CURRENCY CONVERTOR

java Mini project report

Submitted by

T.Harshavardhan – 99210041292
T.Shantan - 9921004711
U.Sasi kumar - 99210041646
V.Prasanth - 99210042163

Under the guidance of

MR.K.VIGNESH(M.E)

(Assistant professor,Department of Computer Science and Engineering) in partial fulfilment for the award of degree

Of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

Of

Kalasalingam Academy of Research and Education

(Deemed to be university)
Anand nagar, krishnankovil- 626126
Academic year 2022-2023

KALASALINGAM
ACADEMY OF RESEARCH AND EDUCATION
(DEEMED TO BE UNIVERSITY)
Under sec. 3 of UGC Act 1956. Accredited by NAAC with "A" Grade

BONAFIDE CERTIFICATE

This is to certify that the mini project titled “CURRENCY CONVERTOR” is a bonafide record of workdone by T.Harshavardhan(99210041292), T.Shantan(9921004711), U.Sasi kumar(99210041646) and Prasanth() in partial fulfilment of requirements for the award of the degree of Bachelor of Technology in specialisation of the Computer Science and Engineering during the academic year 2022-2023.

MR K.VIGNESH

Supervisor

Assistant Professor/ CSE

Dr.A.FRANCIS SAVIOUR DEVARAJ

Head of the Department

Department of CSE

Submitted for the University Viva voce Examination held on__________2021.

Internal Examiner

External Examiner

ACKNOWLEDGEMENT

First and foremost, I wish to thank god for his grace to complete this “text editor” project successfully. I would also give special thanks to my dear parents from bottom of my heart for their support to make our project complete.

I express a deep sense of gratitude to “kalvivallal” thiru. t. Kalasalingam, b.com. Founder chairman, “ilaya vallal” and Dr.K.SRIDHARAN ph.d., chancellor, Dr.S.SHASIANANTH, PhD., Vice President (Academic), Mr.S.ARJUNKALASALINGAM M.S.,Vice President (administration), Dr.R.NAGARAJ PhD., vice- chancellor, Dr.V.VASUDEVAN PhD., registrar , Dr.P.DEEPALAKSHMI M.E., PhD., Dean (school of computing). And also a special thanks to Dr. A.FRANCIS XAVIER DEVARAJ Professor & head department of CSE, Kalasalingam Academy of Research and Education for granting the permission and providing necessary facilities to carry out project work.

I am highly indebted to Kalasalingam Academy Of Research And Education (K.A.R.E) and express my special and sincere thanks to my passionate project supervisor Mr K.VIGNESH(M.E) assistant professor, CSE of Kalasalingam Academy of Research And Education [K.A.R.E] for her inspiring guidance, constant encouragement with my work during all stages. I am feeling very glad to do projects under her guidance, which truly practices and appreciates deep thinking. I will be forever indebted to my guide for supporting me in difficult times, when the start of this project and during compilation errors, he gave moral support and let this project move on.

Table of Contents:

Currency Converter

CHAPTER-1

INTRODUCTION:

- Our proposed project is a Currency convertor that makes an easy way to convert the currency into another types.
- Currency converter is a tool used to convert one country's currency to another.
- In this project by the knowledge of usage of currency converter a program is designed in java language.
- In this Currency Converter application, it is going to display a web page where you can choose to display the converter or the exchange rate of one currency with all other currencies in the form of table.
- To change over one cash into another, a user first select the types of currency i.e FROM currency and TO currency and the enter the amount in from currency and then clicks on the button converts then it converts to the currency type of which it has to be converted.
- An unfamiliar exchange expense is a charge required by your credit or check card guarantor or ATM network on a similar exchange. The unfamiliar exchange expense may incorporate the cash change charge, contingent upon whether the card guarantor or ATM network gives that expense to you. (A few cards don't charge unfamiliar exchange expenses.)
- Dynamic cash change (DCC) is normally more expensive than money transformation through the Visa processor, yet it lets you see the expense of your exchange in US dollars when you cause it instead of when you to get your Visa bill.
- The application uses: java programming for front end and MS access as the database and as the backend.

Existing System:

- ➢ In the existing system we can convert a single type of currency into another type according to our selection.
- ➢ It is difficult to do more conversions and it takes time to perform the different types of conversions. Like if we want to do multiple conversions it takes time.
- ➢ We can overcome this by doing small changes in the existing system.

Proposed System:

- The main frame contains three buttons as single conversion, multiple conversion and history.
- Single conversion button converts one type of currency into another type of the currency as per the user input.
- A Button named “Multiple Conversion” converts one type of currency into some of the important currencies.
- Another button named “History” shows the value of currency in the respective year in the respective country.
- By this we can overcome the disadvantages of existing system

CHAPTER-2

FEATURES:

- Different countries use different currency, and there is daily variation in these currencies relative to one another.
- Those who transfer money from one country to another (one currency to another) must be updated with the latest currency exchange rates in the market.
- Currency converter mini project is built keeping this thing in mind.
- This application can be used by any user, but it is mainly useful for business, shares, and finance related areas where money transfer and currency exchange takes place on a daily basis.
- It is a useful tool which gives us the value of certain amount of one currency to be converted in to a different currency.
- In this project we are going to display web window that contains three buttons like single conversion ,multiple conversion and history.
- The respective Buttons provide the output as per the functions allotted to the buttons.
- The main frame contains three buttons as single conversion, multiple conversion and history.
- Single conversion button converts one type of currency into another type of the currency as per the user input.

- A Button named “Multiple Conversion” converts one type of currency into some of the important currencies.
- Another button named “History” shows the value of currency in the respective year in the respective country.
- By this we can overcome the disadvantages of existing system.

CHAPTER-3

SYSTEM ANALYSIS:

The Systems Development Life Cycle (SDLC), or Software Development Life Cycle in systems engineering, information systems and software engineering, is the process of creating or altering systems, and the models and methodologies that people use to develop these systems. In software engineering the SDLC concept underpins many kinds of software development methodologies.

ADVANTAGES:

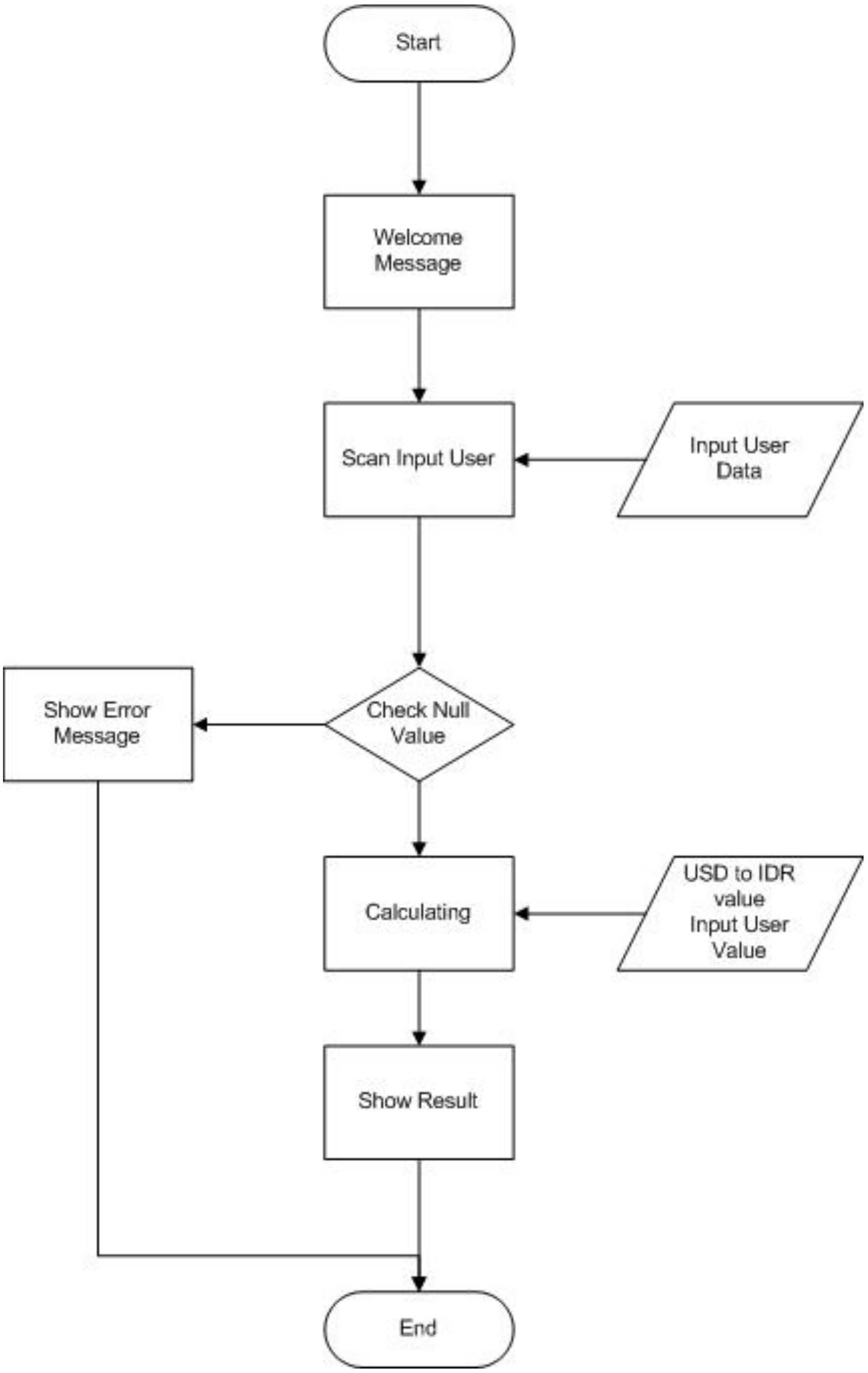
- Less time consumption.
- Easy to convert one type of currency into many other types of currencies.
- Currency history of the country can be calculated easily.
- Easy to understand and use.

