**Name: Abdul Ghaffar Kalhoro**

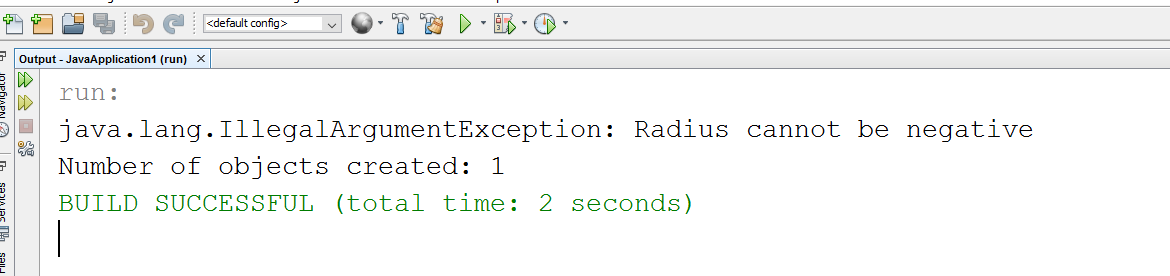
**Registration # 194699**

**OOP Lab # 9**

**Activity # 1:**

1. What will be the output?

Ans: The output for this code is given below:

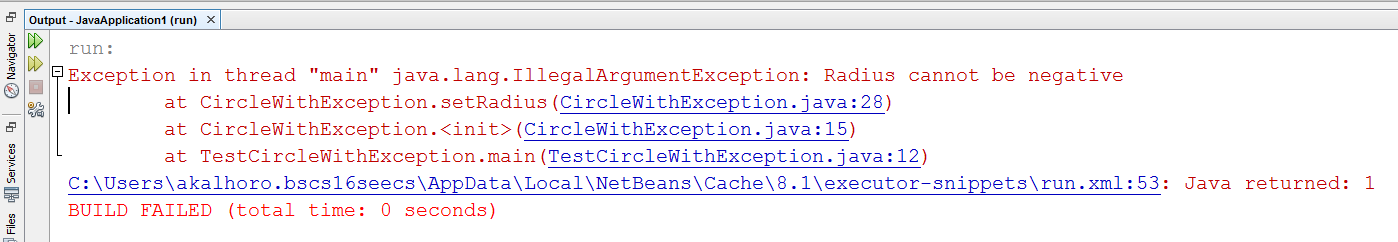


1. What happens if we remove the clause *throws IllegalArgumentException* from the *setRadius* method declaration, and re-compile the *CircleWithException* class? Would it compile? If so, why?

Ans: If we remove the clause *throws illegalArgumentException* from *setRadius* method declaration then it means there would not be any exception that could be thrown from the setRadius method and when we compile it would compile correctly but here we are not dealing with this Exception then at some particular points it would give exception message in our output. And it would compile correctly until the exception occurs.

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

Ans: If we do not handle the *IllegalArgumentException* in our test class then the compiler would show the Exception of radius that the radius can not be negative, this Exception occurs because we are not handling the Exception for the value of radius to be always +ve, the output is shown below:



**Activity # 2**

Q.1: If no exception occurs, will statement4 be executed, and will statement5 be executed?

**Ans:** Yes both of these statement would be executed because the statement4 is in the finally block which always executed. And the statement5 would be executed because after finally block the control would be passed to the end.

Q.2: If the exception is of type Exception1, will statement4 and statement5 be executed?

**Ans:** Yes both of these statement would be executed because the Exception1 catches the exception which after that passes to the finally block where the statement4 executed after that the control would transfer out of the try block where statement5 would be executed.

Q.3If the exception is of type Exception2, will statement4 and statement5 be executed?

**Ans:** The statement4 would be executed as the control would be towards the final block after catch but the statement5 would not be executed as the expression2 rethrows it.

Q.If the exception is not Exception1 nor Exception2, will statement4 and statement5 be executed?

Ans: Here both the statement would be executed as mentioned in the Question 1.

