

Przykłady do wykonania – Stałe, zmienne i operatory

Przedmiot: Programowanie aplikacji internetowych

Klasa: 1 Technik Informatyk

Forma: Samodzielna praca – wykonaj wszystkie przykłady

Instrukcja

Wykonaj poniższe zadania w edytorze kodu (np. VS Code) lub w konsoli przeglądarki. Każdy przykład zapisz w osobnym pliku lub wykonaj po kolei. Sprawdź wyniki w konsoli używając `document.write()` lub `console.log()`.

PRZYKŁAD 1: Tworzenie zmiennych

Zadanie:

Utwórz zmienne przechowujące różne typy danych:

- Zmienną 'wiek' z wartością 18
- Stałą 'imie' z wartością 'Anna'
- Zmienną 'czyPienoletni' z wartością true
- Wyświetl wszystkie zmienne w konsoli

Kod do napisania:

```
// Utwórz zmienne tutaj
```

```
// Wyświetl je w konsoli
```

Sprawdź wynik:

W konsoli powinny pojawić się: 18, 'Anna', true

PRZYKŁAD 2: Podstawowe operacje arytmetyczne

Zadanie:

Utwórz dwie zmienne z liczbami, a następnie wykonaj wszystkie podstawowe operacje:

- Dodawanie (+)
- Odejmowanie (-)
- Mnożenie (*)
- Dzielenie (/)
- Modulo (%)

Kod początkowy:

```
let a = 15;
```

```
let b = 4;
```

```
// Tutaj wykonaj operacje i wyświetl wyniki
```

Oczekiwane wyniki:

- $a + b = 19$
- $a - b = 11$
- $a * b = 60$
- $a / b = 3.75$
- $a \% b = 3$

PRZYKŁAD 3: Operatory przypisania

Zadanie:

Użyj operatorów przypisania, aby zmienić wartość zmiennej 'liczba' (zaczynając od 20):

- Dodaj 5 (użyj `+=`)
- Odejmij 3 (użyj `-=`)
- Pomnóż przez 2 (użyj `*=`)
- Podziel przez 4 (użyj `/=`)
- Wyświetl końcową wartość

Kod początkowy:

```
let liczba = 20;
```

```
// Tutaj użyj operatorów przypisania
```

```
// Wyświetl końcową wartość
```

Oczekiwany wynik:

Końcowa wartość powinna być 11

PRZYKŁAD 4: Inkrementacja i dekrementacja

Zadanie:

Utwórz zmienną 'licznik' z wartością 5. Użyj ++ i --, aby zmienić wartość:

- Zwiększ o 1 (użyj ++)
- Wyświetl wartość
- Zwiększ o 1 jeszcze raz
- Wyświetl wartość
- Zmniejsz o 1 (użyj --)
- Wyświetl wartość

Kod początkowy:

```
let licznik = 5;
```

```
// Tutaj użyj ++ i --
```

Sprawdź różnicę między prefix i postfix:

```
let x = 5;
```

```
let y = 5;
```

```
console.log(++x); // Co się stanie?
```

```
console.log(y++); // Co się stanie?
```

```
console.log(y); // Jaka jest wartość y teraz?
```

PRZYKŁAD 5: Łączenie tekstów

Zadanie:

Utwórz zmienne z imieniem i nazwiskiem, a następnie połącz je na 3 sposoby:

- Używając operatora +
- Używając template literals (backticków)
- Z dodatkowym tekstem: 'Witaj, [imie] [nazwisko]!'

Kod początkowy:

```
let imie = 'Jan';
```

```
let nazwisko = 'Kowalski';
```

```
// Sposób 1: operator +
```

```
let pelnelmie1 = /* wypełnij */;
```

```
// Sposób 2: template literals
```

```
let pelnelmie2 = /* wypełnij */;
```

```
// Sposób 3: z dodatkowym tekstem
```

```
let powitanie = /* wypełnij */;
```

```
console.log(pelnelmie1);
```

```
console.log(pelnelmie2);
```

```
console.log(powitanie);
```

PRZYKŁAD 6: Kolejność wykonywania operacji

Zadanie:

Bez użycia nawiasów i z użyciem nawiasów oblicz wyrażenia:

- $2 + 3 * 4$
- $(2 + 3) * 4$
- $10 / 2 + 3$
- $10 / (2 + 3)$
- $2 ** 3 + 1$
- $(2 ** 3) + 1$

Kod do napisania:

```
// Oblicz każdy przykład i wyświetl wynik
```

```
console.log(2 + 3 * 4);
```

```
console.log((2 + 3) * 4);
```

```
// Dodaj pozostałe...
```

Zapisz swoje przewidywania przed uruchomieniem, a potem sprawdź wyniki!

