

## Zajęcia 5 – Ajax i geolokacja

Zainstaluj Firebug w przeglądarce Firefox: <http://getfirebug.com/> lub skorzystaj z narzędzi dla programistów w innych przeglądarkach.  
(Pomoże ci to w kontroli kodu JavaScript).

### 1. Zadania do wykonania:

- 1.1. Na własnej stronie zastosuj funkcje ajaksowe do ładowania zawartości innych stron (patrz punkt 2 poniżej) oraz wymiany treści fragmentu strony z plików XML (wykorzystaj przykład 2 poniżej tworząc własny plik XML).
- 1.2. Utwórz także podstronę z geolokacją (wpisz współrzędne geograficzne oraz wyświetl na mapce (z Google Maps). Możesz zastosować kod JavaScript lub jQuery.

### 2. Parsowanie XML - wczytywanie zawartości pliku XML (jQuery)

Utwórz plik **wpisy.xml** (zawiera linki do stron) o następującej strukturze i zamieść 3, 4 wpisy:

```
<?xml version="1.0" encoding="utf-8" ?>
<wpisy>
  <wpis>
    <tytul> jakiś tytuł </tytul>
    <opis> tu opis tego co na stronie </opis>
    <data> data wpisu</data>
    <link> http://tu adres strony</link>
  </wpis>
  <wpis>
    <tytul> ... </tytul>
    <opis> ... </opis>
    <data> ... </data>
    <link> ... </link>
  </wpis>
  .....
</wpisy>
```

Dokument XML musi znajdować się na tej samej domenie ponieważ w przeciwnym wypadku napotkamy błąd typu *cross domain*.

Następnie utwórz prostą stronę HTML5 **wpisy.html**.

W sekcji <head> musi się znaleźć kod:

```
<script src="http://code.jquery.com/jquery-1.7.1.js"></script>
```

ponieważ załadujesz zawartości stron z linków za pomocą funkcji ajaksowych z jQuery.

```
<script>
$(document).ready(function () {
  $.ajax({
    type: "GET",
    url: "wpisy.xml",
    dataType: "xml",
    success: fParsujXml
  });
});
function fParsujXml(xml) {
```

```

$(xml).find("wpis").each(function () {
    $(".wpisy").append('<div class="wpisy"><div class="tytul">'
    + $(this).find("tytul").text() + '</div><div class="opis">'
    + $(this).find("opis").text() + '</div><div class="data">Dnia '
    + $(this).find("data").text() + '</div><div class="link"><a href="'
    + $(this).find("link").text() + '"target="_blank">link</a></div></div>');

    $(".wpisy").fadeIn(1000);
});
}
</script>

```

W sekcji <body> natomiast załaduj zawartość strony:

```
<div class="wpisy"> </div>
```

Opis działania skryptu:

- W przypadku poprawnego załadowania pliku (metodą GET, ale może być oczywiście POST), wywoływana jest funkcja *fParsujXml*.
- Po jej uruchomieniu przeszukiwany jest dokument w poszukiwaniu znacznika *wpis*.
- Po jego znalezieniu, dodawane są do elementu HTML-owego o klasie „wpisy” poszczególne części składowe znalezionego elementu: tytuł, opis, data i link, (funkcja jQuery *find*).
- Gdy skrypt zakończy przeszukiwanie element zostanie pokazany z wykorzystaniem funkcji *fadeIn*, która wprowadza dodatkowy element animacji, ale nie jest ona konieczna.
- Zawartości stron są tutaj ładowane w nowym oknie *target="\_blank"*.  
Inne możliwości patrz np.: [http://www.w3schools.com/html5/att\\_a\\_target.asp](http://www.w3schools.com/html5/att_a_target.asp)

### 3. Ćwiczenie dotyczy zastosowania Ajax (**JavaScript**) do odczytu danych z pliku tekstowego .

Utwórz plik tekstowy tekst.txt. Jego zawartość zostanie odczytana i wyświetlona w nowym oknie (**alert box**).

