

ITI 1520 - Devoir 1

Date de remise: Dimanche 29 Septembre, 22h00

Ce devoir est noté sur 20 points au total et forme 5% de la note finale.

Vous devez faire ce travail individuellement. Il faut soumettre un répertoire compressé **Devoir1_<numEtudiant>.zip** (remplacez _<numEtudiant> par votre numéro d'étudiant) contenant les fichiers:

- Nom du fichier à soumettre pour Question 1 : **Devoir1Q1.py**
- Nom du fichier à soumettre pour Question 2 : **Devoir1Q2.py**
- Nom du fichier à soumettre pour Question 3 : **Devoir1Q3.py**
- Nom du fichier à soumettre pour Question 4 : **Devoir1Q4.py**

Tous vos fichiers .py doivent avoir les lignes suivantes au début :

```
#Auteur : Votre nom et prénom  
#Numéro d'étudiant : <Votre numéro d'étudiant>
```

Tous vos fichiers .py doivent aussi avoir les lignes suivantes au début de votre programme:

```
print("Auteur: votre nom et prénom")  
print("Numéro d'étudiant: votre numéro d'étudiant ")
```

Aucun devoir en retard ne sera accepté. Un programme qui donne une erreur d'exécution (quand on clique sur Run) aura la note zéro.

Quand la question dit écrire une fonction, il **faut respecter le nom de la fonction exactement et les noms des paramètres donnés par l'énoncé**.

Le but de ce devoir est de pratiquer la conception et l'implémentation d'algorithmes et l'utilisation des variables de différents types comme on a vu dans Module 1, pratiquer les fonctions vues au Module 2 et les branchements 'if' qui font l'objet de Module 3.

Aucun concept en dehors de ces trois modules n'est accepté. Donc n'utilisez pas des concepts (comme les boucles ou les listes, etc.) qui ne font pas l'objet de Module 1, 2 ou 3.

Question 1 (3 points)

Écrire une fonction Python *interactionUtilisateur()* sans paramètres qui lit au clavier le *prenom*, le *nom* et l'*identifiant* de l'utilisateur. Créer une variable pour chaque donné de l'utilisateur. Vous devez vérifier que l'utilisateur a entré le bon identifiant (un nombre à 4 chiffres).

Affichez le message suivant à l'écran: **Bonjour <prénom> <nom>. J'espère que vous allez bien. Bienvenue dans "mon programme". Votre identifiant est: "<identifiant>"**.

Dans le cas où l'utilisateur a entré un mauvais identifiant, affichez le message suivant à l'écran : **Bonjour <prénom> <nom>. Veuillez vérifier votre identifiant**.

Remplacez <prénom> <nom> et <identifiant> dans le message par les valeurs de ces variables.

Le message doit être affiché sans retour à la ligne.

ITI 1520 - Devoir 1

L'apostrophe dans: j'espère, et dans: l'utilisateur, ainsi que les doubles guillemets de "mon programme" doivent faire partie du message affiché. Le numéro identifiant doit aussi être affiché entre deux guillemets. Le message doit être exactement comme l'indique l'exemple ci-dessous.

Exemple 1:

```
>>> interactionUtilisateur()
Entrez votre prénom: Marie
Entrez votre nom: Curie
Entrez votre identifiant à 4 chiffres: 5624
Bonjour Marie Curie. J'espère que vous allez bien. Bienvenue dans "mon programme". Votre
identifiant est: "5624"
```

Exemple 2 :

```
>>> interactionUtilisateur()
Entrez votre prénom: Marie
Entrez votre nom: Curie
Entrez votre identifiant à 4 chiffres: 12
Bonjour Marie Curie. Veuillez vérifier votre identifiant.
```

Question 2 (2 points)

Écrire une fonction *jourSemaine(jour)* qui prend en paramètres un entier entre 1 et 7 et qui retourne une chaîne de caractères représentant le jour de la semaine. Il faut vérifier que le chiffre est entre 1 et 7. Si le numéro entré est non valide, cette fonction retourne la chaîne de caractères *Jour non valide*

Exemples:

```
>>> jourSemaine(2)
Lundi.

>>> jourSemaine(12)
Jour non valide.
```

Question 3 (3 points)

Écrire une fonction Python *mesure(monrayon)* qui prend en paramètre un rayon et permet de calculer et afficher à l'écran la surface et le volume d'une sphère de rayon *monrayon*. Cette fonction elle ne retourne aucun paramètre.

Voici les formules pour calculer le volume et la surface d'une sphère complète :

$$surface = 4\pi r^2$$

$$volume = \frac{4\pi r^3}{3}$$

L'affichage doit se faire sur deux lignes différentes (en suivant l'exemple ci-dessous).

