**Muhammad Rizwan Khalid**

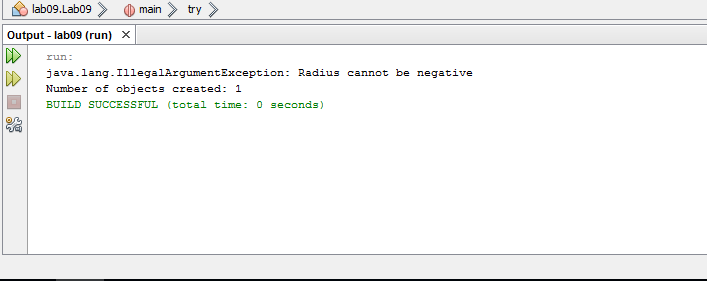
**180459**

**BSCS 6-A**

**Lab no: 09**

**Activity One:**

**Output:**



**Question:** What happens if we remove the clause throws IllegalArgumentException from the setRadius method declaration, and re-compile the CircleWithException class? Would it compile? If so, why?

**Answer:** If we remove the clause throws IllegalArgumentException from method setRadius then program will terminate in the same manner. By using throws keyword we explicitly tells the compiler that this method has an exception. If we remove this statement, compiler is still aware that setRadius has an exception.

**Question:** What happens if we do not handle the IllegalArgumentException in the TestCircleWithException class by not using the try statement?

**Answer:** If we remove the try statement then compiler gives following error: “catch without try”. Catch is the part of try statement and is written just after the catch block. Catch block without try is illegal and redundant. If we remove both try and catch then program terminates abnormally. Exception is used to make program robust and tolerant.

**Last part of Activity One: -**

**Updated code:**

package lab09;

public class Lab09 {

public static void main(String[] args) {

try {

CircleWithException c1 = new CircleWithException(5);

CircleWithException c2 = new CircleWithException(-5);

CircleWithException c3 = new CircleWithException(0);

CircleWithException c4 = new CircleWithException();

}

catch (InvalidRadiusException ex) {

System.out.println(ex);

}

System.out.println("Number of objects created: " +

CircleWithException.getNumberOfObjects());

}

}

class CircleWithException{

/\*\* The radius of the circle \*/

private double radius;

/\*\* The number of the objects created \*/

private static int numberOfObjects = 0;

/\*\* Construct a circle with radius 1 \*/

public CircleWithException()throws InvalidRadiusException {

this(1.0);

numberOfObjects++;

}

/\*\* Construct a circle with a specified radius \*/

public CircleWithException(double newRadius)throws InvalidRadiusException {

setRadius(newRadius);

numberOfObjects++;

}

/\*\* Return radius \*/

public double getRadius() {

return radius;

}

/\*\* Set a new radius \*/

public void setRadius(double newRadius) throws InvalidRadiusException {

if (newRadius >= 0)

radius = newRadius;

else

throw new InvalidRadiusException(newRadius);

}

/\*\* Return numberOfObjects \*/

public static int getNumberOfObjects() {

return numberOfObjects;

}

/\*\* Return the area of this circle \*/

public double findArea() {

return radius \* radius \* 3.14159;

}

}

class InvalidRadiusException extends Exception {

private double radius;

/\*\* Construct an exception \*/

public InvalidRadiusException(double radius) {

super("Invalid radius " + radius);

this.radius = radius;

}

/\*\* Return the radius \*/

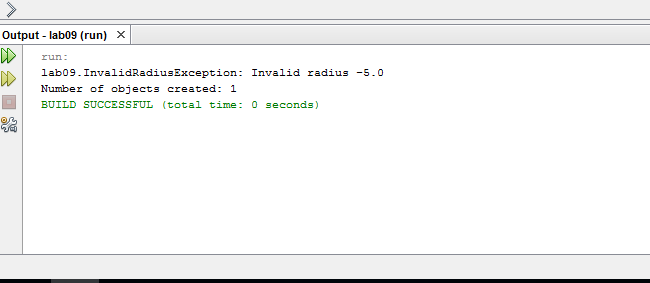
public double getRadius() {

return radius;

}

}

**Updated Output:**



**Activity two:**

**Question:** If no exception occurs, will statement4 be executed, and will statement5 be executed?

**Answer:** finally block will be executed whatever there is an exception or not. Let’s say the no exception occurs then statement 4 and statement 5 will execute.

**Question:** If the exception is of type Exception1, will statement4 and statement5 be executed?

**Answer:**  In this case statement4 and statement5 will be executed.

**Question:** If the exception is of type Exception2, will statement4 and statement5 be executed?

**Answer:** In this case statement4 will be executed while statement5 will not be executed because program is terminating.

**Question:** If the exception is not Exception1 nor Exception2, will statement4 and statement5 be executed?

**Answer:** In this case statement4 will be executed while statement5 will not be executed because program is terminated.

