Currency Converter Using Java

A PROJECT REPORT

Submitted by

Subatra Tirwa(22BCA11132)

in partial fulfilment for the award of the degree of

BACHELOR OF COMPUTER APPLICATION

Chandigarh University

April 2025

CHANDIGARH UNIVERSITY

Discover. Learn. Empower.

BONAFIDE CERTIFICATE

Certified that this project report "**Currency Converter Using Java** is the bonafide work of **Subatra Tirwa** who carried out the project work under my/our supervision.

SIGNATURE	**SIGNATURE**
Dr. Kavita Gupta	**Mr Suma Acharya(E17492)**
HEAD OF THE DEPARTMENT	**SUPERVISOR**
BCA	

Submitted for the project viva-voce examination held on ……………………..

Contents

1. Introduction

In today’s globalized world, currency exchange has become an essential activity, especially for international trade, travel, and finance. With varying exchange rates and multiple currencies in use, a currency converter plays a significant role in helping users determine the equivalent value of one currency in terms of another.

This project, titled **"Currency Converter Using Java (GUI-Based)"**, is developed to provide users with an easy and efficient way to convert currencies using a simple graphical user interface. The application is built using **Java Swing**, a lightweight GUI toolkit in Java, which offers components such as buttons, text fields, labels, and dropdown menus to create a user-friendly interface.

The application supports conversion between five major currencies: **USD (United States Dollar), EUR (Euro), GBP (British Pound), JPY (Japanese Yen), and INR (Indian Rupee)**. The currency values are converted based on predefined exchange rates hardcoded within the program.

2. Objective

The primary objective of this project is to design and implement a **currency converter application** using Java that allows users to convert an entered amount from one currency to another in a quick and accurate manner.

Key objectives include:

- Developing an interactive **Graphical User Interface (GUI)** using Java Swing.
- Implementing logic for currency conversion using **predefined exchange rates**.
- Handling **user input validation** and error checking for smooth operation.
- Creating a beginner-friendly Java project demonstrating real-world use.

3. Tools and Technologies Used

The following tools and technologies were used to develop this project:

Tool/Technology	**Description**
Java SE (Standard Edition)	Core programming language used
Swing (Java GUI Toolkit)	Used for building the GUI interface
JDK 8 or above	Java Development Kit for compiling/running code
IDE (Eclipse / IntelliJ / NetBeans)	For writing and executing the program

Tool/Technology	**Description**
Operating System	Windows / Linux / macOS

4.Features of the Project

The project offers the following features:

- **Graphical User Interface (GUI):** Clean and intuitive layout using Swing components.
- **Bidirectional Conversion:** Allows users to choose any two currencies from the list.
- **Amount Entry:** Accepts user input for the amount to convert.
- **Convert Button:** Executes conversion when clicked.
- **Error Handling:** Detects and informs users about invalid inputs.
- **Real-Time Result Display:** Shows the converted value instantly.

5. Working of the Project

The working of the Currency Converter can be summarized in the following steps:

1. The user **launches the application** and is presented with the GUI.
2. The user selects:
   - A **"From Currency"** (e.g., USD)
   - A **"To Currency"** (e.g., INR)
3. The user then **enters the amount** to be converted in a text field.
4. Upon clicking the **"Convert" button**, the application:
   - Validates the input (checks for valid numeric value)
   - Retrieves the predefined **conversion rate**
   - Calculates the converted amount
   - Displays the result in a readable format with the target currency
5. If the input is invalid, an **error message** is shown using a dialog box.

6. Conversion Rates Used

In this project, the currency conversion is performed using predefined exchange rates. These rates are hardcoded within the program and can be modified as required. The rates used are as follows:

Currency Pair	Conversion Rate
USD → INR	83.12
EUR → INR	89.23
GBP → INR	104.45
JPY → INR	0.56
INR → INR	1.00

7. Code Summary

The currency converter is built using **Java Swing** for the GUI. Below is a brief summary of how the application works:

