

**Zad.1.**

Utwórz dokument HTML5 (Listing 1) podobny jak na zajęciach poprzednich z *RWD* tyle, że tym razem zamiast nawigacji opartej na hiperłączach, nawigacja będzie realizowana za pomocą przycisków (element *button*), które będą obsługiwane przez odpowiednie funkcje JavaScript. Zadaniem funkcji JS będzie modyfikacja zawartości elementu o **id="blok"**, w zależności od tego, który przycisk wybierze użytkownik. Będziemy teraz pracować z jednym dokumentem HTML (zamiast tworzyć 4 różne dokumenty jak należało zrobić w zadaniu poprzednim).

Zwróć uwagę na znacznik:

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

który łączy skrypt JS z definicjami funkcji, które wykorzystamy do obsługi przycisków (plik *funkcje.js* znajduje się w folderze *js*).

Listing 2 przedstawia kod pliku *style.css*.

**Listing 1. Plik index.html z przyciskami akcji w elemencie nav**

```
<!DOCTYPE html>
<html>
  <head>
    <title>Przykłady Java Script</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="css/style.css" />
    <script src="js/funkcje.js"></script>

  </head>
  <body>
    <header>
      <h1>Przykład strony z JavaScript</h1>
      <nav>
        <button onclick="pokaz(1)">0 mnie </button>
        <button onclick="pokaz(2)">Moja galeria</button>
        <button onclick="pokaz(3)">Dodaj post</button>
        <button onclick="pokaz(4)">Kontakt</button>
      </nav>
    </header>

    <section id="blok">
      <h2><br />0 mnie</h2>
      <p>
        Lorem ipsum ... </p>
      <p class="srodek">
        
      </p>
      <article>
        <h2>Moje hobby</h2>
        <p> Lorem ipsum ... </p>
      </article>
    </section>
    <footer>&copy;BP</footer>
  </body>
</html>
```

