

Classes

TP 1 du Module 03 – La programmation orienté objet

Avant de démarrer ce TP, il convient d'avoir suivi les vidéos des modules 1 à 3 jusqu'à la séquence sur les classes abstraites.

Durée estimée

2 heures

Énoncé

- Exercice 1

Écrire une classe représentant une ville. Elle doit avoir les méthodes `setNom()` et `setDépartement()` et une méthode affichant « la ville X est dans le département Y ». Cette classe n'a pas de constructeur.

Créer des instances de `Ville`, affecter leurs propriétés et utiliser la méthode d'affichage.

- Exercice 2

Créer une autre classe représentant une ville mais sans les accesseurs et avec un constructeur à la place.

Réaliser les mêmes opérations de création d'instances et d'affichage.

- Exercice 3

Créer une classe nommée `VilleAvecRegion` héritant de la classe `Ville` affichant « la ville X est dans le département Y de la région Z ».

- Exercice 4

Modifier la classe `Ville` avec constructeur pour que l'on puisse connaître la ville ayant le nom le plus long.

- Exercice 5

Créer une classe nommée `Form` représentant un formulaire HTML. Le constructeur doit créer le code d'en-tête du formulaire en utilisant les éléments `<form>` et `<fieldset>`.

Une méthode `setText()` doit permettre d'ajouter une zone de texte.

Une méthode `setSubmit()` doit permettre d'ajouter un bouton d'envoi.

Les paramètres de ces méthodes doivent correspondre aux attributs des éléments HTML correspondants.

La méthode `getForm()` doit retourner tout le code HTML de création du formulaire.

Créer des objets `Form` et y ajouter deux zones de texte et un bouton d'envoi.

Tester l'affichage obtenu.

- Exercice 6

Créer une sous-classe nommée `Form2` en dérivant la classe `Form` de l'exercice précédent.

Cette nouvelle classe doit permettre de créer des formulaires ayant en plus des boutons radio et des cases à cocher. Elle doit donc avoir les méthodes supplémentaires qui correspondent à ces créations.

Créer des objets et tester le résultat.

- Exercice 7

Créer une classe abstraite représentant une personne. Elle déclare les propriétés `nom` et `prenom` et un constructeur.

Créer une classe `Client` dérivée de la classe `Personne` en y ajoutant la propriété `adresse`, une méthode `setcoord()` et la méthode `__toString()` qui retourne les coordonnées complètes de la personne.

Créer une classe `Electeur` dérivée de la même classe abstraite et y ajouter deux propriétés `bureau_de_vote` et `vote` ainsi qu'une méthode `avote()` qui enregistre que la personne a voté dans la propriété `vote`.

Solution

Une solution est proposée pour ce TP sous la forme d'un PDF commenté avec le code associé. Ces éléments sont disponibles dans les ressources à télécharger.