

Développement d'un site web dynamique (UAA12)

Exercices : cookies et sessions

Objets d'apprentissage :

- ✓ Elaborer un formulaire sur la base d'une structure donnée.
- ✓ Référencer un site.
- ✓ Intégrer du contenu multimédia.
- ✓ Construire une page WEB dynamique à l'aide du langage Javascript.
- ✓ Construire une page WEB dynamique à l'aide du langage PHP.
- ✓ Concevoir un formulaire répondant aux exigences d'un cahier des charges.
- ✓ Dynamiser un site WEB exclusivement à l'aide du langage Javascript.
- ✓ Associer des balises HTML de formulaires à leur sémantique.
- ✓ Décrire le rôle d'un cookie.
- ✓ Décrire le rôle du référencement en ligne.
- ✓ Enumérer les fonctionnalités du langage Javascript.
- ✓ Caractériser les attaques WEB les plus courantes.

Exercice 1 : cookies colorés

L'objectif de cet exercice est de proposer à l'utilisateur une page HTML personnalisable (couleur de texte et couleur de fond). Les préférences de personnalisation seront stockées dans des cookies.

Etape 1

Télécharge la page HTML disponible sur Classroom. Observe attentivement le code et ainsi que l'apparence de la page.

Etape 2

Ajoute le **script PHP** adéquat afin que le bouton « Mise à jour » ajoute deux cookies (`couleurTexte` et `couleurFond`) contenant la couleur de texte et la couleur de fond entrées par l'utilisateur (les données sont envoyées au serveur via la méthode POST).

Teste ton code en soumettant le formulaire avec deux couleurs de ton choix puis en consultant les cookies avec l'instruction `document.cookie` dans la console JavaScript.

```
>> document.cookie  
← "couleurTexte=red; couleurFond=blue"
```

Les navigateurs web (Firefox, Chrome, etc.) proposent tous une fonctionnalité permettant de visualiser les cookies stockés par un site internet particulier.

Voici les étapes à suivre pour visualiser les cookies d'un site :

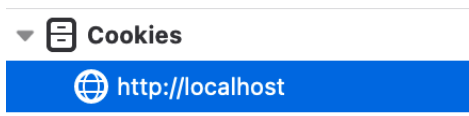
1. Rends-toi dans l'inspecteur
2. Clique sur l'onglet « Stockage »



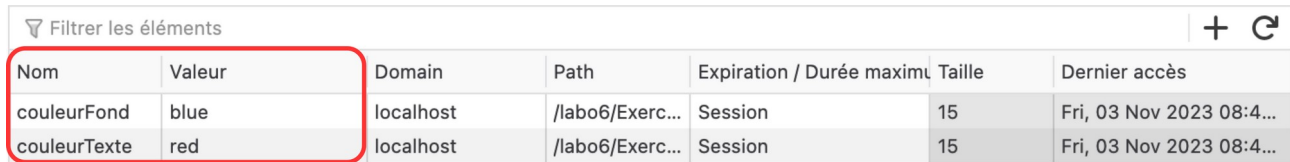
3. Sur le coté gauche, clique sur l'onglet « Cookies »



4. Sélectionne le site web concerné (ici localhost)



5. Un tableau contenant toutes les informations sur les cookies stockés apparaît alors



Nom	Valeur	Domain	Path	Expiration / Durée maximum	Taille	Dernier accès
couleurFond	blue	localhost	/labo6/Exerc...	Session	15	Fri, 03 Nov 2023 08:4...
couleurTexte	red	localhost	/labo6/Exerc...	Session	15	Fri, 03 Nov 2023 08:4...

Etape 3

Ajoute le **script JavaScript** adéquat afin que le bouton « Rafraichir » affiche le contenu des cookies dans le paragraphe identifié par `pCookies`.

Etape 4

Ajoute un **script PHP** qui initialise les variables `$couleurTexte` et `$couleurFond` contenant respectivement la couleur de texte et la couleur de fond du paragraphe. Les valeurs par défaut (c'est-à-dire lorsque l'utilisateur arrive pour la première fois sur la page) devront être noir pour la couleur du texte et blanc pour la couleur de fond.

Voici l'algorithme écrit en pseudo-code (la logique est la même pour la couleur de fond) :

```
if (Une donnée POST est reçue pour la couleur du texte)
// Cas où l'utilisateur envoie le formulaire
Stocker cette donnée dans un cookie
Initialiser la variable couleurTexte avec cette donnée
else
if (il existe un COOKIE pour la couleur du texte)
// Cas où l'utilisateur arrive sur la page
// et qu'il a déjà envoyé le formulaire précédemment
Initialiser la variable couleurTexte avec ce cookie
else
// Cas où l'utilisateur arrive sur la page pour la première fois
Initialiser la variable couleurTexte avec la valeur par défaut
```

Attention : veille à placer les instructions au bon endroit dans ton code !