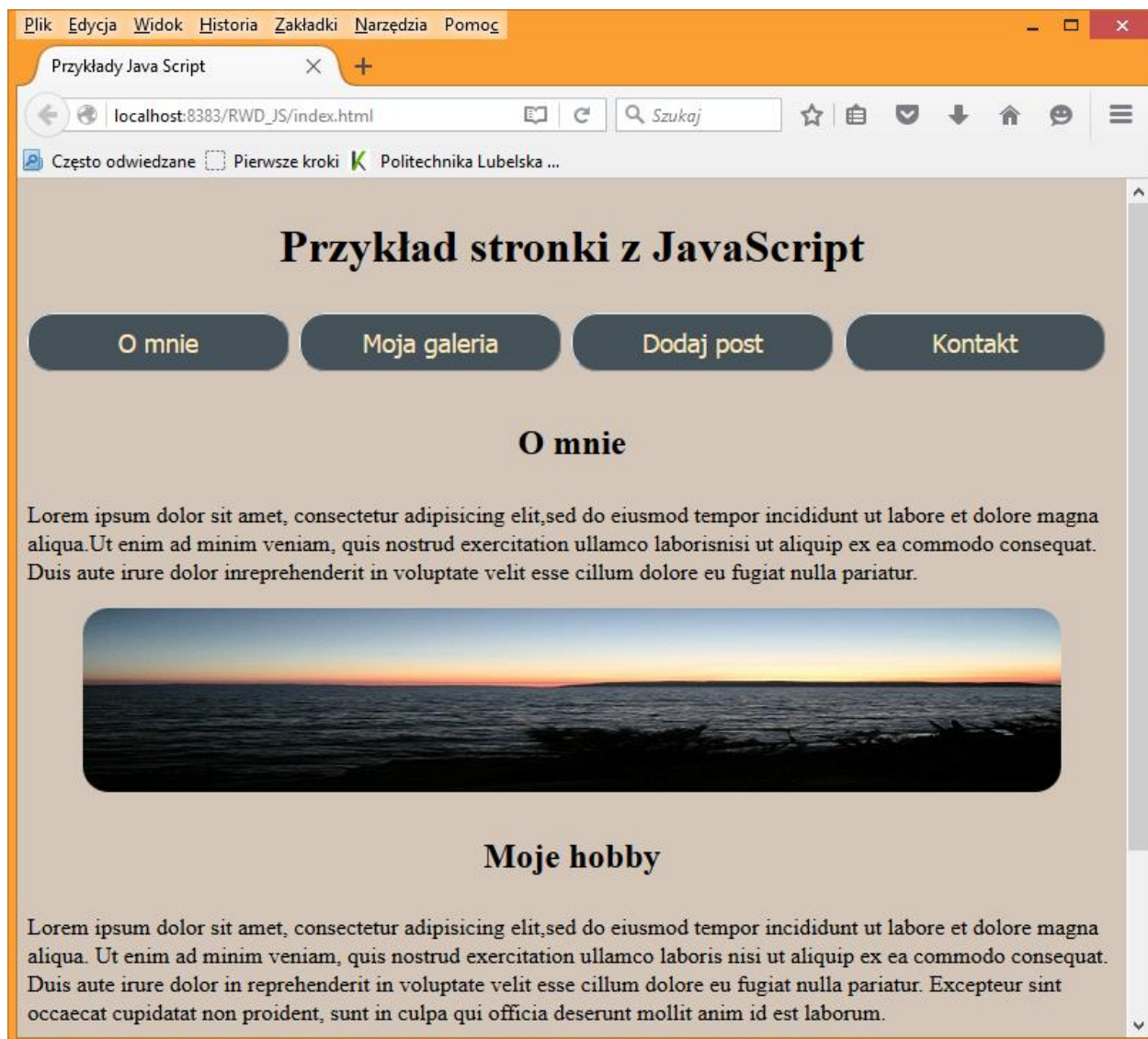
**Listing 2. Arkusz style.css**

```
body { background:#d4c7b9; font-size:18px; }
h1,h2,footer { padding:0.2em; text-align:center; }
.srodek { text-align:center; }
```

```
nav{ float:left; width:100%; }
footer { background: #45525a; color:#ffe9ba; }
nav button {
    background: #45525a;
    color:#ffe9ba;
    float:left;
    display:block;
    width:24%;
    margin-right: 1%;
    padding:1% 0;
    font-size:110%;
    text-align:center;
    border-radius:20px;
}
section { clear:both; }
img { max-width: 100%; height: auto; border-radius:20px; }
.slajd {
    float:left;
    display:block;
    width:13%;
    text-align:center;
    border-radius:20px;
    background:#ffe9ba;
    padding:1% 0.5%;
    margin:1%;
}
.slajd:hover {
    background:#45525a;
}
footer { clear:both; border-radius:20px; }
.galeria{ margin:auto; width:90%; }

@media screen and (max-width: 720px) {
    nav button { width: 48%;
        padding: 10px 0;
        font-size:140%;
    }
    .slajd {width: 45%; }
}
@media screen and (min-width: 481px) and (max-width: 720px) {
    body { font-size: 13px; }
    h1 { font-size: 23x; }
    h2 { font-size: 18px; }
}
@media screen and (max-width: 481px) {
    body { font-size: 16px; }
    h1 { font-size: 24px; }
    h2 { font-size: 20px; }
    nav button{ font-size:20px;
        padding:10px 0;
        border-bottom: 3px solid white;
        width: 98%;
    }
    .slajd {width: 95%; }
}
```

Rysunek 1 przedstawia efekt widoczny w przeglądarce (strona powinna się zachowywać responsywnie).



Rysunek 1.

Naszym zadaniem będzie teraz dodanie skryptu JS z funkcjami, które umożliwią interakcję z użytkownikiem – w przykładzie będzie to modyfikacja treści sekcji o identyfikatorze `id="blok"` z główną zawartością strony.

