

TP 5 : Exceptions

Exercice 1

Considérons une classe `Personne` ayant les attributs suivants :

- `num` : un attribut privé de type `int`
- `nom` : un attribut privé de type `string`
- `prenom` : un attribut privé de type `string`

1. Créez la classe `Personne`
2. Définissez les getters et setters de tous les attributs
3. Définissez la méthode `__toString()`.
4. Définissez une méthode `public getNomPrenom() : string` qui retourne le nom concaténé au prénom
5. Dans `index.php`, créez quelques instances de la classe `Personne` et ajoutez les à un tableau associatif `personnes` : la clé étant la concaténation des nom et prénom et la valeur étant l'instance de `Personne`.
6. Dans une boucle `for`, et pour chaque itération, affichez la clé du tableau associatif si l'indice de l'itération est pair, la valeur sinon.

Exercice 2

Considérons la classe `Personne` de l'exercice 1.

1. Créez une classe `ListePersonnes` contenant un attribut `personnes` : tableau associatif comme indiqué dans l'exercice 1.
2. Écrivez une méthode `ajouterPersonne(Personne $personne) : void` qui permet d'ajouter `$personne` au tableau associatif `personnes` : la clé étant la concaténation des nom et prénom et la valeur étant l'instance de `Personne`.
3. Modifiez la méthode précédente pour qu'elle lève une exception de type `PersonneException` (à créer) si les nom et prénom de l'objet `personne` à ajouter existent déjà dans le dictionnaire.
4. Dans `index.php` :
 - On instancie un objet de la classe `ListePersonnes` au tout début de la page
 - On crée un formulaire qui permettra à l'utilisateur de remplir trois champs : numéro, nom et prénom.
 - En cliquant sur le bouton de soumission du formulaire, on construit un objet de type `Personne` avec les valeurs saisies et on l'ajoute au tableau associatif `personnes` de l'objet de la classe `ListePersonnes`.
 - L'exception levée sera affichée sous forme d'un message d'erreur dans la page.
 - La liste des personnes doit être affichée en dessous du formulaire

Exercice 3

Considérons une classe `Nombre` ayant les attributs suivants :

- `var1` : un attribut privé de type `int`
- `var2` : un attribut privé de type `int`

1. Créez la classe `Nombre` et définissez les getters et setters de tous les attributs

Considérons une deuxième classe appelée `Operation` ayant l'attribut suivant :

- `nombre` : un attribut privé de type `Nombre`

2. Créez la classe `Operation` et définissez les getters et setters de tous les attributs

3. Définissez une première méthode `division(): float` qui retourne le résultat de la division de `var1` par `var2` si ce dernier est différent de zéro. Sinon, elle lève une exception `OperationException`. (une classe à créer)

4. Définissez une deuxième méthode `racineDeLaSomme()` qui retourne la racine carrée de la somme de `var1` et `var2` si la somme est positive. Sinon, elle lève une exception `OperationException`.

5. Dans `index.php`, créez un formulaire (avec deux boutons : un pour la division et un deuxième pour la racine carrée de la somme) qui permettra à l'utilisateur de saisir deux entiers. En cliquant sur un de ces deux boutons, on crée un objet de la classe `Nombre` avec les deux valeurs saisies. Ensuite, en utilisant la classe `Operation`, on calcule le résultat et on l'affiche.