

Programmation orientée objet avec Python

Héritage & Exception

Exercice 12.

- Écrivez un programme en Python qui demande à l'utilisateur de saisir un nombre.
- Si un nombre n'a pas été saisi, alors le programme doit lever une exception en indiquant qu'il a une erreur et de redemander la saisie d'un nombre. Le programme s'arrête à l'affichage du nombre saisi.

Exercice 13.

- Écrivez un programme en Python avec la fonction **dire_bonjour_a(nom)** qui affiche sur la console "Bonjour" suivi du nom passé en paramètre.
- Complétez la fonction ci-dessous pour que l'appel échoue avec une exception si le paramètre nom n'est pas une chaîne de caractères ou si la chaîne de caractères est vide :

```
def dire_bonjour_a(nom):  
    # TODO  
    print("Bonjour", a).
```

Exercice 14.

- Créer une sous-classe fille "**Parallelepiped**" héritant de la classe "**Rectangle**".
- La sous-classe fille est dotée d'un attribut **hauteur**.
- La sous-classe fille possède une méthode **Volume()** pour le calcul du volume d'un Parallélépipède.
- Ajoutez une méthode **Affiche()** pour retourner les données de la classe "**Parallelepiped**".
- Sécuriser les attributs de la classe "**Parallelepiped**" en les rendant privés.
- Ajouter les **accesseurs** (*getters*) et les **mutateurs** (*setters*) pour sécuriser l'accès aux données de la classe.
- Écrire une application Python qui utilise la fonction **input()** pour la saisie des données pour les attributs de la classe depuis le clavier. Sécuriser la saisie vis-à-vis des éventuelles exceptions.
- Visualiser les données affectées aux attributs.
- Afficher le résultat du calcul du volume.
- Effectuer des tests avec un jeux de données au minimum trois.

```
Entrez la longueur: 7  
Entrez la largeur: 5  
Entrez la hauteur: 9  
  
Les attributs de la classe Rectangle sont : Longueur 7, Largeur 5, Hauteur 9  
Le périmètre de mon rectangle est : 24  
La surface de mon rectangle est : 35  
  
Les attributs de la classe Parallelepiped sont : Longueur 7, Largeur 5, Hauteur 9  
Le volume de mon Parallelepiped est : 315
```

Exercice 15.

Décrire une personne dans une classe Python. (Sauvegarder dans le fichier `personne.py`)

1. Écrire une classe "**Personne**" en Python. La classe contient les attributs privés suivants :
 - **nom** (String);
 - **prenom** (String);
 - **age** (int).

2. Écrire le constructeur de la classe **"Personne"** sans paramètres mais prévoir une surcharge (*args).
3. Écrire la méthode **"Affiche()"** qui renvoie dans une liste des 3 attributs de la classe.
4. Ajouter les **accesseurs** (*getters*) et les **mutateurs** (*setters*) pour sécuriser l'accès aux données de la classe.
 - **set_Nom()** pour modifier l'attribut Nom,
 - **set_Prenom()** pour modifier l'attribut Prenom,
 - **set_Age()** pour modifier l'attribut Age,
 - **get_Nom()** pour extraire le contenu de l'attribut Nom,
 - **get_Prenom()** pour extraire le contenu de l'attribut Prenom,
 - **get_Age()** pour extraire le contenu de l'attribut Age.

Exercice 16. (héritage)

1. Écrire une classe **"Etudiant"** qui hérite de la classe **"Personne"**. La nouvelle classe comprend des attributs spécifiques à cette classe et sont déclarés en accès privé :
 - **annee** (int) pour l'année d'étude après le bac;
 - **matiere** (String) pour le libellé d'une matière enseignée avec ses plusieurs modules (Informatique, Math);
 - **moyenne** (float) pour la note moyenne obtenue des différentes épreuves sur les modules de la matière.
2. Définir le constructeur qui prend en compte les renseignements nécessaires pour l'étudiant à la création d'un objet de type **"Etudiant"**.
3. Ajouter les **accesseurs** (*getters*) et les **mutateurs** (*setters*) pour sécuriser l'accès aux données de la classe.
 - **set_Annee()**,
 - **get_Annee()**,
 - **set_Matiere()**,
 - **get_Matiere()**,
 - **set_Moyenne()**,
 - **get_Moyenne()**.
4. Écrire l'application Python **"testEtudiant"** qui fait la saisie au clavier des données pour les attributs de l'objet **"Etudiant"** en prenant en charge le traitement des éventuelles exceptions à la suite des entrées des données.
5. Afficher le contenu de l'objet **"Etudiant"** par le retour de la méthode **"Affiche()"**.
6. Prévoir dans l'application une modification de la moyenne de l'étudiant par la méthode mutateur de l'attribut.
7. Réafficher le contenu de l'objet **"Etudiant"** par le retour de la méthode **"Affiche()"**.

```

ivmad@macivmad-1 Etudiants % python testEtudiants.py
Nom: Azerty
Prénom: Qwerty
Age: T
Erreur dans les données
Reprendre
Fermeture de l'application
Nom: Azerty
Prénom: Qwerty
Age: 19
Année: 2
Matière: reso
Moyenne: 11.08
['Azerty', 'Qwerty', 19, 2, 'reso', 11.08]
Fermeture de l'application
Correction pour la Moyenne: 12.89
['Azerty', 'Qwerty', 19, 2, 'reso', 12.89]
ivmad@macivmad-1 Etudiants %
  
```

Exercice 17. Une liste d'étudiants.

