

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

Téléchargez l'archive **ct-18-3-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.

**Exercice 1.** Créer un script **Ville.php** contenant une classe **Ville** tel que le chargement de **test-ville.php** affiche la page web suivante :

```
Saumur / Pays de la Loire / 27486 /
Toulon / Provence-Alpes-Côte d'Azur / 165514 / Préfecture
Laval est une ville palindrome
```

Respecter les conditions suivantes :

- **Ville** implémente l'interface définie dans **IVille.php** ;
- les méthodes **setNom(\$nv)** et **setRegion(\$nr)** suppriment tout espace apparaissant en début ou fin de la chaîne argument ;
- **setNom(\$nv)** et **setRegion(\$nr)** remplacent par un seul espace toute séquence interne de plus d'un espace dans la chaîne argument ;
- **setNom(\$nv)** lance une exception si la chaîne **\$nv** est un palindrome indépendamment de la casse (par exemple, *Laval*).

**Exercice 2.** On considère une classe **Population** dotée

- d'une propriété privée représentant un tableau d'instances de **Ville** initialement vide,
- d'un constructeur sans arguments,
- d'une méthode **importerXML(\$f)** qui initialise le tableau des villes en lisant les données d'un fichier XML **\$f** validant **population.dtd**,
- d'une méthode **importerJSON(\$f)** qui initialise le tableau des villes en lisant les données d'un fichier JSON **\$f**,
- et d'une méthode **afficher()** qui génère le tableau HTML correspondant au tableau des villes de l'instance.

Créer un script **population.php** qui implémente cette classe mais avec une seule des deux méthodes d'importation (au choix) de sorte que le chargement de **test-population-xml.php** ou de **test-population-json.php** affiche la page HTML5 suivante :

Nom	Region	Population	Préfecture
Saumur	Pays de la Loire	27486	non
Nantes	Pays de la Loire	282047	oui
Avignon	Provence-Alpes-Côte d'Azur	89592	oui
Marseille	Provence-Alpes-Côte d'Azur	850602	oui
Toulon	Provence-Alpes-Côte d'Azur	165514	oui

**Exercice 3.** Ajouter une méthode **trier()** à la classe **Population** qui trie le tableau d'instances de **Ville** de sorte que les villes apparaissent par ordre alphabétique décroissant des noms de régions, et, pour chaque région, par population décroissante. Amender la méthode **afficher** pour afficher le tableau ainsi trié.

Nom	Region	Population	Préfecture
Marseille	Provence-Alpes-Côte d'Azur	850602	oui
Toulon	Provence-Alpes-Côte d'Azur	165514	oui
Avignon	Provence-Alpes-Côte d'Azur	89592	oui
Nantes	Pays de la Loire	282047	oui
Saumur	Pays de la Loire	27486	non

**Exercice 4.** Créer une base de données MySQL nommée `l3info_ct_18_3_population` et composée d'une table `VILLE` contenant les champs `id_ville` (clé primaire), `nom_ville` (nom de la ville), `nom_region` (nom de la région), `population` (nombre d'habitants) et `prefecture` (statut de préfecture).

**Exercice 5.** Ajouter la méthode `inserer($pdo)` à la classe `Ville` qui prend en entrée une instance `$pdo` de la classe `PDO` et crée un enregistrement dans la table `VILLE` correspondant aux données de l'objet `Ville`.

**Exercice 6.** Ecrire un fichier HTML5 `saisir.html` qui affiche un formulaire permettant de saisir les données d'une ville :

Ville	<input type="text"/>
Région	<input type="text"/>
Population	<input type="text"/>
Préfecture	<input type="radio"/> oui <input checked="" type="radio"/> non
<input type="button" value="Ajouter"/>	

La soumission du formulaire invoque un script **ajouter.php** par méthode HTTP POST. Ce script crée une instance de `Ville` et en invoque la méthode `inserer($pdo)` pour créer l'enregistrement correspondant dans la table `VILLE`.

**Exercice 7.** Ecrire un script **inserer-bdd.php** qui insère dans la table `VILLE` les données du fichier **population.xml** ou bien du fichier **population.json**.

**Exercice 8.** Ecrire un script **rechercher.php** qui affiche un champ texte permettant d'effectuer une recherche dans la table `VILLE` par nom de région. Les villes dont le nom de région contient la chaîne saisie par l'utilisateur seront affichées sous forme de liste HTML. Vous effectuerez le filtrage par script PHP, et non pas par requête SQL.

Chercher par région :	<input type="text" value="Azur"/>
<input type="button" value="rechercher"/>	