

JS – powtórzenie, część I

Anna Gogolińska

Umieszczanie skryptu

- Skrypt można umieścić na stronie lub w zewnętrznym pliku:
 - Na stronie:
 - Kod skryptów należy umieścić między znacznikami:
`<script> </script>`
 - Można go umieścić w `<head>` lub w `<body>`.
 - Kod JS można też umieścić np. w momencie definiowania co ma się stać jak się naciśnie przycisk.

Przykład

```
<!DOCTYPE html>
<html>

<head>
<script>
function myFunction() {
    document.getElementById("demo").innerHTML = "Paragraph changed.";
}
</script>
</head>
<body>

<h1>A Web Page</h1>
<p id="demo">A Paragraph</p>
<button type="button" onclick="myFunction()">Try it</button>

</body>
</html>
```

Przykład

```
<!DOCTYPE html>
<html>
<body>

<h1>A Web Page</h1>
<p id="demo">A Paragraph</p>
<button type="button" onclick="myFunction()">Try it</button>

<script>
function myFunction() {
  document.getElementById("demo").innerHTML = "Paragraph changed.";
}
</script>

</body>
</html>
```

Składnia języka

- Zmienna:
 - Tworzenie: `var nazwaZmiennej;`
 - `var a = "aaa";`
`var b = "bbb";`
`var z = a + b;`
 - `var x = 5;`
`var y = 2;`
`var z = x + y;`
`x = x - y;`
`y = x * z;`

Składnia języka

- Warunek i co ma się stać jeśli zachodzi i jeśli nie zachodzi:

```
if (warunek) {  
    // co się ma stać jeśli zachodzi  
} else {  
    // co się ma stać jeśli nie zachodzi  
}
```
- **for (zainicjowanie_licznika; warunek_kontynuowania; zmiana_licznika)**

```
{  
    kod który zostanie wykonany pewną ilość razy  
}
```

Wypisywanie

- W JS jest wiele sposobów wypisywania informacji:
 - Pisanie w prezentowanej stronie HTML – rzadko używane.
 - Użycie okna alertu (o tym później)
 - Pisanie na konsolę przeglądarki
 - Zmianie tekstu w elemencie HTML

Konsola

- Używane w celach sprawdzenia działania skryptów (debuggowania).
- Składnia:
 - `console.log(5 + 6);`
 - `console.log('Ala ma kota');`

Odwołanie się do elementu strony

- Elementom można ustawić **id** – pokazane to już było w celu ustawiania im indywidualnie reguł stylu.
- Dzięki id do elementu można się odwołać również wewnątrz skryptu:
document.getElementById("mojId")

Zmiana tekstu – sposób na pisanie

- Używane dla elementów które zawierają w swojej treści tekst, np. `div`, `p`, `span`, `a`.
- Zawartość elementu można zmodyfikować poprzez zmianę jego *innerHTML*.
- Przykład:

```
document.getElementById('demo').innerHTML = 'Hello  
JavaScript';
```
- https://www.w3schools.com/js/tryit.asp?filename=tryjs_intro_inner_html_quotes

Zmiana atrybutów

- Można zmienić inne rzeczy, nie tylko tekst w elemencie: atrybuty i style.
- Do elementu również odwołujemy się poprzez id, ale nie zmieniamy jego wewnętrznego htmla ale jakiś atrybut, np. src, href, type.
- Przykład:
`document.getElementById('myImage').src='pic.jpg'`
- https://www.w3schools.com/js/tryit.asp?filename=tryjs_intro_lightbulb

Zmiana stylu

- Podobnie jak poprzednio, uzyskujemy dostęp poprzez id, ale zmieniany style.

- Przykład:

```
document.getElementById('demo').style='font-size: 35px',,
```

Lub:

```
document.getElementById("demo").style.fontSize = "35px";
```

- https://www.w3schools.com/js/tryit.asp?filename=tryjs_intro_style

Widoczność

- Widoczność ustawiamy poprzez style więc:
 - Sprawienie, że element przestanie być widoczny:
`document.getElementById("demo").style.display = "none";`
 - Sprawienie, że element się pojawi:
`document.getElementById("demo").style.display = "inline-block";`
 - https://www.w3schools.com/js/tryit.asp?filename=tryjs_intro_hide
 - https://www.w3schools.com/js/tryit.asp?filename=tryjs_intro_show

Zdarzenia

- Pewne akcje, które zrobi przeglądarka lub użytkownik powodują powstanie zdarzenia. Na przykład kliknięcie elementu na stronie (kliknięcie tekstu, kliknięcie przycisku), najechanie myszką na element na stronie, załadowanie się strony.
- Można używając JS zdefiniować co ma się stać jeśli dane zdarzenie zajdzie dla danego elementu.

Obsługa zdarzenia

- Podstawowa składnia jest:
- *<element event='kod JavaScript'>*
- *<element event="kod JavaScript">*
- Przykład:

https://www.w3schools.com/js/tryit.asp?filename=tryjs_event onclick1

Zdarzenia

Zdarzenie	Opis
onchange	Zmiana elementu HTML – chodzi tu o formularze, kiedy zmieniony zostanie zaznaczony element w select, checkbox itd.
onclick	Element został kliknięty.
onmouseover	Użytkownik przesunął kursor myszki nad element.
onmouseout	Użytkownik odjechał kursorem myszki znad elementu.
onkeydown	Użytkownik nacisnął klawisz na klawiaturze – dla elementów w których można pisać, jak input.
onload	Przeglądarka załadowała stronę.

Definicja funkcji

- Aby zdefiniować funkcję (blokowi kodu nadać właściwości funkcji) trzeba użyć odpowiedniej składni.
- Najpierw słowo **function**, dalej nazwa funkcji i nawiasy.
- W nawiasach można podać parametry – o nich za chwilę.
- Jeśli nie ma parametrów to puste nawiasy.
- Kod funkcji (instrukcje) są umieszczane wewnątrz { }.

Parametry

- Często do funkcji chce się przekazać jakieś wartości, aby zostały one użyte w instrukcjach wewnątrz funkcji.
- Wartości te oznaczane są przez kolejne symbole (parametry) i oddzielone przecinkami.
- Później wywołując funkcję – używając jej nazwy w nawiasach wpisuje się wartości, które mają być podstawione pod parametry.

Wywołanie

- Wywołanie funkcji polega na tym, że kod który jest wewnątrz funkcji (instrukcje) są wykonywane.
- Aby wywołać funkcję należy wpisać jej nazwę wraz z nawiasami.
- Jeśli funkcja nie ma parametrów to piszemy puste nawiasy. Jeśli ma parametry to wpisujemy wartości (mogą być to zmienne).

Przykład

```
<!DOCTYPE html>
<html>
<body>
<p id="demo"></p>
<script>
function toCelsius(f) {
    document.getElementById("demo").innerHTML = (5/9) * (f-32);
}
toCelsius(77);
</script>
</body>
</html>
```

Przykład

```
<!DOCTYPE html>
<html>

<head>
<script>
function myFunction() {
    document.getElementById("demo").innerHTML = "Paragraph changed.";
}
</script>
</head>
<body>

<h1>A Web Page</h1>
<p id="demo">A Paragraph</p>
<button type="button" onclick="myFunction()">Try it</button>

</body>
</html>
```

Przykład

```
<!DOCTYPE html>
<html>
<body>

<h1>A Web Page</h1>
<p id="demo">A Paragraph</p>
<button type="button" onclick="myFunction()">Try it</button>

<script>
function myFunction() {
  document.getElementById("demo").innerHTML = "Paragraph changed.";
}
</script>

</body>
</html>
```