Exercices PHP Objet

Exercices d'introduction

CONTENU

CONSIGNES

Créer un répertoire « php_objet_exos » dédié aux exercices de ce document.

Dans cette série d'exercice, il vous sera demandé de :

- Modéliser des diagrammes UML
- Implémenter ces diagrammes avec PHP

1 – VÉHICULES

Créez un répertoire « Véhicules ». Les fichiers des exercices suivants doivent être placés dans ce répertoire.

VOITURE

Donnez une représentation UML et implémentez avec PHP la classe Voiture décrite ci-dessous :

Une Voiture est caractérisée par une marque et un modèle. Il est également possible de définir le poids du véhicule en kilogrammes mais ce n'est pas obligatoire (dans ce cas, le poids par défaut d'une voiture est 1000 Kg).

Une opération permet de retourner l'ensemble des informations (ex : « Renault Mégane, 750 Kg »).

Tous les attributs sont « protected » et sont disponibles via des accesseurs publics (« getters » et « setters »).

MOTEUR

Une **Voiture** possède toujours un **Moteur**. Chaque moteur est caractérisé par une marque et une vitesse maximale exprimée en kilomètres par heure (km/h). Chaque moteur ne peut-être rattaché qu'à une seule Voiture. Une Voiture peut accepter des Moteur de toutes marques.

Une voiture possède une opération qui permet de retourner la vitesse maximale de la Voiture.
La vitesse maximale de la voiture est calculée selon la vitesse maximale du moteur et le poids de la voiture.

Formule simplifiée de calcul de la vitesse maximale d'une voiture :
Voiture.vitesseMax = Moteur.vitesseMax - (Voiture.poids x 30%).

VOITURE DE COURSE

Une VoitureDeCourse est une Voiture performante. Elle diffère d'une voiture classiques pour 2 raisons :

- Une **VoitureDeCourse** n'accepte que des Moteurs de même marque
  - Une Voiture de course « Renault » accepte des Moteurs « Renault » uniquement.

- La formule de calcul de la vitesse maximale de la Voiture de course est différente :
  - **Voiture.vitesseMax =** Moteur.vitesseMax - (Voiture.poids x 5%).

Une opération permettra de retourner les informations complètes d'une voiture de course :
ex : « Renault F1, 450 Kg. Vitesse max : 317km/h.

2 – PERSONNES

Créez un répertoire « Personnes ». Les fichiers des exercices suivants doivent être placés dans ce répertoire.

PERSONNE

Donnez une représentation UML et implémentez avec PHP la classe Personne décrite ci-dessous :

Une **Personne** est caractérisée par son nom, son prénom et son âge. Les attributs de la classe sont privés ; le nom, le prénom ainsi que l'âge de la personne doivent être accessibles en lecture par des opérations publiques.

Un objet de la classe Personne peut uniquement être créé à partir du nom, prénom et date de naissance. Il est possible de changer le nom d'une personne mais pas son prénom ni sa date de naissance.

CLIENT

Enrichissez la représentation précédente pour prendre en compte ces nouveaux éléments :

Un **Client** est une **Personne** identifiée par un numéro de client et une **Adresse**. Une adresse est caractérisée par un numéro de rue, un nom de rue, un code postal et un nom de commune. Le numéro de client est défini à la construction de l'objet et n'est plus modifiable ensuite.

INTERVENANT

Un **Intervenant** est une **Personne** employée par l'entreprise.

Un intervenant peut perçoit des revenus. Deux types de revenus annuels sont envisagés : d'une part le salaire de l'intervenant et d'autre part toutes les autres sources de revenus. Les deux types de revenus sont représentés par des nombres réels (float). Il est possible de modifier les revenus d'un intervenant. Une opération permet de calculer et retourner les charges que qu'il devra payer en fin d'année. Pour calculer ces charges, on applique une taxe de 20% sur les salaires et une taxe de 15% sur les autres revenus. Le pourcentage des taxes est divisé par 2 si l'intervenant est âgé de plus de 55 ans.

Formule de calcul de base des charges :
charges = (salaire x 20%) + (autres-revenus x 15%).

INTERVENTIONS

Une Intervention permet de mettre en relation un **Client** et un **Intervenant**. Une instance de chacune de ces 2 classes est attendue dans les arguments du constructeur d'un Intervention.

Pour chaque intervention, on doit pouvoir définir et lire la date et heure d'intervention ainsi qu'une description de l'intervention.

--- FIN DU DOCUMENT ---