Object Oriented Programming

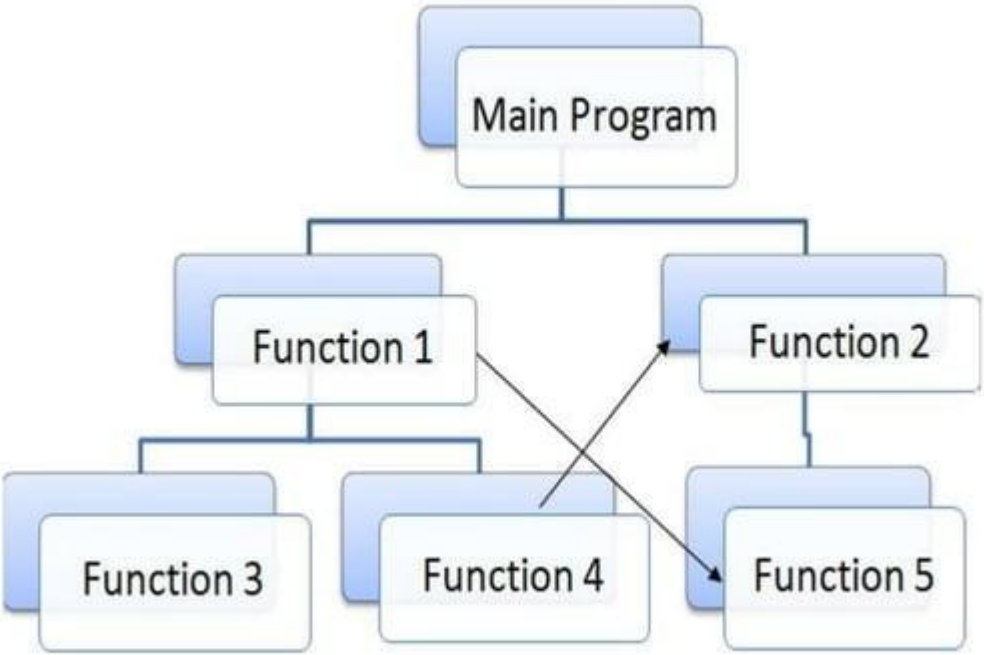
Introduction to Java

Lecture# 00

OOP

Procedure Oriented Programming

- It means "***a set of procedures***" which is a "***set of subroutines***" or a "***set of functions***".
- functions are called repeatedly in a program to execute tasks performed by them. For example, a program may involve collecting data from user (reading), performing some kind of calculations on the collected data (calculation), and finally displaying the result to the user when requested (printing). All the 3 tasks of reading, calculating and printing can be written in a program with the help of 3 different functions which performs these 3 different tasks.


Main Program
Function 1
Function 2
Function 3
Function 4
Function 5

A Real-World Example

- Let's say that you are working for a vehicle parts manufacturer that needs to update it's online inventory system. Your boss tells you to program two similar but separate forms for a website, one form that processes information about cars and one that does the same for trucks.
- **For cars,** we will need to record the following information:
  - Color, Engine Size, Transmission Type, **Number of doors**
- **For trucks**, the information will be similar, but slightly different. We need:
  - Color, Engine Size, Transmission Type, **Cab Size, Towing Capacity**

Scenario 1

- Suppose that we suddenly need to add a bus form, that records the following information:

  Color, Engine Size, Transmission Type,

  Number of passengers

- ***Procedural:*** We need to recreate the entire form, repeating the code for Color, Engine Size, and Transmission Type.
- ***OOP:*** We simply extend the vehicle class with a bus class and add the method,numberOfPassengers.

Scenario 2

- Instead of storing color in a database like we previously did, for some strange reason our client wants the color emailed to him.
- ***Procedural:*** We change three different forms: cars, trucks, and buses to email the color to the client rather than storing it in the database.
- ***OOP:*** We change the color method in the vehicle class and because the car, truck, and bus classes all extend (or inherit from, to put it another way) the vehicle class, they are automatically updated.

Scenario 3

- We want to move from a generic car to specific makes, for example: Nissan and Mazda.
- ***Procedural:*** We create a new form for each make, repeating all of the code for generic car information and adding the code specific to each make.
- ***OOP:*** We extend the car class with a Nissan class and a Mazda class and add methods for each set of unique information for that car make.

