

# Datové typy a aritmetické operace

Lesson 3

Java Developer

Lektor  
Radek Hübner

The logo consists of the lowercase letters 'r\_d' in a white, sans-serif font, positioned inside a dark gray rounded rectangle that has a folded top-right corner.

# LESSON PLAN



- Teorie datových typů
- Celočíselné datové typy
- Číselné typy s plovoucí desetinnou čárkou
- Char a String - Datové typy pro ukládání textu.
- Aritmetické operace

# Co je to ten bit a byte

- bit - "binary digit"
  - Nejmenší jednotka v počítači.
  - Nabývá pouze hodnot 1 a 0.
  - Analogie k žárovce, svítí (1), nesvítí (0)
  - Používá se například k vyjádření rychlosti internetu 10 Mbit/s
  - Jednotka se zapisuje malým písmenem "b"
- Byte - Seskupení 8 bitů
  - Možnost vyjádřit až 256 různých hodnot / stavů
  - Nejčastěji jako čísla v rozsahu:
    - 0-255
    - -128 to 127
  - Používá se například k vyjádření velikosti souboru 10MB
  - Jednotka se zapisuje velkým písmenem "B"

# Datový typ

- Definuje s jak velkými čísly pracujeme
  - Určuje, zda-li budeme počítat i se zapornými čísly.
- Pomáhá počítači rychle a přesně počítat
- Nedovolí nám zaměnit například číslo za text
  - Pomeranče se nevejdou do obalu na vejce.
- Java je silně typovaný objektově orientovaný jazyk.

# Rozdělení datových typů

- primitivní
  - boolean, byte, short, int, long, double, float,
- Neprimitivní, Objektové, ...
  - String, ...

# Celočíselné datové typy

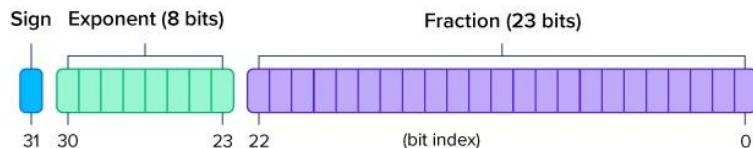
	Velikost v bitech	Rozsah hodnot
boolean	1	0 ... 1, true ... false
byte	8	-128 ... 127
short	16	$-2^{15} \dots 2^{15}-1$ (-32,768...32,767)
int	32	$-2^{31} \dots 2^{31}-1$ (-2,147,483,648...2,147,483,647)
long	64	$2^{63} \dots 2^{63}-1$ -9,223,372,036,854,775,808 ... 9,223,372,036,854,775,807

# Dvojkový doplněk

- Způsob uložení záporných čísel v paměti počítače
- Složitější pro čtení člověkem, jednodušší pro procesor počítače
- Eliminuje kladnou a zápornou nulu.

# Datové typy s pohyblivou řádovou čárkou

- Někdy též nazývány s “plovoucí řádovou čárkou”
- Neukládají číslo přesně
  - $1 + 100,000,000 \neq 100,000,001$
- Malá čísla uložena s větší přesností
- Uložena jako mantisa + exponent.
  - $152853,5047 \Rightarrow 1,528535047 \times 10^5$
- Existuje kladná a záporná nula
- NaN - Not a Number - Speciální hodnota.
- Pozor na operace s penězi !!! Hrozí nepřesnosti.





# Datové typy s pohyblivou řádovou čárkou

	Počet bitů	Počet bitů mantisy	Počet bitů exponentu
float	32	23	8
double	64	52	11

# IEEE 754

- Normal pro reprezentaci čísel pohyblivou řádovou čárkou
- Definuje
  - Datová typy
  - Aritmetické operace
  - Zaokrouhlování !!!
  - Přesnost
  - Reprezentaci čísel v paměti
- Definuje speciální hodnoty
  - NaN - Not a Number
  - Kladná a zaporná nula.
  - INF - Nekonečno

# IEEE 754 -Nepřesnosti

- $0.1 + 0.2 \Rightarrow 0.300000000000000004$
- $1.0000001 - 1.0 \Rightarrow 1.0000000005838672\text{E-}7$
- Nevhodné pro výpočty kde potřebujeme absolutní přesnost
  - Operace s měnou
  - Vědecké výpočty s vysokou přesností
  - Kryptografie
  - Porovnání :  $0.1+0.2 == 0.3$
  - Algoritmy vyžadující deterministické chování

# Datové typy pro práci se znaky a texty

- Char
  - 16 bitů, neznamenkový
  - Reprezentuje jeden znak.
  - Nedoporučuje se používat pro cokoliv jiného než pro reprezentaci znaků
- String
  - Datový typ pro řetězce/texty.

# **/LIVE CODING:**



# Aritmetické operace

- + sčítání
- - odečítání
- \* násobení
- / celočíselné dělení
- / reálné dělení
- % Dělení modulo - zbytek po celočíselném dělení

# Celočíselné vs reálné dělení

- Záleží na datovém typu
- Stačí aby jeden z datových typů byl reálný, výsledek bude také reálný
- Občas je potřeba explicitní přetypování.

# Unární mínus

- Zapisuje se stejně jako v matematice pro reprezentaci záporných čísel.
- Občas je potřeba přidat závorky.

```
int i = -1;  
double d = 10 * (-15);
```



# % Modulo - Zbytek po celočíselném dělení

- Používá se na celých číslech
- Nejčastější použití uvnitř smyček či podmínek
  - $i \% 2 == 0$  - test zdali je číslo sudé

# Speciální unární operátory

- Decrement
  - `i--`
  - Dekrementuje proměnnou o 1
  - Náhrada za `i = i - 1;`
- Increment
  - `i++`
  - Incrementuje hodnotu o 1
  - Náhrada za `i = i + 1;`

# Přiřazovací operátory

- Zjednodušení zápisu
  - $i = i + 10 \rightarrow i += 10$
  - $i = i - 10 \rightarrow i -= 10$
  - $i = i * 10 \rightarrow i *= 10$
  - ...

# Pořadí operací

- Java má definované pořadí vyhodnocování operací
- Jako v matematice, násobení se vypočte před sčítáním.
- Nejsem si jistý, zavorkuji.
  - Zavorky program nezpomalí, v době překladu se odstraní.

# **/LIVE CODING:**



Q&A

???