

Cobra Hunt

version 1.0



DIVERSITY
by EPITECH

I. Introduction

Vous êtes un chasseur à l'arc confirmé depuis aussi longtemps que vous pouvez vous souvenir.

C'est une véritable passion qui vous anime, lorsque vous vous trouvez dans la nature, face à un canard, et que vous n'entendez que la bruit de votre poigne qui se resserre sur votre arc en voyant apparaître une proie dans votre ligne de mire.

Du moins c'était le cas avant, mais ces temps-ci vous vous sentez comme lassé des proies habituelles.

C'est alors que lors d'une balade en voiture en ville, passant devant le bâtiment Hololive, vous apercevez une horde d'oiseau passer juste au-dessus de vous. Mais vous reconnaissez à leur casquette que ces oiseaux n'ont rien de commun, ce sont des Shuba Duck !

Surpris par votre flamme qui se ranime en vue d'une proie aussi rare qu'inattendue, vous bondissez de votre voiture et agrippez votre fidèle arc que vous promenez partout afin de chasser du Shuba Duck.



** "Déjà vu; I've just come to this place before; higher on the street; and I know it's my time to go~*

Calling you; and the search is a mystery; standing on my feet; it's so hard when I try to be me Whooooooooaaaaa! ~"

II. Consignes

- * Vous allez devoir compléter des fonctions du jeu Pong sous Python avec la librairie Pygame (spoiler : ça va être dur section)
- * Si vous avez des questions, demandez à votre voisin de gauche, puis de droite. Coopérez !
- * En cas de blocage, et après avoir tenté la consigne précédente, n'hésitez pas à demander à un Cobra, ils sont très gentils et sont là pour vous aider.
- * Vous avez entièrement le droit d'utiliser internet comme vous le souhaitez pour vous renseigner, nous le conseillons fortement.
- * Si vous n'avez jamais fait de Python ou de programmation auparavant et que vous souhaitez apprendre quelques notions de base avant de vous lancer : demandez à un Cobra de vous assister et rendez [ici](#) pour apprendre les bases.
- * *N'hésitez pas à faire des bonus et ajouter/modifier des fonctionnalités à votre programme, la seule limite est votre imagination ! (Et 17h)*

- * Amusez-vous ! Nous ne sommes pas là pour apprendre mais pour découvrir, vous avez le droit de ne pas réussir à compléter le sujet. 😊

III. Shuba Duck, apparais !

Tous d'abord commencez par ouvrir Visual Code, puis ouvrez « CobraHunt.py ».

Si vous essayez de lancer le jeu, vous remarquerez un manque concernant d'activité à l'écran, ce qui est normal, car pour l'instant les Shuba Duck n'apparaissent pas encore. Enfin, si, mais en dehors de l'écran.

Nous possédons plusieurs variables dans notre jeu actuellement, voici celles qui nous serviront pour cette première partie :

`self.x` : la position sur l'axe X du Shuba Duck

`self.y` : la position sur l'axe Y du Shuba Duck

`window_height` : la taille en hauteur de la fenêtre

Et enfin une fonction qui nous sera indispensable :

`random.randint(valeur1, valeur2)` : génère une valeur aléatoire comprise entre valeur1 et valeur2.

Votre première tâche sera de, grâce à tous éléments réunis, faire apparaître les Shuba Duck dans notre écran à position Y aléatoire.

Dirigez-vous à cet endroit, puis enlevez les guillemets ainsi que le « PARTIE III » afin de coder à ce même emplacement ↓

```
def __init__(self, sprite):
    self.sprite = sprite
    self.width = sprite.get_width()
    self.height = sprite.get_height()
    .....
    PARTIE III
    .....
```

IV. Mason, on bouge

Notre Shuba Duck apparaît enfin sur votre écran, mais il manque grandement de vivacité. Il va falloir le faire bouger désormais.

Voici les variables dont vous allez avoir besoin dans cette partie :

`self.x` : la position sur l'axe X du Shuba Duck

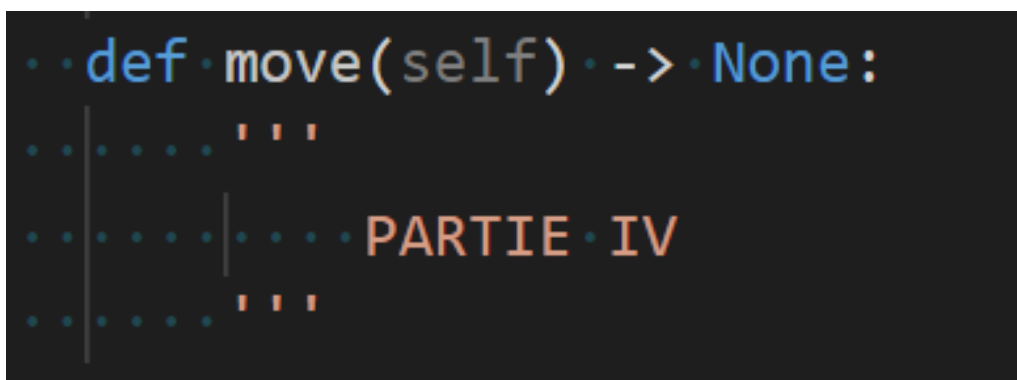
`self.speed` : la vitesse de vol du Shuba Duck

Pour l'instant `self.speed` vaut 0, notre canard n'avancera pas très vite. Vous allez tout d'abord devoir attribuer une valeur à la vitesse du Shuba Duck (dont vous déciderez vous-même) **au même endroit** et de la même manière **que la partie précédente**.

