

Datový typ

Datový typ definuje interpretaci binárních dat (nul a jedniček) uložených v paměti. Bez datového typu by počítač nevěděl, zda 8 bitů v paměti představuje číslo, písmeno nebo část obrázku.

1. Základní (primitivní) datové typy

Tyto typy jsou vestavěné přímo v základu většiny programovacích jazyků:

Celočíselné (Integer)

Slouží k ukládání celých čísel bez desetinné čárky.

- **Příklady:** 1, -50, 2025.
- **Varianty:** short, int, long (liší se rozsahem, tedy kolik paměti zabírají).

Desetinná čísla (Floating-point)

Používají se pro reálná čísla s plovoucí desetinnou čárkou.

- **Příklady:** 3.14, -0.001.
- **Varianty:** float (nižší přesnost), double (dvojitá přesnost).

Znaky a řetězce (Character & String)

- **Char:** Jeden jediný znak (písmeno, číslice, symbol). Např. 'A'.
- **String:** Posloupnost znaků (text). Např. „Ahoj světe“.

Logické hodnoty (Boolean)

Může nabývat pouze dvou stavů: **Pravda (True)** nebo **Nepravda (False)**. Je to základ pro veškeré rozhodování v algoritmech.

2. Typová kontrola (Typování)

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

- **Statické typování (C++, Java):** Typ proměnné musíte uvést při psaní kódu (např. `int x = 5;`) a nelze jej později změnit. Je to bezpečnější, chyby se odhalí před spuštěním.
- **Dynamické typování (Python, JavaScript):** Typ se určí automaticky podle vložené hodnoty. Proměnná může během běhu programu změnit typ (např. nejdřív v ní je číslo, pak text). Je to

flexibilnější, ale náchylnější k chybám.

3. Složené datové typy

Umožňují seskupovat základní typy do složitějších struktur:

- **Pole (Array):** Seznam prvků stejného typu (např. seznam 10 čísel).
- **Záznam/Struktura (Struct/Object):** Skupina různých typů (např. objekt „Uživatel“ obsahuje String *jméno* a Int *věk*).

4. Proč je volba typu důležitá?

- **Paměťová efektivita:** Pokud potřebujete uložit věk člověka (0–120), stačí malý datový typ (např. byte). Použití velkého typu (long) by zbytečně plýtvalo pamětí.
- **Správnost výpočtů:** Pokud se pokusíte sečíst text „5“ a číslo 10, v některých jazycích dojde k chybě, v jiných dostanete nesmyslný výsledek „510“.

Zajímavost: Existuje také speciální typ **void** (prázdný). Používá se u funkcí, které provádějí nějakou činnost, ale nevracejí žádnou výslednou hodnotu (např. funkce, která jen vytiskne text na obrazovku).

[Zpět na Proměnné](#)

From:
<https://serviceit.cz/> - IT ENCYKLOPEDIIE

Permanent link:
https://serviceit.cz/doku.php?id=datovy_typ

Last update: **2025/12/31 17:17**

