

CODING CLUB

EPITECH



Le Prix Juste

VERSION 2.1.0

I. Introduction

Arthur et Sophie sont invités pour participer à la nouvelle émission nommé "Le Prix Juste" qui passera sur TF9 à 19h30. Le but de cette émission est simple : Trouver le prix exact d'un produit afin de le remporter. Mais ce ne sera pas si facile car nos deux participants devront tenter de deviner le prix du produit à tour de rôle !

Malheureusement, la chaîne de télé organisant cette nouvelle émission a oublié de dire au joueur que cette dernière se passait à Paris. Arthur étant à Lille et Sophie à la Réunion, ils sont dans l'impossibilité de venir tous les deux pour la première de ce soir !

Vincent Faigaf ne veut pas annuler cette première. Il t'a donc confié la tâche de dématérialiser le plateau TV pour assurer l'émission de ce soir. Vous avez une heure et demi avant le début !

II. Consignes

- Pour l'installation, suis le tutoriel « Installation de Python et ses outils ».
- Lis tout avant de commencer !
- Demande de l'aide aux Cobras en cas de problème d'installation. Si plus rien ne va, recommence depuis le début en faisant bien attention à toutes les étapes !
- Si tu bloques, rappelle-toi que tu es accompagné ! Demande de l'aide à tes camarades ou à un Cobra, ceux-là ne mordent pas.
- Internet est un outil formidable pour découvrir le fonctionnement des choses, sers-t'en régulièrement !

⚠ Attention: Le code des exemples est incomplet, tu devras rajouter/modifier quelques éléments pour que cela fonctionne.

Un `...` dans le code signifie que tu dois compléter le code par toi-même en utilisant les informations du sujet.

Un `#` est un commentaire pour t'aider à comprendre. Ce qui se trouve après, sur la même ligne, est ignoré par le programme.

III. Inscription des joueurs

a. Les répliques

Dans un premier temps, il faudra recréer le présentateur TV, la voix-off et le public. Nous avons de la chance car les répliques sont les mêmes à chaque émission. Ce serait dommage qu'Arthur et Sophie ne puissent pas suivre le jeu en étant guidés par notre présentateur favori !

```
Dialogue = [  
    "",  
    "voix-off : Bonjour et bienvenue au Prix Juste, avec pour présentateur, Vir  
    "Vincent Faigaf : BIP BIP",  
    "Le public : OUAIIIIIIIS !",  
    "Vincent Faigaf : Et on commence tout de suite avec nos premiers candidats  
    "Notre premier candidat est : ",  
    "Il affrontera : ",  
    "Nous allons tout de suite commencer avec le premier produit."  
]
```

[Plus d'information sur les listes.](#)

b. Création des joueurs

Il faut ensuite entrer le nom des joueurs pour les ajouter au dialogue.

Dans la fonction `Main()`, nous allons donc commencer par demander le nom de chacun des joueurs puis les stocker dans les variables `playerOne` et `playerTwo` grâce à la fonction `input()`.

```
def Main():  
    playerOne = ...  
    playerTwo = ...
```

c. Deux noms à l'oreillette

Les deux candidats sont maintenant inscrits, mais notre cher **Vincent Faigaf** n'a pas leur nom dans son script.

Il va donc falloir modifier les cinquième et sixième phrases de **Dialogue** dans notre fonction **RegisterPlayer(player1, player2)** pour ajouter les noms à la chaîne de caractère déjà existante.

```
def RegisterPlayer(player1, player2):  
    Dialogue[...] = ...  
    Dialogue[...] = ...
```

[Plus d'information sur la concaténation](#)

IV. Un Story Telling de qualité !

a. Une façon différente d'afficher

Arthur et Sophie mettent du temps à lire, il faudra donc afficher le texte petit à petit.

