

## CT JS/PHP : Session 1 - 2h - Sur machine

Téléchargez l'archive **ct-19-1-sujet.tgz** de l'espace Moodle. Après extraction, renommez le répertoire obtenu **nom-prenom** où **nom** et **prenom** sont vos nom et prénom écrits en minuscules et sans espaces (par ex. **delafontaine-jean**). A l'issue de l'examen, archivez ce répertoire (`tar czf nom-prenom.tgz nom-prenom`) et déposez l'archive sur Moodle.

On propose un mini-site affichant un tableau de pays et permettant les opérations suivantes :

- Affichage au choix des noms de pays, de leur code ou de leurs drapeaux (voir figures 1, 2 et 3);
- Filtrage des pays par continent (voir Figure 4).
- Affichage de l'espérance de vie et du régime politique d'un pays en cliquant sur la cellule correspondante du tableau (voir Figure 5).

Le dossier contient les sous-dossiers et fichiers suivants - les fichiers en **rouge** seront à compléter et les fichiers en **bleu** seront automatiquement écrasés par votre script **country.php** (des copies - fichiers **.bak** - vous sont fournies) :

- **data/country-codes.xml** : les codes des pays au format XML.
- **data/l3info\_ct\_19\_1\_pays.sql** : le script SQL pour reconstituer la base de données.
- **src/country-codes.js** : les codes des pays sous forme d'un tableau d'objets JS.
- **src/country-codes.js.bak** : copie du fichier précédent.
- **src/country-continents.js** : les continents des pays sous forme d'un tableau d'objets JS.
- **src/country-continents.js.bak** : copie du fichier précédent.
- **src/country-flags.js** : les drapeaux des pays sous forme d'un tableau d'objets JS.
- **connexpdo.inc.php** : une fonction de connexion à la base de données par l'API PDO.
- **country.css** : feuille de style importée par la page du site.
- **country.js** : fichier définissant les gestionnaires d'évènements de la page du site.
- **country.php** : fichier générant la page du site et les fichiers **src/country-codes.js** et **src/country-continents.js**.
- **features.php** : fichier renvoyant les données d'un pays en réponse à requête Ajax.
- **gabarit.php** : le gabarit pour la génération de la page du site.
- **utils-obf.js** : un fichier obfusqué de fonctions JS.
- **utils.js** : un fichier décrivant les différentes fonctions obfusquées implémentées dans **utils-obf.js**.
- **test.html** : un fichier pour tester votre script **features.php** et vos gestionnaires d'évènements dans **country.js**.

Le sujet comporte deux parties (PHP et JavaScript). Les exercices peuvent tous être faits indépendamment les uns des autres.

## Partie PHP

Créez sous phpMyAdmin une base de données MySQL de nom `l3info_ct_19_1_pays` en choisissant le moteur InnoDB et le jeu d'interclassement `utf8mb4_bin`. Importez y le script `l3info_ct_19_1_pays.sql`. L'import crée les tables `COUNTRY` et `FEATURES`.

`COUNTRY` est dotée des colonnes suivantes :

- `ID` : clé primaire auto-incrémentée.
- `NAME` : nom du pays (`varchar(44)`).

`FEATURES` est dotée des colonnes suivantes :

- `ID_COUNTRY` : identifiant d'un pays (clé étrangère sur `COUNTRY.ID`).
- `CONTINENT` : le continent du pays (`varbinary(13)`).
- `EXPECTANCY` : l'espérance de vie du pays (`varchar(4)`).
- `GOVERNMENT` : le régime politique du pays (`varchar(44)`).
- `LANGUAGES` : l'une des langues parlées dans le pays (`varchar(25)`).

`FEATURES` répertorie toutes les langues parlées dans un pays. Autrement dit, plusieurs enregistrements de `FEATURES` peuvent être associés à un même enregistrement de `COUNTRY`. Adaptez le script `connexpdo.inc.php` pour vous connecter au SGBD MySQL.

