

Exercices pour la séance de révision

Abiola Adeye: abiola.adeye@epfl.ch

2022/10/30

1 Semaine 1

Question 1: (🕒 10 minutes) **Conversion**

1. Convertir le nombre $FFFFFF_{(16)}$ en base 10.
2. Convertir le nombre $4321_{(5)}$ en base 10.
3. Convertir le nombre $ABC_{(16)}$ en base 2.
4. Convertir le nombre $254_{(10)}$ en base 15.
5. Convertir le nombre $11101_{(2)}$ en base 10.

2 Semaine 2

Question 2: (🕒 5 minutes) Sous Linux et MacOS, laquelle de ces commandes modifie le **filesystem** ?

1. `ls -la`
2. `sudo rm -rf ~/nano`
3. `sudo kill -9 3531`
4. `more nano.txt`
5. Aucune réponse n'est correcte.

3 Semaine 3

Question 3: (🕒 10 minutes) **Floating point**

Voici la représentation en binaire d'un nombre à virgule flottante :

signe	exposant	mantisse
0	10110101	01000001000000000000001

Que vaut cette représentation en base 10 ? Utiliser la représentation des floating points (avec un biais de 127). Arrondir les résultats intermédiaires et la valeur finale au 3ème chiffre significatif après la virgule.

4 Semaine 4

Question 4: (🕒 5 minutes) Manipulation d'un dictionnaire - 2

Créez un dictionnaire nommé `fr_eng` contenant les éléments suivants :

```
1 "chat": "cat", "chien": "dog", "oiseau": "bir", "poule": "chicken", "papillon": "butterfly", "souris": "mouse", "ours": "bear",  
  "mouton": "sheep", "cochon": "pig"
```

La traduction du mot “oiseau” est mal orthographiée, modifiez la valeur associée à “oiseau” pour qu’elle devienne “bird”. Ajoutez un nouvel élément au dictionnaire en associant le mot “horse” au mot “cheval”.

Utilisez ce code pour avoir un aperçu des changements :

```
1 for x in fr_eng :  
2     print(x + " : " + fr_eng[x])
```

5 Semaine 5

Question 5: (🕒 10 minutes) Manipulation des listes en Java

Créez une liste nommée `ma_liste` contenant les nombres 1,2,3,4 et 5. Affichez le deuxième élément de la liste ainsi que la taille de la liste.

Créez une liste `ma_liste_m` liée à la liste `ma_liste`. Ajoutez le chiffre 6 à la fin de la liste, et le chiffre 0 au début de cette dernière.

Ajoutez ceci au début de votre code :

```
1 import java.util.List;  
2 import java.util.LinkedList;
```

Ajoutez ceci à la fin de votre code :

```
1 for(int i=0;i<ma_liste_m.size();i++){  
2     System.out.println(ma_liste_m.get(i));  
3 }
```

6 Semaine 6

Question 6: (🕒 10 minutes) Création de classe et encapsulation

Commencez par créer une nouvelle classe `Dog` dans votre projet. Ensuite, créez les attributs suivants :

1. Un attribut `public String` nommé `name`
2. Un attribut `private List` nommé `tricks`
3. Un attribut `private String` nommé `race`
4. Un attribut `private int` nommé `age`
5. Un attribut `private int` nommé `mood` initialisé à 5 (correspondant à l’humeur du chien)
6. Un attribut de classe (`static`) `private int` nommé `nb_chiens`

Créez une méthode publique du même nom que la classe (`Dog`). Cette méthode est appelée le **constructeur**, elle va servir à initialiser les différentes instances de notre classe. Un **constructeur** en **Java** aura le même nom que la classe, et le **constructeur** en **Python** sera défini par la méthode `__init__`. Cette méthode prendra en argument les éléments suivants qui seront utilisés pour initialiser les attributs de notre instance :

1. Une chaîne de caractères `name`,
2. Une liste `tricks`,

3. Une chaîne de caractères **race**,

4. Un entier **age**.

Pour finir, cette méthode doit incrémenter l'attribut de classe **nb_chiens** qui va garder en mémoire le nombre d'instances créées.