FACULTY OF Engineering & Technology

Object-Oriented Programming is a methodology for designing a program using classes and objects. Features of OOPS:

1. Object
2. Class
3. Inheritance
4. Polymorphism
5. Abstraction
6. Encapsulation
7. Coupling
8. Cohesion
9. Association
10. Aggregation
11. Composition

Class:

1. *Class is a collection of similar type of objects* .
2. It is a logical entity.
3. A acts as a blueprint from which you can create an individual object

Objects:

1. An object is a run time entity
2. It is also an instance of the class.
3. An object contains an address and takes up some space in memory.

Example: A cat is an object because it has states like color, name, breed, etc

Inheritance:

1. Inheritance is the mechanism of acquiring the properties and behaviour of a class by another class, retaining similar implementation.
2. Also defined as deriving new classes from existing ones such as super class or base class and then forming them into a hierarchy of classes
3. It is used to achieve runtime polymorphism.

Polymorphism:

1. If *one task is performed in different ways*, it is known as polymorphism.
2. **Polymorphism** is the ability of an object to take on many forms.
3. Any Java object that can pass more than one IS-A test is considered to be **polymorphic**
4. Polymorphism means "many forms", and it occurs when we have many classes that are related to each other by inheritance.

Abstraction

1. *Hiding internal details and just showing functionality* is known as abstraction.
   example phone call, we don't know the internal processing.
2. In Java, abstract class and abstract interface are used for abstraction

Encapsulation

1. Wrapping up of data members and methods in a single unit called class is called Encapsulation
2. example, as like a capsule, it is wrapped with different medicines.

Coupling

1. Coupling refers to the knowledge or information or dependency of another class.