Travail pratique # 1
C'est en forgeant qu'on devient forgeron !

Date de remise : au plus tard le 06 octobre 2024 à 23h59
Pondération de la note finale : 6 %

« *Le secret pour exceller en programmation réside dans la pratique constante et répétée.* »

1 Objectifs

Ce travail a pour principal objectif de vous aider à assimiler les concepts vus dans les premières semaines de cours : l'affectation de variables, les opérations arithmétiques de base, les conversions de types, les entrées/sorties, les structures conditionnelles et les structures itératives.

- Ce travail pratique est à faire **OBLIGATOIREMENT** de façon **INDIVIDUELLE**.
- Interdiction d'utiliser des fonctions prédéfinies (« *built-in* ») Python exceptées celles vues en classe durant les trois premiers modules (vous pouvez par exemple utiliser les fonctions `print`, `input`, `range`, `len`, `int`, `str`, `float`, etc).
- Interdiction d'utiliser les listes Python (`list`), les dictionnaires Python (`dict`), ainsi que leurs méthodes respectives.
- Interdiction d'utiliser les mots-clés `break` et `continue`.
- Autre que cela, vous avez le droit d'utiliser toutes les notions vues en classe durant les trois premiers modules : les *f-strings*, la méthode `format` disponible pour les chaînes de caractères, les structures conditionnelles `if-elif-else`, la boucle `while`, la boucle `for`, les tuples, etc. Vous avez également le droit d'utiliser les méthodes `strip` et `join` disponibles pour les chaînes de caractères.
- Bien que nous n'ayons pas encore abordé ce sujet en classe, vous êtes libre de créer vos propres fonctions si cela peut améliorer la lisibilité de votre programme. Cependant, il est important de noter que cela ne vous accordera aucun point supplémentaire.
- Il se peut que la description du TP1 présentée ci-dessous manque de précision sur certains aspects. Si c'est le cas, il est de votre responsabilité de le remarquer et de demander l'information qui manque.

2 Exercice 1 : IFT-1004 Airlines

Vous devez écrire un programme Python destiné à un(e) agent(e) de la compagnie aérienne *IFT-1004 Airlines*, chargé(e) de réserver des billets pour les passagers. Le prix de base d'un billet est fixé à **150$**, quelle que soit la destination. Le programme devra gérer différentes catégories de passagers et appliquer des réductions personnalisées en fonction de leur âge. Il devra également permettre de traiter plusieurs clients et afficher le montant total cumulé pour l'ensemble des réservations lorsque l'agent(e) décide de quitter le programme.

Règles de tarification :

- Si la personne a 3 ans ou moins, le billet est gratuit.
- Si la personne a 65 ans ou plus, un rabais aléatoire entre **15% et 55%** est appliqué au prix du billet.
- Sinon, le billet est vendu au prix normal de 150$.

2.1 Instructions

Votre programme doit suivre les étapes suivantes :

1. Afficher la bannière de bienvenue, ainsi que le slogan de IFT-1004 Airlines.

```
#######################################
          IFT-1004 Airlines
#######################################

La meilleure compagnie aérienne au monde!
```

2. Demander à l'agent(e) combien de billets il/elle souhaite réserver pour un client (*c'est-à-dire, le nombre de passagers*).
3. Pour chaque billet, afficher le numéro du passager sous le format suivant :

```
============
Passager X/Y
============
```

   - `X` correspond au numéro du passager dans l'ordre de réservation ;
   - `Y` correspond au nombre total de passagers pour un client donné.

4. Pour chaque billet, demander l'âge et le nom complet du passager.
5. Calculer le prix du billet en fonction de l'âge du passager et afficher ce prix immédiatement après que l'agent(e) ait saisi les informations du passager.
6. Si un rabais est appliqué (pour les passagers de 65 ans et plus), afficher le rabais sous la forme suivante :

```
Rabais appliqué pour {nom} : {pourcentage}%
```

7. À la fin de la réservation pour un client, afficher le montant total à encaisser pour l'ensemble des billets achetés. **Si le nombre de billets entré est 0, le programme ne doit pas afficher le montant total à encaisser pour ce client**.
8. À la fin de la réservation pour un client, afficher un menu avec deux options :
   - **1. Enregistrer un autre client**
   - **2. Quitter**
9. Si l'agent(e) choisit d'enregistrer un autre client (*en tapant 1*), répétez les étapes ci-dessus.
10. Si l'agent(e) choisit de quitter (*en tapant 2*), afficher le montant total cumulé pour tous les clients enregistrés et le message `"Bye bye!"`.
11. Si l'agent(e) entre un choix différent de 1 ou 2, afficher `"Choix invalide. Veuillez entrer 1 ou 2."` et redemander un choix valide.

