

# Technologie Informacyjne

## JavaScript - wprowadzenie

**Katedra Technologii Informacyjnych**  
Uniwersytet Ekonomiczny w Poznaniu  
[www.kti.ue.poznan.pl](http://www.kti.ue.poznan.pl)

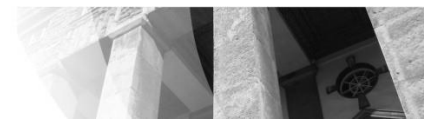


# Czym jest JavaScript?

---

JavaScript– skryptowy język programowania umożliwiający tworzenie interaktywnych stron internetowych

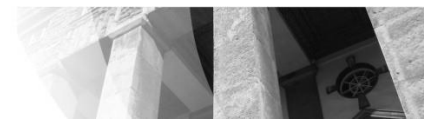
- **Java** – obiektowy język programowania ogólnego przeznaczenia (ale JS nie ma z nim wiele wspólnego)
- **Script** – języki skryptowe są językami wewnętrznymi aplikacji i w nich są wykonywane



# Do czego służy?

---

- Zwiększenie interakcyjności – reagowanie na zdarzenia
- Sprawdzanie poprawności formularzy
- Tworzenie prostych aplikacji wbudowanych na stronach (np. kalkulator, proste gry)
- Dynamiczne strony www (AJAX) – eliminacja konieczności przeładowywania stron



# Umieszczanie skryptów

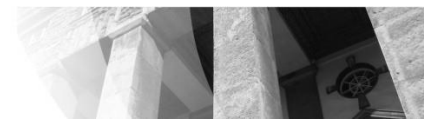
---

## 1. W kodzie HTML

```
<script>  
    alert("Dzień dobry");  
</script>
```

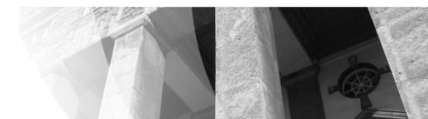
## 2. W pliku zewnętrznym

```
<script src="code.js"></script>
```



# Zmienne (1)

- Zmienne przechowują wartości w programie, np.:
  - Liczby (wynik obliczeń)
  - Teksty (fragmenty wyświetlanego tekstu)
  - Daty
- W JS są one dynamicznie typowane
  - Nie musimy określać sami ich typu
- Zalecane jest ich deklarowanie ze słowem `var`



# Zmienne (2)

---

- **Deklaracja** - wyrażenie chęci użycia zmiennej

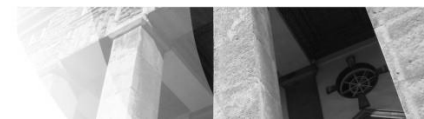
```
var a;
```

- **Inicjalizacja** – nadanie początkowej wartości

```
var b = "text";
```

```
var c = 12;
```

```
var d = new Date();
```



# Wykorzystanie zmiennych

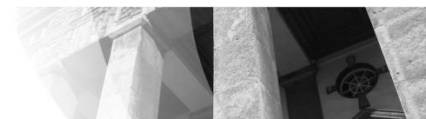
---

```
var pensja = 1000;
```

```
var podatek = 0.19;
```

```
var kwotaPodatku;
```

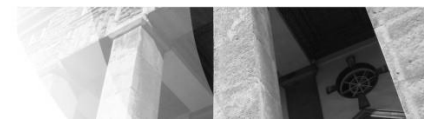
```
kwotaPodatku = pensja * podatek;
```



# Podstawowe typy danych

---

- **Boolean** – typ logiczny (true, false)
- **Number** – liczby (rzeczywiste)
- **String** – łańcuchy znaków (teksty)
- **Undefined** (niezdefiniowany) – wartość zmiennej niezadeklarowanej
- **Null** (wartość pusta) – wartość zmiennej niezainicjalizowanej
- **Object** – typ obiektowy

