17. Create a Graphics package that has classes and interfaces for figures Rectangle, Triangle, Square and Circle. Test the package by finding the area of these figures.

Q1.java

```
import package_graphics.*;
import java.util.Scanner;
public class Q1
{
public static void main(String []args)
{
package_graphics testObj = new package_graphics();
int l,h,r,a,c,d;
Scanner s=new Scanner(System.in);
System.out.println("\n JERIL JOY \n ROLLNO:34");
System.out.println("Enter the length for rectangle");
l=s.nextInt();
System.out.println("Enter the breadth for rectangle");
h=s.nextInt();
System.out.println("Enter the radius of circle");
r=s.nextInt();
System.out.println("Enter the side for Square");
a=s.nextInt();
System.out.println("Enter the breadth for triangle");
c=s.nextInt();
System.out.println("Enter the height for triangle");
d=s.nextInt();
System.out.println("Area of rectangle="+testObj.recArea(l,h));
System.out.println("Area of circle="+testObj.cirArea(r));
System.out.println("Area of square="+testObj.squArea(a));
System.out.println("Area of triangle="+testObj.triArea(c,d));
}
}
```

package_graphics.java

```
package package_graphics;
interface interface_graphics
{
public float recArea(int l, int h);
public float cirArea(int r);
```

```
public float squArea(int a);
public float triArea(int l, int h);
}
public class package_graphics implements interface_graphics
{
public float recArea(int l, int h)
{
return l*h;
}
public float cirArea(int r)
{
return r*r*(float)3.14;
}
public float squArea(int a)
{
return a*a;
}
public float triArea(int l, int h)
{
return l*h*(float)(.5);
}
}
```

Output:

```
mca@mca-HP-Z238-Microtower-Workstation:~/JAVA34/CYCLE_4/CO4Q1/graphics$ javac Q1.java
mca@mca-HP-Z238-Microtower-Workstation:~/JAVA34/CYCLE_4/CO4Q1/graphics$ java Q1

 JERIL JOY
 ROLLNO:34
Enter the length for rectangle
4
Enter the breadth for rectangle
2
Enter the radius of circle
3
Enter the side for Square
4
Enter the breadth for triangle
3
Enter the height for triangle
2
Area of rectangle=8.0
Area of circle=28.26
Area of square=16.0
Area of triangle=3.0
```

18. Create an Arithmetic package that has classes and interfaces for the 4 basic arithmetic operations. Test the package by implementing all operations on two given numbers.

ArithmeticMain.java

```
import Arithmetic.ArithmeticOperations;
import java.util.Scanner;
public class ArithmeticMain {
public static void main(String[] args) {


ArithmeticOperations operations = new ArithmeticOperations();
Scanner scanner = new Scanner(System.in);
System.out.print("JERIL JOY\n ROLLNO:34\n");
System.out.print("Enter the first number: ");
double num1 = scanner.nextDouble();
System.out.print("Enter the second number: ");
double num2 = scanner.nextDouble();
System.out.println("Addition: " + operations.add(num1, num2));
System.out.println("Subtraction: " + operations.subtract(num1, num2));
System.out.println("Multiplication: " + operations.multiply(num1, num2));
System.out.println("Division: " + operations.divide(num1, num2));
}
}
```

Addition.java

```
package arithmetic;
public interface Addition {
public double add(double num1, double num2);
}
```

Subtraction.java

```
package arithmetic;
public interface Subtraction {
public double subtract(double num1, double num2);
}
```

Multiplication.java

```
package arithmetic;
public interface Multiplication {
public double multiply(double num1, double num2);
}
```

Division.java

```
package arithmetic;
public interface Division {
public double divide(double num1, double num2);
}
```

ArithmeticOperations.java

```
package arithmetic;
public class ArithmeticOperations implements Addition, Subtraction, Multiplication, Division {
@Override
public double add(double num1, double num2) {
return num1 + num2;
}
@Override
public double subtract(double num1, double num2) {
return num1 - num2;
}
@Override
public double multiply(double num1, double num2) {
return num1 * num2;
}
@Override
public double divide(double num1, double num2) {
if (num2 == 0) {
throw new ArithmeticException("Division by zero error!");}
return num1 / num2;
}}
```

