1) Try implementing Two interfaces with same method name & signature and same return type?

sol.) Did it and its working fine.

2.) Try implementing Two interfaces with same method name with different return type ?

sol.) Tried to implement 2 interfaces with same method name and different types. But found out that we cannot implement interfaces in such manner.

3) Try adding a static block and instance block inside interface ?

sol.) Tried to do that but found out that interface cannot define an initializer.

4) Add a static block and instance block inside abstract class and identify the sequence of execution ?

sol)

Static block is called first. Then Instance block and constructor are called.

But found from resources that the constructor is called first.But static and instance blocks are executed after super command. As there is nothing in higher class

the static block and instance blocks are executed as they are in the first line of execution and the constructor class is executed.

5) Create a class static block, instance block and default constructor. create object for the same class and let me know the sequence of execution ?

sol.) Executed as mentioned and found that static block is called first. Then Instance block and then constructor. But infact the constructor is called first and

static, Instance blocks are place after super. As they are first 2 lines of constructor block, they are called first.

6) Create object for a class which implements two interfaces and an abstract class and verify the visibility of all the methods(both dummy and concrete).

sol.) Interface methods are public, abstract. Even if we donot assign them Java compiler adds public, Abstract before them. So if we do not assign public when

implementing, we get an error because it is assigned public in interface and it changes in the implementation class. So it tells us not toreduce it's visibility.

For concrete methods static and default are allowed since Java 8.

In Abstract class :

Dummy class : It seems only public and protected are allowed.

Concrete class: it can its visibility private too. But as it is private the object of it's child class cannot access that method.

7.)create a parent class ‘P’ and child classs ‘C’. P having methods variables and 2 methods. In child class override all variables and override only one method.

create Class ‘C’ specific methods and variables as wels.

Sol.) Did it.

8.)create a parent reference variable and assign child class. Verify scope of all parent and child member through parent reference variable.

sol.) While doing this I observed that only parent class methods are being overridden, even if we are overriding the parent class variables when doing this

only parent class variables are being called.

9) Come up with an awesome your Own object and build interfaces and abstract classes around.

Sol.)I implemented Course as an interface and built the different subjects such as Mathematics,Science and Social studies as abstract classes around them.

10) Can we override a construtors?

sol.) No.

11) Can we overload a construtor?

Sol.) Yes.

12.) Does java follow pass by value or pass by reference?

Sol.) Java follows pass by value not pass by reference. I tried working with a swap method and passing arguments to it. It neither works when we pass arguments directly

nor when we pass the object reference variables. It's because java makes a copy of the argument and passes the copy.

13.) Apply all access specifiers to class, methods and variables. Identify by writing which ones not allowed and which ones allowed?

Sol.) To the class in which main resides only public,abstract and final are allowed. We already know that main method must be public static. Because it is the start of the

program and making it static will cause JVM to run it first. All modifiers are permitted for class variables or instance varibles. but the variables inside static main

only final modifier is allowed. For methods in the class public,private,protected and final are allowed.