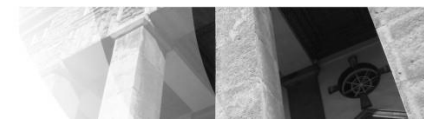




# Funkcje (1)

---

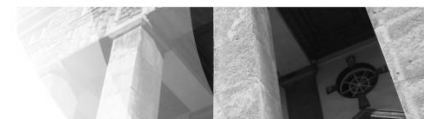
- Funkcja służy do wyodrębnienia fragmentu skryptu i nadania mu określonej nazwy
- Cel - wielokrotne wykorzystanie tego samego fragmentu i uporządkowanie skryptu
- Funkcja może posiadać parametry i zwracać wynik (ale nie musi)



# Funkcje (2)

---

- Funkcję wyróżniają: słowo `function` i nawiasy `{ }`
- Aby zwrócić wynik, wykorzystujemy słowo `return`



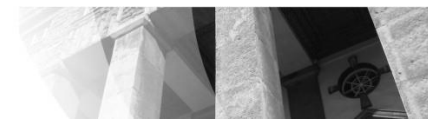
# Funkcje (3)

- **Definicja** – określenie nazwy i ciała funkcji

```
function mnoz (x, y) {  
    var wynik = x * y;  
    return wynik;  
}
```

- **Wywołanie** – użycie funkcji w skrypcie

```
var iloczyn = mnoz (22, 16.7894) ;
```



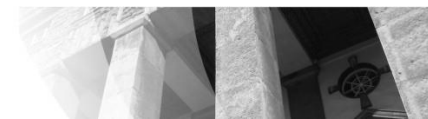
# Zasięg zmiennych

- **Zm. globalne** – widoczne w całym skrypcie

```
var a = 10;
```

- **Zm. lokalne** – widoczne tylko w bieżącym fragmencie (ograniczonym przez { i })

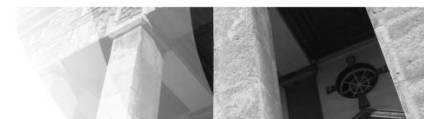
```
function ab() {  
    var b = 9;  
    a = a * b;  
}
```



# Typ obiektowy (1)

---

- Kolekcja **pól** (zmiennych) i **metod** (funkcji)
- Obiekty **standardowe** (wbudowane)
- Obiekty **własne**



# Typ obiektowy (2)

---

- **Tworzenie** (obiekt standardowy):

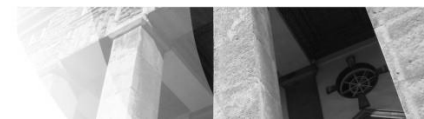
```
var d = new Date();
```

- Dostęp do **pola**:

```
var rok = d.year;
```

- Dostęp do **metody**:

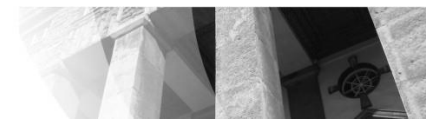
```
var czas = d.now();
```



# Obiekty przeglądarki internetowej

- Dostępne bezpośrednio jako zmienne - nie trzeba ich tworzyć; przykłady:
- **document** – cały dokument HTML
- **window** – okno przeglądarki
- **location** – adres sieciowy bieżącej strony
- Przykładowe użycie:

```
document.write("Napiszmy coś!");
```



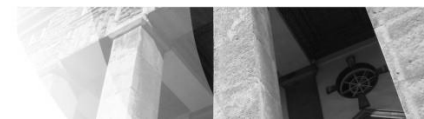
# Komentarze

- Pełnią rolę **objaśniającą**
- Nie są interpretowane przez przeglądarkę
- **Jednowierszowe** – jedna linia

```
var a; // do czego służy a?
```

- **Wielowierszowe** – jedna lub kilka linii

```
/* poniższa funkcja ma na celu  
sprawdzenie formularza */  
function validateForm();
```

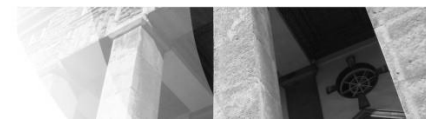




# Instrukcje sterujące

---

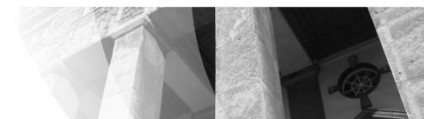
- Instrukcje warunkowe – przebieg skryptu zależy od spełnienia określonych warunków
- Pętle – wykonują dany fragment wielokrotnie



