



Ještě není dokončeno

Principy objektově orientovaného programování

základní koncepce OOP:

Třída

vzor definován uživatelem, který může obsahovat metody i proměnné (např. třída clovek, obsahující proměnné, např:

```
class Clovek{ string jmeno; int vek;
```

a metody, např:

```
dychej(){...}; rekni(string co){...}; }
```

Objekt

instance třídy, vzájemně se liší svými vlastnostmi(atributy) a mají stejné metody jako třída

např .

```
clovek kaja{ jmeno = „Karel Novak“; vek = 55; }
```

a můžeme použít metodu

```
kaja.rekni(„Mam hlad“); kaja.dychej();
```

můžeme si vytvořit více instancí:

```
clovek honza{ jmeno = „Jan Cerny“; vek = 10; }
```

a opět můžeme použít stejné metody...

Základní pilíře OOP

OOP stojí na třech základních pilířích:

Zapouzdření

umožňuje nám skrýt ty metody a atributy, ke kterým nechceme, aby bylo možné přistupovat z vnějšku. Např. u třídy clovek můžeme proměnnou datumNarozeni nechat na skrytou (private) a tím zabráníme, aby mohla být změněna z vnějšku. Tuto vlastnost definujeme pomocí viditelnosti, která může nabývat hodnot public, protected, private. Díky tomu můžou objekty fungovat jako tzv. černé skříňky: můžeme jim dát vstup, přičemž dostaneme nějaký výstup, ale nemusíme se vůbec zajímat o to, jak to vevnitř funguje.

Dědičnost

Usnadňuje vytváření podobných tříd. Z rodičovské třídy si vezme podtřída všechny metody a proměnné (resp. ty, které mají nastavenou viditelnost na public nebo protected) a může si navíc přidat svoje.

např:

```
class Zpevak extends Clovek {(říkáme, že třída pro zpěváky je potomkem člověka) zpivej{}}; tancuj{}};
```

Polymorfismus

Umožňuje použít jednotné rozhraní pro práci s různými typy objektů např: instance třídy mladyClovek bude metodu presunSe(); vykonávat za pomoci metody chůze, ale instance třídy staryClovek bude tu samou metodu presunSe(); vykonávat za pomoci chůze o holi

To znamená, že i když každý objekt tuto metodu vykonává jinak, z vnějšího hlediska se tváří stejně a my nemusíme tedy přemýšlet, jak přesně toho u různých objektů docílit.

From:

<http://wiki.gml.cz/> - **GMLWiki**

Permanent link:

<http://wiki.gml.cz/informatika:maturita:19a?rev=1429972046>

Last update: **25. 04. 2015, 16.27**