### Podstawowe zdarzenia na stronach HTML

Najbardziej podstawowe zdarzenia zachodzące na stronie WWW a związane z akcjami użytkownika są obsługiwane w HTML za pomocą odpowiednich atrybutów dodanych do tagów HTML (w naszym przykładzie będą to przyciski **button**).

Najbardziej podstawowe zdarzenia zachodzące na stronie to:

- kliknięcie na element – zdarzenie **click**, atrybut w elemencie HTML: **onclick**;
- wejście wskaźnikiem myszki w obszar elementu – **mouseover**, atrybut **onmouseover**;
- wyjście wskaźnika poza obszar elementu – **mouseout**, atrybut **onmouseout**

- wysłanie formularza realizowane poprzez kliknięcie na przycisk formularza typu **submit** – zdarzenie **submit**, atrybut **onsubmit**;
- wyczyszczenie formularza realizowane poprzez kliknięcie na przycisk formularza typu **reset** – zdarzenie **reset**, atrybut **onreset**.

**Zad. 2.**

W zadaniu 1 przygotowany został szablon strony wraz z początkową zawartością sekcji o **id="blok"**. Naszym zadaniem będzie teraz dodanie odpowiednich funkcji Java Script, które będą wywoływane w momencie kliknięcia na poszczególne przyciski. Zwróć uwagę na kod:

```
<button onclick="pokaz(1)" >O mnie </button>
```

Dołączenie do elementu **button** atrybutu **onclick="pokaz(1)"** powoduje, że w momencie kliknięcia na przycisk (zachodzi zdarzenie **click**) wywołana zostanie funkcja **pokaz** z jednym parametrem o wartości 1.

Oczywiście teraz trzeba odpowiednio zdefiniować taką funkcję. Wszystkie definicje tej i innych potrzebnych funkcji umieścimy w pliku *funkcje.js*, który już został dołączony do naszego dokumentu *index.html*. Listing 3 przedstawia szablon pliku *funkcje.js*. Zwróć uwagę na instrukcję **switch** oraz na kolejne funkcje, które są wywoływane w zależności od wartości przekazanego funkcji *pokaz* parametru.

Zadaniem funkcji:

- *function pokaz(id)* – na podstawie wartości parametru *id*, generowana jest zawartość sekcji o *id= 'blok'*;
- *function pokazOmnie()* – jest wygenerowanie strony O mnie (funkcja tworzy wynikowy HTML i zwraca go jako wynik w postaci łańcucha – typu string);
- *function pokazGalerie()* – jest wygenerowanie strony z galerią;
- *function pokazKontakt()* – jest wygenerowanie strony z informacjami kontaktowymi;
- *function pokazPost()* – jest wygenerowanie formularza, za pomocą którego użytkownik może przesłać post;
- *function pokazDane()* – ma za zadanie zebranie danych z wypełnionego formularza i pokazanie odpowiedniego okienka do akceptacji przesłania danych.