DISADVANTAGES:

- There is no real time database. The currency value do not change day to day in our convertor.
- Search history of currency convertor is not available.

CHAPTER-4

SYSTEM DIAGRAM:



CHAPTER-5

SYSTEM DESCRIPTION:

- First page of our output contains a label “Currency convertor” and it is declared as main title. The page contains four buttons i.e Single Conversion, Multiple Conversion ,History and Exit. Each button as its own action and own frames. The exit button is helps to come out of the frame.
- After clicking on Single Conversion a new frame opens labelled Currency Convertor. The frame contains two combo boxes and two text fields and three buttons. First combo box is labelled as a FROM and the combo box contains list of 10 countries and another combo box is labelled as TO and it contains the same list of 10 countries. The two text fields are labelled as units and we should enter the currency value in the first text field after selecting the countries in two combo boxes and then click the convert button. The currency that is converted is displayed in the another text field. Reset button is used to reset all the values and selections those are made in combo box and text fields. The exit button is used to exit the frame and go the first frame.
- After clicking on Multiple conversion another frame is another and it contains a from labelled combo box and unit labelled text fields and three buttons as convert, reset and exit and another 8 text fields are used for the output and those are labelled as units. We should select a country in the combo box and enter the currency value in the text field present at the left and then if we click on the convert button the output text fields those are present at the right are filled with the 8 different currency values and the currency names are displayed in the place of units. Reset button is used to reset all the values and selections those are made in combo box and text fields. The exit button is used to exit the frame and go the first frame.
- After clicking on History button a frame is opened that consists of two combo boxes labelled as country and year and frame consists of text field labelled as result and it contains three buttons as search, reset and exit. The country labelled combo box contains 10 different countries and year labelled combo box contains year from 2010 to 2020 after selecting the country and year in the combo box user should click on the search button and then it displays the value of a dollar in selected country in the selected year in the text field labelled as result. Reset button is used to reset all the values and selections those are made in combo box and text fields. The exit button is used to exit the frame and go the first frame.

CHAPTER-6

MODULE:

Packages those are imported are:

- java.awt.*;
- java.awt.event.ActionEvent;
- java.awt.event.ActionListener;
- java.awt.event.ItemEvent;
- javax.swing.*;

