

Technologie internetowe

lista zadań nr 4

Zadanie 1 należy wykonać w czystym JavaScript-cie, natomiast do realizacji pozostałych zachęcam wykorzystać jQuery (lub inną analogiczną bibliotekę).

1. Dany jest następujący dokument HTML:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8" />
<title>Simple Form</title>
<script language="javascript" type="text/javascript">
function checkForm()
{
    var errors = false;
    // sprawdzenie poprawności formularza
    if (errors) document.getElementById('message').innerHTML
        = '<b>Błędy w formularzu</b>';
    else document.getElementById('message').innerHTML
        = '<b>Dane zapisane</b>';
    return false;
}
</script>
</head>
<body>
<div id="message"></div>
<form name="dane" action="#" onsubmit="return checkForm()">
<table>
<tr><td>PESEL</td>
    <td><input type="text" name="Pesel" id="Pesel" /></td></tr>
<tr><td>E-mail</td>
    <td><input type="text" name="Email" id="Email" /></td></tr>
<tr><td>Data urodzenia</td>
    <td><input type="text" name="Data" id="Data" /></td></tr>
<tr><td>Miasto</td>
    <td><select name="Miasto" id="Miasto">
<option value="0" selected="selected">---</option>
<option value="wroclaw">Wrocław</option>
<option value="krakow">Kraków</option>
<option value="poznan">Warszawa</option>
</select></td></tr>
<tr><td>Płeć</td>
    <td><input type="radio" name="Plec" value="m" /> M
        <input type="radio" name="Plec" value="k" /> K</td></tr>
```

```

<tr><td colspan="2"><input type="submit" value="Zapisz" /></td></tr>
</table>
</form>
</body>
</html>

```

Rozszerz powyższy kod o sprawdzenie poprawności powyższego formularza zgodnie z następującymi wytycznymi:

- PESEL powinien być 11 cyfrowy oraz pierwsze 6 cyfr powinno tworzyć poprawną datę (zakładamy, że PESELe są wydane do 1999 roku włącznie),
- e-mail powinien reprezentować w miarę poprawny e-mail,
- data urodzenia powinna tworzyć poprawną datę oraz być taką samą datą, jak ta w PESEL-u,
- wybrane powinny być płeć i miasto.

Dla każdej zależności powinna być utworzona osobna funkcja, należy także skorzystać z możliwości wyrażeń regularnych. Podsumowanie dotyczące błędów należy umieścić w polu o *id*=*'message'*.

[3p]

2. Korzystając z wzorca *Revealing Module Pattern* utwórz obiekt do generowania tabel HTML. Obiekt powinien mieć następujące składowe:

- pola prywatne:
 - data – tabela dwuwymiarowa z danymi do prezentacji,
 - target – id elementu na stronie, w którym ma zostać osadzona tabelka,
 - options – obiekt reprezentujący ewentualne opcje; zakładamy, że można się spodziewać dowolnego podzbioru spośród opcji *border*, *padding* i *spacing*;
- metody prywatne:
 - createCell(content) – zwraca *content* opakowany w znaczniki *<td>*,
 - createRow(content) – zwraca *content* opakowany w znaczniki *<tr>*,
 - createHeader() – zwraca początkowy stały fragment tabeli,
 - createFooter() – zwraca końcowy stały fragment tabeli;
- metody publiczne:
 - init(data, target, options) – ustawia pola prywatne,
 - generate() – tworzy tabelkę i osadza ją w odpowiednim elemencie dokumentu.

Utwórz dokument HTML i wykorzystaj powyższą konstrukcję do utworzenia dwóch różnych tabel HTML. Zadbaj też, żeby kod JavaScript został wykonany po pełnym załadowaniu dokumentu (czyli trzeba ten kod podpiąć pod odpowiednie zdarzenie).

[2p]

3. Utwórz dokument XHTML, w którym będą osadzone 4 miniaturki obrazków (zakładamy, że miniaturki są katalogu *thumbs*). Następnie korzystając z konstrukcji *.each()* w jQuery dla każdego obrazka wykonaj następujące dwie czynności.

- Jeśli brakuje atrybutu `alt` lub jest pusty, ustaw jego wartość na nazwę pliku obrazka bez rozszerzenia.
- Opakuj znacznik `img` w link prowadzący do pliku obrazka, ale w katalogu `images`.

Skrypt ma się wykonać automatycznie po załadowaniu dokumentu. Przykładowo, kod

```

```

ma zostać przekształcony na

```
<a href="images/ferrari.jpg"></a>
```

[2p]

4. Utwórz *slider* z przewijaniem zdjęć, który powinien mieć następujące funkcjonalności:

- zdjęcia powinny się automatycznie zmieniać co pewien czas (powiedzmy 3 sekundy), przy czym nie oczekuję żadnych efektów przy wymianie zdjęcia,
- po najechniu myszką na obszar *slidera* „przewijanie zdjęć” powinno zostać wstrzymane, a po zjechaniu – ponownie włączone.

Można przyjąć założenia upraszczające, jak np. określoną liczbę zdjęć na sztywno wprowadzoną w kod JavaScript, itp. Kluczowe są dwie powyższe funkcjonalności.

[3p]

Paweł Rajba