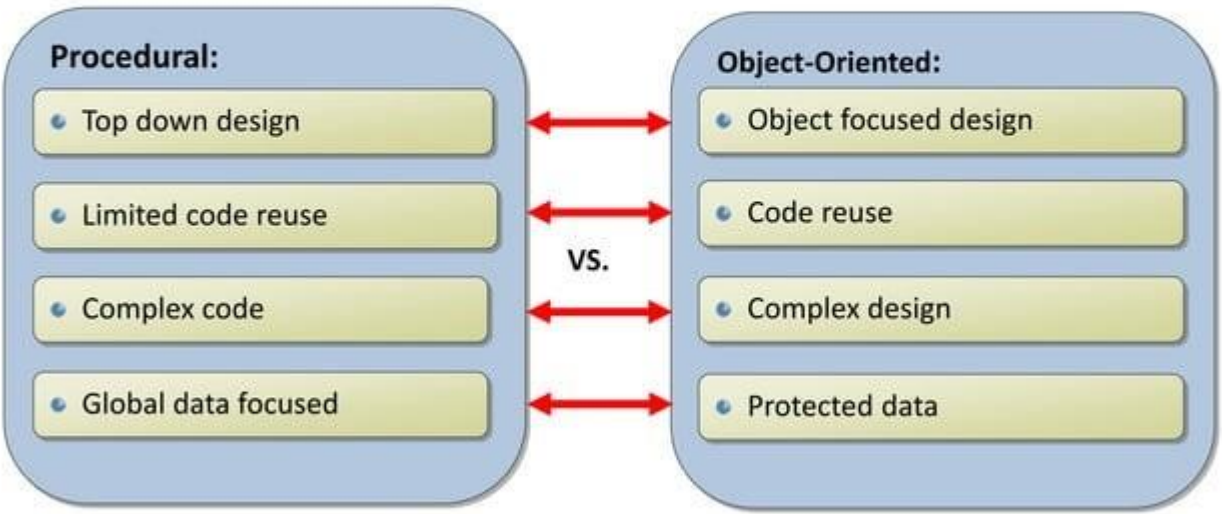
Scenario 4

- We found a bug in the transmission type area of our form and need to fix it.
- ***Procedural:*** We open and update each form.
- ***OOP:*** We fix the transmission Type method in the vehicle class and the change perpetuates in every class that inherits from it.

Procedural vs. Object-Oriented Programming

POP	OOP
In POP, program is divided into small parts called **functions**.	In OOP, program is divided into parts called **objects**.
POP does not have any proper way for hiding data so it is **less secure**.	OOP provides Data Hiding so provides **more security**.
Example of POP are : C, VB, FORTRAN, Pascal.	Example of OOP are : C++, JAVA, VB.NET, C#.NET.

Procedural vs. Object-Oriented Programming



OOP

- **Object Oriented Methodology** is a certain **process** through which software can be developed. The **goals** of this methodology are to achieve Software Systems that are **reliable**, **reusable**, **extensible**; hence, more useful in the long run. The methodology achieves its goals by the help of a *collection of objects that communicate by exchanging messages*.

What is Object-Orientation?

- A technique for system modeling
- OO model consists of several interacting objects

What is a Model?

- A model is an abstraction of something
- Purpose is to understand the product before developing it

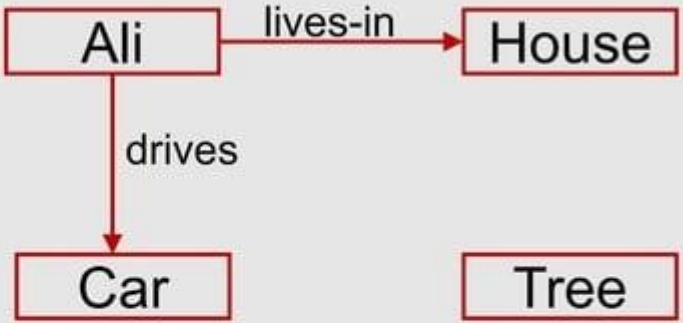
Examples – Model

- Highway maps
- Architectural models
- Mechanical models

Example – OO Model

...Example – OO Model

- **Objects**
  - Ali
  - House
  - Car
  - Tree
- **Interactions**
  - Ali lives in the house
  - Ali drives the car



Object-Orientation - Advantages

- People think in terms of objects
- OO models map to reality
- Therefore, OO models are
  - easy to develop
  - easy to understand

OOP

- **Advantage of Object Oriented Programming**

- a) Objects are modeled on real world entities.
- b) This enables modeling complex systems of real world into manageable software solutions.

