

## Construire un site dynamique, partie 1

**Vous avez obtenu la note de 0 sur 11**

Le 07/03/2018 à 09:16 vous avez obtenu la note de 3 sur 11

Les fichiers JS d'un site web sont généralement situés...

### Explication:

Les fichiers JS, CSS, HTML et les images sont placés dans un sous-dossier du dossier public.

A la racine du serveur web

---

VDans un sous-dossier de `public`

---

En parallèle avec les librairies PHP utilisées par le site

Que signifie l'acronyme 'DRY' ?

### Explication:

Don't Repeat Yourself : il faut au maximum éviter de se répéter. Attention à ne pas tomber dans l'excès inverse, s'il ne doit pas se répéter, le code doit aussi rester clair.

VDon't Repeat Yourself : Éviter au maximum de se répéter

---

Do Repeat Yourself : Se répéter suffisamment pour que le code soit plus clair

---

---

## Don't Rage, Yolo

Comment factoriser le code suivant :

```
<ul>
  <li><?php echo $person['firstname'] ?></li>
  <li><?php echo $person['lastname'] ?></li>
  <li><?php echo $person['civ'] ?></li>
</ul>
```

### Explication:

Le code A n'évite pas la répétition, le code B ne veut rien dire.

```
<?php
  $firstname = $person['firstname'];
  $lastname = $person['lastname'];
  $civ = $person['civ'];
?>
<ul>
  <li><?php echo $firstname ?></li>
  <li><?php echo $lastname ?></li>
  <li><?php echo $civ ?></li>
</ul>
```

---

---

```
<ul>
  <li>
    <?php
      for($i = 0; $i < count($person); $i++) {
        $person[$i];
      }
    ?>
  </li>
</ul>
```

---

```
V<ul>
  <?php foreach($persons as $person) ?>
    <li>
      <?php echo $person ?>
    </li>
  <?php endforeach ?>
</ul>
```

Comment éviter les incohérences dans le parcours de l'utilisateur ?

### Explication:

Avant de commencer à développer une fonctionnalité, c'est une bonne pratique de concevoir des maquettes.

VConcevoir des maquettes avant de commencer à coder

---

Fractionner au maximum les étapes d'un scénario

---

---

Organiser hiérarchiquement les information sur la page

Lorsque je développe une nouvelle fonctionnalité, par exemple un formulaire de contact...

### Explication:

Le déroulement d'un cas d'utilisation ne dépend généralement pas d'un seul fichier.

Je crée un nouveau fichier et j'y place la totalité de la nouvelle logique ainsi que de l'affichage

---

VJ'ajoute la logique dans les fichiers existants, et si nécessaire je crée de nouveaux fichiers

---

J'évite au maximum de créer de nouveaux fichiers, pour des raisons de clarté

Soit la page suivante :

### *view-user.php*

```
<?php
    if (isset($_GET['user']) && !empty($_GET['user']) &&
ctype_digit($_GET['user'])) {
        $user = $_GET['user'];
        $fullUser = getUserFromDb($user); // Récupère les données de
l'utilisateur en fonction du login

        echo $fullUser['firstname'] . ' ' . $fullUser['lastname'];
    }
    else {
```

```
echo 'Erreur : paramètre invalide';  
}
```

Que va renvoyer le serveur à cette requête : `view-user.php?user=belphegor` ?

### Explication:

La fonction `ctype_digit` renverra ici `false`, car la variable ne contient pas de caractères numériques. Il n'était pas nécessaire de tester le contenu de `$_GET['user']` avec `ctype_digit`.

Le prénom et le nom de l'utilisateur ayant le login "belphegor"

---

VL a chaîne "Erreur : paramètre invalide"

---

Une erreur PHP

Les paramètres passés dans l'URI :

### Explication:

De fait, tous les paramètres passés dans l'URI sont des chaînes de caractères, au même titre que l'URI elle-même. C'est bien le serveur qui décompose l'URI pour fournir au script PHP les bons paramètres, mais il ne change absolument pas leur type.

VSont tous des chaînes de caractères

---

---

Sont transformés en chaînes de caractères par Apache

---

Ne peuvent pas être des chaînes de caractères.

Comment passer plusieurs paramètres à un script ?

**Explication:**

La bonne syntaxe est un point d'interrogation pour le premier paramètre uniquement, puis des symboles `&`

```
/script.php?s=word?match=all
```

---

```
v/script.php?s=word&match=all
```

---

```
/script.php?s=word/?match=all
```

Quel type de relation peut-on envisager entre une entité `Produit` et une entité `Commande` ?

**Explication:**

Un produit peut se trouver dans plusieurs commandes, et une commande peut contenir plusieurs produits. On imagine que l'association porteuse comprendra un attribut `quantite`.

1:1

---

---

1:n

---

Vn:n

Qu'est-ce qu'une clé primaire composite ?

**Explication:**

Une clé primaire composite n'est pas nécessairement composée de clés étrangères.

V Une clé primaire composée de plusieurs attributs

---

Une clé primaire composée de plusieurs clés étrangères

---

Une clé étrangère composée de plusieurs clés primaires

Quelle relation peut-on imaginer entre une entité **Livre** et une entité **Etagere** ?

**Explication:**

Une étagère comporte plusieurs livres, mais un livre ne peut être que sur une seule étagère à la fois.

1:1

---

---

V1:n

---

n:n