

# JavaScript

## 1. Pobieranie elementu DOM z HTML:

```
<input type="text" id="object"> // obiekt gdzieś w kodzie HTML
// skrypt JS
// do zmienna przypisujemy wartość pola o danym ID
var zmienna = document.getElementById("object").value
```

## 2. Wpisywanie elementu na stronę za pomocą innerHTML:

```
<div id="do_wpisania"></div> // element gdzieś w kodzie HTML
// skrypt JS
// do podanego obiektu zostanie wpisana wartość „zawartość”
// jeśli w obiekcie coś się znajdowało – zostanie to nadpisane
document.getElementById("do_wpisania").innerHTML = "zawartość"
```

## 3. Funkcje, aktywowanie funkcji JS obiektami *button*:

```
// po wciśnięciu zostanie wykonana funkcja foo()
<button onclick="foo()">Tekst na przycisku</button>
// deklaracja funkcji o nazwie foo
function foo(){...}
```

## 4. Pętle *while* oraz *for*:

```
// for(start; stop; o_ile_zwiększa) {kod} np.:
for(i=0; i<=10; i++){...}

// while(warunek) {kod} np.:
var i = 0
while(i<10){...}
// UWAGA! Pętla while nie inkrementuje iteratora automatycznie!
```

## 5. Warunki:

```
// if(warunek) {kod} np.:
if(9==10){...} // false
else if (11<10) {...} // false
else {...} // dopiero to się wykona
```

## 7. Modyfikacja stylów CSS poprzez JS:

```
// jak poniżej, gdzie:
// property to właściwość którą chcemy zmienić (np.: color)
// value to wartość na którą ma być zmieniony styl (np.: red)
document.getElementById("zmiana").style.property = "value"
// w kodzie HTML musi istnieć obiekt o ID="zmiana,,
```

## 8. Stringi, tablice i inne cuda:

```
var string = "dziki/kuny/jenoty/swinie";
var tab = string.split("/");
// rozłoży napis dzieląc go według separatora „/”
// powstanie tablica: tab["dziki", "kuny", "jenoty", "swinie"]
// iteracja po tablicy i wypisanie elementów w konsoli:
for(i=0; i<tab.length; i++){ console.log(tab[i]) }
// tablica.length – zwraca długość tablicy (int)
```

## 9. Obsługa checkboxów:

```
checkboxObject.checked = true | false
// checkboxObject - nasz checkbox (może być w tablicy)
// .checked - metoda zwracająca true lub false zależnie od
// zaznaczenia checkboxa
```

## 10. Data:

```
var data = new Date(); // zwraca pełną datę
// można użyć metody .toString() aby zrobić z daty typ string
// a potem przeiterować po jego elementach (pkt 8)
```

## 11. Output w JS:

```
document.getElementById("id").innerHTML = "tekst"; // patrz pkt2
console.log("tekst"); // „tekst” pojawi się w konsoli (F12)
alert("tekst"); // „tekst” pojawi się w specjalnym okienku
```

# PHP

## 1. Obsługa formularza:

```
// kod HTML:
<form method="POST">
    <input type="text" name="one">
    <select name="two">
        <option value="opt_o">Opcja 1</option>
        <option value="opt_t">Opcja 2</option>
    </select>
    <input type="button" value="Klik">
</form>
// kod PHP:
if (count($_POST)){ // sprawdzenie czy form jest wypełniony
    $one = $_POST["one"];
    $two = $_POST["two"];
    echo($one . " " . $two);
}
```

## 2. Obsługa wpisywania danych do bazy:

```
// Zakładamy że w HTMLu zostały zapisane inputy „imie” i
// „nazwisko” a także, że istnieje baza danych i w niej właściwa
// tabela. UWAGA! To najprostsza wersja wykonania tej operacji!
if (count($_POST)){
    // łączenie z bazą danych
    $connection = mysqli_connect("localhost", "root", "", "bd_name");

    $imie = $_POST["imie"];
    $nazwisko = $_POST["nazwisko"];
    // formułowanie zapytania do bazy
    $query = "INSERT INTO tabela(imie, nazwisko)
        VALUES ('$imie', '$nazwisko');";
    // wykonanie zapytania do bazy i zamknięcie połączenia
    mysqli_query($connection, $query)
    mysqli_close($connection);
}
```