OOP

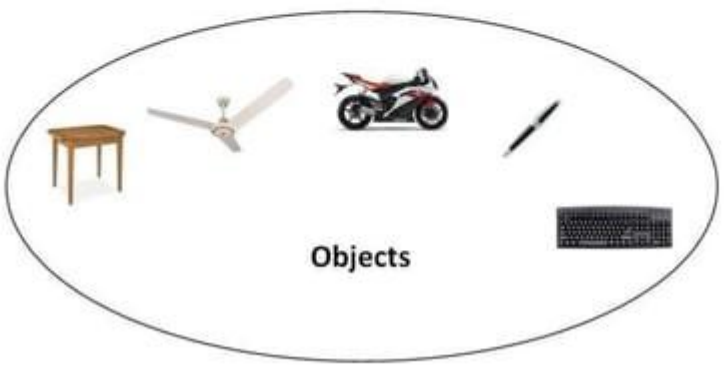
- Java is an Object-Oriented Language. As a language that has the Object-Oriented feature, Java supports the following fundamental concepts –
  - Classes
  - Objects
  - Instance
  - Method
  - Message Parsing
  - Polymorphism
  - Inheritance
  - Encapsulation
  - Abstraction

Object in Java

- An entity that has state and behavior is known as an object e.g. chair, bike, marker, pen, table, car etc. It can be physical or logical (tangible and intangible). The example of intangible object is banking system.
- An object has three characteristics:
  - **state:** represents data (value) of an object.
  - **behavior:** represents the behavior (functionality) of an object such as deposit, withdraw etc.
  - **identity:** Object identity is typically implemented via a unique ID. The value of the ID is not visible to the external user. But, it is used internally by the JVM to identify each object uniquely.

Object in Java

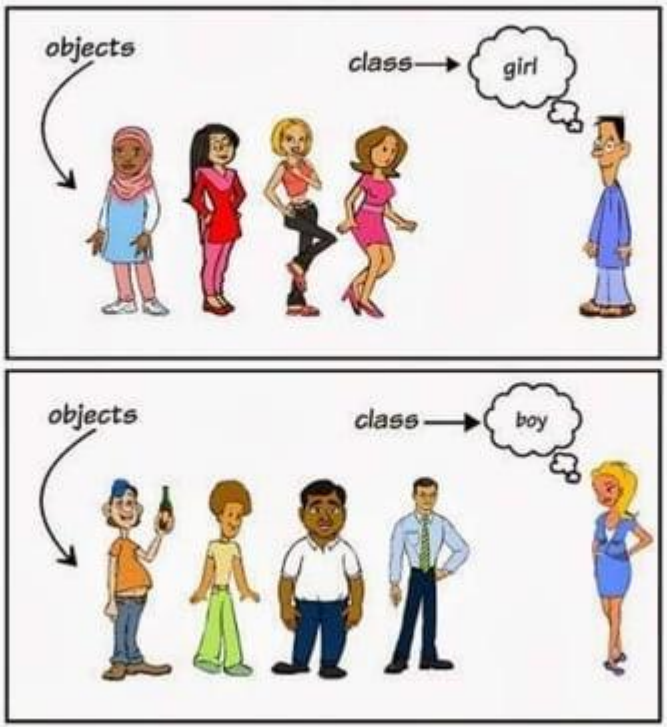
- For Example: Pen is an object. Its name is Reynolds, color is white etc. known as its state. It is used to write, so writing is its behavior.
- **Object is an instance of a class.** Class is a template or blueprint from which objects are created. So object is the instance(result) of a class.



