1) What are the OOPs concepts ?

Answer: OOPs concepts include:

Inheritance

Encapsulation

Polymorphism

Abstraction

Interface

2) What is Inheritance?

Answer: Inheritance means one class can extend to another class. So that the codes can be reused from one class to another class.

The existing class is known as the Super class whereas the derived class is known as a sub class.

3) What is Encapsulation?

Answer: Purpose of Encapsulation :- Protects the code from others. Code maintainability.

In order to overcome the problem we need to follow the steps below: We can make the variable private or protected.

Use public accessor methods such as set<property> and get<property>.

4) What is Polymorphism?

Answer: Polymorphism means many forms.

A single object can refer to the super-class or sub-class depending on the reference type which is called polymorphism.

Using the Manipulation reference type we can call the Addition class “add()” method.

This ability is known as Polymorphism. Polymorphism is applicable for overriding and not for overloading.

5) What is meant by Method Overriding?

Answer: Method overriding happens if the sub-class method satisfies the below conditions with the Super-class method:

Method name should be the same

The argument should be the same

Return type should also be the same

The key benefit of overriding is that the Sub-class can provide some specific information about that sub-class type than the super-class.

6) What is meant by Overloading?

Answer: Method overloading happens for different classes or within the same class.

For method overloading, sub-class method should satisfy the below conditions with the Super-class method (or) methods in the same class itself:

Same method name

Different argument types

There may be different return types

7) What is meant by Interface ?

Answer: Multiple inheritances cannot be achieved in java. To overcome this problem the Interface concept is introduced.

An interface is a template which has only method declarations and not the method implementation.

8) What is meant by Abstract class?

Answer: We can create the Abstract class by using the “Abstract” keyword before the class name.

An abstract class can have both “Abstract” methods and “Non-abstract” methods that are a concrete class.

Abstract method:

The method which has only the declaration and not the implementation is called the abstract method and it has the keyword called “abstract”.

Declarations ends with a semicolon.

An abstract class may have a non- abstract method also.

The concrete Subclass which extends the Abstract class should provide the implementation for abstract methods.