Stwórz prosty plik HTML5 i w sekcji body umieść np.

```

<span
    style="cursor: pointer; text-decoration: underline"
    onclick="makeRequest('data.txt')">
    Przeczytaj
</span>

```

*Pamiętaj, że JS działa tylko lokalnie (w obrębie serwera na którym są pliki) i nie działa zdalnie (nie pobierze informacji z innego serwera).*

W sekcji <head> zamieść poniższy skrypt:

```

<script type="text/javascript">

//<![CDATA[

```

```

var http_request = false;

function makeRequest(url) {
    http_request = false;
    if (window.XMLHttpRequest) { // Mozilla, Safari,...
        http_request = new XMLHttpRequest();
        if (http_request.overrideMimeType) {
            http_request.overrideMimeType('text/xml');
            // Przeczytaj o tym wierszu poniżej
        }
    } else if (window.ActiveXObject) { // IE
        try {
            http_request = new ActiveXObject("Msxml2.XMLHTTP");
        } catch (e) {
            try {
                http_request = new ActiveXObject("Microsoft.XMLHTTP");
            } catch (e) {}
        }
    }
    if (!http_request) {
        alert('Poddaję się :( Nie mogę stworzyć instancji obiektu XMLHTTP');
        return false;
    }
    http_request.onreadystatechange = function() {
        alertContents(http_request); };
    http_request.open('GET', url, true);
    http_request.send(null);
}

function alertContents(http_request) {

    if (http_request.readyState == 4) {
        if (http_request.status == 200) {
            alert(http_request.responseText);
        } else {
            alert('Wystąpił problem z zapytaniem.');
        }
    }
}

//]]>
</script>

```

#### 4. Wykorzystanie *Google Location Services*

Poniżej są dwa przykłady: kod JavaScript i JQuery (**kod na czerwono** – wyświetla mapkę Google z zaznaczoną pozycją) .

Stwórz dwie strony w HTML5 wykorzystując poniższe kody, i uruchom je np. w przeglądarce Firefox z włączonym Firebugiem (*lub w innej przeglądarce ze stosownym narzędziem kontroli czasu*) – porównaj czasy ładowania.

## JavaScript:

```
<script type="text/javascript">
function showLocation(position) {
    var latitude = position.coords.latitude;
    var longitude = position.coords.longitude;
    alert("Latitude : " + latitude + " Longitude: " + longitude);
}

function errorHandler(err) {
    if(err.code == 1) {
        alert("Error: Access is denied!");
    }else if( err.code == 2) {
        alert("Error: Position is unavailable!");
    }
}

function getLocation() {
    if(navigator.geolocation){
        // timeout at 60000 milliseconds (60 seconds)
        var options = {timeout:60000};
        navigator.geolocation.getCurrentPosition(showLocation,
                                                    errorHandler,
                                                    options);
    }else{
        alert("Sorry, browser does not support geolocation!");
    }
}
</script>
```

Oraz w sekcji <body> np.:

```
<form>
    <input type="button" onclick="getLocation();"
           value="Get Location"/>
</form>
```

## jQuery:

```
<script>
    jQuery(window).ready(function() {
        jQuery("#btnInit").click(initiate_geolocation);
    });

function initiate_geolocation() {
    navigator.geolocation.getCurrentPosition(handle_geolocation_query,handle
_errors);
}

function handle_errors(error)
{
    switch(error.code)
    {
```

```

        case error.PERMISSION_DENIED: alert("user did not share
geolocation data");
        break;

        case error.POSITION_UNAVAILABLE: alert("could not detect
current position");
        break;
        case error.TIMEOUT: alert("retrieving position timed out");
        break;
        default: alert("unknown error");
        break;
    }
}

```

```

function handle_geolocation_query(position) {

    var text = "Latitude: " + position.coords.latitude + "<br/>" +
        "Longitude: " + position.coords.longitude + "<br/>" +
        "Accuracy: " + position.coords.accuracy + "m<br/>" +
        "Time: " + new Date(position.timestamp);
    jQuery("#info").html(text); //tu podmiana

    var image_url =
"http://maps.google.com/maps/api/staticmap?sensor=false&center=" +
position.coords.latitude + ',' + position.coords.longitude +
"&zoom=14&size=300x400&markers=color:blue|label:S|"
+ position.coords.latitude + ',' + position.coords.longitude;

    jQuery("#map").remove();
    jQuery(document.body).append(
        jQuery(document.createElement("img")).attr("src",
image_url).attr('id', 'map')

    );
}
</script>

```

Oraz w sekcji <body> np.:

```
<div>         <button id="btnInit" >Find my location</button>         </div>
```

Wykorzystane strony:

<http://blog.alchemycode.pl/2010/04/24/parsowanie-xml-i-w-5-minut-z-wykorzystaniem-jquery/>

Interesujące strony:

<http://webhosting.pl/Kurs.jQuery.Czesc.3.Dynamiczna.witryna.AJAX.i.efekty.interfejsu?page=5>

<http://webhosting.pl/Czas.przestac.placic.podatek.od.jQuery.Tracimy.nawet.350.ms.czasu.ladowania.strony>

<http://www.webskrypty.pl/2010/wysylanie-danych-i-odpowiedz-z-wykorzystaniem-jquery/>