CHAPTER-7

SAMPLE CODE:

```
import java.awt.*;
import java.awt.event.*;
import java.lang.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;

import javax.lang.model.util.ElementScanner14;
//import javax.lang.model.util.ElementScanner14;
import javax.swing.*;
public class convertor extends JFrame implements ActionListener
{
    private static final String BOLD = null;
    JFrame f,f1,f2,f3;
    Label maintitle;
    Button a,b,h;
    JComboBox fromcountry, tocountry,fromyear;
    Button convert,reset,exit;
    Label fromunit,tounit,tounit1,tounit2,tounit3,tounit4,tounit5,tounit6,tounit7,tounit8;
    TextField
fromtext,Answer,totext,totext1,totext2,totext3,totext4,totext5,totext6,totext7,totext8;
    convertor()
    {
        f=new JFrame();
        maintitle = new Label("Currency Converter");
        maintitle.setBounds(150,30,1000,100);
        maintitle.setFont(new Font("Viner Hand ITC",Font.BOLD,100));
        maintitle.setForeground(Color.black);

        a=new Button("Single Conversion");
        a.addActionListener(this);
        b=new Button("Multiple Conversion");
        b.addActionListener(this);
        a.setBounds(200,300,250,60);
        a.setFont(new Font("Times new roman",Font.BOLD, 24));
        b.setBounds(650,300,250,60);
        b.setFont(new Font("Times new roman",Font.BOLD, 24));
        f.add(a);
        f.add(b);
        f.add(maintitle);
        exit=new Button("Exit");
        exit.setBounds(450,500,175,60);
        exit.setFont(new Font("Times new roman",Font.BOLD,24));
        exit.setForeground(Color.black);
        exit.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(ActionEvent evt) {
```

```
            jButton4ActionPerformed(evt);
        }

        private void jButton4ActionPerformed(ActionEvent evt) {
            System.exit(0);
        }
    });
    h=new Button("History");
    h.setBounds(450,400,175,60);
    h.setFont(new Font("Times new roman",Font.BOLD, 24));
    h.addActionListener(this);
    f.add(h);
    f.add(exit);
    f.setSize(2240,1200);
    f.setTitle("Currency Converter");
    f.setLayout(null);
    f.setVisible(true);

}
public void setDefaultCloseOperation(int exitOnClose) {
}
@Override
public void actionPerformed(ActionEvent e) {
    if(e.getSource()==a)
    {
        singleConversion();
    }
    else if(e.getSource()==b){
        multipleConversion();
    }
    else{
        history();
    }

}

public static void main(String args[])
{
    convertor obj=new convertor();
}

public  void history() {
    f3=new JFrame();
    f3.setTitle("Currency Converter");
    f3.setSize(1000,1000);
    f3.setVisible(true);
    f3.setLayout(null);



    maintitle=new Label("History of Currency Converter");
    maintitle.setBounds(130,30,750,100);
    maintitle.setFont(new Font("Viner Hand ITC", Font.BOLD, 50));
    maintitle.setForeground(Color.black);
    f3.add(maintitle);

    JLabel from = new JLabel("Country");
    from.setBounds(10, 160, 150, 50);
    from.setFont(new Font("Mongolian Baiti", Font.BOLD, 20));
    from.setForeground(Color.black);
    f3.add(from);

    String countries[]={"Select One..", "Pakisthan", "India", "Canada", "Kenyan",
    "Nigeria", "Brazil", "Indonesia", "Philippine"};
    fromcountry=new JComboBox(countries);
```

```
fromcountry.setBounds(150, 165, 200, 40);
fromcountry.setFont(new Font("Mongolian Baiti", Font.BOLD, 20));
fromcountry.addItemListener(new java.awt.event.ItemListener() {
    public void itemStateChanged(ItemEvent evt) {
        fromcountryItemStateChanged(evt);
    }

    private void fromcountryItemStateChanged(ItemEvent evt) {
        //int position = fromcountry.getSelectedIndex();
        //fromunit.setText(countries[position]);
    }
});
f3.add(fromcountry);


String years[]={"Select One..", "2010", "2011", "2012", "2013", "2014",
"2015", "2016", "2017", "2018","2019","2020","2021"};
fromyear=new JComboBox(years);
fromyear.setBounds(600, 165, 200, 40);
fromyear.setFont(new Font("Mongolian Baiti", Font.BOLD, 20));
fromyear.addItemListener(new java.awt.event.ItemListener() {
    public void itemStateChanged(ItemEvent evt) {
        fromcountryItemStateChanged(evt);
    }

    private void fromcountryItemStateChanged(ItemEvent evt) {
        //int position = fromyear.getSelectedIndex();
        //fromunit.setText(years[position]);
    }
});
f3.add(fromyear);
```

```
JLabel to = new JLabel("Year");
to.setBounds(850, 160, 150, 50);
to.setFont(new Font("Mongolian Baiti", Font.BOLD, 20));
to.setForeground(Color.black);
f3.add(to);

TextField result=new TextField();
result.setBounds(350, 300, 250, 40);
result.setFont(new Font("Mongolian Baiti", Font.BOLD, 20));
result.setForeground(Color.BLACK);
f3.add(result);

JLabel result1 = new JLabel("Result(1 Dollar):");
result1.setBounds(205, 300, 200, 50);
result1.setFont(new Font("Mongolian Baiti", Font.BOLD, 20));
result1.setForeground(Color.black);
f3.add(result1);

JLabel units = new JLabel("Units");
units.setBounds(600, 300, 400, 50);
units.setFont(new Font("Mongolian Baiti", Font.BOLD, 20));
units.setForeground(Color.black);
f3.add(units);

reset=new Button("Reset");
reset.setBounds(600,450,175,60);
reset.setFont(new Font("Times new roman",Font.BOLD,32));
reset.setForeground(Color.black);
reset.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(ActionEvent evt) {
      jButton4ActionPerformed(evt);
   }
```

```
    }

    private void jButton4ActionPerformed(ActionEvent evt) {
        fromcountry.setSelectedIndex(0);
    fromyear.setSelectedIndex(0);
    result.setText(null);
    units.setText("Units");
    }
});
f3.add(reset);

exit=new Button("Exit");
exit.setBounds(375,550,175,60);
exit.setFont(new Font("Times new roman",Font.BOLD,32));
exit.setForeground(Color.black);
exit.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent evt) {
        f3.dispose();
    }

});
f3.add(exit);

Button search=new Button("Search");
search.setBounds(200,450,175,60);
search.setFont(new Font("Times new roman",Font.BOLD,32));
search.setForeground(Color.black);
search.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }

    private void jButton1ActionPerformed(ActionEvent evt) {
          if (evt.getSource() == search)
        {
          if (fromcountry.getSelectedIndex() == 0 || fromyear.getSelectedIndex() == 0)
               {
             JOptionPane.showMessageDialog(null, "Invalid Input", "Getting Error",
JOptionPane.ERROR_MESSAGE);


          }
          else
          {
             String currencyvalue="";
             switch (fromcountry.getSelectedItem().toString())
             {
                case "Pakisthan":
                if(fromyear.getSelectedIndex()==1)
                currencyvalue="84.38";
                else if(fromyear.getSelectedIndex()==2)
                currencyvalue="86.31";
                else if(fromyear.getSelectedIndex()==3)
                currencyvalue="94.11";
                else if(fromyear.getSelectedIndex()==4)
                currencyvalue="100.47";
                else if(fromyear.getSelectedIndex()==5)
                currencyvalue="101.01";
                else if(fromyear.getSelectedIndex()==6)
                currencyvalue="102.74";
                else if(fromyear.getSelectedIndex()==7)
                currencyvalue="104.62";
                else if(fromyear.getSelectedIndex()==8)
                currencyvalue="105.29";
                else if(fromyear.getSelectedIndex()==9)
                currencyvalue="121.57";
```

```
   else if(fromyear.getSelectedIndex()==10)
   currencyvalue="150.20";
   else if(fromyear.getSelectedIndex()==11)
   currencyvalue="161.83";
   else if(fromyear.getSelectedIndex()==12)
   currencyvalue="162.96";

   else{

   }

   break;
case "India":
if(fromyear.getSelectedIndex()==1)
         currencyvalue="45.73";
      else if(fromyear.getSelectedIndex()==2)
         currencyvalue="46.63";
      else if(fromyear.getSelectedIndex()==3)
         currencyvalue="53.44";
      else if(fromyear.getSelectedIndex()==4)
         currencyvalue="56.57";
      else if(fromyear.getSelectedIndex()==5)
         currencyvalue="62.33";
      else if(fromyear.getSelectedIndex()==6)
      currencyvalue="62.97";
      else if(fromyear.getSelectedIndex()==7)
      currencyvalue="66.46";
      else if(fromyear.getSelectedIndex()==8)
      currencyvalue="67.79";
      else if(fromyear.getSelectedIndex()==9)
      currencyvalue="70.09";
      else if(fromyear.getSelectedIndex()==10)
```

```
    currencyvalue="70.39";
    else if(fromyear.getSelectedIndex()==11)
currencyvalue="76.38";
else if(fromyear.getSelectedIndex()==12)
currencyvalue="74.57";

    else{

    }
    break;
case "Canada":
if(fromyear.getSelectedIndex()==1)
        currencyvalue="1.03";
    else if(fromyear.getSelectedIndex()==2)
        currencyvalue="0.98";
    else if(fromyear.getSelectedIndex()==3)
        currencyvalue="0.99";
    else if(fromyear.getSelectedIndex()==4)
        currencyvalue="1.03";
    else if(fromyear.getSelectedIndex()==5)
        currencyvalue="1.10";
    else if(fromyear.getSelectedIndex()==6)
    currencyvalue="1.27";
    else if(fromyear.getSelectedIndex()==7)
    currencyvalue="1.32";
    else if(fromyear.getSelectedIndex()==8)
    currencyvalue="1.29";
    else if(fromyear.getSelectedIndex()==9)
    currencyvalue="1.29";
    else if(fromyear.getSelectedIndex()==10)
    currencyvalue="1.32";
    else if(fromyear.getSelectedIndex()==11)
```

```
currencyvalue="1.34";
else if(fromyear.getSelectedIndex()==12)
currencyvalue="1.25";


  else{
  }
  break;
  case "Kenyan":
  if(fromyear.getSelectedIndex()==1)
    currencyvalue="79.22";
  else if(fromyear.getSelectedIndex()==2)
    currencyvalue="88.67";
  else if(fromyear.getSelectedIndex()==3)
    currencyvalue="84.49";
  else if(fromyear.getSelectedIndex()==4)
    currencyvalue="86.11";
  else if(fromyear.getSelectedIndex()==5)
    currencyvalue="87.94";
  else if(fromyear.getSelectedIndex()==6)
  currencyvalue="98.23";
  else if(fromyear.getSelectedIndex()==7)
  currencyvalue="101.50";
  else if(fromyear.getSelectedIndex()==8)
  currencyvalue="103.38";
  else if(fromyear.getSelectedIndex()==9)
  currencyvalue="101.28";
  else if(fromyear.getSelectedIndex()==10)
  currencyvalue="101.99";
  else if(fromyear.getSelectedIndex()==11)
currencyvalue="106.48";
else if(fromyear.getSelectedIndex()==12)
currencyvalue="109.66";
```

```
    else{
    }
    break;
    case"Nigeria":
    if(fromyear.getSelectedIndex()==1)
        currencyvalue="122.26";
    else if(fromyear.getSelectedIndex()==2)
        currencyvalue="155.93";
    else if(fromyear.getSelectedIndex()==3)
        currencyvalue="158.79";
    else if(fromyear.getSelectedIndex()==4)
        currencyvalue="159.26";
    else if(fromyear.getSelectedIndex()==5)
        currencyvalue="165.15";
    else if(fromyear.getSelectedIndex()==6)
    currencyvalue="197.87";
    else if(fromyear.getSelectedIndex()==7)
    currencyvalue="257.66";
    else if(fromyear.getSelectedIndex()==8)
    currencyvalue="305.8";
    else if(fromyear.getSelectedIndex()==9)
    currencyvalue="324.2";
    else if(fromyear.getSelectedIndex()==10)
    currencyvalue="325";
    else if(fromyear.getSelectedIndex()==11)
currencyvalue="359.2";
else if(fromyear.getSelectedIndex()==12)
currencyvalue="403.58";

    else{
    }
    break;
    case "Brazil":
    if(fromyear.getSelectedIndex()==1)
        currencyvalue="1.76";
    else if(fromyear.getSelectedIndex()==2)
        currencyvalue="1.67";
    else if(fromyear.getSelectedIndex()==3)
        currencyvalue="1.95";
    else if(fromyear.getSelectedIndex()==4)
        currencyvalue="2.15";
    else if(fromyear.getSelectedIndex()==5)
        currencyvalue="2.35";
    else if(fromyear.getSelectedIndex()==6)
    currencyvalue="3.33";
    else if(fromyear.getSelectedIndex()==7)
    currencyvalue="3.48";
    else if(fromyear.getSelectedIndex()==8)
    currencyvalue="3.19";
    else if(fromyear.getSelectedIndex()==9)
    currencyvalue="3.65";
    else if(fromyear.getSelectedIndex()==10)
    currencyvalue="3.94";
    else if(fromyear.getSelectedIndex()==11)
currencyvalue="5.156";
else if(fromyear.getSelectedIndex()==12)
currencyvalue="5.39";

    else{
    }
    break;
    case "Indonesia":
    if(fromyear.getSelectedIndex()==1)
```

```
        currencyvalue="8991";