Output:

```
mca@mca-HP-Z238-Microtower-Workstation:~/JAVA34/CYCLE_4/CO4Q1/graphics$ javac ArithmeticMain.java
mca@mca-HP-Z238-Microtower-Workstation:~/JAVA34/CYCLE_4/CO4Q1/graphics$ java ArithmeticMain
JERIL JOY
 ROLLNO:34
Enter the first number: 3
Enter the second number: 2
Addition: 5.0
Subtraction: 1.0
Multiplication: 6.0
Division: 1.5
```

19. Write a user defined exception class to authenticate the user name and password.

Program:

```
import java.util.Scanner;
class authException extends Exception
{
public authException(String s) {
super(s);
}
}
public class Q3
{
public static void main(String[] args) {
System.out.println("JERIL JOY");
System.out.println("ROLL NO:34");
System.out.println();
String username = "student";
String passcode = "student123";
String user_name,password;
Scanner sc = new Scanner(System.in);
try
{
System.out.println("Enter the username:");
user_name = sc.nextLine();
System.out.println("Enter the password:");
password = sc.nextLine();
if(username.equals(user_name) && passcode.equals(password))
{
System.out.println("Authentication successful...");
}
else
throw new authException("Invalid user credentials");

}
catch(authException e)
{
System.out.println("Exception caught "+e);
}
}
}
```

Output:

```
mca@mca-HP-Z238-Microtower-Workstation:~/JAVA34/CYCLE_4/CO4Q1/graphics$ javac Q3.java
mca@mca-HP-Z238-Microtower-Workstation:~/JAVA34/CYCLE_4/CO4Q1/graphics$ java Q3
JERIL JOY
ROLL NO:34

Enter the username:
student
Enter the password:
student123
Authentication successful...
```

20. Find the average of N positive integers, raising a user defined exception for each negative Input.

Program:

```
import java.util.Scanner;
class NegException extends Exception
{
public NegException(String s)
{
super(s);
}
}
public class Q4 {
public static void main(String[] args)
{
System.out.println("JERIL JOY");
System.out.println("ROLL NO:34");
System.out.println();
int i;
double sum=0,avg=0;
Scanner sc=new Scanner(System.in);
System.out.println("Enter n numbers:");
int n=sc.nextInt();
for(i=1;i<=n;i++)
{
try
{
System.out.println("Enter number"+i);
int a=sc.nextInt();
if(a<0)
{
i--;
throw new NegException("Negative numbers not allowed,Try again");
}
else
{
sum=sum+a;
}
}
catch(NegException e)
{
System.out.println("NEGETIVE EXCEPTION OCCURED:"+e);
```

```
}
}
avg=sum/n;
System.out.println("Average is "+avg);
sc.close();
}
}
```

Output:

```
mca@mca-HP-Z238-Microtower-Workstation:~/JAVA34/CYCLE_4/CO4Q1/graphics$ javac Q4.java
mca@mca-HP-Z238-Microtower-Workstation:~/JAVA34/CYCLE_4/CO4Q1/graphics$ java Q4
JERIL JOY
ROLL NO:34

Enter n numbers:
3
Enter number1
5
Enter number2
6
Enter number3
7
Average is 6.0
```

21. Program to remove all the elements from a linked list.

Program:

```
 import java.util.*;
public class Q11 {
public static void main(String[] args){
System.out.println("JERIL JOY");
System.out.println("ROLL NO:34");
System.out.println();
LinkedList<String> L=new LinkedList<>();
L.add("Gold");
L.add("Silver");
L.add("Bronze");
L.add(0,"Olympics Medals");
System.out.println(L);
L.remove("Bronze");
System.out.println(L);
L.remove(2);
System.out.println(L);
L.removeLast();
System.out.println(L);
L.removeFirst();
System.out.println(L);
}
}
```

Output:

```
mca@mca-HP-Z238-Microtower-Workstation:~/JAVA34/CYCLE_4/CO4Q1/graphics$ javac Q11.java
mca@mca-HP-Z238-Microtower-Workstation:~/JAVA34/CYCLE_4/CO4Q1/graphics$ java Q11
JERIL JOY
ROLL NO:34

[Olympics Medals, Gold, Silver, Bronze]
[Olympics Medals, Gold, Silver]
[Olympics Medals, Gold]
[Olympics Medals]
[]
```

