

Les dominos sont des objets qui possèdent des caractéristiques pouvant être évaluées :
Chaque domino est porteur de deux marques qui représentent les nombres de 0 à 6.
Mais aussi un comportement.

Pour créer un objet, commençons par créer une classe !

I. Une classe pour les instancier tous :

Une classe est une sorte de moule à partir duquel sont générés les objets.
Nous allons commencer par définir une classe d'objet que nous nommerons naturellement domino.

Le mot-clé `class` en Python est utilisé pour créer la définition d'une classe, la nomenclature d'une classe doit suivre le format CamelCase.

Pour cette première classe, lancer spyder, copiez le code ci-dessous et exécutez le :

```
class Domino:
    def __init__(self, A, B):
        """constructeur: méthode spéciale qui est appelée lors de la création d'une instance.
        Elle sert à initialiser un certain nombre d'attributs et à effectuer des opérations lors de l'instanciation d'une
        classe"""
        self.A = A
        self.B = B

    def __str__(self):
        """ La méthode spéciale __str__ permet d'indiquer la représentation en chaîne de caractères d'un objet.
        Cette chaîne de caractères est retournée lors de l'utilisation de la fonction str sur un objet ou lors de
        l'utilisation de la fonction print
        """
        return f"[{str(self.A)} | {str(self.B)}]"

    def valeur(self):
        """ Permet de calculer la somme des points du domino
        Returns INT
        """
        return self.A + self.B

    def peut_placer_apres(self, domino):
        """ Parameters domino : Domino
        Returns BOOL Est-ce que domino a un côté qui est le même que le côté vacant de self
        """
        return domino.B == self.A or domino.B == self.B
```

La fonction `__init__` est le constructeur : C'est la méthode qui "construit" l'objet lors du premier appel.

La méthode `__init__` nécessite ici deux paramètres (et non pas 3), correspondant aux deux marques du domino.
`self` est le nom du paramètre qui correspond à l'instance depuis laquelle la méthode est appelée dans la programmation orientée objet. Le nom de ce paramètre est une convention.

Une classe est constituée :

- d'un ensemble de variables (appelées attributs) qui décrivent la structure des objets ;
- d'un ensemble de fonctions (appelées méthodes) qui sont applicables aux objets, et qui décrivent leurs comportements.

Q1. Quels sont les attributs de la classe Domino ?

Q2. Quelles sont les méthodes de la classe Domino ?



II. *instanciation des objets dominos:*

L'opération de création d'un nouvel objet à partir d'une classe est appelée instanciation.

Q3. Instanciez ces trois dominos (3,4); (1,1) et (2,5) dans les variables référencées `d1`, `d2` et `d3`.

Q4. Quel est l'affichage produit par le programme suivant ?

```
print(d1)
d1.permute()
print(d1)
print(d1.plus_grand(d2))
print(d1.plus_grand(d3))
```

Remarque: Les classes permettent de définir de nouveaux types Que renvoie l'instruction `type(d1)` ?

III. *méthode réservée équivalent:*

Q5. Ajouter la method `permute` qui échange les valeurs A et B du domino

Q6. Ajouter la method spéciale `équivalent`

```
def __eq__(self, Domino):
    ''' Domino, Domino -> bool
    Teste si deux dominos sont egaux. '''
    if (self.A == Domino.A and self.B == Domino.B) or (self.A == Domino.B
and self.B == Domino.A):
        return
    True return
    False
```

Testez la: Par exemple, le Domino (3,4) est le même que le Domino (4,3)

Q7. Ecrivez la méthode `est_double` qui teste si un domino est un double en renvoyant une valeur booléenne adéquate.

IV. *Pour les plus rapide : Créer un jeu de dominos normal*

Nous allons maintenant générer un jeu de dominos. Pour se faire, nous allons créer tous les dominos d'un *jeu normal de dominos*, c'est à dire l'ensemble des dominos dont les marques sont de "blanc" à "six" et sans que deux dominos ne portent des marques identiques de quelque sens qu'on les regarde.

Par exemple, le domino (3,4) est le même que le domino (4,3) et il ne doit apparaître qu'une seule fois dans le jeu.