Exercise 01:

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

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

public interface MyFirstInterface

{

int x=10;

void display();

}

There is no any difference using with and without public static final keywords because in interface variable is treated as constant.

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

No, there is no any difference above approaches, because in interface all methods are implicitly.

1. **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?**

public class InterfaceImplemented implements MyFirstInterface

{

@Override

public void display()

{

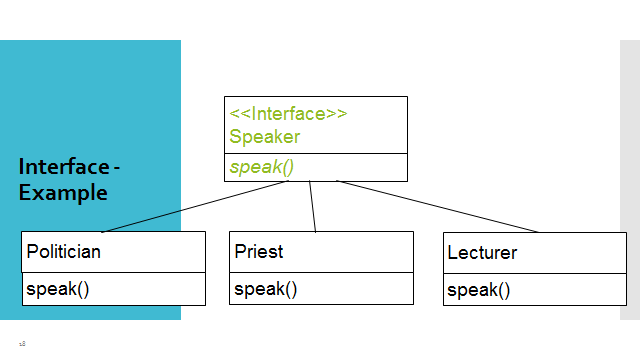
System.out.pritln("value of x is: "+x);

}

}

Exercise 02:

Develop a code base for the following scenario. Recall what we have done at the lecture…



public class Poltician implements Speaker

{

@Override

public void speak() {

System.out.println("Hello");

}

}

public class Priest implements Speaker

{

@Override

public void speak()

{

System.out.println("Every One");

}

}

public class Lecturer implements Speaker

{

@Override

public void speak()

{

System.out.println("Good MORNING");

}

}

public static void main(String[] args)

{

Lecturer L1=new Lecturer();

L1.speak();

Poltician P1=new Poltician();

P1.speak();

Priest PR1=new Priest();

PR1.speak();

}

}

Exercise 03:

Try following code. What is the outcome? Why?

Class 01: Class 02:

final class Student { class Undergraduate extends Student{}

final int marks = 100;

final void display();

}

**Answer**:

public abstract class Student

{

final int marks=100;

final void display()

{

System.out.println("Marks:"+marks);

}

}

public class Undergraduate extends Student

{

}