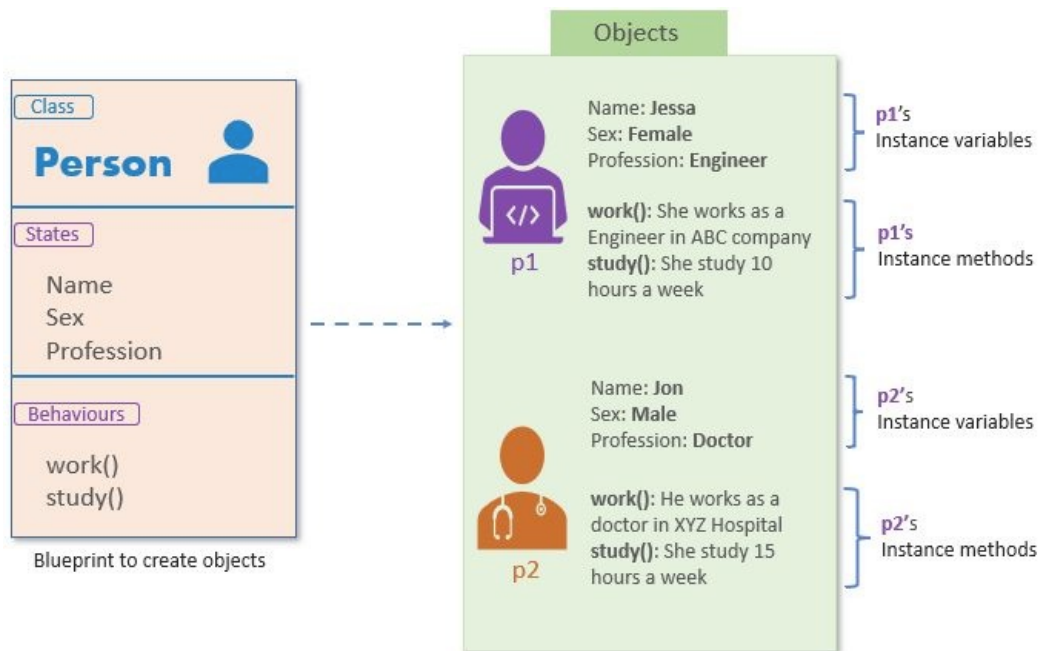


Class and Object

Class is like a blueprint to object. Class is like code template for object creation.



Example:

```
In [1]: #lets see how to create class practically
```

```
class Person:
    def __init__(self,name,sex,profession):
        self.name=name
        self.sex=sex
        self.profession=profession
```

```
#if we do this much we just created class
```

In above we just create the class with name Person and we create states or property of that class-name,sex,profession now, we will create behavior or methods of class.

```
In [3]: class Person:
        def __init__(self,name,sex,profession):
            self.name=name
            self.sex=sex
            self.profession=profession

        #we created methods
        def printshows(self):
            print("His/Her name is",self.name, "and her sex is:",self.sex)
```

```
In [5]: #Above we created blueprint with name Person which is class, inside that we keep few variables name,
        # sex and profession and we create methods like function
```

```
In [9]: #Now lets create object
        person1=Person("ram", "male", "army")
```

```
In [10]: print(person1)

<__main__.Person object at 0x000001E2DD397940>
```

```
In [11]: print(person1.sex)

male
```

```
In [12]: #now lets use with methods
        person1.printshows()

His/Her name is ram and her sex is: male
```

SECOND EXAMPLE

```
In [13]: #lets create class student
```

```
class Students:
    def __init__(self,name,age,gender,section="A"):
        self.name=name
        self.age=age
        self.gender=gender
        self.section=section
```

In [15]: *#we have given "A" section as default but we can modify it.
#now lets create methods*

```
class Students:
    def __init__(self,name,age,gender,section="A"):
        self.name=name
        self.age=age
        self.gender=gender
        self.section=section

    def shows(self):
        print("name:",self.name,"age:",self.age,"gender:",self.gender,"section:",self.section)
```