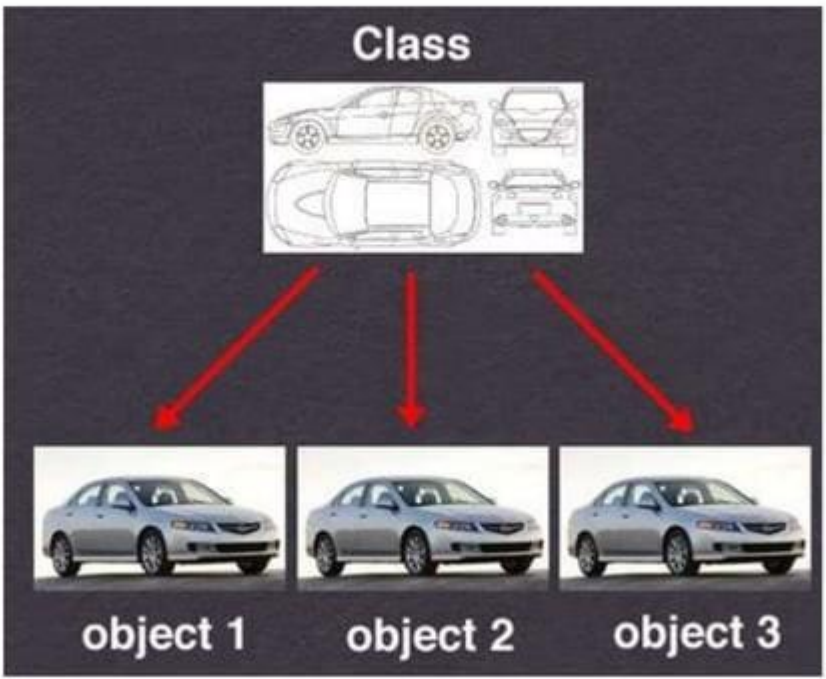
Class in Java

- A class is a group of objects that has common properties. It is a template or blueprint from which objects are created.
- A class in java can contain:
  - **data member**
  - **method**
  - **constructor**
  - **block**
  - **class and interface**

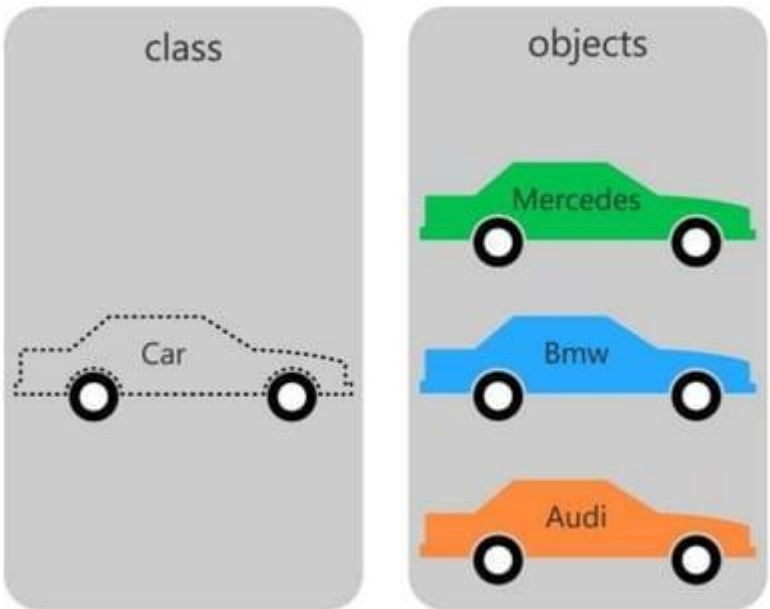
Object and Class in Java



Object and Class in Java

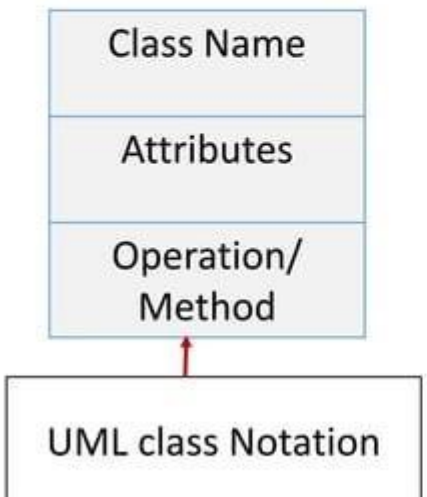


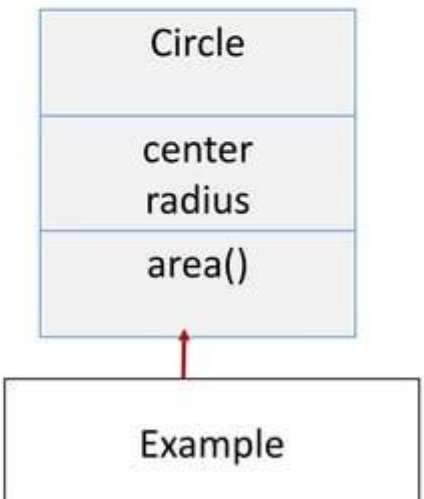
Object and Class in Java



Class in Java

- A *class* is a collection of *fields* (data) and *methods* (procedure or function) that operate on that data.





Class in Java

- A class is a user defined data type.
- The instance of the class are called **objects**.
- Class is a place where we can define the properties and functionalities of the objects.

Class Declaration in Java

- Declaration of class must start with the **keyword *class*** followed by the class name and class members are declared within braces.
- **Syntax of declaring class**

```
access specifier class class_name
  {
  // some data/some fields
  // some functions/methods
  }
```