

Développement d'applications web

Solution Exercices Cours 5

Vincent Simonet, 2015-2016

Exercice 1 : RATP

Question 1.1

Un appel AJAX est soumis à la *same origin policy* qui requiert que l'appel soit effectué sur le même domaine que celui de la page courante. Il ne pourra donc fonctionner puisque l'appel est effectué sur le domaine ratp.fr alors que la page est servie par le domaine upmc.fr.

Question 1.2

```
function callRatp(line, station, direction, cbName) {
    var script = document.createElement('script');
    script.setAttribute(
        'src',
        'http://api.ratp.fr/nexttrain?line=' + line +
        '&station=' + station + '&direction=' + direction +
        '&callback=' + cbName);
    document.body.appendChild(script);
}

function insertMessage(message) {
    var p = document.createElement('p');
    p.textContent = message;
    document.getElementById('metro').appendChild(p);
}

function showMetro() {
    callRatp(11, 'Jussieu', 'Boulogne', 'callback1');
}

function callback1(data) {
    insertMessage('Le prochain train pour Boulogne sur la ligne 11 ' +
        'passe à ' + data.time);
}
```

Question 1.3

```
function showMetro() {
    callRatp(7, 'Jussieu', 'Ivry', 'callback1');
    callRatp(7, 'Jussieu', 'Villejuif', 'callback2');
}

var data1 = null;
var data2 = null;
```

```

function callback1(data) {
    insertMessage('Le prochain train pour Boulogne sur la ligne 11 ' +
        'passe à ' + data.time);
}

function callback2(data) {
    insertMessage('Le prochain train pour Austerlitz sur la ligne 11 ' +
        'passe à ' + data.time);
}

```

Question 1.4

```

function showMetro() {
    callRatp(7, 'Jussieu', 'Ivry', 'callback1');
    callRatp(7, 'Jussieu', 'Villejuif', 'callback2');
}

var data1 = null;
var data2 = null;

function callback1(data) {
    data1 = data;
    finalize();
}

function callback2(data) {
    data2 = data;
    finalize();
}

function finalize() {
    if (data1 && data2) {
        var txt = 'Le prochain train vers le sud sur la ligne 7 ' +
            'a pour terminus ';
        if (data1.date + data1.time < data2.date + data2.time) {
            insertMessage(txt + 'Ivry');
        } else {
            insertMessage(txt + 'Villejuif');
        }
    }
}

```

Exercice 2 : Client Web et RPC

Question 2.1 Sélecteurs CSS

	Sélecteur CSS	Numéro(s) de ligne des sélecteurs
a.	<code>#flight-number</code>	16
b.	<code>.field</code>	9, 15, 18, 21
c.	<code>#status .field</code>	15, 18, 21
d.	<code>.field > #flight-number</code>	16
e.	<code>#status #flight-number</code>	16
f.	<code>#status > #flight-number</code>	aucun
g.	<code>span.form input</code>	aucun
h.	<code>input[type="button"]</code>	12
i.	<code>body #status > .field span</code>	16, 19, 22
j.	<code>.form .field, #status .field</code>	9, 15, 18, 21

Question 2.2: Fonction submitForm

La fonction submitForm agit en insérant dynamiquement un élément `<script>` dans la page HTML. Cette insertion provoque une requête HTTP GET contenant la valeur entrée par l'utilisateur dans le formulaire en argument. Le serveur répond par un script JavaScript contenant un appel à la fonction callback. Ce script est exécuté, ce qui provoque la mise à jour des données dans la page HTML.

Il s'agit du protocole JSONP.

Le développeur n'a pas choisi de faire un appel AJAX à cause de la *same-origin policy* : le serveur contacté (airservice.com) n'est pas le même que celui qui sert la page HTML (www.flightstatus.com).

Question 2.3: Réponse

```
HTTP/1.1 200 OK
Content-Type: application/jsonp; charset=utf-8

callback({
  "flight_number": "AF007",
  "current_status": "On time",
  "arrival_time": "10:25"
```

```
});
```

Question 2.4: AJAX

On suppose désormais que l'appel pour obtenir l'état d'un vol est fait vers le même serveur (flightstatus.com). La réponse est retournée au format XML, par exemple :

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
  <item id="flight-number">AF007</item>
  <item id="current-status">On time</item>
  <item id="arrival-time">10:25</item>
</response>
```

Le code de la fonction submitForm peut alors être modifié comme suit :

```
function submitForm(event) {
  var xhr = new XMLHttpRequest();
  xhr.open('get', 'http://www.flightstatus.com?q=' + query.value);
  xhr.responseType = "document";

  xhr.onreadystatechange = function() {
    // Ready state 4 means the request is done
    if (xhr.readyState === 4) {
      if(xhr.status === 200){
        var root = xhr.responseXML.documentElement;
        callback({
          flight_number: root.getElementById('flight-number').textContent,
          current_status: root.getElementById('current-status').textContent,
          arrival_time: root.getElementById('arrival-time').textContent
        });
      } else {
        alert('Error: ' + xhr.status);
      }
    }
  };

  xhr.send(null);
}
```