[Ceci pourra t'aider.](#)

Le premier paramètre `text` servira à afficher les dialogues et le second paramètre `sleep_time` servira à donner le temps entre chaque message.

```
def CustomPrint(text, sleep_time):  
    ...
```

b. Il est temps de donner la parole au programme

Ici, vous vous chargerez d'afficher les phrases du dialogue avec 2 secondes de temps de pause entre l'affichage des messages grâce à la fonction que vous avez précédemment créée.

```
def StoryTeller():  
    CustomPrint(Dialogue[0], ...)  
    ...
```

V. Un lot à faire gagner

a. Récupérer le prix du lot

La régie est venue nous annoncer qu'ils possédaient déjà quelques produits en stock. Voici la liste de produits sous la forme d'un dictionnaire en Python.

```
Products = {
    "Ecran" : {
        "name" : "Ecer Gaming",
        "price" : random.randint(180, 230)
    },
    "Console" : {
        "name" : "Nantendo Swatch",
        "price" : random.randint(250, 320)
    },
    "Unité centrale" : {
        "name" : "NSA Gaming",
        "price" : random.randint(1000, 1200)
    },
    "Télévision" : {
        "name" : "Somsing QLED",
        "price" : random.randint(1000, 1500)
    },
    "Enceinte" : {
        "name" : "JPL PoomPox",
        "price" : random.randint(100, 180)
    },
}
```

[Plus d'information sur les dictionnaires](#)

b. Choisir un lot au hasard

- Récupère un produit au hasard grâce à la fonction `random()`

```
nb_product = random.choice(list(Products.keys()))  
product = Products.get(nb_product)
```

- Affiche le nom du produit sélectionné aléatoirement

```
print("Vous devez trouver le prix de ce produit : ", product.get(...))
```

- Récupère le prix du produit sélectionné aléatoirement

```
price = product.get(...)
```


VI. Boucle de jeu

a. Le début de la fin

- Initialise deux variables pour récupérer les entrées des utilisateurs
- Crée la boucle de jeu: tant que la valeur entrée par les deux joueurs est différente du prix
- Récupère l'entrée du premier joueur et transforme la valeur texte en entier numérique
- Laisse une ligne pour appeler la fonction `PlayerTurnVerification()`
- Vérifie si l'entrée du premier joueur est égale au prix, si oui, arrête la boucle à l'aide de `break` et affiche « Le joueur 1 a gagné ! »
- Récupère l'entrée du second joueur et transforme la valeur en entier
- Laisse une ligne pour appeler la fonction `PlayerTurnVerification()`
- Vérifie si l'entrée du second joueur est égale au prix, si oui, affiche « Le joueur 2 a gagné ! »

b. La fin du début

L'heure est venue de terminer la dernière partie du jeu afin que l'émission puisse commencer ! Il faudra maintenant compléter cette fonction :

```
def PlayerTurnVerification(valuePlayer, price, nextPlayerName):  
    ...
```

Ici, il faudra vérifier si la valeur entrée par le joueur `valuePlayer` est supérieure ou est inférieure au prix du produit.

- Si la valeur entrée par le joueur est inférieure au prix, affiche :

```
print("C'est plus ! Au tour de " + nextPlayerName)
```

- Si la valeur entrée par le joueur est supérieure au prix, affiche :

```
print("C'est moins ! Au tour de " + nextPlayerName)
```

Pour terminer le programme, il faut remplir les lignes laissées tout à l'heure en appelant cette fonction avec les bons arguments.

Enfin, il faut appeler la fonction `Main()` à la fin du programme.

VII. Conclusion

Félicitations, tu as sauvé la première du Prix Juste ! Arthur et Sophie vont enfin pouvoir commencer à jouer !

Pour aller plus loin, voici quelques idées :

- Afficher le nom du joueur victorieux
- Ajouter un timer pour gérer le temps de jeu de chaque joueur
- Créer un générateur de produit
- Ajouter un affichage plus coloré pour les textes d'affichage
- Créer une interface graphique pour le jeu grâce à PyGame