Etape 5

Ajoute deux autres petits **scripts PHP** qui écrivent les valeurs de `$couleurTexte` et de `$couleurFond` au bon endroit dans la définition du CSS.

Comme pour le HTML, il est tout à fait possible de rendre le CSS dynamique grâce au PHP.

Ainsi, il suffit d'ajouter une balise PHP directement dans la balise `<style>`. Voici un exemple :



```
<style>

#monPara {
    color : <?php echo "black"; ?>;
}

</style>
```

Une fois terminé, vérifie le fonctionnement de ton code.

Etape 6

Ajoute le **script JavaScript** adéquat de telle sorte que le bouton « Recharger la page » recharge la page (utilise `location.reload()`).

Etape 7

Ajoute les **scripts JavaScript et PHP** adéquats afin que le bouton « Supprimer les cookies » supprime les cookies actuellement stockés.

Pour cela, le client (JavaScript) doit envoyer l'instruction au serveur (PHP) de supprimer les cookies. Cette information peut être transmise à l'aide de la méthode GET. Ainsi, lorsque le bouton est cliqué, l'utilisateur est redirigé vers `cookies.php?delete=true` (tu peux utiliser une autre clé si tu préfères). Lorsque le serveur reçoit la clé « delete » dans `$_GET`, il est informé qu'il doit supprimer les cookies.

Exercice 2 : panier

Dans cet exercice, tu vas mettre en place une page WEB permettant à un client de choisir plusieurs articles à acheter. Le contenu du panier sera conservé sur le serveur, en utilisant les données de session.

Etape 1

Télécharge le fichier `achats.html` ainsi que le fichier `games.php` disponibles sur Classroom. Observe attentivement le code et ainsi que l'apparence de la page.

Etape 2

Ajoute le CSS adéquat (au fur et à mesure de ton avancement) afin d'obtenir le résultat suivant :

Jeux de rôle

Autres jeux

Passer commande

Vider votre panier

Commande : Neverwinter Nights 2 Complete ajouté !

Jeu	Prix	Genre	Éditeur	
Planescape: Torment	9.09€	RPG	Black Isle Studios	Commander
Neverwinter Nights 2 Complete	22.78€	RPG	Obsidian Entertainment	Commander
Legend of Grimrock II	21.79€	RPG	Almost Human	Commander
Shadowrun returns	13.69€	RPG	Harebrained Schemes	Commander
Pillars of Eternity: Hero Edition	41.99€	RPG	Obsidian Entertainment	Commander
Arx Fatalis	5.49€	RPG	Arkane Studios	Commander
Wasteland 2: Director's Cut	39.99€	RPG	inXile Entertainment	Commander
Gothic 3: Enhanced Edition	9.09€	RPG	Piranha Bytes	Commander
Baldur's Gate II: Enhanced Edition	18.19€	RPG	Beamdog	Commander

Votre panier :

- Gothic 3: Enhanced Edition (9.09 €)
- Shadowrun returns (13.69 €)
- Neverwinter Nights 2 Complete (22.78 €)

Total : 45.56 euros

Etape 3

Fais en sorte que ton document `achats.html` porte maintenant l'extension `.php`

Ajoute un premier script PHP qui lie `achats.php` avec `games.php` de telle manière que les deux tableaux `$jeuxNonRPG` et `$jeuxRPG` puissent être utilisés dans `achats.php`.

Etape 4

Ajoute un second script PHP qui insère l'ensemble jeux RPG dans l'interface (c'est à dire dans le `<div id="items">`) sous la forme illustrée à l'étape 2 (ne te préoccupe pas du bouton « Commander »).

Etape 5

Modifie le script PHP pour que l'utilisateur puisse choisir la liste des jeux affichée en entrant les URL suivante :

- `achat.php?games=rpg` : affiche la liste des jeux RPG
- `achat.php?games=nonrpg` : affiche la liste des jeux non RPG
- `achat.php` : affiche la liste des jeux RPG (liste par défaut)

Associe les URL correspondantes aux boutons « Jeux de rôle » (RPG) et « Autres jeux » (non RPG).

Etape 6

Occupe-toi désormais des boutons « Commander » et fais en sorte qu'un clic sur l'un deux lance une redirection (`document.location`) vers l'URL : `achat.php?games=X&order=Y`

- X correspond au type de la liste de jeux actuellement affichée (rpg ou nonrpg)
- Y correspond au nom du jeu commandé (ex. Arx Fatalis)

Attention : le nom du jeu peut contenir des espaces ou d'autres caractères problématiques. Il faut donc *encoder* le nom du jeu pour qu'il puisse être inséré dans l'URL. Pour cela, utilise la fonction `urlencode()` : <https://www.php.net/manual/fr/function.urlencode.php>

Voici un exemple d'URL qui devrait être générée :

