1. Can abstract class have constructors in Java?
   * Yes, however it will be implemented only when an instance of a subclass of the abstract class is created.
2. Can abstract class implements interface in Java? Do they require to implement all methods?
   * Yes. No they are not required to implement all the methods of that interface. This responsibility is left to the concrete subclass of the abstract class.
3. Can abstract class be final in Java?
   * No an abstract class cannot be final in java because by definition abstract class contains partial implementation of methods and need subclasses to provide implementation of those.
4. Can abstract class have static methods in Java?
   * Yes because a static method does not need an instance and can be called directly using the abstract class itself without instantiation.
5. Can you create instance of abstract class?
   * No, because the abstract class has incomplete method implementations and requires a subclass to complete those implementations.
6. Is it necessary for abstract class to have abstract method?
   * No it is not necessary for an abstract class to have an abstract method.
7. Difference between abstract class and interface in Java?
   * Methods: - Interface can only have abstract methods whereas abstract class can have both abstract and non-abstract methods.
   * Variables: - Variables in an interface are by default final and it can have only static and final variables. Whereas an abstract class can have both static/non-static and final/non-final variables.
   * Implementation: - An abstract class can provide implementation of an interface but an interface cannot provide implementation of an abstract class.
8. When do you favor abstract class over interface?
   * Use of an abstract class is favorable when requirements may change and you may need to add additional methods to incorporate those changes. This can be achieved without any changes to the underlying subclasses. In an interface implementations of all the (new) methods needs to done for the class which implements that interface.
9. What is abstract method in Java?
   * A method without an implementation is called an abstract method.
10. Can abstract class contains main method in Java?
    * Yes an abstract class can contain main method since it is static and does not require an instance.
11. What is static block in java?
    * Static block is used for initializing static variables. This block gets executed when the class gets loaded into the memory.
12. What is the need of static block?
    * Static block is used to execute those operations which needs to be done only once when the class is loaded into memory.
13. Can we overload static methods in java?
    * Yes static methods can be overloaded in java.
14. Can we call super class static methods from sub class?
    * Yes we can call super class static methods from a subclass.
15. What is the difference between final and static keywords?
    * Static Keyword: - static keyword can be applied to instance variables and methods but not to classes (with exception of nested class). When applied, variables and methods can be called without the help of an object
    * Final keyword: - final keyword can be applied to all constructs – variables, methods and classes. A final variable cannot be reassigned. A final method cannot be overridden final class cannot be inherited. Which means no changes can be made.

1. Write a note on covariant return type with example code.
   * Prior to java5 it was not possible to override a method by changing the return type. When we override a parent class method, the name, argument types and return type of the overriding method in child class has to be exactly same as that of parent class method. Overriding method was said to be invariant with respect to return type.

However from java 5 onwards it is possible to have different return type for a overriding method in child class, but child’s return type should be sub-type of parent’s return type. Overriding method becomes covariant with respect to return type.

// Java program to demonstrate that we can have

// different return types if return type in

// overridden method is sub-type

// Two classes used for return types.

class firstClass {

}

class secondClass extends firstClass {

}

class Base

{

firstClass fun()

{

System.out.println("Base fun()");

return new firstClass ();

}

}

class Derived extends Base

{

secondClass fun()

{

System.out.println("Derived fun()");

return new secondClass ();

}

}

public class Main

{

public static void main(String args[])

{

Base base = new Base();

base.fun();

Derived derived = new Derived();

derived.fun();

}

}

1. Write a note on Enum with example code.
   * Enum is a special data type in java which is used to define a collection of constants. Java enums can be thought of as class with fixed set of constants. It can include constants, methods etc.
2. Write a note on use of super keyword and super() method.
   * The super keyword is used to call super class variables and methods by the subclass object when they are overridden by subclass.

Whereas the super () method is used to call super class constructor from subclass constructor.

19)  Write a code to implement abstraction using interface.

20) Write a Java program to sort a numeric array and a string array.

21)Write a Java program to sum values of an array.

22)Write a Java program to remove a specific element from an array.

23)Write a Java program to reverse an array of integer values.

24)Write a Java program to find the duplicate values of an array of integer values.