OBJECT

\* An entity that has state and behavior is known as object

E.g:- Chair, Bike, Pen Table, car

* Objects are instances of classes, which serve as blueprint for creating objects. Each object has its own State (Attributes) and behavior (Methods)

Inheritance

* It is a mechanism in which one object acquires all the properties and behaviors of parent object
* Inheritance represents the IS-A relationship which is also known as a parent – child relationship
* Sub class/ child class: sub class is a type of class that takes on the features of another class.it gets all the properties and behaviors of the other class, referred to as the parent class.
* Super class is also known as parent class, the subclass inherits all the properties and behaviors from the super class

Polymorphism

* Polymorphism in java is like doing one thing in many ways. The word comes from Greek – “poly” means many, and “morphs” means forms. So polymorphism is about having many forms for single action.

Abstraction

* **Abstraction**is a process of hiding the implementation details and showing only functionality to the user.
* A class which is declared as abstract is known as an abstract class.
* An abstract class must be declared with an abstract keyword.
* It can have abstract and non- abstract methods.
* It cannot be instantiated.
* It can have constructors and static method also.
* It can have final methods

Encapsulation

* Encapsulation in java is a process of wrapping code and data together into a single unit, for example, a capsule which is mixed of several medicines.
* We can create a fully encapsulated class in java by making all the data members of the class private. Now we can use setter and getter methods to set and get the data in it.

Association

* It is a connection or relation between two separate classes.
* It has a Has- A relationship.
* There are four types of association between the objects:

One to one

One to many

Many to one

Many to many

* E.g: one country can have one prime Minister (one to one) and a prime minister can have ministers (one to many). Also, many MP’s can have one prime minister (many to one), and many ministers can have many departments (many to many).

Aggregation

* Aggregation in java is like having connection b/w two things where one thing contains the other as a part of itself but the relationship is not too strong.
* It is also termed as a has-a relationship in Java.
* Where one thing can have another as part, but if the main thing goes away, the part can still exist on its own

Composition

* There is a strong relationship between the containing object and the dependent object.
* If we delete the parent object, all the child objects will be deleted automatically.
* E.g: House- rooms