**Listing 3. Plik funkcje.js**

```
function pokaz(id)
{
    tresc="";
    switch (id)
    {
        case 2: tresc += pokazGalerie();break;
        case 3: tresc += pokazPost(); break;
        case 4: tresc += pokazKontakt();break;
        default: tresc += pokazOmnie();
    }
    //pobierz element o wskazanym id i ustaw jego nową zawartość:
    document.getElementById('blok').innerHTML = tresc;
}

function pokazOmnie(){
    tresc = '<h2><br />O mnie</h2> ';
    //operator += uzupełnia łańcuch kolejną porcją znaków:
    Tresc += '<p>Lorem ipsum dolor pariatum,...</p>'+
        '<p class="srodek"></p>'+
        '<article><h2>Moje hobby</h2><p>'+
        'Lorem ipsum dolor sit amet,...'+
        'mollit anim id est laborum.</p></article>';
    //przekaż wynik - gotową zawartość - do miejsca wywołania funkcji:
```

```

    return tresc;
}

function pokazGalerie()
{
    tresc='<h2><br />Moja galeria</h2>';
    tresc+=' <div class="galeria">';
    //wygeneruj kod HTML z obrazami za pomocą pętli for:
    for(i=1;i<=10;i++)
    {
        tresc+='<div class="slajd"> <img ... /></div>';
    }
    return tresc+'</div>';
}

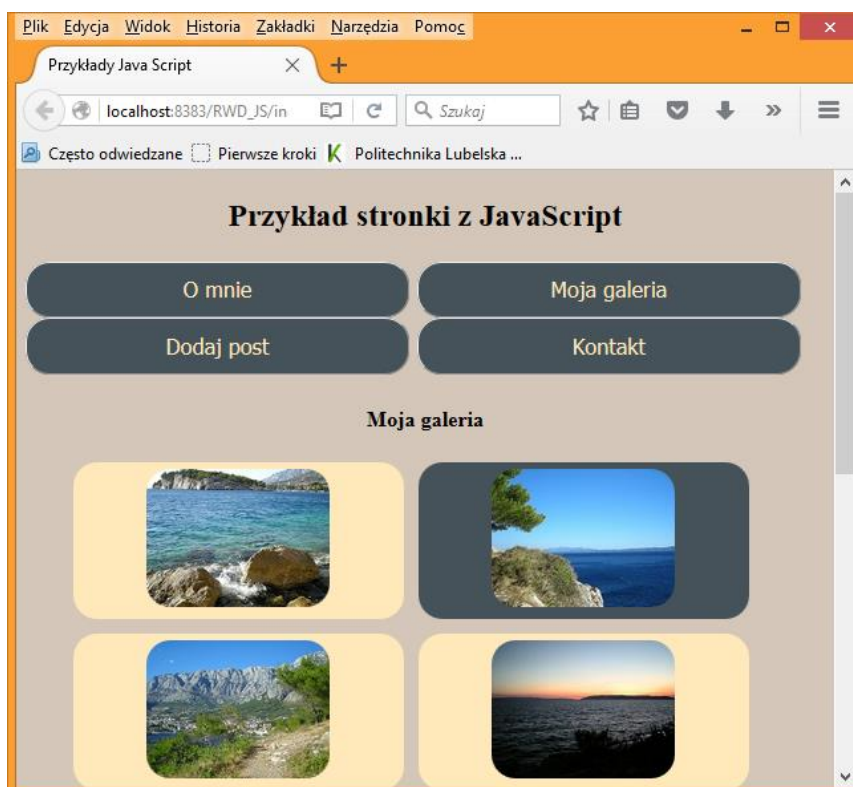
function pokazKontakt()
{
    tresc='<h2><br />Kontakt</h2>';
    //uzupełnij treść:
    // tresc+= ...
