Lab Session 05

29 May 2023

Exercise 01:

Declare an interface called “MyFirstInterface”. Decalre integer type variable called “x”. Declare an abstract method called “display()”.

```
public interface MyFirstInterface {
// Public static final variable
int x = 10;
// Abstract method
void display();
}
public class InterfaceImplemented implements MyFirstInterface {
@Override
public void display() {
// Error: Cannot assign a value to a final variable
x = 20;
System.out.println("Value of x: " + x);
}
}
```

1. **Try to declare the variable with/without public static final keywords. Is there any difference between these two approaches? Why?**

   When declaring a Variable in an interface,it is by default publicstatic final.So whether you explicitly mention these keywords or not, the variable is treated as public static final.There is no difference between the two approaches.

2. **Declare the abstract method with/without abstract keyword. Is there any difference between these two approaches? Why?**

   When declaring a method in an interface,it is by default public abstract.So whether you explicitly mention the abstract key word or not,the method is treated as public abstract. There is no difference between the two approaches.

3. **Implement this into a class called “IntefaceImplemented” . Override all the abstract methods. Try to change the value of x inside this method and print the value of x. Is it possible for you to change x? why?**

   In the InterfaceImplemented class, you're attempting to change the value of the variable x inside the display() method. However, you will encounter an error because x is declared as a public static final variable in the interface, making it a constant. Constants cannot be modified once they are assigned a value.

   So, it is not possible to change the value of x inside the display() method. If you want to modify the value of x, you would need to remove the final keyword from its declaration in the interface.