**Activity # 3**

Q. modify the given program to take three numbers as input and compute their sum?

Ans: Modified code:

**import** javax.swing.JOptionPane; // program uses JOptionPane

**public** **class** Addition

{

**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,

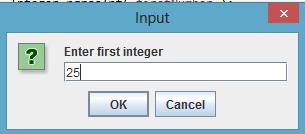
"Sum of Three Integers",

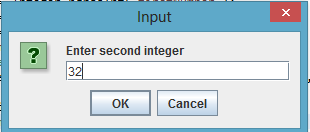
JOptionPane.***PLAIN\_MESSAGE*** );

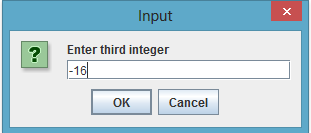
} // end method main

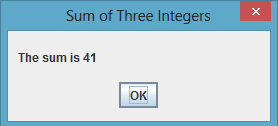
} // end class Addition

**OUTPUT**



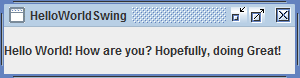






**Activity # 4**

**OUTPUT**



**Task # 1**

**import** java.awt.\*;

**import** java.awt.event.\*;

**import** javax.swing.\*;

**public** **class** Conversion **implements** ActionListener {

//data fields

JFrame converterFrame;

JPanel converterPanel;

JTextField GivenValue;

JLabel changedLabel, conversionLabel;

JButton convertTemp;

JComboBox CollectionUnits;

//constructor

**public** Conversion() {

//Create and set up the window.

converterFrame = **new** JFrame("Convert between Different Quantities");

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

converterFrame.setSize(**new** Dimension(450, 150));

//Create and set up the panel.

converterPanel = **new** JPanel(**new** GridLayout(4, 4));

String[] units = {"Celsius To Fahrenheit" , "Meters to Feet",

"Kilograms(Kgs) to Pounds(LBs)", "Radians to Degrees(Angles)"};

CollectionUnits = **new** JComboBox(units);

//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**);

//a new class is created to handle the items in the list

CollectionUnits.addItemListener(

**new** ItemListener(){

**public** **void** itemStateChanged(ItemEvent event){

**if**(event.getStateChange() == ItemEvent.***SELECTED***){

//switch statement to find the correct selection

**switch** (CollectionUnits.getSelectedIndex()) {

**case** 0:

conversionLabel.setText(" Fahrenheit");

changedLabel.setText("Celsius");

**break**;

**case** 1:

conversionLabel.setText(" Feet");

changedLabel.setText("Meters");

**break**;

**case** 2:

conversionLabel.setText(" Pounds");

changedLabel.setText("Kilograms(KGs)");

**break**;

**default**:

conversionLabel.setText(" Degrees");

changedLabel.setText("Radians");

**break**;

} //end switch

} //end if.

} //end method

}

);

}

/\*\*

\* Create and add the widgets.

\*/

**private** **void** addWidgets() {

//Create widgets.

GivenValue = **new** JTextField(2);

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

convertTemp = **new** JButton("Convert");

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

//Listen to events from the Convert button.

convertTemp.addActionListener(**this**);

//Add the widgets to the container.

converterPanel.add(GivenValue);

converterPanel.add(changedLabel);

converterPanel.add(convertTemp);

converterPanel.add(conversionLabel);

converterPanel.add(CollectionUnits);

changedLabel.setBorder(BorderFactory.*createEmptyBorder*(8,8,8,8));

conversionLabel.setBorder(BorderFactory.*createEmptyBorder*(8,8,8,8));

}

**public** **void** actionPerformed(ActionEvent event) {

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

//Performing specific functioning on specific items of Drop Down List

**if**(CollectionUnits.getSelectedIndex() == 2){

**double** pounds = Double.*parseDouble*((GivenValue.getText())) \* 2.204;

conversionLabel.setText(pounds + " pounds");

}

**if**(CollectionUnits.getSelectedIndex() == 0){

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

conversionLabel.setText(tempFahr + " Fahrenheit");

}

**if**(CollectionUnits.getSelectedIndex() == 1){

**double** feet;

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

conversionLabel.setText(feet + " feet");

}

**if**(CollectionUnits.getSelectedIndex() == 3){

**double** degrees = Double.*parseDouble*((GivenValue.getText())) \* 57.2958;

conversionLabel.setText(degrees + " Degrees");

}

}

**public** **static** **void** main(String[] args) {

//Make sure we have nice window decorations.

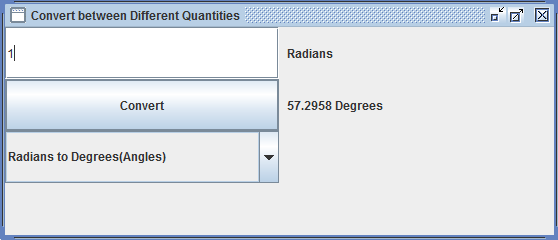
JFrame.*setDefaultLookAndFeelDecorated*(**true**);

Conversion converter = **new** Conversion();

}

}

**OUTPUT**

****