**MarathwadaMitra Mandal’s Polytechnic**

Thergaon Pune-33

**Fifth Semester**

**(Year: 2020-21)**

Micro Project

**Advanced Java Programming (22517)**

**Title of the Project**: Currency Conversion System.

**Branch: Computer Engineering (CO5I)**

**Members of the Group**

1. Avishkar Harishchandra Sontakke Roll No. 180352
2. Sumit Murlidhar Bhoite Roll No. 180353

**Maharashtra State Board of Technical Education, Mumbai**

**CERTIFICATE**

This is to certify that,

**Mr. Avishkar Harishchandra Sontakke** Roll No: **180352** of Fifth Semester of Diploma in **Computer Engineering** of **Marathwada Mitra Mandal’s Polytechnic** has completed the Micro Project satisfactorily in course **Advanced Java Programming (22517)** for the academic year 2020-21 as prescribed in the curriculum.

Place: Thergaon Enrollment No: 1809890173

Date: 04 /11/20 Exam Seat No:

**Course Coordinator HOD Principal**

**Institute Seal**

**Maharashtra State Board of Technical Education, Mumbai**

**CERTIFICATE**

This is to certify that,

**Mr.** **Sumit Murlidhar Bhoite** Roll No: **180353** of Fifth Semester of Diploma in **Computer Engineering** of **Marathwada Mitra Mandal’s Polytechnic** has completed the Micro Project satisfactorily in course **Advanced Java Programming (22517)** for the academic year 2020-21 as prescribed in the curriculum.

Place: Thergaon Enrollment No: 1809890174

Date: 04/11/20 Exam Seat No:

**Course Coordinator HOD Principal**

**Institute Seal**

**Maharashtra State Board of Technical Education, Mumbai**

**MICRO PROJECT**

**Progress Report / Weekly Report**

Name of the Project: Currency Conversion System

Course: AJP (22517) Program: Computer Engineering (CO5I) Roll No: 180352

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Week No** | **Date** | **Duration in Hrs.** | **Work / Activity Performed** | **Sign of the Faculty** |
| 1 | 08/07/20 | 1 | Researched All About Our project. |  |
| 2 | 15/07/20 | 1 | We distributed the work among the group members. |  |
| 3 | 22/07/20 | 1 | We wrote the Acknowledgement. |  |
| 4 | 29/07/20 | 1 | We wrote the Abstract. |  |
| 5 | 05/08/20 | 1 | We wrote the Introduction of our project. |  |
| 6 | 12/08/20 | 1 | We wrote the description of AWT, SWING & EVENT. |  |
| 7 | 19/08/20 | 1 | We Developed the program for our project. |  |
| 8 | 26/08/20 | 1 | Found Some Errors and solved them with help of Teacher. |  |
| 9 | 09/09/20 | 1 | Then did a Recheck by Teacher. |  |
| 10 | 16/09/20 | 1 | Copied the Code with output and pasted it on the project report. |  |
| 11 | 23/09/20 | 1 | Wrote the Conclusion of our project. |  |
| 12 | 30/09/20 | 1 | Wrote the References Took by us for our project. |  |
| 13 | 07/10/20 | 1 | Wrote the Format pages of our project. |  |
| 14 | 14/10/20 | 1 | Teacher Did Final Checkup of our project. |  |
| 15 | 21/10/20 | 1 | Did Some Corrections and rechecked the project. |  |
| 16 | 29/10/20 | 1 | Submitted the project to Teacher. |  |

**MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION, MUMBAI**

**MICRO PROJECT**

**Progress Report / Weekly Report**

Name of the Project: Currency Conversion System

Course: AJP (22517) Program: Computer Engineering (CO5I) Roll No: 180353

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Week No** | **Date** | **Duration in Hrs.** | **Work / Activity Performed** | **Sign of the Faculty** |
| 1 | 08/07/20 | 1 | Researched All About Our project. |  |
| 2 | 15/07/20 | 1 | We distributed the work among the group members. |  |
| 3 | 22/07/20 | 1 | We wrote the Acknowledgement. |  |
| 4 | 29/07/20 | 1 | We wrote the Abstract. |  |
| 5 | 05/08/20 | 1 | We wrote the Introduction of our project. |  |
| 6 | 12/08/20 | 1 | We wrote the description of AWT, SWING & EVENT. |  |
| 7 | 19/08/20 | 1 | We Developed the program for our project. |  |
| 8 | 26/08/20 | 1 | Found Some Errors and solved them with help of Teacher. |  |
| 9 | 09/09/20 | 1 | Then did a Recheck by Teacher. |  |
| 10 | 16/09/20 | 1 | Copied the Code with output and pasted it on the project report. |  |
| 11 | 22/09/20 | 1 | Wrote the Conclusion of our project. |  |
| 12 | 30/09/20 | 1 | Wrote the References Took by us for our project. |  |
| 13 | 07/10/20 | 1 | Wrote the Format pages of our project. |  |
| 14 | 14/10/20 | 1 | Teacher Did Final Checkup of our project. |  |
| 15 | 21/10/20 | 1 | Did Some Corrections and rechecked the project. |  |
| 16 | 29/10/20 | 1 | Submitted the project to Teacher. |  |

**MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION, MUMBAI**

**Teacher Evaluation Sheet for Micro Project**

**Course Title and Code: - Advanced Java Programming (22517)**

**Title of the Project:-Currency Conversion System**

**Group No:-21**

**COs addressed by the Micro Project:**

|  |  |
| --- | --- |
| **CO\_\_1\_:** | Develop program using GUI framework (AWT and Swing). |
| **CO\_\_2\_:** | Handle events of AWT and Swing components. |
| **CO\_\_3\_:** | Develop programs to handle events in java programming. |

**Marks:-**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Roll No.** | **Name Of Student** | **Marks for Group Work (06)** | **Marks obtained by the individual based on viva (04)** | **Total Marks (10)** |
| 180352 | Avishkar Harishchandra Sontakke |  |  |  |
| 180353 | Sumit Murlidhar Bhoite |  |  |  |

**Name and designation of Faculty Member**: Mrs. Chavan.P. P.

Lecturer in Computer Department

**Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Micro Project of Advance JAVA programming**

**Currency Conversion System**

**Group No 21:**

|  |  |
| --- | --- |
| **Roll No** | **Group Members** |
| 180352 | Avishkar Harishchandra Sontakke |
| 180311 | Sumit Murlidhar bhoite |

* **Contents**
* Acknowledgement
* Abstract
* Introduction
* Description
* program
* output
* Conclusion
* References

**Acknowledgement**

In performing our micro project, we had to take the help and guidance of respected teacher, who deserve our greatest gratitude, the completion of this project gives us much pleasure. We would like to show our gratitude Mrs. Pragati Chavan for giving us a good guidance for Micro Project.

In addition, thank you Mr. Vikas Solanke (HOD), who introduced us to the Methodology of work.We also thank to Mrs. G.S.Joshi (Principal) for her valuable guidance by accepting proposal which gave us an inspiration to improve our Micro Project.

We thank the entire class and the team members itself for their help directly and indirectly to complete our Micro Project.

**Abstract**

The title of the project is **Currency Conversion System**. In this project we demonstrate that how the Currency changes Depending on the Different Countries using AWT and Swing components.

In this micro project AWT, swing, event handling, exception handling is included.

By using frame when we run the program, one form is displayed. In that form we have to choose from which currency we want to convert and to which currency we want to convert then fill the Amount which we have to convert. It will give the Result you want.

**Introduction**

Here is the project we developed a **micro project to design Application of Currency Conversion System using AWT & swing in Advance JAVA**, it is complete and totally error-free. Swing API is set of extensible GUI Component to ease developer’s life to create JAVA based Front End/GUI Application. It is build upon top of AWT API and acts as replacement of AWT API as it has almost every control corresponding to AWT controls. Swing component follows a Model-View-Controller architecture to full fill the following criteria:

* + A Single API is to be sufficient to support multiple look and feel.
  + API is model driven so that highest level API is not required to have the data.

.

In this project we can use class, constructor, JLabel, JTextField, JButton, JFrame, for, if, switch case etc.

A Currency Conversion System helps an Entrepreneur or a businessman to get an easy way to convert all data from one currency to another currency. This data must be Converted properly to give the Proper output. It leads to a great experience for the user which can give us Satisfaction and Inspiration of creating more such projects.