    else if(fromyear.getSelectedIndex()==2)
        currencyvalue="9068";
    else if(fromyear.getSelectedIndex()==3)
        currencyvalue="9670";
    else if(fromyear.getSelectedIndex()==4)
        currencyvalue="12189";
    else if(fromyear.getSelectedIndex()==5)
        currencyvalue="12440";
    else if(fromyear.getSelectedIndex()==6)
    currencyvalue="13795";
    else if(fromyear.getSelectedIndex()==7)
    currencyvalue="13436";
    else if(fromyear.getSelectedIndex()==8)
    currencyvalue="13548";
    else if(fromyear.getSelectedIndex()==9)
    currencyvalue="14481";
    else if(fromyear.getSelectedIndex()==10)
    currencyvalue="13901";
    else if(fromyear.getSelectedIndex()==11)
currencyvalue="14105";
else if(fromyear.getSelectedIndex()==12)
currencyvalue="14269";

    else{
    }
    break;
    case"Philippine":
    if(fromyear.getSelectedIndex()==1)
        currencyvalue="45.09";
    else if(fromyear.getSelectedIndex()==2)
        currencyvalue="43.28";
        else if(fromyear.getSelectedIndex()==3)
            currencyvalue="42.21";
        else if(fromyear.getSelectedIndex()==4)
            currencyvalue="42.46";
            else if(fromyear.getSelectedIndex()==5)
    currencyvalue="44.29";
        else if(fromyear.getSelectedIndex()==6)
            currencyvalue="45.52";
        else if(fromyear.getSelectedIndex()==7)
        currencyvalue="47.49";
        else if(fromyear.getSelectedIndex()==8)
        currencyvalue="50.37";
        else if(fromyear.getSelectedIndex()==9)
        currencyvalue="52.66";
        else if(fromyear.getSelectedIndex()==10)
        currencyvalue="51.76";
        else if(fromyear.getSelectedIndex()==11)
        currencyvalue="51.00";
        else if(fromyear.getSelectedIndex()==12)
    currencyvalue="49.27";

        else{
        }
        break;

}
result.setText(currencyvalue);
switch (fromcountry.getSelectedItem().toString())
{
    case "Pakisthan":
    units.setText("Pakisthan_rupee");
    break;
```

```
                case "India":
                units.setText("Indian_rupee");
                break;
                case "Canada":
                units.setText("Canadian_dollor");
                break;
                case "kenyan":
                units.setText("Kenyan_Shilling");
                break;
                case "Nigeria":
                units.setText("Nigerian_Naira");
                break;
                case "Brazil":
                units.setText("Braziliana_Real");
                break;
                case "Indonesia":
                units.setText("Indonesian_Rupiah");
                break;
                case "Philippine":
                units.setText("Philippine_Pisco");
                break;
              }
            }}}});
            f3.add(search);
    }


    public void singleConversion()
    {

      String[] currencyUnits = { "units", "Indian Rupee", "Pakistani Rupee", "US Dollar",
"Canadian Dollar",
        "Kenyan Shilling", "Nigerian Naira", "Brazilian Real", "Indonesian Rupiah",
"Philippine_Pisco", };


    double Indian_Rupee = 93.20;

    double Pakistani_Rupee = 248.67;

    double US_Dollar = 1.13;

    double Canadian_Dollar = 1.55;

    double Kenyan_Shilling = 137.27;

    double Nigerian_Naira = 493.51;

    double Brazilian_Real = 5.95;

    double Indonesian_Rupiah = 17514.43;

    double Philippine_Pisco = 66.66;



        f1=new JFrame();

        f1.setTitle("Currency Converter");

        f1.setSize(1000,1000);

        f1.setVisible(true);

        f1.setLayout(null);



        maintitle=new Label("Currency Converter");

        maintitle.setBounds(250,30,650,100);

        maintitle.setFont(new Font("Viner Hand ITC", Font.BOLD, 50));

        maintitle.setForeground(Color.black);

        f1.add(maintitle);


        JLabel from = new JLabel("From");

        from.setBounds(10, 160, 50, 50);

        from.setFont(new Font("Mongolian Baiti", Font.BOLD, 20));

        from.setForeground(Color.black);

        f1.add(from);
```

```
String country[]={"Select One..", "India", "Pakistan", "USA", "Canada", "Kenyan",
"Nigeria", "Brazil", "Indonesia", "Philippine"};
fromcountry=new JComboBox(country);
fromcountry.setBounds(100, 165, 200, 40);
fromcountry.setFont(new Font("Mongolian Baiti", Font.BOLD, 20));
fromcountry.addItemListener(new java.awt.event.ItemListener() {
    public void itemStateChanged(ItemEvent evt) {
        fromcountryItemStateChanged(evt);
    }

    private void fromcountryItemStateChanged(ItemEvent evt) {
        int position = fromcountry.getSelectedIndex();
        fromunit.setText(currencyUnits[position]);
    }
});
f1.add(fromcountry);

JLabel to = new JLabel("To");
to.setBounds(10, 310, 50, 50);
to.setFont(new Font("Mongolian Baiti", Font.BOLD, 20));
to.setForeground(Color.black);
f1.add(to);

String countries[]={"Select One..", "India", "Pakistan", "USA", "Canada", "Kenyan",
"Nigeria", "Brazil", "Indonesia", "Philippine"};
tocountry=new JComboBox(countries);
tocountry.setBounds(100, 310, 200, 40);
tocountry.setFont(new Font("Mongolian Baiti", Font.BOLD, 20));
tocountry.addItemListener(new java.awt.event.ItemListener() {
    public void itemStateChanged(ItemEvent evt) {
         tocountryItemStateChanged(evt);
    }

    private void tocountryItemStateChanged(ItemEvent evt) {
        int position = tocountry.getSelectedIndex();
        tounit.setText(currencyUnits[position]);
    }
});
f1.add(tocountry);

fromunit=new Label("Units");
fromunit.setBounds(800,160,100,50);
fromunit.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
fromunit.setForeground(Color.black);
f1.add(fromunit);

fromtext=new TextField();
fromtext.setBounds(600, 165, 250, 40);
fromtext.setFont(new Font("Mongolian Baiti", Font.BOLD, 20));
fromtext.setForeground(Color.BLACK);
f1.add(fromtext);

tounit=new Label("Units");
tounit.setBounds(800,310,100,50);
tounit.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
tounit.setForeground(Color.black);
f1.add(tounit);

Answer=new TextField();
Answer.setBounds(600, 310, 250, 40);
Answer.setFont(new Font("Mongolian Baiti", Font.BOLD, 20));
Answer.setForeground(Color.BLACK);
f1.add(Answer);
```

```
reset=new Button("Reset");
reset.setBounds(600,450,175,60);
reset.setFont(new Font("Times new roman",Font.BOLD,32));
reset.setForeground(Color.black);
reset.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(ActionEvent evt) {
      jButton3ActionPerformed(evt);
   }

   private void jButton3ActionPerformed(ActionEvent evt) {
      fromcountry.setSelectedIndex(0);
   tocountry.setSelectedIndex(0);
   fromtext.setText(null);
   Answer.setText(null);
   }
});
f1.add(reset);

exit=new Button("Exit");
exit.setBounds(375,550,175,60);
exit.setFont(new Font("Times new roman",Font.BOLD,32));
exit.setForeground(Color.black);
exit.addActionListener(new ActionListener() {
   public void actionPerformed(ActionEvent evt) {
      f1.dispose();
   }

});
f1.add(exit);
```

```
        f1.setLayout(null);
        f1.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        convert =new Button("Convert");
        convert.setBounds(150,450,175,60);
        convert.setFont(new Font("Times new roman",Font.BOLD,32));
        convert.setForeground(Color.BLACK);
        convert.addActionListener(new java.awt.event.ActionListener() {
          public void actionPerformed(ActionEvent evt) {
            jButton1ActionPerformed(evt);
          }

          private void jButton1ActionPerformed(ActionEvent evt) {
            if (evt.getSource() == convert)
          {
            if (fromcountry.getSelectedIndex() == 0 || tocountry.getSelectedIndex() == 0
                || fromtext.getText().equals(""))
                {
              JOptionPane.showMessageDialog(null, "Invalid Input", "Getting Error",
JOptionPane.ERROR_MESSAGE);

            }
            else
            {
              double amountToChange = Double.parseDouble(fromtext.getText());
              double amountInPounds = 0.0;
              // From Ammout change into pounds
              switch (fromcountry.getSelectedItem().toString())
              {
                case "India":
                  amountInPounds = amountToChange / Indian_Rupee;
                  break;
```

```
    case "Pakistan":
        amountInPounds = amountToChange / Pakistani_Rupee;
        break;
    case "USA":
        amountInPounds = amountToChange / US_Dollar;
        break;
    case "Canada":
        amountInPounds = amountToChange / Canadian_Dollar;
        break;
    case "Kenyan":
        amountInPounds = amountToChange / Kenyan_Shilling;
        break;
    case "Nigeria":
        amountInPounds = amountToChange / Nigerian_Naira;
        break;
    case "Brazil":
        amountInPounds = amountToChange / Brazilian_Real;
        break;
    case "Indonesia":
        amountInPounds = amountToChange / Indonesian_Rupiah;
        break;
    case "Philippine":
        amountInPounds = amountToChange / Philippine_Pisco;
        break;
    default:
        amountInPounds = 0.0;
}

double newamount = 0.0;
switch (tocountry.getSelectedItem().toString()) {
    case "India":
        newamount = amountInPounds * Indian_Rupee;
```

```
                break;
            case "Pakistan":
                newamount = amountInPounds * Pakistani_Rupee;
                break;
            case "USA":
                newamount = amountInPounds * US_Dollar;
                break;
            case "Canada":
                newamount = amountInPounds * Canadian_Dollar;
                break;
            case "Kenyan":
                newamount = amountInPounds * Kenyan_Shilling;
                break;
            case "Ngeria":
                newamount = amountInPounds * Nigerian_Naira;
                break;
            case "Brazil":
                newamount = amountInPounds * Brazilian_Real;
                break;
            case "Indonesia":
                newamount = amountInPounds * Indonesian_Rupiah;
                break;
            case "Philippine":
                newamount = amountInPounds * Philippine_Pisco;
                break;
            default:
                newamount = amountInPounds = 0.0;
        }
        String amount = String.format("%.2f", newamount);
        Answer.setText(amount);
    }
}
```

```
        }


        });


        f1.add(convert);
    }




    public void multipleConversion()
    {


        String[] currencyUnits = { "units", "Indian Rupee", "Pakistani Rupee", "US Dollar",
"Canadian Dollar",
            "Kenyan Shilling", "Nigerian Naira", "Brazilian Real", "Indonesian Rupiah",
"Philippine_Pisco", };


            double Indian_Rupee = 93.20;
            double Pakistani_Rupee = 248.67;
            double US_Dollar = 1.13;
            double Canadian_Dollar = 1.55;
            double Kenyan_Shilling = 137.27;
            double Nigerian_Naira = 493.51;
            double Brazilian_Real = 5.95;
            double Indonesian_Rupiah = 17514.43;
            double Philippine_Pisco = 66.66;
        f2=new JFrame();
        f2.setTitle("Currency Converter");
        f2.setSize(1000,1000);
        f2.setLayout(null);
        f2.setVisible(true);


        maintitle=new Label("Currency converter");
```

```
maintitle.setBounds(250,30,650,100);
maintitle.setFont(new Font("Viner Hand ITC", Font.BOLD, 50));
maintitle.setForeground(Color.black);
f2.add(maintitle);



JLabel from = new JLabel("From");
from.setBounds(10, 160, 50, 50);
from.setFont(new Font("Mongolian Baiti", Font.BOLD, 20));
from.setForeground(Color.black);
f2.add(from);

String country[]={"Select One..", "India", "Pakistan", "USA", "Canada", "Kenyan",
"Nigeria", "Brazil", "Indonesia", "Philippine"};
fromcountry=new JComboBox(country);
fromcountry.setBounds(150, 165, 200, 40);
fromcountry.setFont(new Font("Mongolian Baiti", Font.BOLD, 20));
fromcountry.addItemListener(new java.awt.event.ItemListener() {
    public void itemStateChanged(ItemEvent evt) {
        fromcountryItemStateChanged(evt);
    }

    private void fromcountryItemStateChanged(ItemEvent evt) {
        int position = fromcountry.getSelectedIndex();
        fromunit.setText(currencyUnits[position]);
    }
});
f2.add(fromcountry);

fromtext = new TextField();
fromtext.setBounds(150, 350, 200, 40);
fromtext.setFont(new Font("Mongolian Baiti", Font.BOLD, 20));