ITI 1520 - Devoir 1

Exemple:

```
>>> mesure(4)
La surface de votre sphère est : 201.06193.
Le volume de votre sphère est : 268.08.
```

Question 4 (12 points)

Pour cet exercice, veuillez écrire les algorithmes pour vous-même et faire l'implémentation du programme python pour chaque algorithme. Donc, il faut remettre les codes python dans le fichier *Devoir1Q4.py*

Vous faites la gestion d'un magasin qui vend des articles dans le tableau ci-dessous au prix et quantités suivants:

Produits	Prix	Quantité
Bureau	75.9\$	9
Chaise	50.9\$	25
Imprimante	32.5\$	46
Scanneur	28.0\$	17

On veut faire l'implémentation d'un algorithme qui prend les données concernant 3 articles que le client veut acheter et leurs quantités, calcule les prix de chaque article selon la quantité commandée, et ensuite calcule le prix total des 3 articles, et à la fin affiche au client le prix total de sa commande. Avant de valider le prix d'un article, il faut vérifier si l'article et la quantité demandée est disponible (selon le tableau ci-dessous).

Pour ce faire, on va diviser, cet algorithme en sous-algorithmes qui vont s'occuper chacun de tâches différentes :

- a) Le premier sous-algorithme s'occupe du calcul du prix d'un article selon la quantité disponible.

Écrire une fonction Python *calculPrix(article, quantite)*, qui prend en entrée le nom de l'article et la quantité demandée. L'article est de type *string* (une chaîne de caractères) et la quantité est de type *int* (un nombre entier). Cette fonction retourne le prix de type *float* (un nombre réel) calculé en utilisant les prix définis dans le tableau ci-dessus. Pour cette fonction, on suppose que l'article et la quantité choisie sont déjà valides. Tester la fonction.

Exemple:

```
>>> calculPrix("bureau", 1)
50.9
```

- b) Le deuxième sous-algorithme permet de calculer le prix total pour la commande demandée par le client. Il utilise le sous-algorithme précédent pour calculer le prix total pour un article. Le résultat de ce sous-algorithme est le prix total calculé.

ITI 1520 - Devoir 1

Écrire une fonction Python *calculTotal()* qui prend en entrée 6 paramètres

calculTotal(article1, quantite1, article2, quantite2, article3, quantite3)

Elle appelle la fonction *calculPrix()* autant de fois que nécessaire pour obtenir les prix des 3 articles selon les quantités. *calculTotal()* s'occupe ensuite du calcul du prix total de la commande incluant les trois articles. La fonction *calculTotal()* n'est pas autorisée à utiliser les prix directement. Les prix ne peuvent pas être utilisés que par la fonction *calculPrix()*.

La fonction *calculTotal()* retourne le prix total de la commande.

Pour cette fonction, on suppose que l'article et la quantité choisie sont déjà valides.

Exemple:

```
>>> calculTotal("chaise",2,"bureau",1,"scanneur",1)
205.7
```

- c) Le troisième sous-algorithme permet de vérifier si les articles demandés par le client sont vendus au magasin et si les quantités sont disponibles. Écrire la fonction *validerCommande(article1, quantite1, article2, quantite2, article3, quantite3)* qui prend en entrée 6 variables (les noms des articles avec les quantités associés). Elle retourne une variable de type *bool* (booléen) qui sera *true* si la commande est valide et *false* sinon. Une commande est valide si tous les articles existent dans le magasin et toutes les quantités sont disponibles.

Exemple:

```
>>> validerCommande("bureau",15,"imprimante",2,"scanneur",3)
0
>>> validerCommande("bureau",1,"imprimante",2,"scanneur",3)
1
>>> validerCommande("bureau",1,"souris",2,"scanneur",3)
0
```

- d) Écrire le programme principal qui demande au client d'entrer la liste de trois articles avec les quantités associés, et utilise les sous-algorithmes déjà définis pour retourner le prix total de la commande. Après l'entrée de chaque article, le programme appelle la fonction *validerCommande()* afin de vérifier la disponibilité des articles demandés. Si la commande contient n'est pas valide, la commande sera annulée (dans ce cas pas besoin de faire le calcul de prix total). Après la vérification de tous les articles, le programme appelle les fonctions implémentées en a) et b) afin de calculer et afficher le prix total de la commande du client. Des messages appropriés doivent être affichés (voir les exemples d'exécution ci-dessous).

Exemple d'exécution 1:

```
Entrez le premier article: chaise
Entrez la quantité de votre 1ere article: 1
Entrez le deuxième article: bureau
Entrez la quantité de votre 2eme article: 1
```

ITI 1520 - Devoir 1

Entrez le troisième article: trombone

Entrez la quantité de votre 3eme article: 3

Votre commande est annulée. SVP, vérifier les articles ou les quantités.

Exemple d'exécution 2:

Entrez le premier article: chaise

Entrez la quantité de votre 1ere article: 4

Entrez le deuxième article: bureau

Entrez la quantité de votre 2eme article: 18

Entrez le troisième article: scanneur

Entrez la quantité de votre 3eme article: 3

Votre commande est annulée. SVP, vérifier les articles ou les quantités.

Exemple d'exécution 3:

Entrez le premier article: chaise

Entrez la quantité de votre 1ere article: 2

Entrez le deuxième article: bureau

Entrez la quantité de votre 2eme article: 1

Entrez le troisième article: scanneur

Entrez la quantité de votre 3eme article: 1

Le prix total de votre commande est : 205.7\$. Merci et à la prochaine.

Bon chance