**Description:**

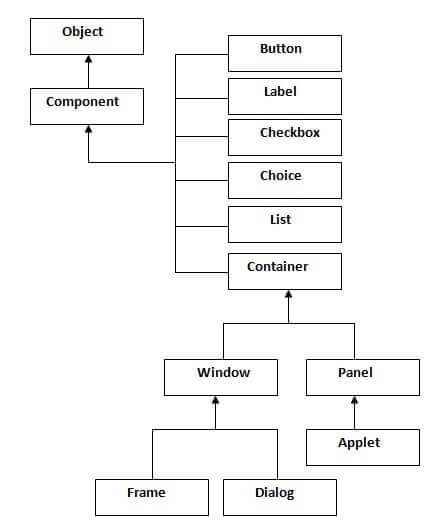
**AWT**

* Java AWT (Abstract Window Toolkit) is an API to develop GUI or window-based applications in java.
* Java AWT components are platform-dependent i.e. components are displayed according to the view of operating system.
* AWT is heavyweight i.e. its components are using the resources of OS.
* The AWT is part of the Java Foundation Classes (JFC) — the standard API for providing a graphical user interface (GUI) for a Java program.

**AWT Components**

The java.awt package provides classes for AWT API such as Text Field, Label, Text Area, Radio Button, Checkbox, Choice, List etc.

**Java AWT Hierarchy**



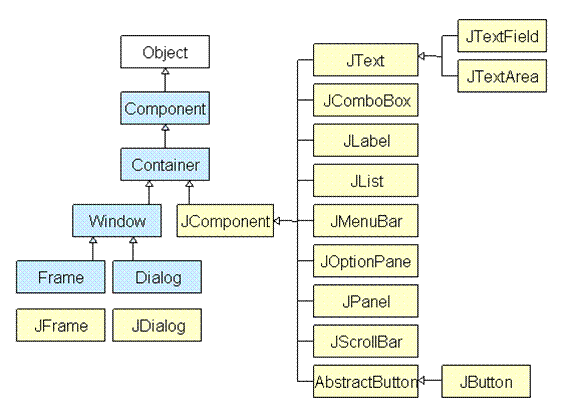
**Swing**

Swing API is set of extensible GUI Components to create JAVA based Front End/ GUI Applications. It is build upon top of AWT API and acts as replacement of AWT API as it has almost every control corresponding to AWT controls.

**Features**

* **Light Weight** - Swing component are independent of native Operating System's API as Swing API controls are rendered mostly using pure JAVA code instead of underlying operating system calls.
* **Rich controls** - Swing provides a rich set of advanced controls like Tree, Tabbed Pane, slider, colour picker, table controls
* **Pluggable look-and-feel**- SWING based GUI Application look and feel can be changed at run time based on available values.

**Hierarchy of Java Swing classes**



**Action Event**

* An Action Event is generated when a button is pressed, a list item is double-clicked, or a menu item is selected.
* The Action Event class defines four integer constants that can be used to identify any modifiers associated with an action event:
* public static final int ALT\_MASK
* public static final int SHIFT\_MASK
* public static final int CTRL\_MASK
* public static final int META\_MASK
* Constructs an Action Event object with modifier keys.

**public ActionEvent(Object source, int id,String command, int modifiers)**

Parameters:

* **source** - the object that originated the event
* **id** - an integer that identifies the event
* **command** - a string that may specify a command (possibly one of several)associated with the event
* **modifiers** - the modifier keys held down during this action
* Constructs an Action Event object with the specified modifier keys and timestamp.

**public ActionEvent(Object source, int id, String command, long when, int modifiers)**

Parameters:

* **source** - the object that originated the event
* **id** - an integer that identifies the event
* **command** - a string that may specify a command (possibly one of several)associated with the event
* **when** - the time the event occurred
* **modifiers** - the modifier keys held down during this action.

**Methods of Action Event Class**

* public String getActionCommand()
* Returns the command string associated with this action.
* public long getWhen()
* Returns the timestamp of when this event occurred.
* intgetModifiers()
* Returns the modifier keys held down during this action event.
* String paramString()
* Returns a parameter string identifying this action event.