    return tresc;
}

function pokazPost()
{
    //funkcja generuje kod formularza - dane wpisane w odpowiednie pola przez
    //użytkownika zostaną przekazane mailem na wskazany adres, ale w
    //zajścia zdarzenia submit - zostanie wywołana funkcja pokazDane()
    //submit zostanie najpierw
    który będzie wysłany mailem
    tresc='<h2><br />Dodaj post</h2>';
    tresc+='<article class="srodek" ><form action="mailto:b.panczyk@pollub.pl"
        method="post" onsubmit="return pokazDane();">'+
        'Twój email:<br /> <input type="email" name="email" id="email"
        required /><br />'+
        // dodaj kolejne 2 pola formularza
        'Komentarz: <br /><textarea rows="3" cols="20" id="wiadomosc"
        name="wiadomosc" required></textarea>'+
        '<br /> <input type="submit" name="wyslij" value="Wyślij" />'+
        '</form></article>';
    return tresc;
}

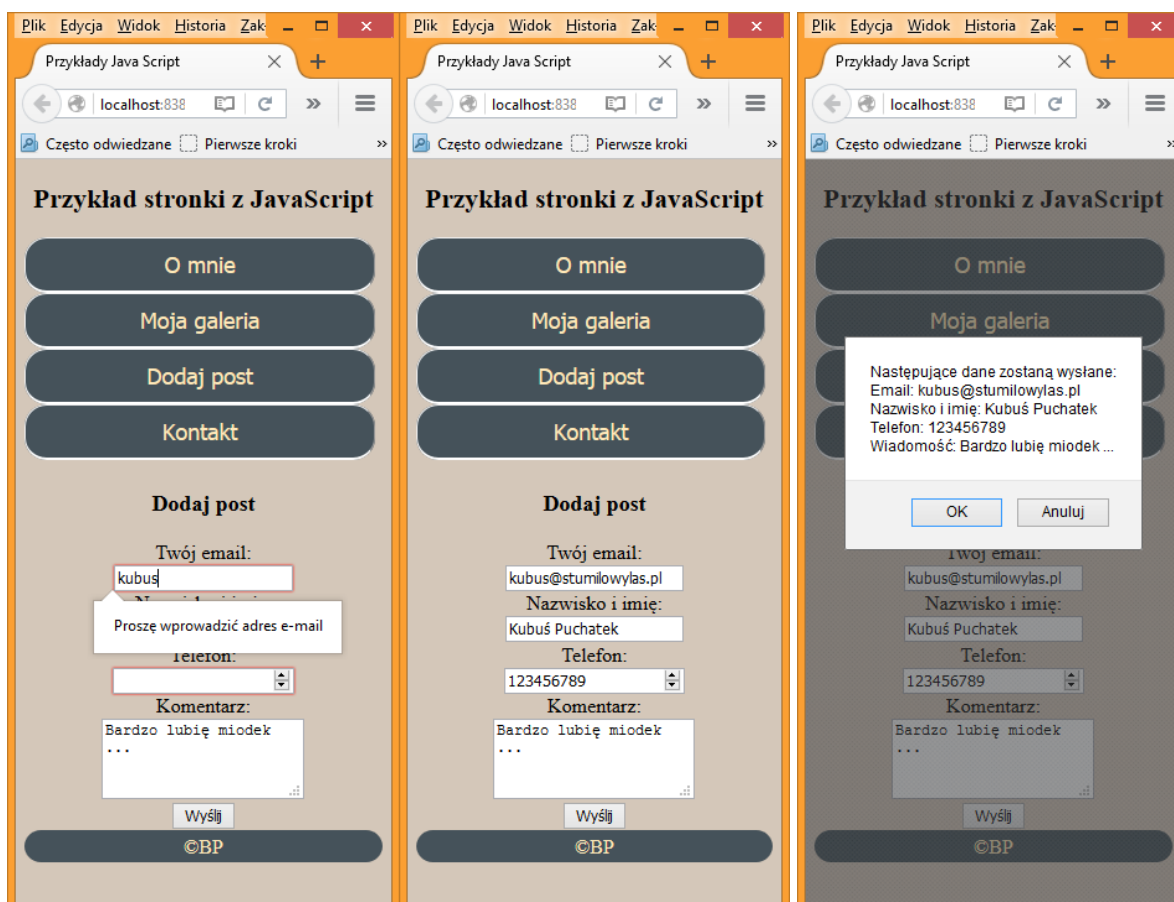
function pokazDane()
{
    //Funkcja zbiera dane wpisane w pola formularza i wyświetla okienko
    //typu confirm do zatwierdzenia przez użytkownika:
    dane="Następujące dane zostaną wysłane:\n";
    dane+="Email: "+document.getElementById('email').value+"\n";
    // uzupełnij dane ...
    if (window.confirm(dane)) return true;
    else return false;
}