**Activity Three:**

package lab09;

import javax.swing.JOptionPane;

public class Lab09 {

public static void main(String[] args){

// obtain user input from JOptionPane input dialogs

String firstNumber =

JOptionPane.showInputDialog( "Enter first integer" );

String secondNumber =

JOptionPane.showInputDialog( "Enter second integer" );

String thirdNumber =

JOptionPane.showInputDialog( "Enter third integer" );

// convert String inputs to int values for use in a

// calculation

int number1 = Integer.parseInt( firstNumber );

int number2 = Integer.parseInt( secondNumber );

int number3 = Integer.parseInt(thirdNumber);

int sum = number1 + number2 + number3; // add numbers

// display result in a JOptionPane message dialog

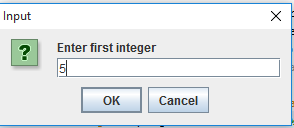
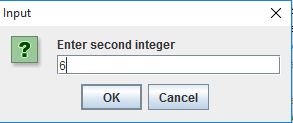
JOptionPane.showMessageDialog( null, "The sum is " + sum,

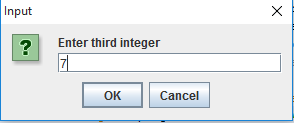
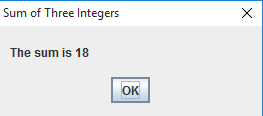
"Sum of Two Integers",JOptionPane.PLAIN\_MESSAGE );

}

}

**Output:**

**Activity four:**

package lab09;

import javax.swing.\*;

public class Lab09 {

public static void main(String[] args){

//Make sure we have nice window decorations.

JFrame.setDefaultLookAndFeelDecorated(true);

//Create and set up the window.

JFrame frame = new JFrame("HelloWorldSwing");

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

//Add the ubiquitous "Hello World" label.

JLabel label = new JLabel("Hello World! How are you? Hopefully, doing Great!");

frame.getContentPane().add(label);

//Display the window.

frame.pack();

frame.setVisible(true);

}

}

**Output:**



**Task one:**

package lab09;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

public class Lab09 {

public static void main(String[] args) {

//Make sure we have nice window decorations.

JFrame.setDefaultLookAndFeelDecorated(true);

CelsiusConverter converter = new CelsiusConverter();

}

}

class CelsiusConverter implements ActionListener {

String items[] = {"Centigrade to Farenheit","Meters to Feet","Kgs to lbs","Radian to Degree"};

JComboBox comboBox = new JComboBox(items);

JFrame converterFrame;

JPanel converterPanel;

JTextField tempCelsius;

JLabel celsiusLabel, fahrenheitLabel;

JButton convertTemp;

public CelsiusConverter() {//Create and set up the window.

converterFrame = new JFrame("Unit Converter");

converterFrame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

converterFrame.setSize(new Dimension(600, 150));

//Create and set up the panel.

converterPanel = new JPanel(new GridLayout(2, 2));

//Add the widgets.

addWidgets();

//Set the default button.

//converterFrame.getRootPane().setDefaultButton(convertTemp);

//Add the panel to the window.

converterFrame.getContentPane().add(converterPanel,BorderLayout.CENTER);

//Display the window.

converterFrame.pack();

converterFrame.setVisible(true);

}

/\*\*

\* Create and add the widgets.

\*/

private void addWidgets() {

//Create widgets.

tempCelsius = new JTextField(2);

celsiusLabel = new JLabel("Celsius", SwingConstants.LEFT);

convertTemp = new JButton("Convert");

fahrenheitLabel = new JLabel("Fahrenheit",SwingConstants.LEFT);

//Listen to events from the Convert button.

convertTemp.addActionListener(this);

//Add the widgets to the container.

converterPanel.add(tempCelsius);

converterPanel.add(celsiusLabel);

converterPanel.add(comboBox);

converterPanel.add(convertTemp);

converterPanel.add(fahrenheitLabel);

celsiusLabel.setBorder(BorderFactory.createEmptyBorder(5,5,5,5));

fahrenheitLabel.setBorder(BorderFactory.createEmptyBorder(5,5,5,5));

}

public void actionPerformed(ActionEvent event) {

//Parse degrees Celsius as a double and convert to Fahrenheit.

if(comboBox.getSelectedIndex() == 0) {

int tempFahr = (int)((Double.parseDouble(tempCelsius.getText())) \* 1.8 + 32);

fahrenheitLabel.setText(tempFahr + " Fahrenheit");

}

if(comboBox.getSelectedIndex() == 1) {

double feet = ((Double.parseDouble(tempCelsius.getText())) \* 3.28);

celsiusLabel.setText("Meters");

fahrenheitLabel.setText(feet + " Feet");

}

if(comboBox.getSelectedIndex() == 2) {

double lb = ((Double.parseDouble(tempCelsius.getText())) \* 2.204);

celsiusLabel.setText("Kgs");

fahrenheitLabel.setText(lb + " Pounds");

}

if(comboBox.getSelectedIndex() == 3) {

double degree = ((Double.parseDouble(tempCelsius.getText())) \* 57.29);

celsiusLabel.setText("Radians");

fahrenheitLabel.setText(degree + " Degrees");

}

}

}

**Output:**

