

Exo1

1.1 Identifiez les éléments suivants: 1) "du constructeurs", 2) "class", 3) "attributes", 4) "methods" et 5) "state" dans le code ci-dessous (CODE Exo1). Par exemple, pour class, on attend la réponse Dog.

1.2. Modifiez le code afin que la race du chien devienne un attribut de type "string" fixé lors de l'instanciation de la classe.

1.3. Modifiez le code afin que l'humeur du chien devienne un entier entre 0 (grincheux) et 10 (heureux). Lors de l'instanciation du chien la valeur est à 5. Créez une méthode "eat" qui incrémente l'humeur de 3 (l'humeur plafonne à 10) et une méthode "leash" (mettre la laisse) qui diminue l'humeur de 1.

CODE Exo1

```
class Dog:
    """
    This is a dog
    """

    def __init__(self, name):
        self.name = name
        self.tricks = [] # creates a new empty list for each dog

    def add_trick(self, trick):
        self.tricks.append(trick)

    def bark(self, intensity=3):
        if intensity==1:
            print('whouaf')
        elif intensity==2:
            print('Whouaf!')
        else:
            print('WHOUAF!')

fido = Dog('Fido')
buddy = Dog('Buddy')
fido.add_trick('roll over')
fido.add_trick('play dead')
buddy.bark(2)
```

Exo2

Soit le code ci-dessous "CODE Exo2":

- 2.1. En utilisant la classe "Dog", instanciez trois nouveaux chiens, chacun avec un âge différent.
- 2.2. Ecrivez une fonction appelée `get_biggest_number()`, qui prend un nombre quelconque d'âge (`*args`) et renvoie le plus ancien.
- 2.3. Indiquez ensuite l'âge du chien le plus âgé comme suit: Le chien le plus âgé a 7 ans.
- 2.4. définir une méthode d'instance 'description' qui renvoie le nom et l'âge du chien
- 2.5. définir une méthode d'instance 'speak' qui prend comme argument un son et renvoie le nom du chien et le son quelle forme?

CODE Exo2:

```
class Dog:
    # Class Attribute
    species = 'mammal'

    # Initializer / Instance Attributes
    def __init__(self, name, age):
        self.name = name
        self.age = age
```

Exo3

Soit le code ci-dessous "CODE Exo3":

- 3.1. Write a child class `Teacher`, which inherits from the `SchoolMember` class with an additional argument `salary`
- 3.2. Ecrivez une classe "Student" qui hérite de la classe `SchoolMember` avec un argument supplémentaire "note"
- 3.3. Pour "Teacher" et "Student", ajoutez une méthode 'tell' qui appelle la méthode `tell` de la classe parente et affiche le salaire pour un "Teacher" et la note pour un "Student".

CODE Exo3:

```
class SchoolMember:
    """Represents any school member."""
    def __init__(self, name, age):
        self.name = name
        self.age = age
        print('(Initialized SchoolMember: {})'.format(self.name))

    def tell(self):
        """Tell my details."""
        print('Name:"{}" Age:"{}"'.format(self.name, self.age), end=" ")
```

```

class Teacher(SchoolMember):
    """Represents a teacher."""
    def __init__(self, name, age, salary):
        super().__init__(self, name, age)
        self.salary = salary
        print('(Initialized Teacher: {})'.format(self.name))

    def tell(self):
        SchoolMember.tell(self)
        print('Salary: "{:d}"'.format(self.salary))

```

Exo4

Construisez une classe "Student" qui gère les informations suivantes pour un étudiant: nom, année académique et liste de notes.

1. Définissez un constructeur qui initie un nouvel étudiant avec un nom et l'année académique.
2. Écrivez une méthode pour obtenir le nom et l'année académique d'un étudiant.
3. Écrivez une méthode pour ajouter une nouvelle note.
4. Implémentez une méthode qui calcule la note moyenne.
5. Ajoutez une nouvelle méthode qui fait passer l'étudiant (c'est-à-dire que l'étudiant passe à l'année suivante et commence avec de nouvelles notes) si la note moyenne est supérieure à un seuil.

Exo5

Définissez une classe "Classe" contenant des étudiants. Il est possible d'ajouter de nouveaux étudiants à une classe, d'obtenir la liste des noms d'étudiants, ainsi que de récupérer le nom et la note finale du meilleur étudiant.

1. Construisez le constructeur pour lancer une nouvelle classe sans élève.
2. Implémentez des méthodes pour ajouter un nouvel étudiant et obtenir la liste du nom des étudiants ajoutés.
3. Implémentez une méthode qui renvoie le nom et le grade du meilleur élève d'une classe.