```

```
 fromtext.setForeground(Color.BLACK);
 f2.add(fromtext);

 fromunit = new Label("Unit");
 fromunit.setBounds(10, 350, 100, 40);
 fromunit.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
 fromunit.setForeground(Color.BLACK);
 f2.add(fromunit);

totext1 = new TextField();
 totext1.setBounds(500, 160, 250, 30);
 totext1.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
 totext1.setForeground(Color.BLACK);
 f2.add(totext1);

 tounit1 = new Label("Unit");
 tounit1.setBounds(800, 155, 200, 30);
 tounit1.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
 tounit1.setForeground(Color.BLACK);
 f2.add(tounit1);

 totext2 = new TextField();
 totext2.setBounds(500, 200, 250, 30);
 totext2.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
 totext2.setForeground(Color.BLACK);
 f2.add(totext2);

 tounit2 = new Label("Unit");
 tounit2.setBounds(800, 200, 200, 30);
 tounit2.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
 tounit2.setForeground(Color.BLACK);
 f2.add(tounit2);
```

```
totext3 = new TextField();
totext3.setBounds(500, 250, 250, 30);
totext3.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
totext3.setForeground(Color.BLACK);
f2.add(totext3);

tounit3 = new Label("Unit");
tounit3.setBounds(800, 250, 200, 30);
tounit3.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
tounit3.setForeground(Color.BLACK);
f2.add(tounit3);



totext4 = new TextField();
totext4.setBounds(500, 300, 250, 30);
totext4.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
totext4.setForeground(Color.BLACK);
f2.add(totext4);

tounit4 = new Label("Unit");
tounit4.setBounds(800, 300, 200, 30);
tounit4.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
tounit4.setForeground(Color.BLACK);
f2.add(tounit4);



totext5 = new TextField();
totext5.setBounds(500, 350, 250, 30);
totext5.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
totext5.setForeground(Color.BLACK);
f2.add(totext5);
```

```
tounit5 = new Label("Unit");
tounit5.setBounds(800, 350, 200, 30);
tounit5.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
tounit5.setForeground(Color.BLACK);
f2.add(tounit5);


totext6 = new TextField();
totext6.setBounds(500, 400, 250, 30);
totext6.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
totext6.setForeground(Color.BLACK);
f2.add(totext6);

tounit6 = new Label("Unit");
tounit6.setBounds(800, 400, 200, 30);
tounit6.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
tounit6.setForeground(Color.BLACK);
f2.add(tounit6);


totext7 = new TextField();
totext7.setBounds(500, 450, 250, 30);
totext7.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
totext7.setForeground(Color.BLACK);
f2.add(totext7);

tounit7 = new Label("Unit");
tounit7.setBounds(800, 450, 200, 30);
tounit7.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
tounit7.setForeground(Color.BLACK);
f2.add(tounit7);
```

```
totext8 = new TextField();
totext8.setBounds(500, 500, 250, 30);
totext8.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
totext8.setForeground(Color.BLACK);
f2.add(totext8);

tounit8 = new Label("Unit");
tounit8.setBounds(800, 500, 200, 30);
tounit8.setFont(new Font("Mongolian Baiti", Font.BOLD, 15));
tounit8.setForeground(Color.BLACK);
f2.add(tounit8);

reset=new Button("Reset");
reset.setBounds(300,500,100,40);
reset.setFont(new Font("Times new roman",Font.BOLD,28));
reset.setForeground(Color.black);
reset.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(ActionEvent evt) {
        jButton3ActionPerformed(evt);
    }

    private void jButton3ActionPerformed(ActionEvent evt) {
        fromcountry.setSelectedIndex(0);
        fromtext.setText(null);
        totext1.setText(null);
        totext2.setText(null);
        totext3.setText(null);
        totext4.setText(null);
        totext5.setText(null);
        totext6.setText(null);
        totext7.setText(null);
        totext8.setText(null);
    }
});
f2.add(reset);

exit=new Button("Exit");
exit.setBounds(175,550,100,40);
exit.setFont(new Font("Times new roman",Font.BOLD,32));
exit.setForeground(Color.black);
exit.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent evt) {
        f2.dispose();
    }

});
f2.add(exit);

convert =new Button("Convert");
convert.setBounds(50,500,150,40);
convert.setFont(new Font("Times new roman",Font.BOLD,28));
convert.setForeground(Color.BLACK);
convert.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }

    private void jButton1ActionPerformed(ActionEvent evt) {
        if (evt.getSource() == convert)
    {
        if (fromcountry.getSelectedIndex() == 0 || fromtext.getText().equals(""))
                {
```

```
  JOptionPane.showMessageDialog(null, "Invalid Input", "Getting Error",
   JOptionPane.ERROR_MESSAGE);
}
else
{
  double amountToChange = Double.parseDouble(fromtext.getText());
  double amountInPounds = 0.0;
  double newamount=0.0,newamount1=0,newamount2=0,newamount3=0,
   newamount4=0,newamount5=0,newamount6=0,newamount7=0;
  // From Ammout change into pounds
  switch (fromcountry.getSelectedItem().toString())
  {
    case "India":
      amountInPounds = amountToChange / Indian_Rupee;
      newamount=amountInPounds*Pakistani_Rupee;
      tounit1.setText("pakisthan");
      newamount1=amountInPounds*US_Dollar;
      tounit2.setText("US_Dollar");
      newamount2=amountInPounds*Canadian_Dollar;
      tounit3.setText("Canadian_Dollar");
      newamount3=amountInPounds*Kenyan_Shilling;
      tounit4.setText("Kenyan_Shilling");
      newamount4=amountInPounds*Nigerian_Naira;
      tounit5.setText("Nigerian_Naira");
      newamount5=amountInPounds*Brazilian_Real;
      tounit6.setText("Brazilian_Real");
      newamount6=amountInPounds*Indonesian_Rupiah;
      tounit7.setText("Indonesian_Rupiah");
      newamount7=amountInPounds*Philippine_Pisco;
      tounit8.setText("Philippine_Pisco");

      break;
```

```
case "Pakistan":
    amountInPounds = amountToChange / Pakistani_Rupee;
    newamount=amountInPounds*Indian_Rupee;
    tounit1.setText("Indian_Rupee");
    newamount1=amountInPounds*US_Dollar;
    tounit2.setText("US_Dollar");
    newamount2=amountInPounds*Canadian_Dollar;
    tounit3.setText("Canadian_Dollar");
    newamount3=amountInPounds*Kenyan_Shilling;
    tounit4.setText("Kenyan_Shilling");
    newamount4=amountInPounds*Nigerian_Naira;
    tounit5.setText("Nigerian_Naira");
    newamount5=amountInPounds*Brazilian_Real;
    tounit6.setText("Brazilian_Real");
    newamount6=amountInPounds*Indonesian_Rupiah;
    tounit7.setText("Indonesian_Rupiah");
    newamount7=amountInPounds*Philippine_Pisco;
    tounit8.setText("Philippine_Pisco");
    break;
case "USA":
    amountInPounds = amountToChange / US_Dollar;
    newamount=amountInPounds*Indian_Rupee;
    tounit1.setText("Indian_Rupee");
    newamount1=amountInPounds*Pakistani_Rupee;
    tounit2.setText("Pakisthan_rupee");
    newamount2=amountInPounds*Canadian_Dollar;
    tounit3.setText("Canadian_Dollar");
    newamount3=amountInPounds*Kenyan_Shilling;
    tounit4.setText("Kenyan_Shilling");
    newamount4=amountInPounds*Nigerian_Naira;
    tounit5.setText("Nigerian_Naira");
    newamount5=amountInPounds*Brazilian_Real;
    tounit6.setText("Brazilian_Real");
    newamount6=amountInPounds*Indonesian_Rupiah;
    tounit7.setText("Indonesian_Rupiah");
    newamount7=amountInPounds*Philippine_Pisco;
    tounit8.setText("Philippine_Pisco");
    break;
case "Canada":
    amountInPounds = amountToChange / Canadian_Dollar;
    newamount=amountInPounds*Indian_Rupee;
    tounit1.setText("Indian_Rupee");
    newamount1=amountInPounds*US_Dollar;
    tounit2.setText("US_Dollar");
    newamount2=amountInPounds*Pakistani_Rupee;
    tounit3.setText("Pakistani_Rupee");
    newamount3=amountInPounds*Kenyan_Shilling;
    tounit4.setText("Kenyan_Shilling");
    newamount4=amountInPounds*Nigerian_Naira;
    tounit5.setText("Nigerian_Naira");
    newamount5=amountInPounds*Brazilian_Real;
    tounit6.setText("Brazilian_Real");
    newamount6=amountInPounds*Indonesian_Rupiah;
    tounit7.setText("Indonesian_Rupiah");
    newamount7=amountInPounds*Philippine_Pisco;
    tounit8.setText("Philippine_Pisco");
    break;
case "Kenyan":
    amountInPounds = amountToChange / Kenyan_Shilling;
    newamount=amountInPounds*Pakistani_Rupee;
    tounit1.setText("pakisthan");
    newamount1=amountInPounds*US_Dollar;
    tounit2.setText("US_Dollar");
    newamount2=amountInPounds*Canadian_Dollar;
    tounit3.setText("Canadian_Dollar");
    newamount3=amountInPounds*Indian_Rupee;
    tounit4.setText("Indian_Rupee");
    newamount4=amountInPounds*Nigerian_Naira;
    tounit5.setText("Nigerian_Naira");
    newamount5=amountInPounds*Brazilian_Real;
    tounit6.setText("Brazilian_Real");
    newamount6=amountInPounds*Indonesian_Rupiah;
    tounit7.setText("Indonesian_Rupiah");
    newamount7=amountInPounds*Philippine_Pisco;
    tounit8.setText("Philippine_Pisco");
    break;
case "Nigeria":
    amountInPounds = amountToChange / Nigerian_Naira;
    newamount=amountInPounds*Pakistani_Rupee;
    tounit1.setText("pakisthan");
    newamount1=amountInPounds*US_Dollar;
    tounit2.setText("US_Dollar");
    newamount2=amountInPounds*Canadian_Dollar;
    tounit3.setText("Canadian_Dollar");
    newamount3=amountInPounds*Kenyan_Shilling;
    tounit4.setText("Kenyan_Shilling");
    newamount4=amountInPounds*Indian_Rupee;
    tounit5.setText("Indian_Rupee");
    newamount5=amountInPounds*Brazilian_Real;
    tounit6.setText("Brazilian_Real");
    newamount6=amountInPounds*Indonesian_Rupiah;
    tounit7.setText("Indonesian_Rupiah");
    newamount7=amountInPounds*Philippine_Pisco;
    tounit8.setText("Philippine_Pisco");
    break;
case "Brazil":
    amountInPounds = amountToChange / Brazilian_Real;
    newamount=amountInPounds*Pakistani_Rupee;
    tounit1.setText("pakisthan");
    newamount1=amountInPounds*US_Dollar;
    tounit2.setText("US_Dollar");
    newamount2=amountInPounds*Canadian_Dollar;
    tounit3.setText("Canadian_Dollar");
    newamount3=amountInPounds*Kenyan_Shilling;
    tounit4.setText("Kenyan_Shilling");
    newamount4=amountInPounds*Nigerian_Naira;
    tounit5.setText("Nigerian_Naira");
    newamount5=amountInPounds*Indian_Rupee;
    tounit6.setText("Indain_Rupee");
    newamount6=amountInPounds*Indonesian_Rupiah;
    tounit7.setText("Indonesian_Rupiah");
    newamount7=amountInPounds*Philippine_Pisco;
    tounit8.setText("Philippine_Pisco");
    break;
case "Indonesia":
    amountInPounds = amountToChange / Indonesian_Rupiah;
    newamount=amountInPounds*Pakistani_Rupee;
    tounit1.setText("pakisthan");
    newamount1=amountInPounds*US_Dollar;
    tounit2.setText("US_Dollar");
    newamount2=amountInPounds*Canadian_Dollar;
    tounit3.setText("Canadian_Dollar");
    newamount3=amountInPounds*Kenyan_Shilling;
    tounit4.setText("Kenyan_Shilling");
    newamount4=amountInPounds*Nigerian_Naira;
    tounit5.setText("Nigerian_Naira");
    newamount5=amountInPounds*Brazilian_Real;
    tounit6.setText("Brazilian_Real");
```

```
        newamount6=amountInPounds*Indian_Rupee;
        tounit7.setText("Indain_rupee");
        newamount7=amountInPounds*Philippine_Pisco;
        tounit8.setText("Philippine_Pisco");
        break;
    case "Philippine":
        amountInPounds = amountToChange / Philippine_Pisco;
        newamount=amountInPounds*Pakistani_Rupee;
        tounit1.setText("pakisthan");
        newamount1=amountInPounds*US_Dollar;
        tounit2.setText("US_Dollar");
        newamount2=amountInPounds*Canadian_Dollar;
        tounit3.setText("Canadian_Dollar");
        newamount3=amountInPounds*Kenyan_Shilling;
        tounit4.setText("Kenyan_Shilling");
        newamount4=amountInPounds*Nigerian_Naira;
        tounit5.setText("Nigerian_Naira");
        newamount5=amountInPounds*Brazilian_Real;
        tounit6.setText("Brazilian_Real");
        newamount6=amountInPounds*Indonesian_Rupiah;
        tounit7.setText("Indonesian_Rupiah");
        newamount7=amountInPounds*Indian_Rupee;
        tounit8.setText("Indian_Rupee");
        break;
    default:
        amountInPounds = 0.0;
}
String amount1 = String.format("%.2f", newamount);
String amount2= String.format("%.2f", newamount1);
String amount3= String.format("%.2f", newamount2);
String amount4= String.format("%.2f", newamount3);
String amount5= String.format("%.2f", newamount4);
```