**ActionListener Interface**

* This interface defines the actionPerformed() method that is invoked when an action event occurs.
* Its general form is shown here:

**void actionPerformed(ActionEvent ae)**

**Event classes and Listener interfaces:**

|  |  |
| --- | --- |
| **Event Classes** | **Listener Interfaces** |
| ActionEvent | ActionListener |
| MouseEvent | MouseListener and MouseMotionListener |
| MouseWheelEvent | MouseWheelListener |
| KeyEvent | KeyListener |
| ItemEvent | ItemListener |
| TextEvent | TextListener |
| AdjustmentEvent | AdjustmentListener |
| WindowEvent | WindowListener |
| ComponentEvent | ComponentListener |
| ContainerEvent | ContainerListener |
| FocusEvent | FocusListener |

**Program:**

package converter;

import java.awt.EventQueue;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.border.EmptyBorder;

import javax.swing.JLabel;

import javax.swing.JComboBox;

import javax.swing.JTextField;

import javax.swing.JButton;

import javax.swing.JMenuBar;

import javax.swing.JMenu;

import javax.swing.JMenuItem;

import java.awt.event.ActionListener;

import java.awt.event.KeyEvent;

import java.text.DecimalFormat;

import java.util.ArrayList;

import java.awt.event.ActionEvent;

import javax.swing.SwingConstants;

import java.util.ResourceBundle;

public class MainWindow extends JFrame

{

private static final ResourceBundle BUNDLE = ResourceBundle.getBundle("localization.translation"); //$NON-NLS-1$

private JPanel contentPane;

private JTextField fieldAmount;

private ArrayList<Currency> currencies = Currency.init();

/\*\*

\* Create the mainWindow frame

\*/

public MainWindow()

{

setTitle(BUNDLE.getString("MainWindow.this.title")); //$NON-NLS-1$

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setBounds(100, 100, 589, 300);

setLocationRelativeTo(null);

setResizable( false );

// Create menu bar

JMenuBar menuBar = new JMenuBar();

setJMenuBar(menuBar);

// "File" menu

JMenu mnFile = new JMenu(BUNDLE.getString("MainWindow.mnFile.text"));

//$NON-NLS-1$

mnFile.setMnemonic(KeyEvent.VK\_F);

menuBar.add(mnFile);

// "Quit" menu item

JMenuItem mntmQuit = new JMenuItem(BUNDLE.getString("MainWindow.mntmQuit.text")); //$NON-NLS-1$

mntmQuit.setMnemonic(KeyEvent.VK\_Q);

mntmQuit.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent arg0)

{

System.exit(0);

}

});

mnFile.add(mntmQuit);

// "Help" menu

JMenu mnHelp = new JMenu(BUNDLE.getString("MainWindow.mnHelp.text")); //$NON-NLS-1$

mnHelp.setMnemonic(KeyEvent.VK\_H);

menuBar.add(mnHelp);

// "About" menu item

JMenuItem mntmAbout = new JMenuItem(BUNDLE.getString("MainWindow.mntmAbout.text"));

//$NON-NLS-1$

mntmAbout.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent arg0)

{

EventQueue.invokeLater(new Runnable()

{

public void run()

{

try

{

AboutWindow.getInstance().setVisible(true);

}

catch (Exception e)

{

e.printStackTrace();

}

}

});

}

});

mntmAbout.setMnemonic(KeyEvent.VK\_A);

mnHelp.add(mntmAbout);

// Window components

contentPane = new JPanel();

contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

setContentPane(contentPane);

contentPane.setLayout(null);

// Label "Convert"

JLabel lblConvert = new JLabel(BUNDLE.getString("MainWindow.lblConvert.text")); //$NON-NLS-1$

lblConvert.setHorizontalAlignment(SwingConstants.RIGHT);

lblConvert.setBounds(0, 14, 92, 15);

contentPane.add(lblConvert);

// ComboBox of the first currency

final JComboBox<String> comboBoxCountry1 = new JComboBox<String>();

comboBoxCountry1.setBounds(147, 7, 240, 28);

populate(comboBoxCountry1, currencies);

contentPane.add(comboBoxCountry1);

// Label "To"

JLabel lblTo = new JLabel(BUNDLE.getString("MainWindow.lblTo.text")); //$NON-NLS-1$

lblTo.setHorizontalAlignment(SwingConstants.RIGHT);

lblTo.setBounds(66, 54, 26, 15);

contentPane.add(lblTo);

// ComboBox of the second currency

final JComboBox<String> comboBoxCountry2 = new JComboBox<String>();

comboBoxCountry2.setBounds(147, 47, 240, 28);

populate(comboBoxCountry2, currencies);

contentPane.add(comboBoxCountry2);