```

Rysunki 3-4 przedstawiają przykładowe widoki poszczególnych zakładek.



Rysunek 3. Widok strony z galerią

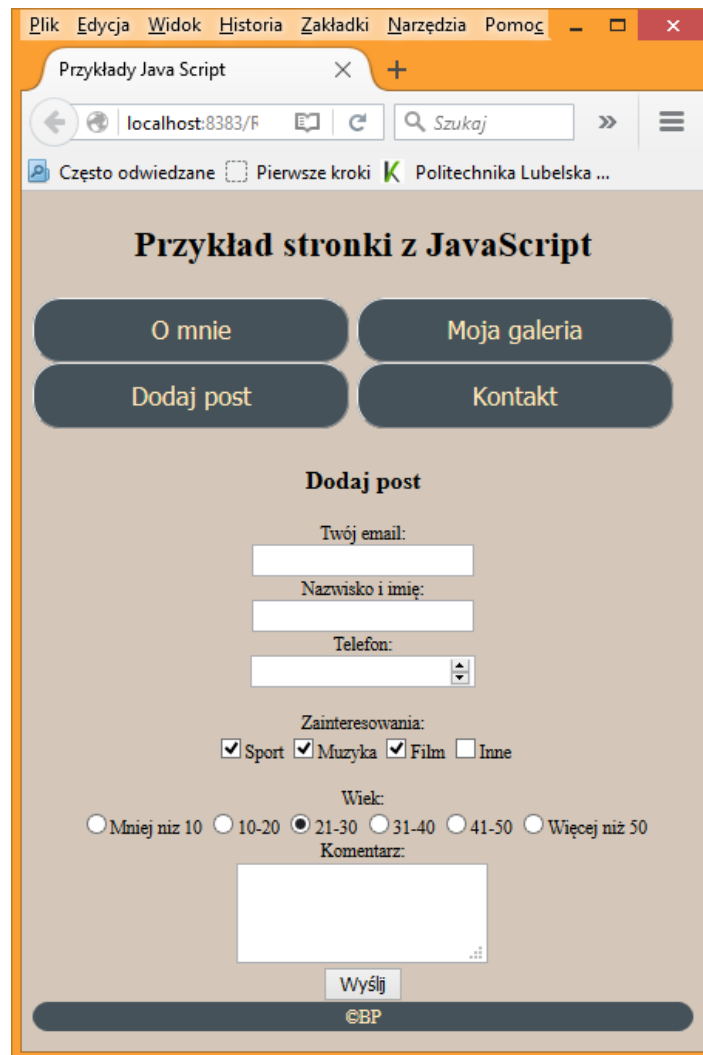


Rysunek 4. Widok strony z postem



**Zad. 3.**

Uzupełnij formularz dodatkowymi polami wyboru na temat zainteresowań (grupa przycisków typu *checkbox* – można zaznaczyć kilka z nich) oraz wieku (grupa przycisków typu *radio* – możliwy jest wybór tylko jednej opcji spośród wielu) (Rysunek 5). Dodaj te informacje o wyborze użytkownika do wyświetlanego okienka z potwierdzeniem. Przykładowe funkcje sprawdzające, które opcje zostały wybrane przez użytkownika możesz znaleźć w wykładzie 3 z Java Script.



Przykłady Java Script

localhost:8383/F Szukaj

Często odwiedzane Pierwsze kroki Politechnika Lubelska ...

### Przykład strony z JavaScript

O mnie Moja galeria

Dodaj post Kontakt

#### Dodaj post

Twój email:

Nazwisko i imię:

Telefon:

Zainteresowania:

☒ Sport ☒ Muzyka ☒ Film ☐ Inne

Wiek:

☐ Mniej niż 10 ☐ 10-20 ☒ 21-30 ☐ 31-40 ☐ 41-50 ☐ Więcej niż 50

Komentarz:

Wyślij

©BP

Rysunek 5. Widok formularza z dodatkowymi polami