22. Program to remove an object from the Stack when the position is passed as parameter.

Program:

```
import java.util.Stack;
public class Q12
{
public static void removeElementAtPosition(Stack<String> stack, int position)
{
if (position >= 1 && position <= stack.size())
{
Stack<String> tempStack = new Stack<>();
// Remove elements from the original stack until the desired position is reached
for (int i = 1; i < position; i++)
{
tempStack.push(stack.pop());
}
// Remove the element at the desired position
stack.pop();
// Restore the remaining elements back to the original stack
while (!tempStack.isEmpty())
{
stack.push(tempStack.pop());
}
System.out.println("Element at position " + position + " removed successfully.");
} else
{
System.out.println("Invalid position. Please provide a valid position within the stack range.");
}
}
public static void main(String[] args)
{
System.out.println("JERIL JOY\n ROLL NO:34");
System.out.println();
Stack<String> stack = new Stack<>();
stack.push("Element 1");
stack.push("Element 2");
stack.push("Element 3");
stack.push("Element 4");
stack.push("Element 5");
int positionToRemove = 3;
System.out.println("Before removal: " + stack);
removeElementAtPosition(stack, positionToRemove);
```

```
System.out.println("After removal: " + stack);
}
}
```

Output:

```
mca@mca-HP-Z238-Microtower-Workstation:~/JAVA34/CYCLE_4/CO4Q1/graphics$ javac Q12.java
mca@mca-HP-Z238-Microtower-Workstation:~/JAVA34/CYCLE_4/CO4Q1/graphics$ java Q12
JERIL JOY
 ROLL NO:34

Before removal: [Element 1, Element 2, Element 3, Element 4, Element 5]
Element at position 3 removed successfully.
After removal: [Element 1, Element 2, Element 4, Element 5]
mca@mca-HP-Z238-Microtower-Workstation:~/JAVA34/CYCLE_4/CO4Q1/graphics$
```

23. Write a Java program to compare two hash set.

Program:

```
import java.util.HashSet;
import java.util.Scanner;
import java.util.Set;
public class Q16
{
public static void main(String[] args)
{
System.out.println("JERIL JOY\n ROLLNO:34 \n 15-04-2024");
System.out.println();
Set<Integer> set1 = new HashSet<>();
Set<Integer> set2 = new HashSet<>();
Scanner scanner = new Scanner(System.in);
// Input for Set 1
System.out.print("Enter the number of elements in Set 1: ");
int numElements1 = scanner.nextInt();
System.out.println("Enter the elements for Set 1:");
for (int i = 0; i < numElements1; i++)
{
int element = scanner.nextInt();
set1.add(element);
}
// Input for Set 2
System.out.print("Enter the number of elements in Set 2: ");
int numElements2 = scanner.nextInt();
System.out.println("Enter the elements for Set 2:");
for (int i = 0; i < numElements2; i++)
{
int element = scanner.nextInt();
set2.add(element);
}
// Comparison
boolean isEqual = set1.equals(set2);
// Output
System.out.println("Set 1: " + set1);
System.out.println("Set 2: " + set2);
if (isEqual)
{
System.out.println("Set 1 and Set 2 are equal.");
} else
```

```
{
System.out.println("Set 1 and Set 2 are not equal.");
}
scanner.close();
}
}
```

Output:

```
mca@mca-HP-Z238-Microtower-Workstation:~/JAVA34/CYCLE_4/CO4Q1/graphics$ javac Q16.java
mca@mca-HP-Z238-Microtower-Workstation:~/JAVA34/CYCLE_4/CO4Q1/graphics$ java Q16
JERIL JOY
 ROLLNO:34
 15-04-2024

Enter the number of elements in Set 1: 3
Enter the elements for Set 1:
23
24
76
Enter the number of elements in Set 2: 3
Enter the elements for Set 2:
65
76
88
Set 1: [23, 24, 76]
Set 2: [65, 88, 76]
Set 1 and Set 2 are not equal.
```