

## 1. Programování – datové typy a pole

- proměnné jsou základním stavebním kamenem v programování
- datový typ určuje, **co lze** do proměnné **ukládat** a jaké **operace** s ní lze provádět
- v závislosti na **typové kontrole** jazyka může být nutné předem určit typ proměnné (**statická** typová kontrola - **Java, C, C++, C#**) nebo se typ určuje až za běhu (**dynamická** typová kontrola tj. duck typing - **Python, JavaScript, PHP**, pohodlnější, ale náročnější)

### Základní datové typy:

1. **boolean (bool)** - **logická hodnota**, nabývá pouze hodnot true (1) nebo false (0) - tyhle né vždy i v číslech, zabírá 1 bit paměti, používá se v např. ve for a while cyklech, logické operátory AND, NOT, OR, XOR... (nezabírá 1 bit paměti, provádíme mezi dvěma bity, ale výsledek jeden)
2. **integer (int)** - **malé celé číslo**, zpravidla short zabírá 2 bajty, int 4 a long 8 bajtů, lze provádět sčítání, odčítání a násobení, celočíselné dělení (//) a modulo (%), konvertování pomocí int(hodnota)
3. **char** - **znak**, ukládán kódován jako celé číslo podle znakových sad ASCII nebo Unicode, zabírá 21 bitů
4. **float/real/double** - **desetinné číslo**, ukládáno v dvojkové soustavě => může způsobit nepřesnost, konvertování pomocí float(hodnota)

### Složené datové typy: - iterovatelné (můžu procházet po znacích)

1. **String** - **textový řetězec**, zpravidla založený na poli znaků (char)
2. **array (pole)** - pouze **jeden datový typ**, může být **vícerozměrné**, prvky dostupné pomocí **indexu** (0 až délka-1), pecivo = ["chleba", "makovka", "houska", "rohlík"] → pecivo[1] == "makovka", lze také odzadu → pecivo[-1] == "rohlík", počet prvků získáme pomocí metody len(pole) → len(pecivo) == 4, část pole získáme pomocí rozsahu → pecivo[1:2] == ["makovka", "houska"]
3. **tuple (n-tice)** - podobné poli, ale **neměnné**, tj. po inicializaci nelze měnit prvky, inicializace pomocí kulatých () závorek
4. **seznam** - může mít více datových typů, lze pouze procházet od začátku do konce, výhodou oproti poli je snadnější mazání prvků uprostřed
5. **dictionary (slovník)** - podobné seznamu, ale místo indexu se používá **definovaný klíč**, tj. uloženy jsou **dvojice klíč-hodnota**, klíč a hodnota mohou být stejné i jiné datové typy, inicializace pomocí složených {} závorek, klíč a hodnota odděleny dvojtečkou :, JSON
6. **záznam** - může být složen z více datových typů, chová se jako celek, pevně definován, tj. obsahuje konkrétní datové typy

### Další datové typy:

1. **file (soubor)** - přístup k souborům (jak místním, tak síťovým)
2. **complex** - imaginární číslo
3. **pointer (ukazatel)** - ukazuje na místo v paměti
4. **enum** - výčtový typ s programátorem definovanými hodnotami, kterých může nabývat, v Pythonu enumerate
5. **objekt** - objekty jsou programátorem definované datové typy

Zdroje:

<https://cs.wikipedia.org/>

<https://www.w3schools.com/> |

Commented [1]: zanořování, slovníky JSON

<https://stackoverflow.com/>  
<https://docs.oracle.com/>