



CENTRE SCOLAIRE SAINTE-JULIENNE

Exercices Ajax – Bases

Série 1 - Enoncés

I- Mise en situation


Tu es web master dans une société et tu dois passer un test sur les techniques AJAX. A travers une série d'exercices, tu dois comprendre et maîtriser ces techniques pour obtenir la prime salariale.

II- Travail à réaliser

1. Effectuer l'exercice proposé par le professeur.
2. Analyser l'énoncé du point III correspondant au numéro de l'exercice demandé.
3. Réaliser l'exercice.
4. Visualiser le travail.
5. Sauvegarder le document suivant les instructions données.
6. Imprimer le(s) document(s).

III- Enoncés

1. Ex1 – Objet XHR
Créer une page PHP vierge;
Nommer cette page: index.php.
Sauvegarder l'exercice dans un dossier nommé: Ex1 – Objet XHR.
Titrer la page avec ce même nom.

AJAX - Exercice 1 - Objet XHR	
<input type="text" value="Frodon"/>	<input type="text" value="bit"/>
<input type="text" value="Frodon"/> <input type="text" value="Gandalf le Gris"/> <input type="text" value="Aragorn"/>	
<i>Classe</i>	<input type="text" value="Aventurier"/>
<i>Provenance</i>	<input type="text" value="La Comté"/>

Créer la liste dynamiquement à partir du tableau associatif suivant contenu dans une page PHP : donnees.php :

```
$tabPersonnages=array("Frodon"=>array("Hobbit",
                                        "Aventurier",
                                        "La Comté",
                                        "Frodon.png"),
                      "Gandalf le Gris"=>array("Humain",
                                                "Magicien",
                                                "Inconnue",
                                                "GandalfGris.png"),
                      "Aragorn"=>array("Humain",
                                        "Rodeur",
                                        "Fondcombe",
                                        "Aragorn.png"));
```

Créer un tableau qui reprend la race, la classe, la provenance et la photo du personnage.

Créer une image identifiée loader qui contient un gif animé appelé ajax-loader.gif.

Associer au chargement de la page et à la sélection d'un élément de la liste l'appel à la fonction Javascript request(readData) où readData est le nom d'une fonction callback.

Créer une page JS vierge nommée Ex01.js qui contient:

- une fonction getXMLHttpRequest() qui retourne un objet xhr;
- la fonction request(callback) qui:
 1. initialise l'objet xhr,
 2. teste son exécution avec gestion du loader,
 3. traite son envoi vers une page PHP: pageServeur.php avec le nom du personnage choisi avec la méthode GET.
 4. traite le retour d'information : callback(xhr.responseText);
- la fonction readData(sData) où sData est une chaîne de caractères où chaque info est séparée par un *. Cette fonction garni les champs texte du tableau via le DOM.

Créer la page PHP : pageServeur.php qui contient le code suivant:

```
<?php
    include("donnees.php");

    $personnage = (isset($_GET["perso"]))?
htmlspecialchars($_GET["perso"]) : NULL;
    if($personnage) echo implode(" ",$tabPersonnages[$personnage]);
    else echo "Inconnu";

?>
```

2. Ex2 – DSL & XML

Reprendre le contenu de l'Ex1.

Sauvegarder l'exercice dans un dossier nommé: Ex2 – DSL & XML.

Titrer la page avec ce même nom.

Le fichier donnees.php inclus le fichier db.php et contient les fonctions suivantes:

```
<?php      include("db.php");

            function listerPersonnages()
            {
                ouvrir();
                $tabPersonnages=lire("SELECT * FROM personnages
ORDER BY nom;");
                fermer();
                return $tabPersonnages;
            }

            function afficherPersonnage($nom)
            {
                ouvrir();
                $personnage=lire("SELECT * FROM personnages WHERE
nom='".$nom."'");
                fermer();
                return $personnage;
            }
?>
```

Les fichiers db.php et config.ini vous sont donnés.

Créer un fichier DSLPersonnage.php qui reprend l'en-tête suivant: <?php header("Content-type: text/javascript"); ?> et les instructions javascript qui permettent de garnir avec le DOM les champs texte.

Modifier le fichier pageServeur.php avec comme en-tête :

```
header("Content-Type: text/xml");
include("donnees.php");
```

et le contenu qui génère un document xml à partir des informations renvoyée par la fonction afficherPersonnage(\$personnage) où \$personnage est le nom du personnage choisi.

