Experiment 5

```
class MyException extends Exception{
	String a;
	MyException(String b){
		a=b;
	}
	public String toString() {
		return("Exception:"+ a);
	}
}

import java.util.*;
public class Exp5 {
   public static void main(String[] args) {
		int bal,amt;
		Scanner sc = new Scanner(System.in);
		System.out.println("Enter the balance:");
		bal=sc.nextInt();
		System.out.println("Enter the withdrawl amount:");
		amt=sc.nextInt();
		try {
			bal=bal-amt;
			if(bal<0)
			throw new MyException("Insufficient Balance");
			System.out.println("Remaining Balance"+ bal);
		}
		catch(MyException e) {
			System.out.println(e);
		}

	}
  }
```

```
Enter the balance:
10000
Enter the withdrawl amount:
12000
Exception:Insufficient Balance
```

Experiment 4

```
import java.util.*;
public class Experiment4 {
   public static void main(String[] args) {
      figure b=new triangle();
      double ar=b.area();
      System.out.println("The area of the triangle is "+ar);
      b=new rectangle();
      ar=b.area();
      System.out.println("The area of the rectangle is "+ar);
   }
}

public abstract class figure {
   abstract double area();
}

public class rectangle extends figure {
   @Override double area(){
       System.out.println("Enter the side1");
      Scanner sc=new Scanner(System.in);
      float a=sc.nextFloat();
      System.out.println("Enter the side2");
      float b=sc.nextFloat();
      double ar=a*b;
      return ar;
   }
}

public class triangle extends figure {
   @Override double area(){
      System.out.println("Enter the base");
      Scanner sc=new Scanner(System.in);
      float a=sc.nextFloat();
      System.out.println("Enter the height");
      float b=sc.nextFloat();
      double ar=0.5*a*b;
      return ar;
   }
}
```

```
Enter the base
2
Enter the height
3
The area of the triangle is 3.0
Enter the side1
4
Enter the side2
5
The area of the rectangle is 20.0
```

Part b

```
public class Experiment4part2 {
   public static void main(String[] args) {
      java j=new java();
      j.op();
   }
}

public interface compiler {
   void op();
}

public interface interpreter {
   void op();
}

public class java implements compiler,interpreter {
  @Override public void op(){
     System.out.println("Multiple inheritence using interface");
  }
}
```

```
Multiple inheritence using interface
```

Experiment-8

```
package exp_7;
import java.applet.*;
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
public class event_h extends Applet implements ActionListener {
   Button b1=new Button("OK");
   public void init(){
      add(b1);
      b1.addActionListener(this);
   }
   public void actionPerformed(ActionEvent e){
      showStatus("Button OK");
   }
}
```

Button OK