Important :

- Le menu à l'étape 8 n'apparaît qu'après l'enregistrement du premier client (et non au tout début du programme).
- **Il est important de s'assurer que tous les montants affichés soient formatés de manière à inclure systématiquement deux décimales.**
- À chacune des entrées utilisateur, vous pouvez supposer que l'agent(e) entrera une chaîne valide. En d'autres termes, si on demande par exemple le nombre de billets à réserver ou l'âge et que l'agent(e) entre la chaîne `"bonjour"`, il est tout à fait normal que votre programme plante ou agisse de manière imprévisible. Autrement dit, vous n'avez pas à gérer les exceptions à ce stade.

2.2 Exemple d'exécution

Voici un exemple d'exécution du programme :

```
#####################################
          IFT-1004 Airlines
#####################################

La meilleure compagnie aérienne au monde!

Combien de billets souhaitez-vous réserver ? 3

============
Passager 1/3
============
Nom du passager 1 : Alice Lamalchance
Âge du passager 1 : 70
```

```
Rabais appliqué pour Alice Lamalchance : 17%
Prix du billet réservé pour Alice Lamalchance : 124.50$

============
Passager 2/3
============
Nom du passager 2 : Bob Lachance
Âge du passager 2 : 65
Rabais appliqué pour Bob Lachance : 40%
Prix du billet réservé pour Bob Lachance : 90.00$

============
Passager 3/3
============
Nom du passager 3 : Charlie Heureux
Âge du passager 3 : 0
Prix du billet réservé pour Charlie Heureux : 0.00$

Montant total à encaisser pour cette réservation : 214.50$

1. Enregistrer un autre client
2. Quitter
Entrez votre choix : 1
Combien de billets souhaitez-vous réserver ? 1

============
Passager 1/1
============
Nom du passager 1 : Eve Malheureuse
Âge du passager 1 : 30
Prix du billet réservé pour Eve Malheureuse : 150.00$

Montant total à encaisser pour cette réservation : 150.00$

1. Enregistrer un autre client
2. Quitter
Entrez votre choix : 1
Combien de billets souhaitez-vous réserver ? 0

1. Enregistrer un autre client
2. Quitter
Entrez votre choix : 0
Choix invalide. Veuillez entrer 1 ou 2.

1. Enregistrer un autre client
2. Quitter
Entrez votre choix : 2
```

```
Montant total cumulé pour tous les clients enregistrés : 364.50$
Bye bye!
```

3 Exercice 2 : Dessin de pyramides de nombres

Vous devez écrire un programme Python qui demande à l'utilisateur d'entrer la hauteur d'une pyramide de nombres. Le programme dessinera ensuite cette pyramide, où chaque ligne contient des nombres croissants commençant par 1 jusqu'au numéro de la ligne, centrés pour former une pyramide. La hauteur de la pyramide doit être un **entier positif supérieur ou égal à 4**. Si la hauteur saisie est inférieure à 4, le programme devra afficher un message `"Hauteur incorrecte"` et ne pas procéder au dessin.

3.1 Spécifications

- Chaque ligne de la pyramide contient des nombres de 1 jusqu'au numéro de la ligne, séparés par des espaces.
- Les nombres doivent être centrés pour former une pyramide.
- Si la hauteur spécifiée est inférieure à 4, affichez `"Hauteur incorrecte"`.

3.2 Exemples d'exécution

Voici un exemple d'exécution :

```
Entrez la hauteur: 5
    1
   1 2
  1 2 3
 1 2 3 4
1 2 3 4 5
```

Voici un autre exemple d'exécution :

```
Entrez la hauteur: 8
       1
      1 2
     1 2 3
    1 2 3 4
   1 2 3 4 5
  1 2 3 4 5 6
 1 2 3 4 5 6 7
1 2 3 4 5 6 7 8
```

Un autre pour la route :

```
Entrez la hauteur: 10
          1
         1 2
        1 2 3
       1 2 3 4
      1 2 3 4 5
     1 2 3 4 5 6
    1 2 3 4 5 6 7
   1 2 3 4 5 6 7 8
  1 2 3 4 5 6 7 8 9
1 2 3 4 5 6 7 8 9 10
```

Et un dernier exemple où la hauteur est incorrecte :

```
Entrez la hauteur: 3
Hauteur incorrecte
```

Important : À chacune des entrées utilisateur, vous pouvez supposer que l'utilisateur entrera une chaîne valide. En d'autres termes, si on demande par exemple la hauteur et que l'utilisateur entre la chaîne `"bonjour"`, il est tout à fait normal que votre programme plante ou agisse de manière imprévisible. Autrement dit, vous n'avez pas à gérer les exceptions à ce stade.

4 Modalités d'évaluation

Ce travail sera évalué sur **35 points**, et la note sera ramenée sur 6. Vous trouverez ci-après le barème de notation détaillé.

Notez qu'un programme qui n'est pas fonctionnel (qui ne s'exécute pas ou qui plante à l'exécution) pourrait recevoir une note de 0.

