EASYLEARNING.GURU

Classes

What is Class?

EASYLEARNING.GURU

Class

A user defined prototype for an object that defines a set of attributes that characterize any object of the class. The attributes are data members and methods.

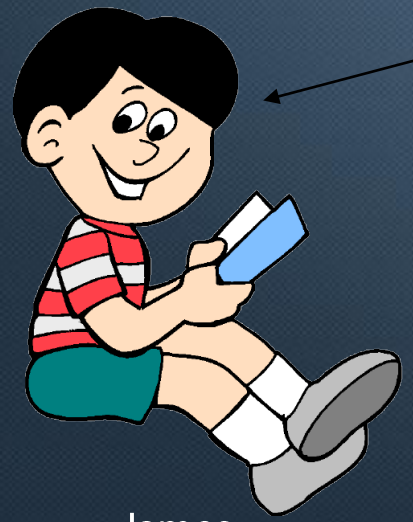
Instance

A user defined prototype for an object that defines a set of attributes that characterize any object of the class. The attributes are data members and methods.

Syntax

```
class class_name:
statement1
statement2
statement3
statement4
etc.
```

Class: Student
Attributes: Name and Rollno
Methods: setName() and setRollno()



James
110

John
114

Sam
142

Maria
122

Why Classes are useful?

EASYLEARNING.GURU

Multiple instances

- Classes are essentially factories for generating one or more objects. Every time we call a class, we generate a new object.

Customization via Inheritance

- We can extend a class by redefining its attributes outside the class in new components coded as subclasses.

Operator overloading

- Operator overloading allows same operator to have different meaning according to the context.

Hands-On

Class Attributes

EASYLEARNING.GURU

1

Data member

A class variable or instance variable that holds data associated with a class and its objects.

Class variable

A variable that is shared by all instances of a class.

Instance Variable

Defined inside a method and belongs only to the current instance of a class.

Class Attributes

EASYLEARNING.GURU

2

Method

A special kind of function that is defined in a class definition.

```
class class_name:
        data=42                  #class variable
        def getdata(self):      #method
            data = 40           #instance variable
#'self' references the instance object being
processed
```

Example

EASYLEARNING.GURU

Define a class

```
>>>class First:
        def setdata(self , value):
            self.data = value
        def getdata(self):
            print self.data
```

Creating instances of class

```
>>>x = First()
>>>y = First()
```

Example

EASYLEARNING.GURU

Calling methods

```
>>>x.setdata('hello')
>>>y.setdata(21)
>>>x.getdata()
     hello
>>>y.getdata()
     21
```

Setting attribute outside class

```
>>>x.data = 'hi!'
>>>x.getdata()
hi!
>>>x.anothername = 'it is
                   possible'
>>>x.anothername
'it is possible'
```

Hands-On