

# Java 1 • lekce 3

Filip Jirsák

21.2.2023

online

# Proměnné

Proměnná je pojmenované místo v paměti, ve kterém je uložený nějaký údaj nebo soubor údajů.

```
String jmeno;  
int vek;  
LocalDate datum;
```

Hodnota se nastavuje pomocí `=`.

```
jmeno = "Eva";  
vek = 23;  
datum = LocalDate.now();
```

Spolu s vytvořením proměnné se může rovnou nastavit i hodnota. Následující dva zápisy jsou ekvivalentní:

```
String jmeno;  
jmeno = "Eva";
```

```
String jmeno = "Eva";
```

# Proměnné – klíčové slovo `var`

Pokud do proměnné rovnou přiřazujeme hodnotu, můžeme místo typu použít klíčové slovo `var`.

Následující dva zápisy jsou ekvivalentní:

```
String jmeno = "Eva";
```

```
var jmeno = "Eva";
```

# Platnost proměnných (scope)

Proměnná platí v rámci bloku, kde je deklarována.

- Pokud je deklarována na úrovni celé třídy, existuje v rámci celé třídy → členská proměnná, field.
- Pokud je deklarována v metodě, platí od místa deklarace do konce metody, s ukončením metody zanikne → (běžná) proměnná.

# Parametry metod

Parametr metody je speciální proměnná, jejíž hodnota se předává při volání metody.

```
public class Osoba {  
    private LocalDate datum;  
  
    public void nastavDatumNarozeni(LocalDate datumNarozeni) {  
        this.datumNarozeni = datumNarozeni;  
    }  
  
}  
  
// použití  
Osoba uzivatel = new Osoba();  
var datum = LocalDate.of(2001, 2, 21);  
uzivatel.nastavDatumNarozeni(datum);
```

# Objekt

Objekt – data (údaje, hodnoty) spolu s akcemi (činnostmi), které se s daty dají provádět.

- členská proměnná (field) – data, údaje, hodnoty
- metoda (method) – akce, činnost, výkonný kód

```
public class Osoba {  
  
    private String jmeno;  
    private String prijmeni;  
    private LocalDate datumNarozeni;  
  
    public String getCeleJmeno() {  
        return jmeno + " " + prijmeni;  
    }  
  
    public int getVek() {  
        Period period = datumNarozeni.until(LocalDate.now());  
        return period.getYears();  
    }  
}
```

# Třída

Třída – objekty, které mají stejné akce.

- Definuje se klíčovým slovem `class`.
- Jednotlivé objekty, vznikající z dané třídy, se nazývají **instance třídy**.

Různé objekty stejného typu se liší údaji, metody mají stejné.

```
var jenicek = new Osoba("Jan", "Myslivec");  
var marenka = new Osoba("Marie", "Myslivcová");  
  
jenicek.getCeleJmeno();  
marenka.getCeleJmeno();
```

# Konstruktor

Konstruktor je speciální metoda, která slouží pro vytvoření objektu.

- Jmenuje se stejně, jako název třídy.
- Nemá žádný návratový typ
- Konstruktor se volá s klíčovým slovem `new`.

```
public class Osoba {  
  
    private String jmeno;  
    private String prijmeni;  
    private LocalDate datumNarozeni;  
  
    public Osoba(String jmeno, String prijmeni, LocalDate datumNarozeni) {  
        this.jmeno = jmeno;  
        this.prijmeni = prijmeni;  
        this.datumNarozeni = datumNarozeni;  
    }  
  
}  
  
//použití  
var jenicek = new Osoba("Jan", "Myslivec");  
var marenka = new Osoba("Marie", "Myslivcová");
```



# Datové typy

- primitivní typy – pouze hodnota (číslo, znak, logická hodnota ano/ne)
- objektové typy – vše ostatní (včetně `String`)
- speciální hodnota objektových typů: `null`
  - znamená „prázdnó, nic, nezadaná nebo neznámá hodnota“
  - je to něco jiného než `0` nebo `""`

Primitivní typ	Třída	Popis	Rozsah	Zápis
byte	Byte	celé číslo	-128 až 127	zapisuje se jako int
short	Short	celé číslo	-32 768 až 32 767	zapisuje se jako int
int	Integer	celé číslo	cca $\pm 2,1$ miliardy ( $-2^{31}$ až $2^{31} - 1$ )	123, -123
long	Long	celé číslo	cca $\pm 9$ trilionů ( $-2^{63}$ až $2^{63} - 1$ )	123l, 123L, -123L
float	Float	desetinné číslo	$-\infty$ až $+\infty$ (nepřesné!)	15f, 15.0f, -15.23f
double	Double	desetinné číslo	$-\infty$ až $+\infty$ (nepřesné!)	15d, 15.0d, -15.23d
char	Character	jeden znak	65 536 variant	'x', nelze "
boolean	Boolean	logická hodnota	true, false	true, false
<i>neexistuje</i>	String	text, řetězec znaků		"text" "" Víceřádkový text ""