In [16]: *#now lets create objects, above we create the blue print with attributes and methods*

In [17]:

```
student1=Students("ram",21,"Male")
student2=Students("geeta",32,"Female","B")
```

```
print(student1.name)
print(student2.section)
print(student1.section)
```

```
ram
B
A
```

In [18]: *#now lets run methods*

```
student1.shows()
```

```
name: ram age: 21 gender: Male section: A
```

In [21]: *#This above is very basic example of OOP- class, objects and methods. Now, lets do slightly advance which have #inside methods.*

```
class Room:
    def __init__(self,length,breadth):
        self.length=length
        self.breadth=breadth

    def area(self):
        area=self.length*self.breadth
        print("Area of room is: ",area)
```

In [22]: *#now lets create object*

```
studyroom=Room(31.5,40)
livingroom=Room(25,40)

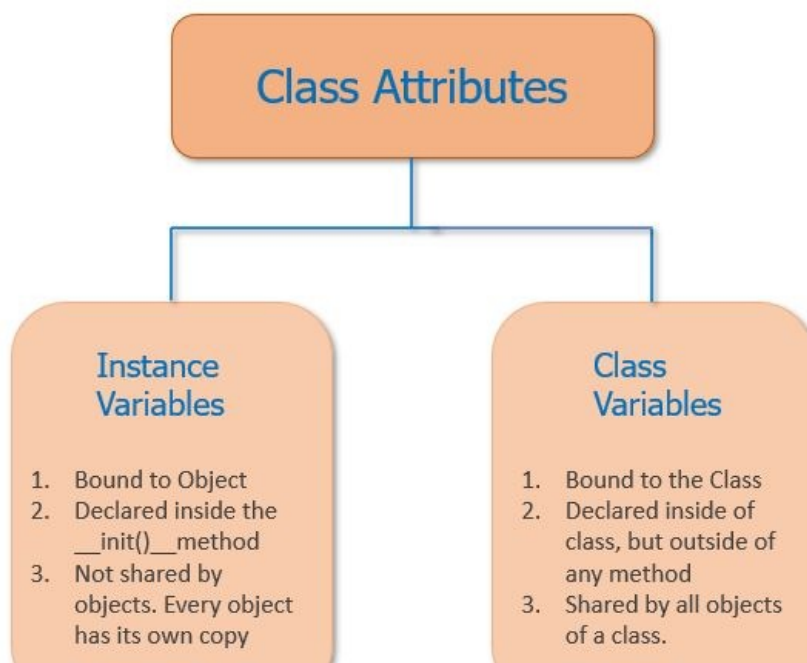
print(studyroom.length)
print(livingroom.breadth)

livingroom.area()
```

```
31.5
40
```

```
Area of room is: 1000
```

What is instance and class variables



In [1]: *#lets see with simple example*
#create class

```
class student:
    #class variable
    schoolname="Pentagon"

    #constructor
    def __init__(self,name,age,gender):
        #instance variable
        self.name=name
        self.age=age
        self.gender=gender

    #methods
    def show(self):
        print("He/She is",self.gender,"and his/her name is",self.name)

#create object
student1=student("ram",21,"male")

print(student1.gender)
```

male

In [2]: student1.show()

He/She is male and his/her name is ram

In [4]: *#now what about class variable, how we can use that?*
print(student1.schoolname)

Pentagon

In [5]: *#if we want to use it in methods?*

```
class student:
    #class variable
    schoolname="Pentagon"

    #constructor
    def __init__(self,name,age,gender):
        #instance variable
        self.name=name
        self.age=age
        self.gender=gender

    #methods
    def show(self):
        print("He/She is",self.gender,"and his/her name is",self.name,"also school name is",student.schoolname)

#create object
student1=student("ram",21,"male")
```

In [6]: student1.show()

He/She is male and his/her name is ram also school name is Pentagon