

## Funkcja tradycyjna

Definicja funkcji:

```
function kliknij() {  
    alert('Kliknięto!');  
}
```

Wywołanie funkcji:

```
button.addEventListener('click', kliknij);
```



## Funkcja anonimowa

```
button.addEventListener('click', function() {  
    alert('Kliknięto!');  
});
```

## Funkcja strzałkowa / lambda

```
button.addEventListener('click', () => alert('Kliknięto!')); // Może być wielowierszowa
```

x = 5

```
lambda x: x + 1 // Jest jednowierszowa
```

# Zasady tworzenia funkcji

## Funkcja tradycyjna (z nazwą)

Gdy chcemy ją wywołać wiele razy w różnych miejscach kodu; jest dosyć rozbudowana.

## Funkcja anonimowa (bez nazwy)

Gdy chcemy ją wywołać tylko w tym jednym miejscu; może zawierać wiele wierszy z instrukcjami; używana zazwyczaj jako **parametr** w innych funkcjach / metodach (w jej miejsce zwracana jest jakaś konkretna wartość); występuje prawie we wszystkich nowoczesnych językach programowania.

## Funkcja strzałkowa

Gdy chcemy jeszcze bardziej skrócić zapis (funkcja nadal może zawierać wiele wierszy z instrukcjami, choć zazwyczaj wykonuje jedną instrukcję). W językach *Java/Kotlin* występuje jako „->”, zaś w *JavaScript/C#* jako „=>”. *PHP* nie posiada funkcji strzałkowych, zaś w języku *C++* występuje w postaci: `[](int x){ return x + 1; }`. Koncepcja strzałki pojawiła się w 2007 roku.

Ogólny schemat: `parametryFunkcji => wynikJejDziałania.`

## Funkcja lambda

Funkcja charakterystyczna dla *Python/Scheme*, pochodzi z matematyki (Alonzo Church, lata '30 XX wieku, np.  $\lambda x . x + 1$ ).