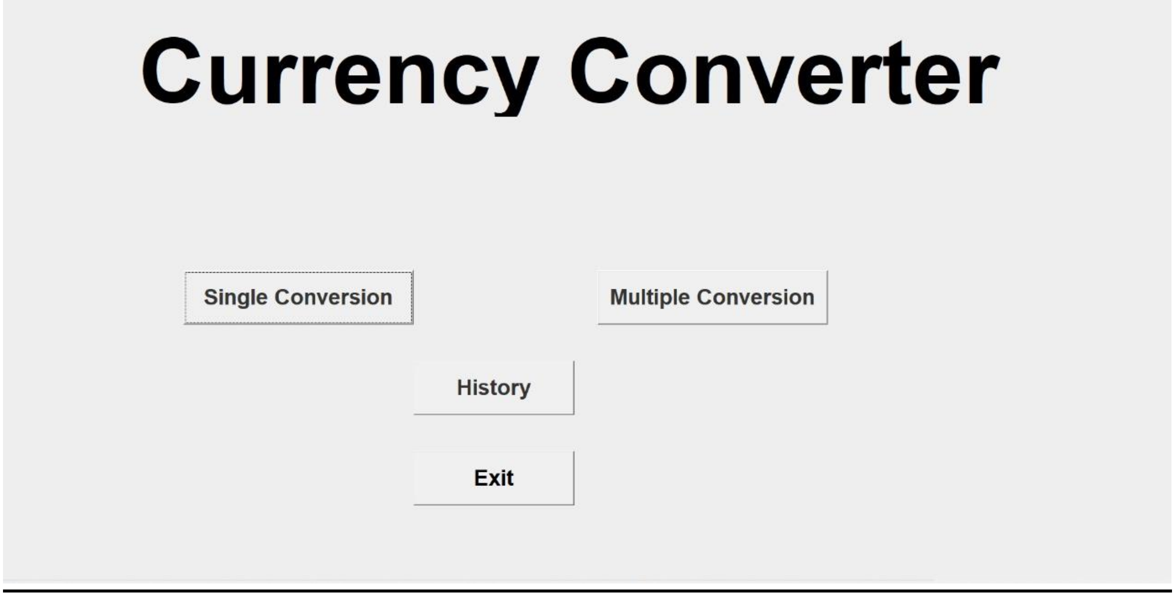
```
          String amount6= String.format("%.2f", newamount5);
          String amount7= String.format("%.2f", newamount6);
          String amount8= String.format("%.2f", newamount7);
          totext1.setText(amount1);
          totext2.setText(amount2);
          totext3.setText(amount3);
          totext4.setText(amount4);
          totext5.setText(amount5);
          totext6.setText(amount6);
          totext7.setText(amount7);
          totext8.setText(amount8);
        }
    }
 }

    });
    f2.add(convert);
  }
}
```

