Object-Oriented Programming

➢ **Principles Of Object-Oriented Programming**

- Principles of Object-Oriented programming are
- 1.Abstraction.
- 2.Encapsulation.
- 3.Inheritance.
- 4.Polymorphism.

1.Abstraction

- Abstraction means hiding internal details and showing the required things.

For Example

Consider a man driving a car, while driving he focus on using of steering, gear, accelerator etc.
He does not require to know the inner mechanism of the car.

2.Encapsulation

- Encapsulation is the process of grouping data in a single section.

For Example

Complete television is single box where all the mechanism are hidden inside the box all are capsuled.

3.Inheritance

- Inheritance means designing an object or a class by re-using the properties of the existing class and object.
- Inheritance is same as specialization.

For Example

A old style television (idiot box) is transformed with extra features into slim and smart television where it re-used the properties of old television.

- **4.Polymorphism**

- Polymorphism is a concept in which we can execute a single operation in different ways.
- polymorphism is same as generalization.

➢ **Class VS Object**

- Object is defined in terms of its **properties** and **behaviour.**
- Operation of behaviours will affect the properties.
- Anything in the world can be defined in the terms of properties and behaviour.
- For a single class wee can have many objects.
- Multiple number of objects can be created by one single class

For example

A house or a car or a television is an object but the design or blueprint of the objeect is a class.

Example Program

```
class Television

{
 private int channel;
 private int volume;

 public void changechannel()
 public void changevolume()
}
class test
{
 public static void main()
  {
   Television t=new Television();
   t.changechannel(10);
  }
}
```

- In java there is an area inside main memory which is known as method area which contains all the methods.
- The definitions of the will be present inside the heap, as the objects will be based on the definitions so the objects are also present in heap.

➢ **Data Hiding**

- Usually data is hidden and the operations are made visible and operations or methods are performed over the data

For example

Actual operation of the television is performed in the circuitry which is done by pressing a button.so the circuitry is data and operations are methods where the data is hidden inside the box.

Example Program

```
class Rectangle

{
  public int length;
  public int breadth;

  public int area()
  {
    return length*breadth;
  }
  public int perimeter ()
  {
    Return 2*(length+breadth);
  }
class test
{
 public static void main()
 {
  Rectangle r=new Rectangle();
 }
}
```

- In above example there is given two data members length and breadth which are the properties of the class.
- And the area and perimeter is the method of the class where both the methods are performing the operations on the given data.
- For hiding the data, the data members will have the stricter (private) success modifier.
- So, when the data is made private, we can’t access that data outside the class.

➢ <u>**Types of Properties**</u>

- Read and writable property.
- getLength() method will allow us to read the property and setLength() method will allow us to write the property .
- Read only property.
- When there is no modification to the property then read only property is used.
- Write only property.

- Only set method is used for writing the property where no get method is used.

➢ Constructors

- A method is required for Initialization of properties at the time of construction of an object, this method is known as constructor.
- Constructor is a method of class called when an object is created.
- Every class will have a default constructor provided by java compiler.
- Constructor will not have any return type.
- There are two types of constructors
    - 1. parameterized
    - 2. Non-parameterized.
- Non-parameterized constructors is a replacement for default constructors.
- Constructors can be overloaded.