AHMED BIN KHALID

FA20-BSE-062

BSE-7B

LAB Task 2

1. **Overloading of the main method is possible or not?**

In Java, overloading the main method with different parameter types is possible, but it won't be recognized as the entry point by the JVM. The standard entry point for a Java application is a main method with the signature public static void main(String[] args). While you can create other methods named main with different parameters, they won't serve as entry points for your program.

1. **What is Object Cloning?**

Object cloning refers to the creation of an exact copy of an object. It creates a new instance of the class of the current object and initializes all its fields with exactly the contents of the corresponding fields of this object.

* There are 3 methods for creating Object Cloning in Java that are mentioned below: Using Assignment Operator to create a copy of the reference variable
* Creating a copy using the clone() method
* Usage of clone() method – Deep Copy

1. **What is the purpose of private constructor in java?**

A private constructor in Java is used in restricting object creation. It is a special instance constructor used in static member-only classes. If a constructor is declared as private, then its objects are only accessible from within the declared class. You cannot access its objects from outside the constructor class.

1. **What is the difference between Static binding and dynamic binding in Java?**

Binding is a mechanism creating link between method call and method actual implementation. As per the polymorphism concept in Java, object can have many different forms. Object forms can be resolved at compile time and run time. If linking between method call and method implementation is resolved at compile time then we call it **static binding** and If it is resolved at run time then it **dynamic binding**. Dynamic binding uses object to resolve binding but static binding use type of the class and fields.

1. **What are the different usages of final variable?**

In Java, the final keyword is used to indicate that a variable, method, or class cannot be modified or extended.

Following are some of its characteristics:

**Final variables:** When a variable is declared as final, its value cannot be changed once it has been initialized. This is useful for declaring constants or other values that should not be modified.

**Final methods:** When a method is declared as final, it cannot be overridden by a subclass. This is useful for methods that are part of a class’s public API and should not be modified by subclasses.

**Final classes:** When a class is declared as final, it cannot be extended by a subclass. This is useful for classes that are intended to be used as is and should not be modified or extended.

**Initialization:** Final variables must be initialized either at the time of declaration or in the constructor of the class. This ensures that the value of the variable is set and cannot be changed.

**Performance:** The use of final can sometimes improve performance, as the compiler can optimize the code more effectively when it knows that a variable or method cannot be changed.

**Security:** final can help improve security by preventing malicious code from modifying sensitive data or behavior.