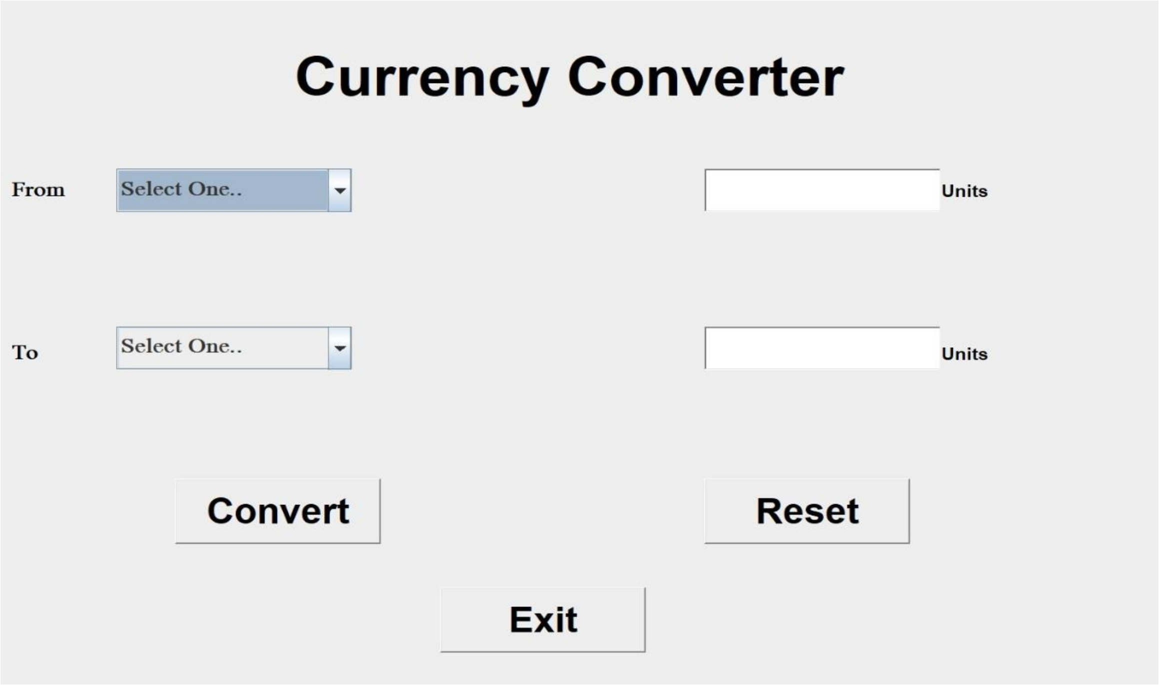
CHAPTER-8

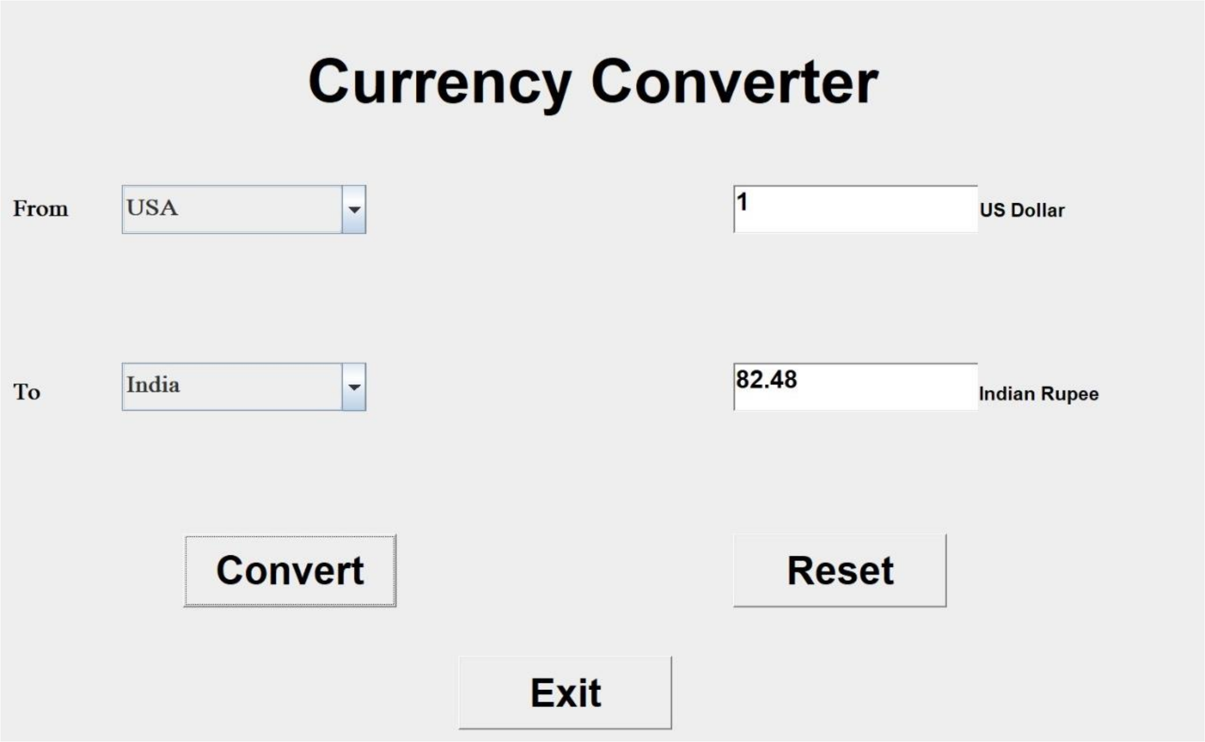
OUTPUT SCREENSHOTS:

Main Frame:



Single Conversion:



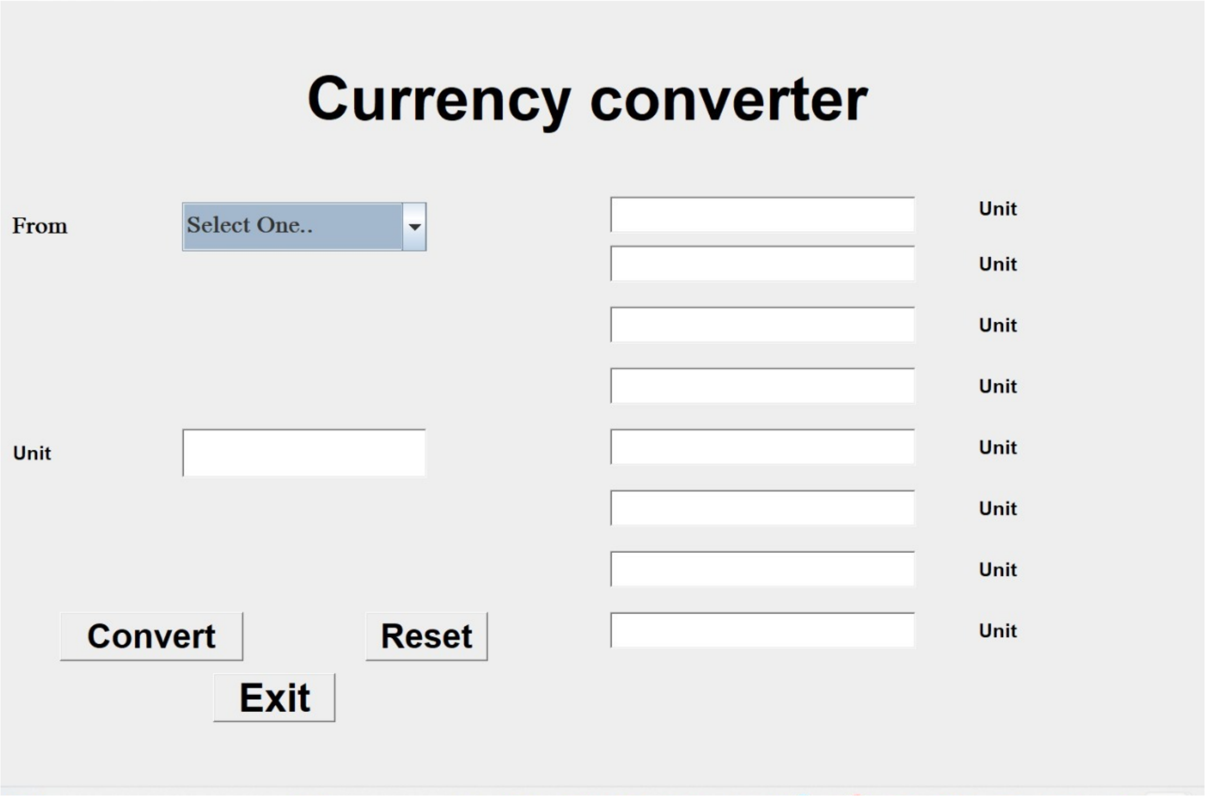


Currency Converter

From USA 1 US Dollar

To India 82.48 Indian Rupee

Multiple Conversion:

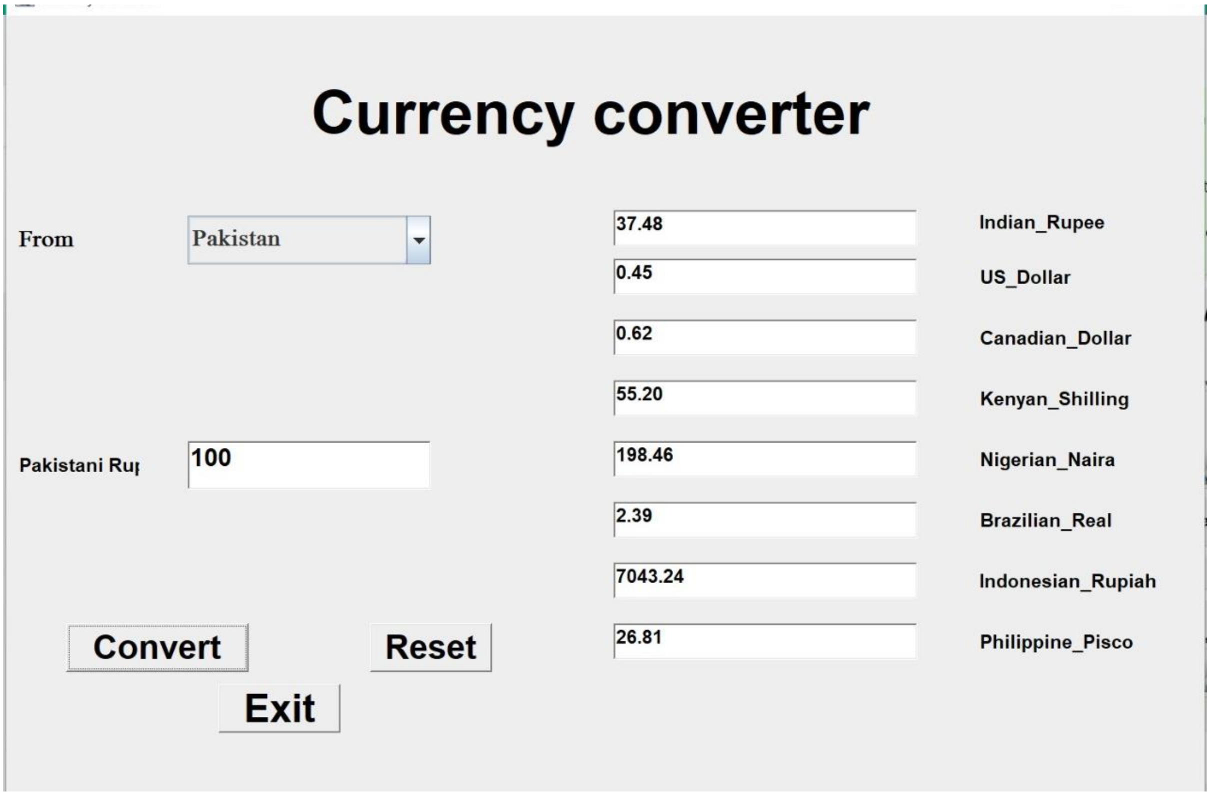


Currency converter

From Select One..

