

```

/*Develop a JAVA program to create an interface
 * Resizable with methods resizeWidth(int width)
 * and resizeHeight(int height) that allow an
 * object to be resized. Create a class Rectangle
 * that implements the Resizable interface and
 * implements the resize methods*/
interface Resizable {
    void resizeWidth(int width);
    void resizeHeight(int height);
}
class Rectangle implements Resizable{
    int width,height;
    Rectangle(int w, int h)
    {
        this.width=w;
        this.height=h;
    }
    System.out.println("Initialized Rectangle Width:
    "+width);
    System.out.println("Initialized Rectangle Height:
    "+height);
    }
    public void resizeWidth(int w) {
        this.width=w;
        System.out.println("Rectangle Width After Resizing");
        System.out.println("Rectangle Width: "+width);
    }
    public void resizeHeight(int h) {
        this.height=h;
        System.out.println("Rectangle Height After Resizing");
        System.out.println("Rectangle Height: "+height);
    }
}
public class InterfaceMain {
    public static void main(String args[]) {
        Rectangle r=new Rectangle(5,6);
        r.resizeWidth(12);
        r.resizeHeight(21);
    }
}

```

```

/*Develop a JAVA program to create an outer class
 * with a function display. Create another class
 * inside the outer class named inner with a
 * function called display and call the two
 * functions in the main class.*/
class Outer{
    int x,y;
    public Outer(int x,int y) {
        this.x=x;
        this.y=y;
    }
    void display() {
        System.out.println("This is from Outer class");
        System.out.println("x is=> " + x + " n is=> " + y);
    }
    public class Inner{
        int m,n;
        public Inner(int m, int n) {
            this.m=m;
            this.n=n;
        }
        void display() {
            System.out.println("This is from Inner class");
            System.out.println("m is=> " + m + " n is=> " + n);
        }
    }
}

public class NestedClassDemo{
    public static void main(String args[]) {
        Outer outerObj=new Outer(21,33);
        outerObj.display();
        Outer.Inner innerObj = outerObj.new Inner(5,15);
        innerObj.display();
    }
}

```