In [1]:
```
# Objects are collection of properties(which can be expressed as variables) and/c

# The (variable/properties) within an object are known as (instance variables/att
# the (functions) within an object are known as (methods)

# once an object is set, it can be put into a variable
```

In [2]:
```
# CLass= its a blue print from which you can make up an object
```

In [3]:
```
# Constructor = a function that allows you to create an object from a class
```

In [11]:
```
class Robot:
    def introduce_self(self):
        print("My name is", self.name)
```

In [12]:
```
 # creating an object from above class

r1 = Robot()
r1.name = "Tom"
r1.color = "Red"
r1.weight = 30

r2 = Robot()
r2.name = "Jerry"
r2.color = "Blue"
r2.weight = 40
```

In [13]:
```
r1.introduce_self()
r2.introduce_self()
```

```
My name is Tom
My name is Jerry
```

In [35]:
```
# Creating a Constructor
class Robot:
    def __init__(self, name, color, weight):
        self.name = name
        self.color = color
        self.weight = weight

    def introduce_self(self):
        print("My name is", self.name)
```

In [36]:
```
r1 = Robot ("Tom", "Red", "30")
r2 = Robot ("Jerry", "Blue", "40")
```

In [37]:
```
r1.introduce_self()
r2.introduce_self()
```

```
My name is Tom
My name is Jerry
```

In [38]:
```
# bool(boolean)== true or false statements
```

In [39]:
```
#Introducing another Class

class Person:
    def __init__(self, name, personality, issitting):
        self.name = name
        self.personality = personality
        self.issitting = issitting
    def sit_down(self):
        self.issitting = True
    def stand_up(self):
        self.issitting = False
```

In [40]:
```
p1 = Person("Alice", "aggressive", False)
p2 = Person("Becky", "talkative", True)
```

In [41]:
```
# to show robot owned
p1.robot_owned = r2
p2.robot_owned = r1
```

In [43]:
```
#confirming
p1.robot_owned.introduce_self()
```

```
My name is Jerry
```

In [ ]:
```
```