4.1 Exercice 1 (23 points)

Le premier exercice sera évalué sur 23 points. Voici la grille de correction détaillée :

Affichage correct de la bannière et du slogan	2 points
Gestion correcte du nombre de billets (y compris le cas où 0 billets sont demandés)	2 points
Affichage correct du numéro du passager pour chaque billet (format : Passager X/Y)	2 points
Calcul et affichage correct du prix du billet (y compris la gestion des rabais)	5 points
Affichage du rabais lorsque applicable	2 points
Affichage du montant total à encaisser pour un client (si le nombre de billets est > 0)	3 points
Affichage correct du menu (avec validation des choix d'entrée utilisateur : 1 ou 2)	3 points
Affichage correct du montant total cumulé pour tous les clients enregistrés	2 points
Qualité du code (noms de variables, lisibilité, respect des consignes, sans « break »)	2 points

Votre programme doit être rédigé dans le fichier python `exercice1.py` présent dans le dossier téléchargé, que vous devez compresser dans une archive Zip (fichier avec extension `.zip`) avec les autres fichiers à rendre.

4.2 Exercice 2 (10 points)

Le second exercice sera évalué sur 10 points. Voici la grille de correction détaillée :

Validation correcte de la hauteur de la pyramide (un minimum de 4)	2 points
Affichage du message d'erreur approprié pour les hauteurs incorrectes	2 points
Exactitude du dessin de la pyramide selon la hauteur valide fournie	4 points
Qualité du code (noms de variables, lisibilité, respect des « interdits » pour ce travail)	2 points

Votre programme doit être rédigé dans le fichier python `exercice2.py` présent dans le dossier téléchargé, que vous devez compresser dans une archive Zip (fichier avec extension `.zip`) avec les autres fichiers à rendre.

Utilisation de Git (2 points)

Vous devez compléter la partie intitulée *« Lien vers le dépôt GitHub PRIVÉ de votre TP : »* dans le fichier `correction.txt` fourni avec cet énoncé. Un total de 2 points sera attribué pour tout dépôt GitHub **privé** illustrant une utilisation de Git tout au long du développement de ce premier travail pratique.

5 Stratégie suggérée

- Résolvez les problèmes dans un premier temps en utilisant du papier et un crayon. Une bonne pratique de programmation consiste à formuler une ébauche de solution avant de se lancer dans l'écriture de code. Réfléchissez donc à un pseudo-code.
- Écrivez ensuite une version simple de votre programme en Python. Exécutez le programme fréquemment et trouvez les sources d'erreur. Ajoutez une fonctionnalité à la fois en répétant le cycle exécution/débogage.

Si vous avez du mal à programmer votre solution, voici quelques questions que vous devez vous poser et ensuite vérifier votre programme :

1. Quelles sont les informations que notre programme doit stocker en mémoire, et comment pouvons-nous les conserver pour les réutiliser plus tard ?
2. Quel devrait être le type de ces données ?
3. Quelle structure étudiée en classe nous permettra d'effectuer les opérations demandées ?

6 Remarques

Plagiat : Tel que décrit dans le plan de cours, le plagiat est strictement interdit. Ne partagez pas votre code source à quiconque. Une politique stricte de tolérance zéro est appliquée en tout temps et sous toute circonstance. Tous les cas détectés seront référés à la direction de la faculté. Des logiciels comparant chaque paire de TPs pourraient être utilisés pour détecter les cas de plagiat.

Retards : Les travaux pratiques doivent être impérativement remis via le portail de cours. Aucune remise par courriel n'est acceptée. Un travail qui ne respecte pas les directives de remise se verra pénalisé de 20%. Tout travail remis en retard se verra pénalisé de 25% par jour de retard. Chaque journée de retard débute dès la limite de remise dépassée (dès la première minute). Un retard excédant 2 jours provoquera le rejet du travail pour la correction et la note de 0 pour ce travail.

Remises multiples : Il vous est possible de remettre votre TP plusieurs fois sur le portail des cours. La dernière version sera considérée pour la correction.

Respect des normes de programmation : Nous vous demandons de prêter attention au respect des normes de programmation établies pour le langage Python, notamment de nommer vos variables et fonctions en utilisant la convention suivante : `ma_variable`, `mon_autre_variable`, etc. Utiliser **PyCharm** s'avère être une très bonne idée ici, car celui-ci nous donne des indications sur la qualité de notre code (en marge à droite, et souligné).

7 Ce que vous devez rendre

Vous devez remettre une archive `.zip` d'un **dossier**, contenant **uniquement** les fichiers suivants :

- `exercice1.py` : Un fichier Python contenant votre programme pour l'exercice 1. Les correcteurs doivent être en mesure d'exécuter ce fichier sans lui apporter de modifications.
- `exercice2.py` : Un fichier Python contenant votre programme pour l'exercice 2. Les correcteurs doivent être en mesure d'exécuter ce fichier sans lui apporter de modifications.
- `correction.txt` : Le fichier de correction fourni avec l'énoncé, que vous devez compléter en y indiquant votre nom, votre NI, le lien vers le dépôt GitHub **PRIVÉ** de votre TP, et le nombre d'heures que vous avez consacré à compléter votre TP.

Cette archive doit être remise via le site Web du cours. Une pénalité de 5% pourra être appliquée si vous remettez des fichiers inutiles.

Bon travail :).