

Lecture 1: -- Objects and Classes: -- JAVA

Procedural Programming (POP)

- Derived from Structural Programming
- A program is divided into small parts called functions.
- Procedural programming follows top down approach
- Function is more important than data
- Based on unreal world.
- No proper way for hiding data, so it is less secure.
- Adding new data and function is not easy
- Examples: C, FORTRAN, Pascal, Basic etc

Object Oriented Programming (OOPs)

- OOPs is based on real world.
- Program is divided into small parts called objects.
- Object oriented programming follows bottom up approach.
- Data is given more importance than functions
- Data hiding is possible and hence more secure.
- Adding new data and function is much easier
- Examples: C++, Java, Python, C# etc.

❖ What are Objects?

What are Objects?

Bank Account
 Prop → Account Type, Number, Balance, Interest
 Behaviour → Credit, Debit

- Anything that has a state and behaviour is known as an object.
- Objects can be:
 Physical: Car, Laptop, Pen etc.
 Logical: Intangible object, like in some banking system.
- They store data in form of variables and operate on this data using methods/functions.
- An object is a real-world entity.
- An object is an instance of a class.



❖ What are Classes?

What are Classes?

➤ A class is a group of objects which have common properties.

➤ A class can be seen as a blueprint from which objects are created.

➤ For example: A car can be a class and we can have various objects. Each car object has attributes(properties), such as weight and color, and methods (behavior), such as accelerate and brake.

Classes

Classes can contain:

Bank Account

*{ Name
Number*

*Address → Pin code
→ State*

*{
→ Landmark
→ Viable Pin code*

- Fields
- Methods
- **Constructors** *to initialize variables*
- **Nested class and interface**

Creation of Objects of a class (Instantiation of a class):

- **Known as instantiation of a class**
- **Objects share the attributes and the behavior of the class**
- **Values of those attributes may be unique for each object**
- **A class may have any number of instances**

