

Les dictionnaires pour l'escape game

Ex 1

Exercice 1 : Créer des objets pour votre escape game

Créez trois dictionnaires représentant les objets suivants :

1.0.1 a) Un livre

- Nom : "livre"
- Position : x=150, y=180
- Dimensions : largeur=80, hauteur=60
- Propriété "lu" : False
- Description : "Un vieux grimoire poussiéreux"

```
1 livre = {  
2  
3  
4  
5  
6  
7  
8 }
```

1.0.2 b) Une lampe

- Nom : "lampe"
- Position : x=500, y=300
- Dimensions : largeur=40, hauteur=80
- Propriété "allumee" : False
- Description : "Une lampe à huile"

```
1 lampe = {  
2  
3  
4  
5  
6  
7  
8 }
```

```
9  
10 }
```

1.0.3 c) Un tiroir

- Nom : “tiroir”
- Position : x=200, y=400
- Dimensions : largeur=100, hauteur=50
- Propriété “ouvert” : False
- Description : “Un tiroir du bureau”

```
1 tiroir = {  
2  
3  
4  
5  
6  
7  
8  
9  
10 }
```

Ex 2

Exercice 2 : Modifier les propriétés

Écrivez le code Python qui effectue les actions suivantes :

a) Allumer la lampe

b) Ouvrir le tiroir

c) Marquer le livre comme lu

d) Eteindre la lampe

Ex 3

Exercice 3 : Interaction simple - Exemple

Voici un exemple d'interaction : cliquer sur un coffre pour l'ouvrir et révéler une clé.

```
1  # Définition des objets
2  coffre = {
3      "nom": "coffre",
4      "x": 300,
5      "y": 350,
6      "visible": True,
7      "ouvert": False
8  }
9
10 cle = {
11     "nom": "clé",
12     "x": 320,
13     "y": 370,
14     "visible": False,  # Cachée au début
15     "ramasse": False
16 }
17
18 # Dans la gestion des clics
19 if event.type == pygame.MOUSEBUTTONDOWN:
20     # Clic sur le coffre
21     if rect_coffre.collidepoint(event.pos) and coffre["visible"]:
22         if not coffre["ouvert"]:
23             coffre["ouvert"] = True
24             cle["visible"] = True  # La clé apparaît !
25             message = "Le coffre contenait une clé !"
```

Question : Que se passe-t-il quand on clique sur le coffre ?

Ex 4

Exercice 4 : Interaction simple - À compléter

Complétez le code suivant pour créer cette interaction :

- Cliquer sur le tableau le fait basculer et révèle un coffre-fort caché derrière

- Le tableau a une propriété "basculer" qui passe à True
- Le coffre-fort a une propriété "visible" qui passe à True

```

1  # Définition des objets
2  tableau = {
3      "nom": "tableau",
4      "x": 200,
5      "y": 150,
6      "rect": pygame.Rect(200, 150, 150, 100),
7      "visible": True,
8      "basculer": False
9  }
10
11 coffre_fort = {
12     "nom": "coffre-fort",
13     "x": 210,
14     "y": 160,
15     "rect": pygame.Rect(210, 160, 130, 80),
16     "visible": False,
17     "ouvert": False
18 }
19
20 # Dans la gestion des clics - À COMPLÉTER
21 if event.type == pygame.MOUSEBUTTONDOWN:
22     pos = event.pos
23
24     # Clic sur le tableau
25     if tableau["rect"].collidepoint(pos):
26         if not tableau["basculer"]:
27             # Complétez ici
28             tableau[_____] = _____
29             coffre_fort[_____] = _____
30             message = "Le tableau bascule ! Un coffre-fort est
caché derrière !"

```

Ex 5

Exercice 5 : Proposez un scénario

En vous inspirant des exercices précédents : Décrivez les objets à l'aide de dictionnaires. Proposez un (extrait de) programme python montrant l'évolution des valeurs stockées dans les dictionnaires décrivant ces objets, en imaginant le mécanisme du jeu.

Ex 6

Exercice 6 : Scénario complet - Analyse (prolongement)

Lisez attentivement le code suivant qui représente un mini escape game complet :

```
1 import pygame
2
3 pygame.init()
4 screen = pygame.display.set_mode((800, 600))
5 clock = pygame.time.Clock()
6 font = pygame.font.Font(None, 28)
7
8 # Définition des rectangles
9 rect_tableau = pygame.Rect(200, 150, 150, 100)
10 rect_coffre = pygame.Rect(550, 350, 80, 60)
11 rect_cle = pygame.Rect(100, 400, 40, 20)
12 rect_porte = pygame.Rect(650, 200, 100, 200)
13
14 # Définition des objets
15 tableau = {
16     "nom": "tableau",
17     "rect": rect_tableau,
18     "visible": True,
19     "examine": False
20 }
21
22 coffre = {
23     "nom": "coffre",
24     "rect": rect_coffre,
25     "visible": False,
26     "ouvert": False
27 }
28
29 cle = {
30     "nom": "clé",
31     "rect": rect_cle,
32     "visible": False,
33     "ramasse": False
34 }
35
36 porte = {
37     "nom": "porte",
38     "rect": rect_porte,
39     "visible": True,
40     "verrouille": True
41 }
42
43 # État du jeu
44 message = "Explorez la pièce..."
```

```
45 victoire = False
46
47 running = True
48 while running:
49     for event in pygame.event.get():
50         if event.type == pygame.QUIT:
51             running = False
52
53         if event.type == pygame.MOUSEBUTTONDOWN and not victoire:
54             pos = event.pos
55
56             # 1. Cliquer sur le tableau
57             if tableau["rect"].collidepoint(pos) and not tableau["examine"]:
58                 tableau["examine"] = True
59                 coffre["visible"] = True
60                 message = "Vous découvrez un coffre caché derrière le tableau !"
61
62             # 2. Cliquer sur le coffre
63             if coffre["rect"].collidepoint(pos) and coffre["visible"]:
64                 if not coffre["ouvert"]:
65                     coffre["ouvert"] = True
66                     cle["visible"] = True
67                     message = "Le coffre contenait une clé !"
68
69             # 3. Ramasser la clé
70             if cle["rect"].collidepoint(pos) and cle["visible"] and not cle["ramasse"]:
71                 cle["ramasse"] = True
72                 cle["visible"] = False
73                 message = "Vous ramassez la clé"
74
75             # 4. Ouvrir la porte
76             if porte["rect"].collidepoint(pos):
77                 if cle["ramasse"]:
78                     if porte["verrouille"]:
79                         porte["verrouille"] = False
80                         message = "Vous ouvrez la porte !"
81
82                 victoire = True
83             else:
84                 message = "La porte est verrouillée"
85
86             # Affichage (simplifié pour l'exercice)
87             screen.fill((50, 50, 50))
88
89             # Afficher le message
90             texte = font.render(message, True, (255, 255, 255))
```

```
90     screen.blit(texte, (50, 550))
91
92     pygame.display.flip()
93     clock.tick(60)
94
95 pygame.quit()
```

6.1 Questions :

a) Listez les 4 objets présents dans le jeu et leur rôle :

1. _____
2. _____
3. _____
4. _____

b) Décrivez les étapes pour gagner le jeu (dans l'ordre) :

Étape 1 : _____

Étape 2 : _____

Étape 3 : _____

Étape 4 : _____

c) Que se passe-t-il si on clique sur la porte avant de ramasser la clé ?

d) Quelle propriété du dictionnaire permet de savoir si le joueur a ramassé la clé ?