`achat.php?games=rpg&order=Baldur's+Gate+II%3A+Enhanced+Edition`

Etape 7

Lorsqu'un jeu est commandé, fais en sorte d'afficher un simple message de confirmation de la manière suivante :

Commande : Baldur's Gate II: Enhanced Edition ajouté !				
Jeu	Prix	Genre	Éditeur	
Planescape: Torment	9.09€	RPG	Black Isle Studios	Commander
Neverwinter Nights 2 Complete	22.78€	RPG	Obsidian Entertainment	Commander
Legend of Grimrock II	21.79€	RPG	Almost Human	Commander
Shadowrun returns	13.69€	RPG	Harebrained Schemes	Commander
Pillars of Eternity: Hero Edition	41.99€	RPG	Obsidian Entertainment	Commander
Arx Fatalis	5.49€	RPG	Arkane Studios	Commander
Wasteland 2: Director's Cut	39.99€	RPG	inXile Entertainment	Commander
Gothic 3: Enhanced Edition	9.09€	RPG	Piranha Bytes	Commander
Baldur's Gate II: Enhanced Edition	18.19€	RPG	Beamdog	Commander

Etape 8

Il est temps de passer au vif du sujet : les sessions !

Les commandes effectuées doivent être stockées sous la forme d'un tableau dans les informations de session. La clé pour y accéder sera « order ».

Dans chacune des cases du tableau, on trouvera un tableau associatif reprenant le nom du jeu (clé « name ») ainsi que le prix (clé « price »).

Voici un exemple de structure :

Clé	Valeur	
order	Clé	Valeur
	0	Clé
		Valeur
		name
		price
	1	name
		price
		Clé
		Valeur
		name
		price

Pour manipuler les informations de session, il faut activer la commande `session_start()` avant le début du contenu HTML. Le script PHP contenant cette instruction doit donc se trouver avant toute balise HTML.

Un fois la commande `session_start()` activée, ajoute au tableau `$_SESSION['order']` une nouvelle case contenant le nom et le prix du jeu.

Attention, n'oublie pas d'initialiser le tableau référencé par la clé `order` !

Etape 9

Ajoute un script PHP qui parcourt la liste des jeux présent dans `$_SESSION['order']` et qui l'affiche sous la forme illustrée à l'étape 2.

Etape 10

Pour terminer, fais en sorte que si l'utilisateur entre l'URL `achat.php?clear=true`, le panier soit vidé. Pour cela, il suffit de supprimer l'ensemble des informations de session à l'aide de `session_destroy()`

Etape 11 (dépassement)

Imaginons que l'utilisateur entre l'URL suivante : `achat.php?games=rpg&order=The+Witcher`

« The Witcher » ne fait pas partie des jeux en vente. Dès lors, ton programme PHP risque très probablement d'afficher un résultat étrange (quel est son prix ?).

Mets à jour ton programme PHP pour que le message « Jeu introuvable ! » s'affiche à l'écran dans le cas où le jeu commandé n'existe pas. Evidemment, ce jeu ne doit pas s'ajouter dans le panier.

Exercice 3 : authentification

Exercice tiré de <https://www.chiny.me/exercice-authentification-7-15.php>

L'objectif de cet exercice est d'utiliser les sessions afin de mettre en place un système d'authentification. On te demande de créer trois pages PHP :

- `login.php` : authentifie l'utilisateur ; elle contient un formulaire renfermant une zone de texte, une zone de mot de passe et un bouton d'envoi.
- `session.php` : représente la page à accès limité ; aucun visiteur n'a le droit de voir son contenu s'il n'a pas été authentifié par la page `login.php`.
- `deconnexion.php` : permet de déconnecter le client (détruire la session) et rediriger le navigateur vers la page `login.php`.

Si le client tente d'accéder directement à la page `session.php` alors qu'il n'est pas authentifié, il sera aussitôt redirigé vers la page `login.php`. S'il fournit un bon login et un bon mot de passe alors il sera redirigé vers la page `session.php` qu'il a désormais le droit de consulter.

Pour simplifier, nous allons définir statiquement le bon login qui est "user" et le bon mot de passe qui est "1234".



Pour rediriger le navigateur automatiquement vers une autre page on peut le faire en HTML, Javascript ou en PHP. Cependant, il est préférable de le faire en PHP grâce à la fonction `header()`. Par exemple, si on veut rediriger le navigateur vers la page `session.php` alors on utilise `header("location:session.php")`.

Notez que la fonction `header()` est une fonction entête comme c'est le cas pour `session_start()`. Elle doit donc être placée dans un script PHP qui se trouve avant `<!DOCTYPE html>`.

Références

Les présents exercices ont été élaborés à l'aide des ressources suivantes :

- Forbes, A. (2013). The Joy of PHP: A Beginner's Guide to Programming Interactive Web Sites.
- Vaswani, V. (2021). PHP A Beginner's Guide.
- Cours de « Développement d'applications WEB » (2018), Hénallux