PRZYKŁAD 7: Sprawdzanie typu danych

Zadanie:

Utwórz różne zmienne i sprawdź ich typy używając typeof:

- Liczba całkowita
- Liczba dziesiętna
- Tekst
- Wartość logiczna (true/false)
- Zmienna bez wartości (undefined)
- Zmienna z wartością null

Kod początkowy:

```
let liczbaCalkowita = 42;  
let liczbaDziesietna = 3.14;  
let tekst = 'Hello';  
let czyPrawda = true;  
let niezdefiniowane;  
let puste = null;
```

```
// Sprawdź typ każdej zmiennej używając typeof  
console.log(typeof liczbaCalkowita);  
// Dodaj pozostałe...
```

PRZYKŁAD 8: Kalkulator prosty

Zadanie:

Utwórz prosty kalkulator, który:

- Przechowuje dwie liczby w zmiennych
- Oblicza ich sumę, różnicę, iloczyn i iloraz
- Wyświetla wszystkie wyniki w czytelny sposób

Kod początkowy:

```
let liczba1 = 25;  
let liczba2 = 5;
```

```
// Oblicz i wyświetl wszystkie operacje  
console.log(`Suma: ${liczba1 + liczba2}`);  
// Dodaj pozostałe operacje...
```

Sprawdź dla różnych wartości liczby1 i liczby2

PRZYKŁAD 9: Operator modulo w praktyce

Zadanie:

Użyj operatora modulo (%), aby sprawdzić:

- Czy liczba jest parzysta ($\text{liczba \% 2} === 0$)
- Resztę z dzielenia przez różne liczby
- Czy liczba dzieli się bez reszty przez 3, 5, 10

Kod początkowy:

```
let liczba = 17;
```

```
// Sprawdź parzystość  
console.log(`Czy ${liczba} jest parzysta? ${liczba \% 2 === 0}`);
```

```
// Sprawdź resztę z dzielenia przez 3, 5, 10  
// Dodaj kod tutaj...
```

Sprawdź dla różnych wartości: 10, 15, 20, 23

PRZYKŁAD 10: Kompleksowy przykład – Kalkulator BMI

Zadanie:

Utwórz kalkulator BMI (Body Mass Index):

- $\text{BMI} = \text{masa (kg)} / (\text{wzrost (m)})^2$
- Użyj const dla wzrostu (np. 1.75 m)

- Użyj let dla masy (np. 70 kg)
- Oblicz BMI
- Wyświetl wynik z dokładnością do 2 miejsc po przecinku
- Użyj template literals do wyświetlenia wyniku

Kod początkowy:

```
// Stałe  
const wzrost = 1.75; // metry  
  
// Zmienne  
let masa = 70; // kilogramy  
  
// Oblicz BMI  
// BMI = masa / (wzrost * wzrost)  
// Lub: BMI = masa / wzrost ** 2
```

// Wyświetl wynik

Sprawdź dla różnych wartości masy i wzrostu

Podsumowanie

Po wykonaniu wszystkich przykładów powinieneś umieć:

- Tworzyć zmienne używając let i const
- Rozróżnić różne typy danych
- Stosować operatory arytmetyczne
- Używać operatorów przypisania (+=, -=, *=, /=, %=)
- Stosować inkrementację i dekrementację
- Łączyć teksty używając + i template literals
- Rozumieć kolejność wykonywania operacji
- Sprawdzać typy danych używając typeof

Sprawdź się:

Spróbuj wymyślić własny przykład wykorzystujący zmienne i operatory. Może to być kalkulator ocen, kalkulator czasu, czy coś związanego z Twoim hobby!