Créer une page JS vierge nommée Ex02.js qui contient:

- une fonction `getXMLHttpRequest()` qui retourne un objet `xhr`;
- la fonction `request(callback)` qui:
 1. initialise l'objet `xhr`,
 2. teste son exécution avec gestion du loader,
 3. traite son envoi vers une page PHP: `pageServeur.php` avec le nom du personnage choisi avec la méthode GET.
 4. traite le retour d'information : `callback(xhr.responseXML)`;
- la fonction `readData(oData)` où `oData` est objet xml qui utilise `oData.getElementsByTagName('race')[0].firstChild.data` pour lire un nœud. La fonction créer dynamiquement un script et appelle la page `DSLPersonnage.php` en passant les paramètres avec la méthode GET.

3. Ex3 – DSL & JSON

Reprendre le contenu de l'Ex2.

Sauvegarder l'exercice dans un dossier nommé: Ex3 – DSL & JSON.

Titrer la page avec ce même nom.

Le fichier `donnees.php` reste inchangé.

Les fichiers `db.php` et `config.ini` vous ont été donnés.

Modifier le fichier `DSLPersonnage.php` qui reprend l'en-tête suivant: `<?php header("Content-type: text/javascript"); ?>` et les instructions javascript qui permettent de garnir avec le DOM les champs texte.

Modifier le fichier `pageServeur.php` avec le code suivant :

```
include("donnees.php");

$personnage = (isset($_GET["perso"]))?
htmlspecialchars($_GET["perso"]) : NULL;
if($personnage) echo
json_encode(afficherPersonnage($personnage));
else echo "Inconnu";
```

Créer une page JS vierge nommée Ex03.js qui contient:

- une fonction `getXMLHttpRequest()` qui retourne un objet `xhr`;
- la fonction `request(callback)` qui:
 1. initialise l'objet `xhr`,
 2. teste son exécution avec gestion du loader,
 3. traite son envoi vers une page PHP: `pageServeur.php` avec le nom du personnage choisi avec la méthode GET.
 4. traite le retour d'information : `callback(xhr.responseText)`;
- la fonction `readData(oJson)` où `oJson` est objet json qui utilise `JSON.parse(oJson.substring(1,oJson.length-1))`; pour lire les infos retournée sous forme de tableau associatif.

La fonction créer dynamiquement un script et appelle la page `DSLPersonnage.php` en passant les paramètres avec la méthode GET.

4. Ex4 – DSL & Cookie Loading

Reprendre le contenu de l'Ex3.

Sauvegarder l'exercice dans un dossier nommé: Ex4 – DSL & Cookie Loading.

Titre la page avec ce même nom.

Le fichier `donnees.php` reste inchangé.

Les fichiers `db.php` et `config.ini` vous ont été donnés.

Le fichier `DSLPersonnage.php` reste inchangé.

Le fichier `pageServeur.php` reste inchangé:

Créer la page `DSLlangue` qui contient:

- l'en-tête suivant: `<?php header("Content-type: text/javascript"); ?>`
- le tableau
`$tabLibelle=array("Français"=>array("Race","Classe","Provenance"),"Neerlandais"=>array("Race","klasse","Van"),"Anglais"=>array("Race","Class","F rom"));`
- la mise à jour des intitulés.

Créer la page `CookiesImageLoading.php` qui:

- utilise l'en-tête: `header("Content-type: image/gif");`
- crée le cookie s'il n'existe pas :
`if(isset($_GET['langue']))
 setcookie("langue", $_GET['langue'], time()+3600);`
- charge l'image: `header("Location: white.gif");`

Créer une page JS vierge nommée Ex04.js qui contient:

- le contenu de Ex03.js;
- la fonction send(langue) qui:
 1. crée un objet Image: `var oImg = new Image();`
 2. place une fonction anonyme sur l'objet lors de son chargement qui crée un script dynamique en appelant la page DSLLangue.php en passant la langue choisie et en utilisant `cleanString(getCookie("langue"))`.
 3. indique la source de l'image en appelant la page CookiesImagesLoading.php et en passant la langue choisie en paramètre avec la méthode GET.
- La fonction getCookie(sName) qui gère une expression régulière:

```
var oRegex = new RegExp("(?:; )?" + sName + "=(^[^;]*)"?");
if (oRegex.test(document.cookie))
    return decodeURIComponent(RegExp["$1"]);
else return null;
```
- La fonction cleanString(sString) qui remplace des caractères dans une chaîne:

```
return sString.replace(/(\+)/g, " ");
```

AJAX - Exercice 4 - DSL & Cookies Loading	
Français - Néerlandais - Anglais	
Aragorn ▾	
Race	<input type="text" value="Humain"/>
klasse	<input type="text" value="Rodeur"/>
Van	<input type="text" value="Fondcombe"/>
	