// Label "Amount"

final JLabel lblAmount = new JLabel(BUNDLE.getString("MainWindow.lblAmount.text")); //$NON-NLS-1$

lblAmount.setHorizontalAlignment(SwingConstants.RIGHT);

lblAmount.setBounds(23, 108, 69, 15);

contentPane.add(lblAmount);

// Textfield where the user

fieldAmount = new JTextField();

fieldAmount.setText("0.00");

fieldAmount.setBounds(147, 101, 103, 29);

contentPane.add(fieldAmount);

fieldAmount.setColumns(10);

fieldAmount.setDocument(new JTextFieldLimit(8));

// Label displaying result of conversion

final JLabel lblResult = new JLabel("");

lblResult.setHorizontalAlignment(SwingConstants.LEFT);

lblResult.setBounds(147, 188, 428, 38);

contentPane.add(lblResult);

// Button "Convert"

JButton btnConvert = new JButton(BUNDLE.getString("MainWindow.btnConvert.text")); //$NON-NLS-1$

btnConvert.setBounds(147, 142, 129, 38);

btnConvert.addActionListener(new ActionListener()

{

@Override

public void actionPerformed(ActionEvent arg0)

{

String nameCurrency1 = comboBoxCountry1.getSelectedItem().toString();

String nameCurrency2 = comboBoxCountry2.getSelectedItem().toString();

String result;

String formattedPrice;

String formattedAmount;

Double price;

Double amount = 0.0;

DecimalFormat format = new DecimalFormat("#0.00");

try

{

amount = Double.parseDouble( fieldAmount.getText() ) ;

}

catch (NumberFormatException e)

{

e.printStackTrace();

amount = 0.0;

}

price = convert(nameCurrency1, nameCurrency2, currencies, amount);

// Format numbers to avoid "E7" problem

formattedPrice = format.format(price);

formattedAmount = format.format(amount);

result = formattedAmount + " " + nameCurrency1 + " = " + formattedPrice + " " + nameCurrency2;

lblResult.setText(result);

}

});

contentPane.add(btnConvert);

}

// Fill comboBox with currencies name

public static void populate(JComboBox<String> comboBox, ArrayList<Currency> currencies)

{

for (Integer i = 0; i < currencies.size(); i++)

{

comboBox.addItem( currencies.get(i).getName() );

}

}

// Find the short name and the exchange value of the second currency

public static Double convert(String currency1, String currency2, ArrayList<Currency> currencies, Double amount)

{

String shortNameCurrency2 = null;

Double exchangeValue;

Double price = 0.0;

// Find shortname for the second currency

for (Integer i = 0; i < currencies.size(); i++)

{

if (currencies.get(i).getName() == currency2)

{

shortNameCurrency2 = currencies.get(i).getShortName();

break;

}

}

// Find exchange value and call convert() to calcul the new price

if (shortNameCurrency2 != null)

{

for (Integer i = 0; i < currencies.size(); i++)

{

if (currencies.get(i).getName() == currency1)

{

exchangeValue = currencies.get(i).getExchangeValues().get(shortNameCurrency2);

price = Currency.convert(amount, exchangeValue);

break;

}

}

}

return price;

}

}

package converter;

import java.util.ArrayList;

import java.util.HashMap;

public class Currency

{

private String name;

private String shortName;

private HashMap<String, Double> exchangeValues = new HashMap<String, Double>();

// "Currency" Constructor

public Currency(String nameValue, String shortNameValue)

{

this.name = nameValue;

this.shortName = shortNameValue;

}

// Getter for name

public String getName()

{

return this.name;

}

// Setter for name

public void setName(String name)

{

this.name = name;

}

// Getter for shortName

public String getShortName()

{

return this.shortName;

}

// Setter for shortName

public void setShortName(String shortName)

{

this.shortName = shortName;

}

// Getter for exchangeValues

public HashMap<String, Double> getExchangeValues()

{

return this.exchangeValues;

}

// Setter for exchangeValues

public void setExchangeValues(String key, Double value)

{

this.exchangeValues.put(key, value);

}

// Set default values for a currency

public void defaultValues()

{

String currency = this.name;

switch (currency)