## 3. Obsługa wypisania z bazy danych:

```
// Zakładamy, że istnieje już właściwa baza danych i tabela w
// niej. UWAGA! To najprostsza wersja wykonania tej operacji!
$connection = mysqli_connect("localhost", "root", "", "bd_name");
$query = "SELECT imie, nazwisko FROM tabela WHERE warunek";
// pozyskanie wyniku zapytania
$result = mysqli_query($connection, $query);
// fetch tablicy i wypisanie wartości (alternatywnie pętlą for)
while($row = mysqli_fetch_array($result)){
    echo("Imię:".$row[0]. "Nazwisko:".$row[1])
}
mysqli_close($connection);
```

## 4. Warunki i pętle:

```
// if(warunek) {kod} np.:
if(9==10) {...} // false
elseif (10<11) {...} // true
else {...} // powyższe jest prawdą – zostanie pominięte

// while(warunek) {kod} np.:
$i = 0
while($i<10){...}
// UWAGA! Pętla while nie inkrementuje iteratora automatycznie!

// for(start; stop; o_ile_zwiększa) {kod} np.:
for($i=0; $i<=10; $i++){...}
```

## 5. Przydatne metody i funkcje PHP:

```
strtoupper($napis) // napis wielkimi literami
is_numeric($zmienna) // zwraca true / false w zależności od tego
// czy zmienna jest liczbą
isset($_POST["name"]) // zwraca true / false w zależności od tego
// czy dany obiekt został zwrócony pomyślnie
```

# HTML + CSS

## 1. Rodzaje pól formularza (wersja minimalna – parametry jedynie wymagane do działania):

```
<input type="button" value="Treść"> Treść
```

```
<input type="text"> TEKST WPISANY
```

```
<input type="password"> .....
```

```
<select name="nazwa">
  <option value="wartosc">Tekst</option> Tekst ▼
  <option value="wartosc2">Tekst 2</option> Tekst
</select>
```

```
<label for="chbx">checkbox label</label> checkbox label ✓
```

```
<input type="checkbox" name="chbx" value="wartosc">
```

```
<input type="number"> 123456
```

## 2. Pomocne instrukcje dla pól formularzy:

```
<input type="text" placeholder="Wartość domyślna" >
  Wartość domyślna
```

```
<select name="nazwa">
  <option selected disabled hidden>Domyślna wartość</option>
  <option value="wartosc2">Opcja 1</option> Domyślna wartość ▼
  <option value="wartosc2">Opcja 2 </option> Opcja 1
</select>
```

```
<label for="chbx">checkbox label</label>
```

```
<input type="checkbox" name="chbx" value="wartosc" checked>
```

```
<label for="chbx">second checkbox label</label>
```

```
<input type="checkbox" name="chbx" value="wartosc">
```

```
<input type="text" required>
```

Wypełnij to pole.

## 4. Podłączanie CSS, linki, obrazy, skrypty:

```
// podłączanie arkusza CSS
<link rel="stylesheet" href="style.css">
// linkowanie
<a href="adres.pl">Tekst</a>
// obrazy

// skrypt JS
<script> // skrypt </script>
// skrypt PHP
<?php // skrypt ?>
```

## 5. Podstawowe działania CSS:

```
.element {...} // stylowanie klasy
pseudoklasa {...} // stylowanie pseudoklasy (przykład)
#element {...} // stylowanie po ID
element:hover {...} // wygląd po najechaniu kursorem myszy
a:visited {...} // wygląd linku odwiedzonego
/* komentarz */
```

## 6. Listy:

```
<ol> <li> Tekst </li> </ol> // lista numerowana: 1. 2. 3.
<ol type="A"> <li> Tekst </li> </ol> // lista numerowana: A. B. C.
<ol type="I"> <li> Tekst </li> </ol> // lista numerowana: I. II. III.

<ul> <li>Coś</li> </ul> // lista nienumerowana, styl określany w CSS
// parametrem „list-style”
```