Unit

Unit
Unit
Unit
Unit
Unit
Unit
Unit
Unit



Currency converter

From: Pakistan

Pakistani Ru: 100

37.48 Indian_Rupee

0.45 US_Dollar

0.62 Canadian_Dollar

55.20 Kenyan_Shilling

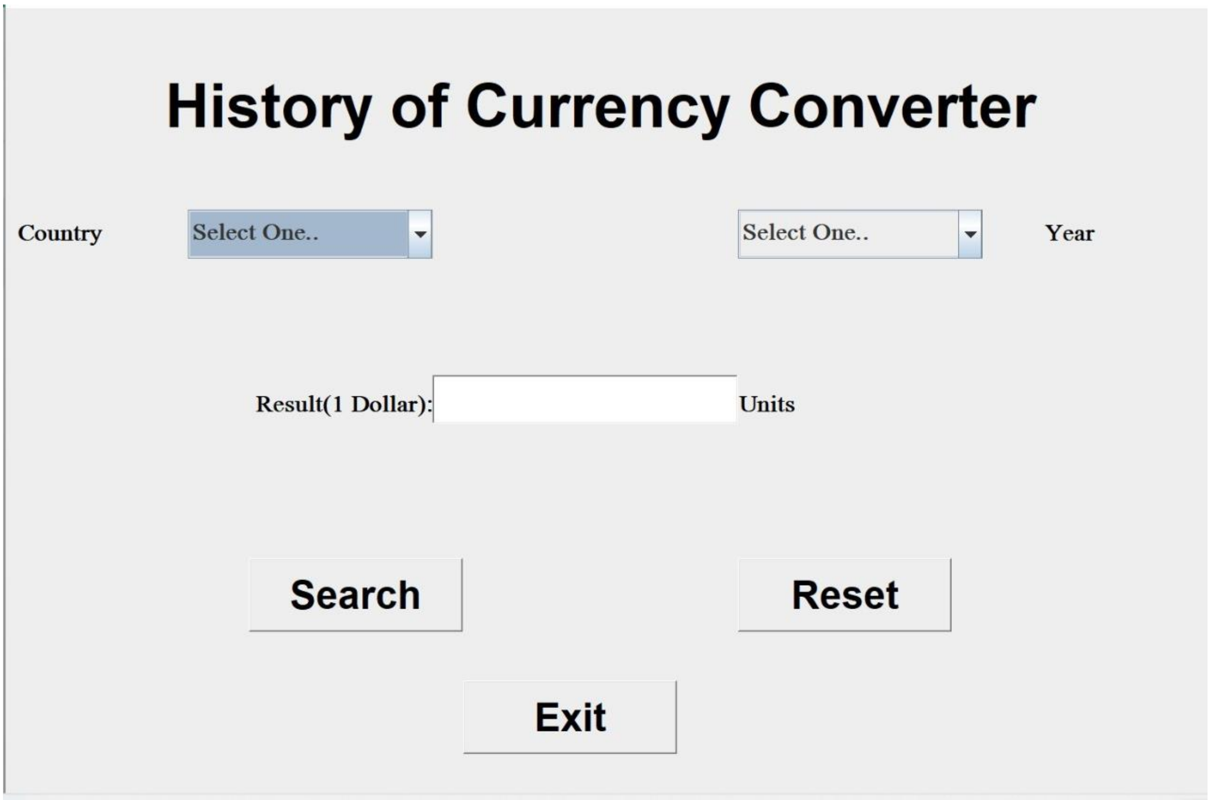
198.46 Nigerian_Naira

2.39 Brazilian_Real

7043.24 Indonesian_Rupiah

26.81 Philippine_Pisco

History:

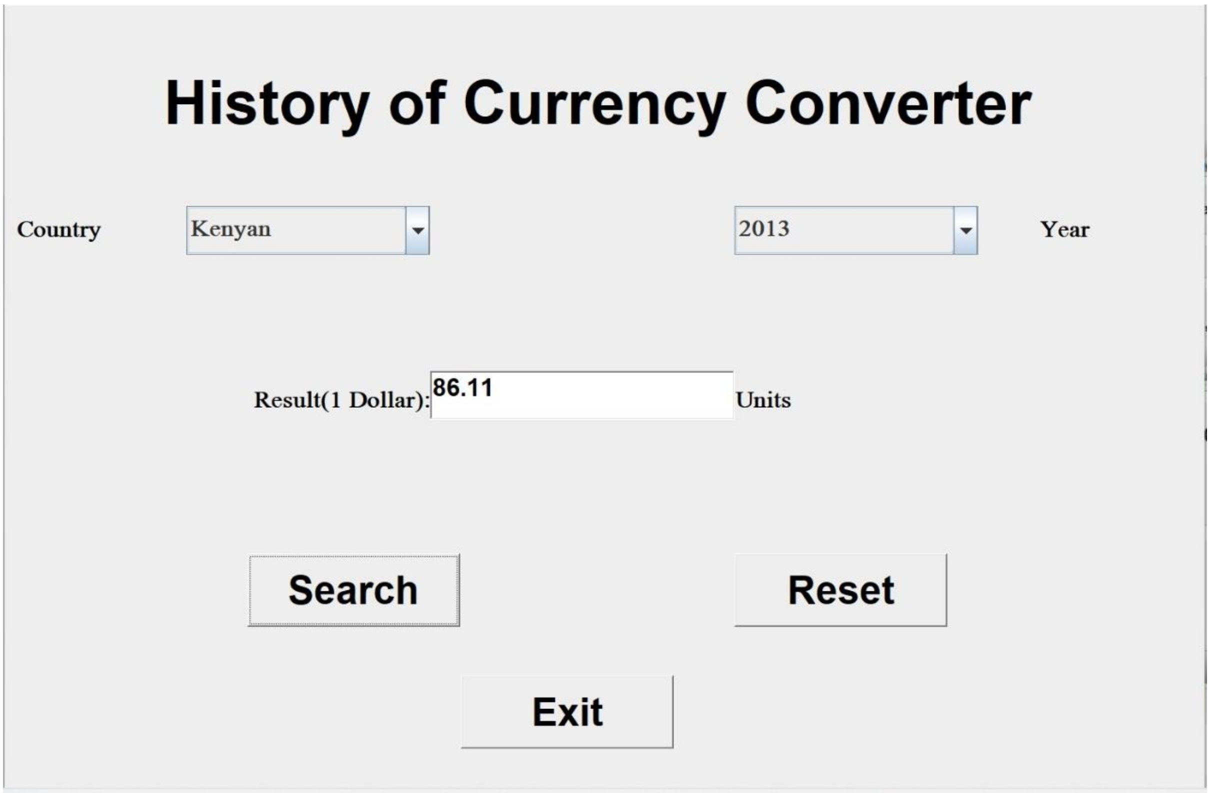


History of Currency Converter

Country: Select One..

Select One.. Year

Result(1 Dollar): Units



History of Currency Converter

Country: Kenyan

Year: 2013

Result(1 Dollar): 86.11 Units

CHAPTER-9

CONCLUSION:

- The currency convertor application helped the people who are in investing in stocks they can verify daily the change in currency in different countries.
- It is also helpful for the people who wants to travel different countries.
- The conversions are fast when you want to convert from one country to many other different countries.
- The application contains three features it can convert single type of currency into another type and it can convert single type in to multiple types of currency and it can provide the history of the currency in the respected country in respected year.
- The Currency convertor application is easy to use and it can be access by everyone from everywhere.

CHAPTER-10

REFERENCES:

- Javatpoint
- Greek for Greeks
- https://projectworlds.in/java-projects-with-source-code/currency-converter-java-mini-project/.
- Amandeep Kaur Research Scholar,“An Empirical Estimation of Exchange rate Determination in India : Since 1980- 2011”, IJSER 1 ISSN 2229-5518, 2012.

- Ahmed Amine Lamzour “Literature Review: Fundamental Analysis and Technical Analysis of the Exchange Rate” IJSER Volume 9, Issue 8, August-2018 ISSN 2229-5518.

- Leyla Ahmed “The Effect of Foreign Exchange Exposure on the Financial Performance of Commercial Banks in Kenya” IJSER, Volume 5, Issue 11, November 2015 ISSN 2250[1]November 2015 ISSN 2250-3153.-3153.

- Khaled Alotaibi “How Exchange Rate Influence a Country’s Import and Export” IJSER, Volume 7, Issue 5, May2016 ISSN 2229-5518.

- Dr. Devajit Mahanta “Indian Currency Futures: An Analytical study ofits performance “ International Journal of Marketing, Financial Services & Management Research Vol.1 Issue 11, November 2012, ISSN 2277 3622.