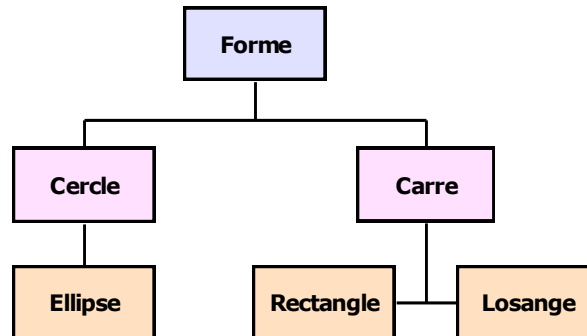
Écrire une classe **"groupeEtudiant"** qui permet d'encapsuler une liste de type **"Etudiant"**.

- Le constructeur reçoit un entier en paramètre pour la taille de la liste (minimum 5);
- ajouter(Etudiant, position) qui ajoute un étudiant à une position donnée dans la liste ;
- etudiantNo(position) qui retourne une référence sur l'étudiant à la position spécifiée ;
- affiche() qui affiche tous les étudiants du groupe ;
- effacer(position) qui met à null la case spécifiée dans la liste.

Exercice 18. (Classes d'objets géométriques)

On souhaite disposer d'un ensemble de classes permettant de manipuler des objets géométriques en 2 dimensions (on s'occupera de leur représentation graphique plus tard !).

1. Écrire les classes reflétant la structure suivante en respectant la dépendance des classes :



- Toutes les classes héritent de la classe parent "Forme"
 - Ainsi Cercle hérite de Forme et Ellipse hérite de Cercle ;
 - Carre hérite de Forme et Rectangle hérite de Carre dont Losange hérite de Carre
2. Écrire la classe **Forme**, avec les attributs : **x**, **y** et **nom** ;
 3. Mettre les attributs en mode privé ;
 4. Écrire les **accesseurs** et les **mutateurs** traitant les attributs de la classe **Forme** ;
 5. Écrire les méthodes vides pour prendre en charge l'**aire** et le **périmètre** d'une forme géométrique 2D ;
 6. Écrire les sous-classes qui dérivent par les relations d'héritage (voir la figure) dans leurs fichiers correspondants ;
 7. Préciser les attributs propres à chaque classe pour pouvoir calculer le périmètre et l'aire ;
 8. Écrire les méthodes qui retournent les calculs pour l'aire et le périmètre des figures géométriques.

À noter :

Cercle: $\text{aire} = \pi \times R^2$, $\text{perimetre} = 2 \times \pi \times R$

Ellipse: $\text{aire} = \pi \times R_1 \times R_2$, $\text{perimetre} = 2 \times \pi \times \sqrt{0.5 \times (R_1^2 + R_2^2)}$

9. Pour manipuler une forme dans un plan, il faudrait ajouter les coordonnées X et Y aux classes.
10. Écrire les méthodes pour accéder aux coordonnées et pour les modifier.
11. Écrire l'implémentation en mettant en conformité les classes appelées.
12. Ajoutez à l'arborescence une nouvelle classe **Triangle**.
 - a. Le triangle est défini par ses côtés : **a**, **b**, **c**.
 - b. Écrire les méthodes calculant le périmètre et l'aire.

A savoir que : Le périmètre des polygones est égal à la somme de la longueur des côtés.

L'aire d'un polygone est le nombre de carrés unités à l'intérieur de ce polygone.

L'aire du triangle est calculée en multipliant la base par la hauteur puis divisé par 2. ($A = \frac{1}{2} \times b \times h$).

```

ivmad@macivmad-1 Formes % python FormesApp.py
0.0 0.0
Entrez des données pour la forme:
X: 5
Y: 7
Nom: Forme
X: 5 Y: 7 Nom: Forme
Entrez des données pour le carré:
X: 4
Y: 5
Nom: Carré
Largeur: 6
L'aire du Carré: 36
Le périmètre du Carré: 24
Entrez des données pour le rectangle:
Nom: Rectangle
Longueur: 7
Largeur: 6
X: 4
Y: 5
L'aire du Rectangle: 42
Le périmètre du Rectangle: 26
Entrez des données pour le cercle:
X: 4
Y: 3
Nom: Cercle
Radius: 5
L'aire du Cercle: 78.54
Le périmètre du Cercle: 157.08
Entrez des données pour le ellipse:
X: 7
Y: 6
Nom: Ellipse
Radius 1: 4
Radius 2: 6
L'aire du Ellipse: 75.4
Le périmètre du Ellipse: 32.04
ivmad@macivmad-1 Formes %
  
```

-
- c. Trouvez la formule calculant la hauteur d'un triangle quelconque à partir des trois côtés. Vérifiez que le triangle existe ! (Traçable à partir des trois côtés).