

Détecter plusieurs collisions avec des carrés

En repartant des exercices précédents nous allons détecter les collisions avec plusieurs carrés.

Rappelle-toi, la méthode que nous utilisons pour détecter les collisions consiste à comparer les positions de deux rectangles entre eux:

```
1 | if player_rect.collidect(carre):
2 |     # Collision détectée
```

Pour détecter les collisions avec tous les objets qu'on a créés, il suffit de passer chacun d'eux en revue à l'aide d'une boucle.

```
1 | for carre in liste_carres:
2 |     if player_rect.collidect(carre):
3 |         print("Le joueur touche une pièce !")
```

Ceci doit se faire bien sûr **dans la boucle de jeu**. Le code a été ajouté à la ligne 56 ci-dessous:

```
1 | import pygame
2 | import random
3 | import sys
4 |
5 | # Initialisation
6 | pygame.init()
7 |
8 | # Fenêtre
9 | largeur_ecran = 800
10 | hauteur_ecran = 600
11 | screen = pygame.display.set_mode((largeur_ecran, hauteur_ecran))
12 | pygame.display.set_caption("Collisions multiples")
13 |
14 | # Joueur (image)
15 | joueur_image = pygame.image.load("joueur.png")
16 | player_x = 100
17 | player_y = 100
18 | vitesse = 5
19 | player_largeur = joueur_image.get_width()
20 | player_hauteur = joueur_image.get_height()
21 |
22 | # Couleur et taille des pièces
23 | couleur_carre = (255, 215, 0) # doré
24 | taille_carre = 20
25 |
26 | # Génération de plusieurs carrés aléatoires
27 | liste_carres = []
28 | for _ in range(5):
29 |     x = random.randint(0, largeur_ecran - taille_carre)
30 |     y = random.randint(0, hauteur_ecran - taille_carre)
31 |     rect_carre = pygame.Rect(x, y, taille_carre, taille_carre)
32 |     liste_carres.append(rect_carre)
33 |
34 | # Boucle de jeu
```

```

35 clock = pygame.time.Clock()
36 running = True
37 while running:
38     for event in pygame.event.get():
39         if event.type == pygame.QUIT:
40             running = False
41
42     # Déplacements
43     touches = pygame.key.get_pressed()
44     if touches[pygame.K_LEFT]:
45         player_x -= vitesse
46     if touches[pygame.K_RIGHT]:
47         player_x += vitesse
48     if touches[pygame.K_UP]:
49         player_y -= vitesse
50     if touches[pygame.K_DOWN]:
51         player_y += vitesse
52
53     # Créer le rectangle du joueur à chaque boucle
54     player_rect = pygame.Rect(player_x, player_y, player_largeur, player_hauteur)
55
56     # Vérifier les collisions avec chaque carré
57     for carre in liste_carres:
58         if player_rect.colliderect(carre):
59             print("Le joueur touche une pièce !")
60
61     # Affichage
62     screen.fill((0, 0, 0)) # fond noir
63
64     # Dessiner les carrés
65     for carre in liste_carres:
66         pygame.draw.rect(screen, couleur_carre, carre)
67
68     # Afficher le joueur
69     screen.blit(joueur_image, (player_x, player_y))
70
71     pygame.display.flip()
72     clock.tick(60)
73
74 pygame.quit()
75 sys.exit()

```