**Exercice 1.** Le script `country.php` produit la page du site en important le gabarit `gabarit.php`. Ne manquez dans ce gabarit que le code PHP générant le tableau HTML des pays. Complétez les deux fichiers pour produire ce tableau HTML en respectant les consignes suivantes :

- Dans `country.php`, extrayez et convertissez les enregistrements de la table `COUNTRY` sous la forme d'un tableau associatif. Chaque élément de ce tableau correspond à un enregistrement de `COUNTRY` et a pour clé le champ `ID` et pour valeur le champ `NOM` du pays (par ex. `1 => "Afghanistan"`).
- Dans `gabarit.php`, générez à partir de ce tableau associatif le tableau HTML en utilisant 16 lignes et 16 colonnes et en omettant les 10 dernières cellules (il n'y a que 246 pays). Chacune des 246 premières cellules correspond à un pays et obéit au format `<td id="France">France</td>` où l'identifiant HTML est le nom du pays lui-même.

**Exercice 2.** Complétez le script `country.php` afin de lire et convertir les données du fichier XML `./data/country-codes.xml` en un tableau indexé numériquement du type :

```
[..., ["country" => "France", "abbreviation" => "FR"], ...].
```

où chaque élément est un tableau associatif représentant une paire pays-abbréviation (code). Créez ensuite la chaîne de caractères résultant de la concaténation, dans l'ordre, de :

- la chaîne `const country_codes=`
- la chaîne renvoyée par l'appel de la fonction `json_encode` sur ce tableau
- et le caractère point-virgule ;

Sauvegardez cette chaîne, par instruction PHP, dans le fichier `./src/country-codes.js` (ce qui écrasera le fichier existant). Le fichier obtenu doit être identique à la copie `./src/country-codes.js.bak` aux espaces et retours à la ligne près.

**Exercice 3.** Complétez le script `country.php` afin d'extraire des tables `COUNTRY` et `FEATURES` toutes les paires `country - continent` (sans doublons) et les convertir en un tableau indexé numériquement du type :

```
[..., ["country" => "France", "continent" => "Europe"], ...].
```

où chaque élément est un tableau associatif représentant une paire pays-continent. Comme dans l'exercice précédent, vous sauvegarderez l'encodage JSON de ce tableau en y ajoutant les fragments `const country_continents=` et un point-virgule à la fin dans le fichier `./src/country-continents.js`. Le fichier obtenu doit être identique à la copie `./src/country-continents.js.bak` (modulo espaces et retours à la ligne).<sup>1</sup>

1. Notez que dans la table `FEATURES`, le continent enregistré pour certains pays (par ex. Bouvet Island) est la chaîne de caractères `"null"` : vous traiterez ces pays comme les autres en conservant cette valeur `"null"` dans le fichier JSON.

**Exercice 4.** Un clic sur une cellule du tableau HTML doit afficher l'espérance de vie et le régime politique du pays correspondant dans l'en-tête de la page. Le gestionnaire de ces événements, qui sera à développer dans la partie Javascript, récupère ces données en soumettant le nom du pays par requête HTTP POST ou GET (selon votre choix) au script **features.php**. Le paramètre soumis suit le format `country_name="France"`.

Complétez le script **features.php** afin de récupérer le nom du pays soumis pour extraire ensuite de la base de données son espérance de vie et son régime politique puis renvoyer ces données au format `{"expectancy": "78.8", "government": "Republic"}`. Utilisez le formulaire figurant en bas de la page **test.html** pour tester **features.php** avec différents pays et la méthode HTTP POST ou GET que vous avez choisie.

On notera que, pour certains pays (par exemple Bouvet island), les champs `EXPECTANCY` et `GOVERNMENT` dans la table `FEATURES` ont pour valeur la chaîne de caractères `"null"`. Dans ce cas, **features.php** renverra cette chaîne `"null"` telle quelle. Par exemple, la réponse à la requête `country_name="Bouvet+Island"` sera `{"expectancy": "null", "government": "null"}`.

### Partie JavaScript

L'objectif est ici de développer dans le fichier **country.js** les gestionnaires d'événements de la page web réagissant au choix d'options dans le menu déroulant et au survol des cellules du tableau HTML. Le gestionnaire des boutons radio est quant à lui déjà implémenté dans le fichier obfusqué **utils-obf.js** et n'a pas à être développé.

Vos gestionnaires d'événements doivent utiliser l'objet `data` qui est défini dans le fichier **country.js** et qui contient les données relatives aux pays, à leurs continents, leurs codes et leurs drapeaux. Ces données sont extraites et reformattées à partir des constantes Javascript déclarés dans les fichiers du dossier **src** en faisant appel à différentes fonctions pré-implémentées dans le fichier obfusqué **utils-obf.js**.

