



Programowanie obiektowe

Wykład: klasa, obiekt, po co używać klas, właściwości, atrybuty, funkcje, zachowania, metody, przykładowe obiekty, definiowanie klasy, obiektu, dostęp do składników klasy, public, private, protected, atrybuty publiczne, prywatne, chronione

Pojęcie klasy

Klasa - jest to pojęcie abstrakcyjne, oznaczające w programowaniu obiektowym własny typ danych.

Zestaw taki składa się z *właściwości* (atrybutów) oraz *funkcji* (*zachowań, metod*) danego obiektu. Na podstawie klasy mogą być wytwarzane konkretne, kolejne egzemplarze obiektów.

Dane klasy nazywane są danymi składowymi, natomiast funkcje – funkcjami składowymi albo metodami.

ⓘ *Ważne: klasa to typ obiektu, a nie sam obiekt!*

Klasy postaci w grach typu RPG 😊

Każda klasa ma własny zestaw atrybutów oraz działań (umiejętności). Pojedyncze atrybuty np. siła fizyczna mogą występować we wszystkich klasach, niemniej jednak zawsze łatwo określić o czyją siłę chodzi, jeżeli tylko wskażemy na *obiekt* (czyli na konkretnego *reprezentanta* klasy) – np. na złodzieja o imieniu JanJansen



wojownik



kapłan



złodziej



mag

Po co w ogóle używać klas?

Odpowiedź jest prosta: uzyskujemy lepsze odwzorowanie rzeczywistości w naszym programie. Zamiast zestawu niepowiązanych zmiennych otrzymujemy uporządkowany zbiór obiektów (mających swoje atrybuty i metody), w którym łatwiej się odnaleźć. W przypadku dużych projektów wprowadzenie takiej naturalnej hierarchii ma olbrzymie znaczenie.

Program komputerowy napisany obiektowo traktować możemy jako *zbiór obiektów komunikujących się pomiędzy sobą w celu wykonywania zadań.*

Podjęcie obiektowe ma oczywiście więcej zalet – zdążymy je wszystkie poznać i docenić podczas kilku nadchodzących zajęć 😊

Przykład 1 – Barbarzyńca w grze Diablo



Atrybuty:

- imię
- siła
- zręczność
- odporność
- charyzma
- udźwig

Metody:

- atak toporem
- tańce wojenne
- berserker

Przykład 2 – pralka automatyczna



Atrybuty:

- temperatura prania
- obr/min
- nr programu
- waga
- gabaryty
- marka

Metody:

- pranie
- wirowanie
- płukanie

Uwaga! Przerywam ten program, aby ponownie przypomnieć!

WHY SO
SERIOUS?



Jeszcze raz: klasa to typ obiektu, a nie sam obiekt!

Opis cech i umiejętności barbarzyńcy
to jeszcze nie człowiek z toporem 😊

Projekt pralki to jeszcze nie pralka 😊

Przepis na ciasto to nie pachnąca szarlotka 😊

Klasy - implementacja w C++

DEFINICJA KLASY

```
class Nasz_Typ //definicja klasy o nazwie Nasz_Typ
{
    //ciało klasy - atrybuty i metody
}; //średnik!
```

DEFINICJA OBIEKTÓW (REPREZENTANTÓW) KLASY

```
int zmienna; //definicja zmiennej typu całkowitego
Nasz_Typ obiekt; //definicja obiektu typu Nasz_Typ
Nasz_Typ &referencja = obiekt; //referencja obiektu
Nasz_Typ *wskaznik = &obiekt; //wskaźnik do obiektu
```


Odwołanie się do składnika klasy (1)

```
class Osoba;  
Osoba McLovin;
```

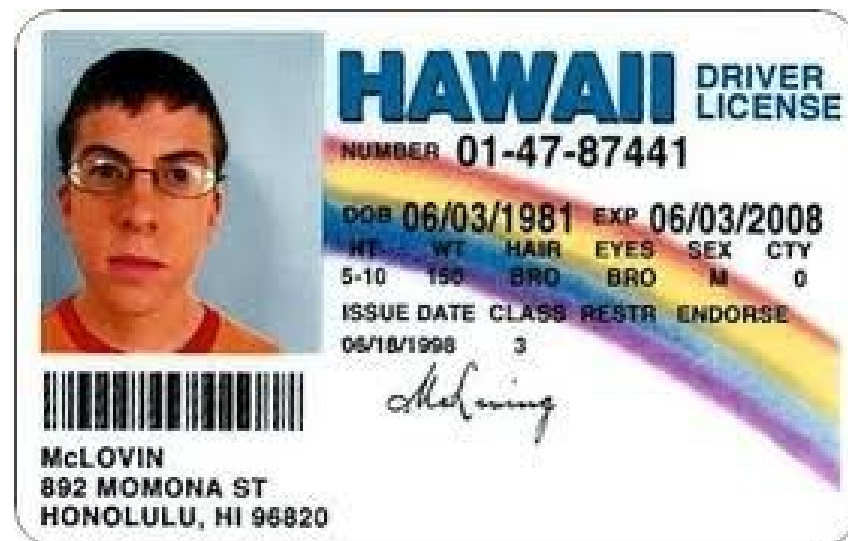
JEŻELI ATRYBUT JEST PUBLICZNY:

```
McLovin.dzien_ur = 6;
```

```
McLovin.miesiac_ur = 3;
```

```
McLovin.rok_ur = 1981;
```

```
McLovin.adres = "Momona ST 892"; //itd...
```



DWUARGUMENTOWY OPERATOR KROPKI DZIAŁA NASTĘPUJĄCO:

```
NAZWA_OBIEKTU.NAZWA_ATRYBUTU_LUB_FUNKCJI_SKŁADOWEJ
```

Dostęp do składników klasy



Czy należysz do rodziny lub przyjaźnisz się z nią? 😊

private – taki składnik jest dostępny tylko dla funkcji składowych danej klasy (oraz dla funkcji zaprzyjaźnionych z tą klasą, o czym później).

protected – składnik jest dostępny tak, jak składnik private, ale dodatkowo jest jeszcze dostępny dla klas wywodzących się (dziedziczących) z danej klasy.

public – składnik jest dostępny bez ograniczeń (jest publiczny). Zwykle są to wybrane funkcje składowe, za pomocą których dokonuje się z zewnątrz operacji na danych prywatnych (czyli interfejs jaki świadomie udostępniamy).

Odwołanie się do składnika klasy (2)

A CO JEŻELI ATRYBUT JEST PRYWATNY I CHCEMY GO ODCZYTAĆ?

Tworzymy w klasie funkcję, która może odczytać wartość składnika prywatnego (bo funkcja ta jest częścią klasy i należy do rodziny)

ZAŁOŻMY ŻE ATRYBUT „ADRES” JEST PRYWATNY:

```
void odczytaj_adres ()  
{  
    cout<<adres;  
}
```

wewnątrz klasy brak kropki!

FUNKCJA ODCZYTUJE PRYWATNY SKŁADNIK:

```
McLovin.odczytaj_adres ();
```

