Name - Shahnoor Maniyar
Roll No - 212
Assignment No - 3

Q 1 - Attempt All of The Following.

1 - Difference between Abstract Class and Interface

Interface -
i- Pure abstract class.
ii- It specifies functionality which a class should have.
iii- Contains only method declarations.
iv- Fields are implicitly final and static.
v- Cannot have constructor.
vi- Used with the implements keyword.
vii- Can be implemented by more than one class.
viii- A class can implement more than one interface.
ix- An interface can be extended by another interface.
x- Can be used to implement multiple inheritance.

Abstract Class -

i- Incomplete Class.

ii- It specifies common attributes and behavior that a class inherits.

iii- Contains method declarations and definitions.

iv- Fields can be static, final as well as instance fields.

v- Can have a constructor.

vi- Used with the extends keyword.

vii- Can be extended by more than one class.

viii- A class can extend only one abstract class.

ix- An abstract class can be extended by another abstract as well as concrete class.

x- A class cannot extend more than one class.

2 - Explain 2 types of Inheritance with suitable Example.

Answer -

Types of Inheritance --

Java supports three types of inheritance single, multilevel and hierarchical inheritance.

1. Single Inheritance In this type of inheritance, a class can have only one super class i.e. it can be inherited from one class only.

2. Multilevel Inheritance In multilevel inheritance, a subclass is further used to derive more classes. For example, class B extends A and C extends class B

3. Hierarchical Inheritance A single superclass class has one or more subclasses. These are further used to create more subclasses.

3 - Explain how to define and implement interfaces

Interfaces -

Interface is an innovative concept in java. It can be thought of as a fully abstract class which does not contain any instance members but only method declarations. That is, using interface, you can specify what a class must do, but not how it does it. A class can specify how the operations should be performed by implementing the interface. An interface can also contain fields, but these are implicitly static and final.

The interface is defined using keyword interface. It can be declared using the public access modifier. Certain rules to remember while writing an interface are as follows -

1- All interface methods are implicitly public and abstract.
2- Interface methods must not be static.
3- All fields defined in an interface must be public, static, and final in other words, interfaces can declare only
constants, not instance variables.
4- Because interface methods are abstract, they cannot be marked final. An interface can extend one or more other
interfaces.
5- An interface can extend another interface.
6- An interface cannot implement another interface or class

4 - Explain the concept of Marker and Functional Interface.

There are some special interfaces in java which do not have any methods i.e. they are emp interfaces. An interface that does not contain methods, fields, and constants is known as marker interface or tag interface.

Example - java.lang.Serializable, java.util.Cloneable

Uses of Marker Interface -

Marker interface is used to tag a class so that some special behavior can be added to the cla implementing it. Thus, a class that wants to allow its objects to be cloned create an exact copy must be tagged by the Cloneable interface. In such a case, the class must override the close method from the Object ciass so that its objects can be cloned. class Student implements Cloneable

Functional Interface -

A functional interface is an interface that contains only one abstract method. A function interface can have any number of default methods and static methods but can contain only c abstract method.Functional Interface is also known as Single Abstract Method Interfaces SAM Interfaces. This feature in Java, which helps to achieve functional programming approach

Examples - Runnable, ActionListener, Comparable are functional interfaces.