

# Czym jest zmienna i jak działa?

## Wprowadzenie

W programowaniu **zmienna** to nazwane miejsce w pamięci, które przechowuje dane. Zmienne pozwalają na przechowywanie, modyfikowanie i wykorzystywanie wartości w programie. W języku C# (używanym w Unity) zmienne dzielą się na dwa główne typy: **Value Types** (typy wartościowe) i **Reference Types** (typy referencyjne). W tej dokumentacji skupimy się na **Value Types**.

---

## Czym są zmienne typu wartościowego?

**Value Types** to typy danych, które przechowują bezpośrednio swoją wartość w pamięci. Oznacza to, że gdy przypisujemy wartość jednej zmiennej do drugiej, tworzona jest **kopia** tej wartości. Zmiany w jednej zmiennej nie wpływają na drugą.

---

## Najważniejsze zmienne typu wartościowego w C#

### 1. Typy liczbowe

- Przechowują liczby całkowite lub zmiennoprzecinkowe.
- Przykłady:
  - `int` - liczba całkowita (np. `5`, `-10`, `100`).
  - `float` - liczba zmiennoprzecinkowa (np. `3.14f`, `-0.5f`).
  - `double` - liczba zmiennoprzecinkowa o większej precyzji (np. `3.14`, `-0.5`).
  - `byte` - liczba całkowita od 0 do 255.

**Przykład:**

```
int wiek = 18;
float wzrost = 1.75f;
double pi = 3.14159;
byte liczba = 255;
```

### 2. Typy znakowe

- Przechowują pojedyncze znaki.
- Przykłady:
  - `char` - pojedynczy znak (np. `'A'`, `'1'`, `'$'`).

**Przykład:**

```
char litera = 'A';
char znakSpecjalny = '$';
```

### 3. Typy logiczne

- Przechowują wartości logiczne: `true` (prawda) lub `false` (fałsz).
- Przykłady:

- `bool` - wartość logiczna.

**Przykład:**

```
bool czyGraJestAktywna = true;
bool czyGraczJestWidoczny = false;
```

#### 4. Typy wyliczeniowe

- Przechowują zestaw stałych wartości.
- Przykłady:
  - `enum` - typ wyliczeniowy.

**Przykład:**

```
enum DzieńTygodnia
{
    Poniedziałek,
    Wtorek,
    Środa,
    Czwartek,
    Piątek,
    Sobota,
    Niedziela
}
```

```
DzieńTygodnia dzis = DzieńTygodnia.Piątek;
```

```
...
```

### Jak działają zmienne typu wartościowego?

#### 1. Deklaracja zmiennej:

- Określamy typ zmiennej i nadajemy jej nazwę.
- Przykład:

```
int liczba;
```

#### 2. Inicjalizacja zmiennej:

- Przypisujemy zmiennej wartość.
- Przykład:

```
int liczba;
liczba = 10;
```

#### 3. Deklaracja i inicjalizacja w jednym kroku:

- Przykład:

```
int liczba = 10;
```

#### 4. Kopiowanie wartości:

- W przypadku Value Types, przypisanie jednej zmiennej do drugiej tworzy kopię wartości.

- Przykład:

```
int a = 5;  
int b = a; // b otrzymuje kopię wartości a  
b = 10;    // Zmiana b nie wpływa na a  
Debug.Log(a); // Wynik: 5  
Debug.Log(b); // Wynik: 10
```