чланова класе. На основу те дефиниције мора да се зна величина потребног меморијског простора за смештање појединих типова класе. Класа се дефинише описом **class**.

Декларацијом класе се само назначи да неки идентификатор преставља класу, али се ништа не каже о садржају класе. Општи облик декларације класе је:

```
class Ime_klase;
```

Јавни и приватни делови класе разграничавају се ознакама **public** и **private**. Иста ознака сме више пута да се користи. Почетни део класе, који се налази пре прве ознаке **public**, приватан је. Чланови класе наведени у приватним деловима су приватни чланови, а чланови наведени у јавним деловима су јавни чланови.

Члан у дефиницији класе може да буде:

- **Дефиниција поља**
- **Дефиниција методе** која се по форми поклапа с дефиницијом обичних функција.

Објекти

После дефинисања неке класе аутоматски стоје на располагању следеће радње над том класом:

- Дефинисање (стварање) објеката
- Дефинисање показивача (*) и упућивача (&) на објекте
- Додељивање вредности (=) једног објекта другом
- Приступ члановима објекта (.)

За свака објект дате класе ствара се засебан комплет поља те класе.