'A Report on Java objects & class and its Properties'

Introduction:

Java is a programming language and a platform. Java is a high level, robust, object-oriented and secure programming language. It is platform independent and is used for building web-applications, stand-alone apps etc.

Class and Objects:

A class is a group of objects which have common properties. It is a template or blueprint from which objects are created. It is a logical entity. It can't be physical.

A class in Java can contain:

i) Fields
ii) Methods
iii) Constructors
iv) Blocks
v) Nested class and interface.

Objects are the real world entities. In Java, an object is created from a class. We have already created the class named 'Main', so now we can the objects.

When an object of a class is created, the class is said to be instantiated. All the instances share the attributes and the behaviour of the class. But the values of those attributes. A single class may have any number of instances.

Inheritance:

Inheritance in Java, is a mechanism in which one object acquires all the properties and behaviours of a parent object. It is an important part of OOP.

The idea behind inheritance in Java is that you can create new classes built upon existing classes. When you inherit from an existing class, you can reuse methods and fields of the parent class. Moreover, you can add new methods and fields in your current class also.

Inheritance represents the IS-A relationship which is also known as a parent-child relationship.

Polymorphism :

Polymorphism in Java is a concept by which we can perform a single action in different ways. Polymorphism is derived from 2 Greek-words : poly and morphs. The word "poly" means many and "morph" means forms. So polymorphism means many forms.

There are two types of polymorphism in Java : compile-time polymorphism and runtime polymorphism. We can perform polymorphism in java by method overloading and method overriding.

In Java polymorphism is mainly divided into two types : i) Compile-time Polymorphism.
ii) Runtime-time Polymorphism.

i) Compile-time Polymorphism :

It is also known as static polymorphism. This type of polymorphism is achieved by function overloading or operator overloading.

ii) Runtime Polymorphism :

It is also known as dynamic method dispatch. It is a process in which a function call to the overridden method is resolved at runtime. This type of polymorphism is achieved by Method overriding.

Encapsulation :

Encapsulation is one of the four fundamentals OOP concepts. The other three are inheritance, polymorphism and abstraction.

Encapsulation in Java is a mechanism of wrapping the data and code acting on the data together as a single unit. In encapsulation, the variables of a class will be hidden from other classes, and can be accessed only through the methods of their current class. Therefore it is known as data hiding.

Abstraction :

Data abstraction is the process of hiding certain details and showing only essential information to the user. Abstraction can be achieved with either abstract classes or interfaces.

The 'abstract' keyword is a non-access modifier, used for classes and methods :

a) Abstract class : it is a restricted class that cannot be used to create objects.

b) Abstract method : It can only be used in an abstract class, and it does not have a body. The body is provided by the sub-class.