Afin de travailler avec des données correctes dans cette partie, écrasez les fichiers `./src/country-codes.js` et `./src/country-continents.js` avec leurs copies de sauvegarde `./src/country-codes.js.bak` et `./src/country-continents.js.bak` si vous les avez modifiés. De plus, utilisez le fichier **test.html** pour tester vos gestionnaires (ou bien le fichier **country.php** si vous l'avez implémenté correctement en faisant la partie PHP).

**Exercice 5.** Complétez le corps de l'IIFE `handleSelectors` dans **country.js**. Lorsqu'un continent est sélectionné dans le menu déroulant, cette fonction doit identifier les pays dudit continent, rendre visible le contenu des cellules correspondantes dans le tableau et masquer les autres. Le filtrage des pays se fait en appelant la fonction `selectCountries` qui est décrite dans le fichier **utils.js** et implémentée dans **utils-obf.js**. Le masquage/affichage des cellules se fait avec la propriété CSS `visibility`. On notera que les cellules du tableau HTML ont pour identifiant les noms de pays et que l'option `*` doit permettre de sélectionner tous les pays.

**Exercice 6.** Complétez le corps de l'IIFE `handleHeader` dans **country.js**. A chaque clic sur une cellule du tableau, cette fonction doit afficher dans l'en-tête de la page le nom du pays correspondant, son espérance de vie et son régime politique. Le formatage du texte attendu est illustré en Figure 5. Les informations sur un pays seront obtenues en soumettant :

- le nom du pays en suivant le format `country_name="France"`
- au script **features.php** (voir l'exercice 4 de la partie PHP)
- par requête Ajax en HTTP POST ou GET
- en utilisant au choix l'API `XMLHttpRequest` ou l'API `fetch`.

**features.php** renverra alors les données au format `{"expectancy": "78.8", "government": "Republic"}` (cas de la France). Si **features.php** renvoie la chaîne de caractères `"null"` comme valeur de `"expectancy"` ou `"government"`, vous afficherez le symbole `?` à la place de cette chaîne.

[illegible]

FIGURE 2 – Codes des pays

FIGURE 3 – Drapeaux des pays

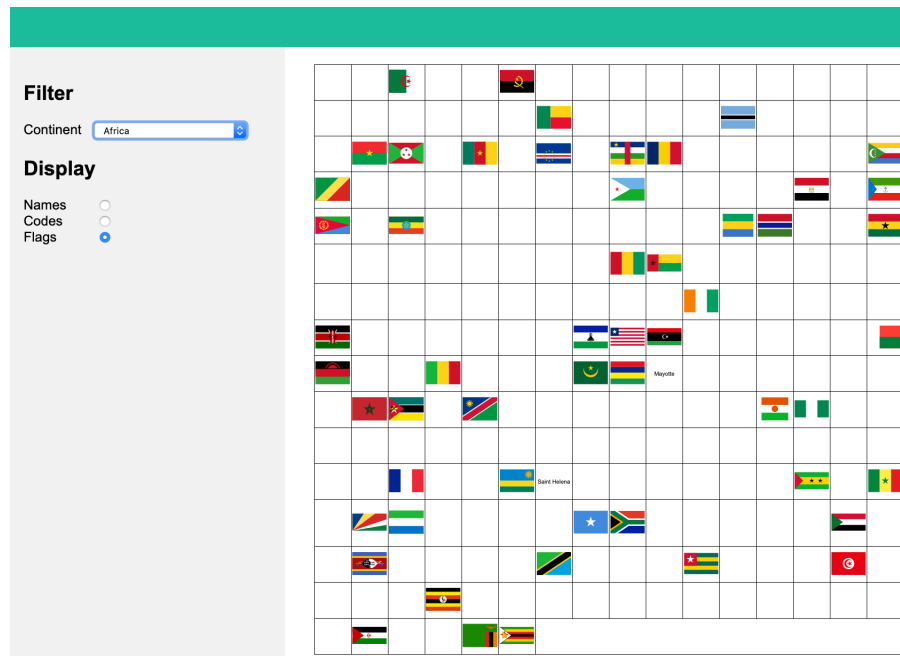


FIGURE 4 – Filtrage par continent



FIGURE 5 – Informations sur un pays par clic sur la cellule correspondante