{

case "US Dollar":

this.exchangeValues.put("USD", 1.00);

this.exchangeValues.put("EUR", 0.86);

this.exchangeValues.put("PS", 0.77);

this.exchangeValues.put("CHF", 0.92);

this.exchangeValues.put("INR", 74.55);

this.exchangeValues.put("JPY", 104.66);

break;

case "Euro":

this.exchangeValues.put("USD", 1.17);

this.exchangeValues.put("EUR", 1.00);

this.exchangeValues.put("PS", 0.90);

this.exchangeValues.put("CHF", 1.07);

this.exchangeValues.put("INR", 87.05);

this.exchangeValues.put("JPY", 122.21);

break;

case "Pound Sterling":

this.exchangeValues.put("USD", 1.29);

this.exchangeValues.put("EUR", 1.11);

this.exchangeValues.put("PS", 1.00);

this.exchangeValues.put("CHF", 1.19);

this.exchangeValues.put("INR", 96.50);

this.exchangeValues.put("JPY", 135.93);

break;

case "Swiss Franc":

this.exchangeValues.put("USD", 1.09);

this.exchangeValues.put("EUR", 0.93);

this.exchangeValues.put("PS", 0.84);

this.exchangeValues.put("CHF", 1.00);

this.exchangeValues.put("INR", 81.21);

this.exchangeValues.put("JPY", 114.01);

break;

case "Indian Rupees":

this.exchangeValues.put("USD", 0.013);

this.exchangeValues.put("EUR", 0.011);

this.exchangeValues.put("PS", 0.010);

this.exchangeValues.put("CHF", 0.012);

this.exchangeValues.put("INR", 1.00);

this.exchangeValues.put("JPY", 1.41);

break;

case "Japanese Yen":

this.exchangeValues.put("USD", 0.0096);

this.exchangeValues.put("EUR", 0.0082);

this.exchangeValues.put("PS", 0.0074);

this.exchangeValues.put("CHF", 0.0088);

this.exchangeValues.put("INR", 0.71);

this.exchangeValues.put("JPY", 1.000);

break;

}

}

// Initialize currencies

public static ArrayList<Currency> init()

{

ArrayList<Currency> currencies = new ArrayList<Currency>();

currencies.add( new Currency("US Dollar", "USD") );

currencies.add( new Currency("Euro", "EUR") );

currencies.add( new Currency("Pound Sterling", "PS") );

currencies.add( new Currency("Swiss Franc", "CHF") );

currencies.add( new Currency("Indian Rupees", "INR") );

currencies.add( new Currency("Japanese Yen", "JPY") );

for (Integer i =0; i < currencies.size(); i++)

{

currencies.get(i).defaultValues();

}

return currencies;

}

// Convert a currency to another

public static Double convert(Double amount, Double exchangeValue)

{

Double price;

price = amount \* exchangeValue;

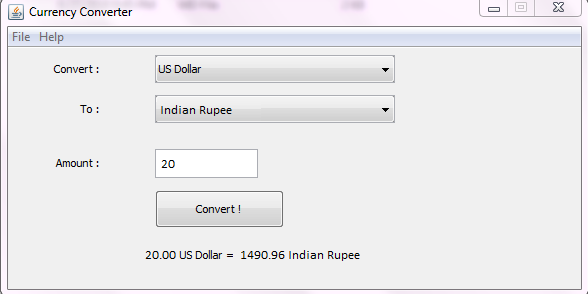
price = Math.round(price \* 100d) / 100d;

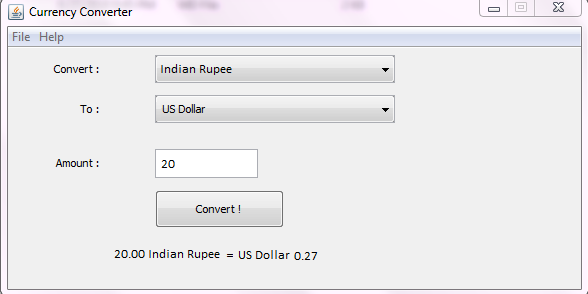
return price;

}

}

**Output:**



****

****

**Conclusion**

We implemented the Currency Conversion System using swing in Advance JAVA programming. In this project we used class, constructor, JLabel, JTextField, JButton, JFrame, for, if, switch case etc. By this Currency Conversion System using swing we successfully implemented the corresponding task.

**References**

* [www.chavanprgatip.wordpress.com](http://www.chavanprgatip.wordpress.com)
* [www.tutorialspoint.com](http://www.tutorialspoint.com)
* <https://www.skyfilabs.com>
* [www.geeksforgeeks.com](http://www.geeksforgeeks.com)
* [www.javatpoint.com](http://www.javatpoint.com)