**-----------------abstract class-----------------------**

1. Abstract class must have only abstract methods. True or false?->

**false**

1. Is it compulsory for a class which is declared as abstract to have at least one abstract method? ->

**Yes, it is compulsory**

1. Can we use “abstract” keyword with constructor? ->

**No**

1. Why final and abstract cannot be used at a time? ->

**when we will used final keyword in abstract class then we will not able to complete that method in concrete class that’s why we cannot used abstract and final keyword at a time**

1. Can we instantiate a class which does not have even a single abstract methods but declared as abstract?

* **Yes we can**

1. Can we declare abstract methods as private?

**No**

1. We can’t instantiate an abstract class. Then why constructors are allowed in abstract class?

1. Can we declare abstract methods as static?

-> **No**

9) Abstract classes can be nested. True or false?

-> **Yes**

10) Can we declare local inner class as abstract?

-> **yes**

11) Can abstract class have constructors in Java?

-> **Yes**

12) Can abstract class implements interface in Java? do they require to implement all methods?

-> **Yes abstract class implements interface in java and required to implement all methods because all method are incomplete so required to implement in abstract class**

13) Can abstract class have static methods in Java

-> **yes but ststic method should be complete**

**-------------------------Inheritance------------------------**

1. What is Inheritance in Java?

->**One class acquire Properties From another class using extends keyword is known as inheritance**

2) What are different types of Inheritance supported by Java?

->**1.single level inheritance**

**One class acquire properties from another class where the class from which properties acquire is superclass and properties are inherited is subclass.**

**2.Multilevel Inheritance:**

**One subclass acquired properties of another superclass and that class acquired properties of another superclass and phenomenon continues this is called as multilevel inheritance.**

**3.Multiple inheritance:**

**One subclass tries to acquire properties of two superclass at a time is known as multiple inheritance.**

**Multiple inheritance is not supported in java but it will support in interface**

**\*Diamond Ambiguity Problem:\***

**As one sub class tries to acquire properties of two super class at a time so there will be some confusion that from which class properties acquired first Hence this is called as diamond ambiguity problem**

**4.Hierarchical inheritance:**

**Multiple subclasses can acquire properties from one superclass is called as hierarchical inheritance.**

1. Why multiple Inheritance is not supported by Java?

* **As one sub class tries to acquire properties of two super class at a time so there will be some confusion that from which class properties acquired first Hence this is called as diamond ambiguity problem**

1. Why Inheritance is used by Java Programmers?

->**reusability**

**reduce length of code and reduce development time**

**and make flexible code**

**----------------------------Polymoprphism-------------------**

1. What is polymorphism in java?

* **One object showing different behavior at different stages of life cycle is Known As polymorphism**

2) Give real-life examples of polymorphism

->**like man at same time is son, brother, husband ,father, friend and employee.**

3) How many types are of polymorphism in java?

->**There are two types**  **of polymorphism**

**1.Compile time polymorphism:**

**Method declaration gets binded to its method definition at compile time so this is called as compile time polymorphism**

**-Binding takes place at compilation time so it is known as early binding.**

**2.Run time polymorphism:**

**Method declaration gets binded to its method definition at run time so this is called as compile time polymorphism.**

**-Binding takes place at run time so it is known as late binding.**

5) What are compile-time and run-time polymorphism?

6) What is method overloading?

**->Declaring multiple methods with same name with different parameters inside same class is known as method overloading**

7) What is method overriding?

**->acquiring super class property into subclass and changing implementation as per sub class specification.**

1. How to achieve Compile-time polymorphism?

->**Method Overloading**

8) How to achieve Runtime-time polymorphism?

->**method overriding**

9) Can we achieve method overloading by changing the return type?

**->yes**

10) Difference between method overloading and overriding?

11) Can we overload main() method?

-> **yes**

**-------------Interface------------------**

1. Can we re-assign a value to a field of interfaces?

**-> No, variable inside interface are by default static and final**

2) Can we declare an Interface with “abstract” keyword?

->**Yes**

3) For every Interface in java, .class file will be generated after compilation. True or false?

->**False**

4) Can we override an interface method with visibility other than public?

->**No**

5) Can interfaces become local members of the methods?

->

6) Can an interface extend a class?

->**No, interface implements class**

7) Like classes, does interfaces also extend Object class by default?

->**No**

8) Can interfaces have static methods?

->**No**

9) Can an interface have a class or another interface as it’s members?

->**yes, interface have** **a class or another interface as its member.**