Maintenant que `self.speed` est supérieur à 0, vous devez coder la fonction qui déplacera Shuba Duck.

En utilisant `self.speed` et `self.x`, vous devriez être capable de changer sa position en X afin de simuler un mouvement.

Dirigez-vous ici pour coder cette partie ↓



V. Chassons du canard

Nous avons un Shuba Duck en pleine forme qui apparaît dans la fenêtre et qui se déplace à toute vitesse.

Toute fois votre objectif est, je vous le rappelle, de chasser les Shuba Duck, c'est de cela dont nous allons nous occuper dans cette partie.

a. On vise avant de tirer

Tout d'abord l'on doit vérifier que lorsque l'on clique, la souris se trouve bien sur un Shuba Duck.

Voici la liste des variables qui vous seront utiles :

`self.x` : la position sur l'axe X du Shuba Duck

`self.y` : la position sur l'axe Y du Shuba Duck

`self.width` : la taille du Shuba Duck en largeur (← →)

`self.height` : la taille du Shuba Duck en hauteur (↑ ↓)

`mouse_x` : la position sur l'axe X du curseur de votre souris

`mouse_y` : la position sur l'axe Y du curseur de votre souris

Grâce à ces variables, vous allez devoir créer une condition qui vérifie si la souris se trouve bien sur le Shuba Duck, et si c'est le cas, renvoyez `True`.

Sinon renvoyez `False`.

Dirigez-vous au même emplacement que dans la capture d'écran ci-dessous ↓

```
def contains(self, mouse_x: int, mouse_y: int) -> bool:
    """
    PARTIE V.a
    """
```

b. Décès du Shuba Duck 😞

Après avoir complété la fonction qui vous permet de détecter que votre souris se positionne bien sur un Shuba Duck, il faut maintenant s'occuper de faire disparaître le Shuba Duck touché du jeu et rajouter des points à votre score.

Voici la liste des éléments qui vous seront utiles :

`duck` : le Shuba Duck qui a été touché

`score` : le nombre total de votre score

`DuckList` : la liste de tous les Shuba Duck présent dans le jeu

`DuckList.remove(duck)` : une fonction qui permet d'enlever le `duck` fourni de la `DuckList`

Encore une fois, votre objectif ici est d'enlever le Shuba Duck touché du jeu, et de rajouter des points (autant que vous le désirez) à votre score.

Si vous avez du mal à comprendre comment utiliser `DuckList` ou `duck`, n'hésitez pas à demander de l'aide à Cobra !

Dirigez-vous au même emplacement que dans la capture d'écran ci-dessous ↓

```
for duck in DuckList:
    if duck.contains(pygame.mouse.get_pos()[0], pygame.mouse.get_pos()[1]):
        '''
        PARTIE V.b
        '''
```

VI. On vous avait dit de bien viser !

Vous êtes désormais dans la dernière ligne droite, qui ne sera que tâche aisée désormais grâce à vos compétences acquises.

Le jeu fonctionne assez bien, vous pouvez être fier. Mais il manque un peu de difficulté. En effet lorsque l'on rate un Shuba Duck, aucune pénalité ne vous est appliquée. C'est presque trop facile.

Vous allez devoir tout d'abord, vérifier que le Shuba Duck sort de l'écran. Si cette condition est respectée, il faudra enlever des points à votre score (autant que vous le désirez), et enlevez le Shuba Duck du jeu. (pour soulager un peu votre ordinateur)

Voici la liste des éléments dont vous devrez vous servir :

`duck.x` : la position sur l'axe X du Shuba Duck

`score` : le nombre total de votre score

`DuckList` : la liste de tous les Shuba Duck présent dans le jeu

`DuckList.remove(duck)` : une fonction qui permet d'enlever le `duck` fourni de la `DuckList`

Attention, vous avez probablement dû remarquer que `self.x` est devenu ici `duck.x`. C'est normal, ce n'est pas une erreur du sujet, si vous souhaitez comprendre pourquoi, n'hésitez pas à poser la question à un cobra.

Dirigez-vous au même emplacement que dans la capture d'écran ci-dessous ↓

```
· · for duck in DuckList:
· ·     · · duck.move()
· ·     · · duck.display(window)
· ·     · ·
· ·     · ·
· ·     · · PARTIE VI
· ·     · ·
· ·     · ·
```


VII. Bonus

Bien joué !

Vous êtes arrivé à la fin de ce sujet, et ce grâce à votre persévérance ainsi que vos compétences acquises durant vos efforts. Vous pouvez être fier !

Une fois que votre folie meurtrière vous est passée, vous vous rendez compte que les Shuba Duck se réunissent en cercle sur le sol dans un tourbillon, et font apparaître nul autre que Oozora Subaru, votre Vtuber préférée !

C'est alors que vous tombez de votre lit et vous comprenez que tout cela n'était qu'un rêve, dieu soit loué, vous n'avez jamais blessé le moindre animal durant toute votre vie.



Mais ce n'est pas fini, vous pouvez encore développer vos compétences en réalisant plusieurs bonus comme :

- * Pimper les Shuba Duck en changeant leur apparence ou en les animant un peu
- * Ajouter un peu de musique grâce à la librairie Pygame
- * Importer des images pour changer le fond d'écran sur votre jeu
- * Modifier certains paramètres comme le paterne de mouvement des Shuba Duck
- * Changez l'apparence de votre curseur grâce à la librairie Pygame
- * *Absolument n'importe quel autre bonus dont la seule limite est votre imagination !*