

Datové typy v programování

Datový typ definuje doménu hodnot, kterých může proměnná nabývat, a množinu operací, které lze s těmito hodnotami provádět. Správná volba datového typu je klíčová pro efektivní správu paměti a stabilitu programu.

1. Základní (primitivní) datové typy

Tyto typy jsou vestavěné přímo v jazyce a odpovídají základnímu způsobu, jakým [binární soustava](#) ukládá data.

Celá čísla (Integers)

Slouží k ukládání čísel bez desetinné čárky. Rozlišují se podle rozsahu (počtu bitů):

- **int / integer:** Standardní celé číslo (obvykle 32 nebo 64 bitů).
- **short / long:** Menší nebo větší rozsah paměti.
- **Signed vs. Unsigned:** Signed typy mohou být záporné, Unsigned pouze nezáporné (nabízejí však dvojnásobný kladný rozsah).

Desetinná čísla (Floating point)

Slouží k ukládání čísel s plovoucí desetinnou čárkou (reálná čísla).

- **float:** Základní přesnost.
- **double:** Dvojitá přesnost (využívá více paměti, ale omezuje zaokrouhlovací chyby).

Logické hodnoty (Booleans)

Přímo vycházejí z [digitální logiky](#).

- Nabývají pouze dvou hodnot: **true** (1) nebo **false** (0).

Znaky (Characters)

Ukládají jeden symbol (písmeno, číslice, interpunkce).

- **char:** Tradičně 1 bajt (ASCII), v moderních jazycích často 2-4 bajty (Unicode/UTF-8).

2. Textové a složené typy

Řetězce (Strings)

Sekvence znaků tvořící slova nebo věty. V mnoha jazycích je string považován za objekt nebo pole znaků.

Pole (Arrays)

Kolekce prvků stejného datového typu uložených v paměti za sebou. Přístup k prvkům probíhá pomocí indexu (obvykle od 0).

3. Typová kontrola (Type System)

Jazyky se liší v tom, jak přísně vyžadují definici datových typů:

Typ systému	Popis	Příklady
Statické typování	Typ proměnné se určuje při psaní kódu a nelze jej měnit.	C++, Java, Rust
Dynamické typování	Typ se určuje až za běhu programu podle obsahu.	Python, JavaScript
Silné typování	Jazyk nedovolí operaci mezi nekompatibilními typy (např. sčítat číslo a text).	Python, Java
Slabé typování	Jazyk se snaží typy automaticky převést, což může vést k chybám.	JavaScript, PHP

4. Typové konverze (Casting)

Někdy je nutné změnit datový typ za běhu:

- Implicitní konverze:** Prohlížeč/kompilátor ji udělá sám (např. přičtení celého čísla k desetinnému).
- Explicitní konverze:** Programátor vynutí změnu (např. převod textu „123“ na číslo 123).

5. Moderní koncepty: Ukazatele a Reference

V nízkourovňových jazycích (C, C++) existují speciální typy:

- Pointer (Ukazatel):** Proměnná, která neobsahuje hodnotu, ale adresu v paměti, kde je hodnota uložena.
- Null / None:** Speciální hodnota značící, že proměnná na nic neodkazuje (častý zdroj chyb typu

`NullPointerException`).

Související články:

- [Binární soustava a bity](#)
- [Přehled programovacích jazyků](#)
- [Datové typy v SQL databázích](#)

Tagy: *programming dev software-engineering data-types coding*

From:

<https://serviceit.cz/> - **IT ENCYKLOPEDIE**

Permanent link:

https://serviceit.cz/doku.php?id=datove_typy

Last update: **2026/01/02 18:42**