# Instrukcja if

---

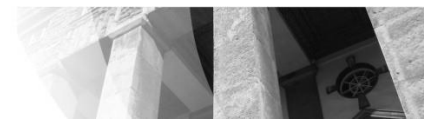
```
if (x != 0) { // warunek
    w = y / x;
} else { // opcjonalne
    w = 0;
}
```



# Pętla while

---

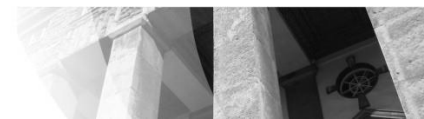
```
x = 10;  
while (x > 0) { // warunek  
    x = x - 1;  
}
```



# Pętla do ... while

---

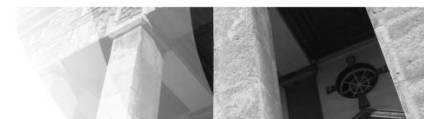
```
x = 10;  
do {  
    x = x - 1;  
} while (x > 0); // warunek
```



# Pętla for

---

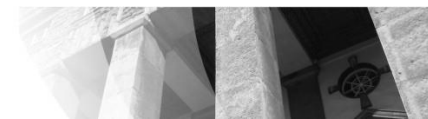
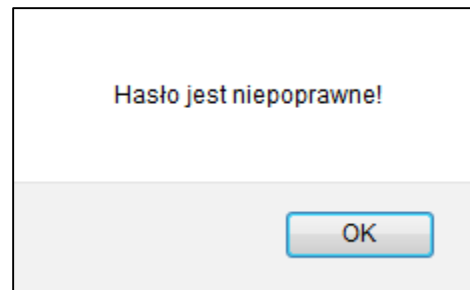
```
x = 10;  
for (var i = 0; i < 10; i++) {  
    x = x - 1;  
}
```



# Wyskakujące okienka (1)

- **Informacyjne** – komunikat i przycisk OK

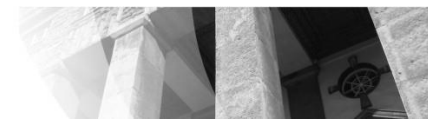
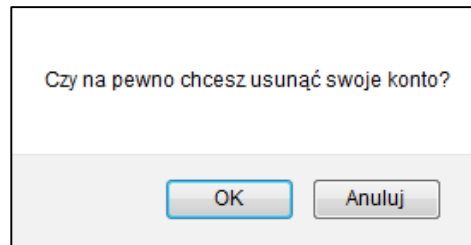
```
alert("Hasło jest niepoprawne!");
```



# Wyskakujące okienka (2)

- **Potwierdzające** – pytanie oraz przyciski OK i Anuluj

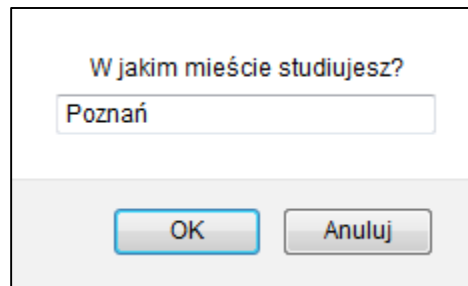
```
confirm("Czy na pewno chcesz  
usunąć swoje konto?");
```



# Wyskakujące okienka (3)

- **Pytające** – pytanie, pole tekstowe oraz przyciski OK i Anuluj

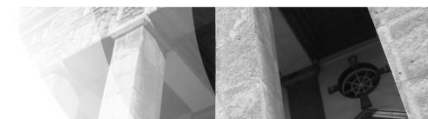
prompt ("W jakim mieście studiujesz?") ;



W jakim mieście studiujesz?

Poznań

OK Anuluj





**Dziękuję za uwagę**

**Katedra Technologii Informacyjnych**  
Uniwersytet Ekonomiczny w Poznaniu  
[www.kti.ue.poznan.pl](http://www.kti.ue.poznan.pl)