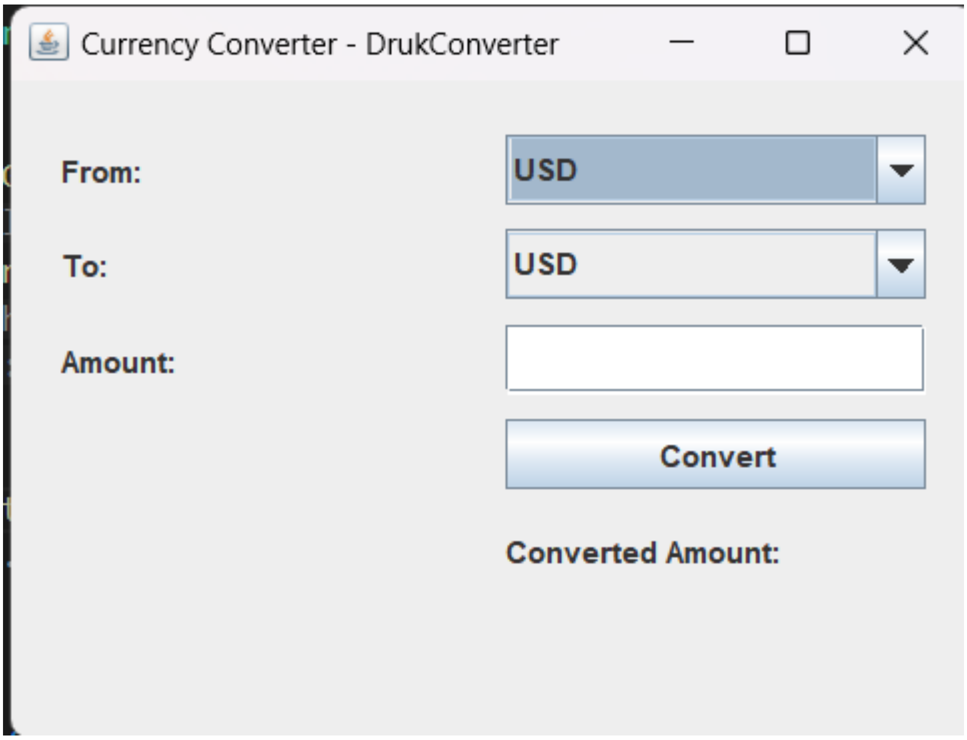
- **JFrame** is used as the main window.
- **JLabel** displays static texts like "Amount", "From", "To", etc.
- **JTextField** takes user input for the amount.
- **JComboBox** allows selection of source and target currencies.
- **JButton** triggers the conversion when clicked.
- A method getConversionRate(String from, String to) is implemented to return the relevant rate.
- The actionPerformed() method handles the click event and performs:
  - Input validation
  - Currency conversion
  - Result display using JOptionPane or another JLabel.

Example code snippet (for illustration):

```
double amount = Double.parseDouble(textField.getText());

String fromCurrency = (String) fromCombo.getSelectedItem();

String toCurrency = (String) toCombo.getSelectedItem();

double rate = getConversionRate(fromCurrency, toCurrency);

double result = amount * rate;

resultLabel.setText("Converted Amount: " + result);
```

8.Output Screenshot

Below is a sample screenshot of the Currency Converter application after a successful currency conversion:



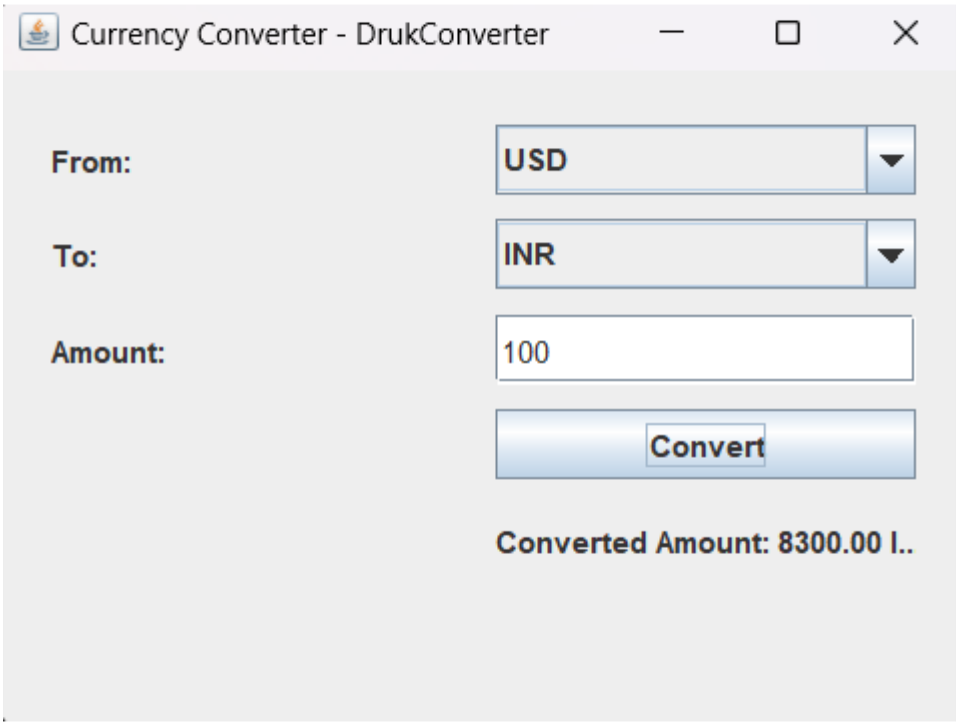
Currency Converter - DrukConverter

From: USD

To: USD

Amount:

Converted Amount:



Currency Converter - DrukConverter

From: USD

To: INR

Amount: 100

Converted Amount: 8300.00 I..

Description of Screenshot:

- The user has entered an amount of **100**.

- The selected source currency is **USD**.
- The selected target currency is **INR**.
- Upon clicking the **Convert** button, the application displays the converted amount: **Converted Amount: 8300.00 INR**

9.Advantages

The Currency Converter project offers several benefits:

- **User-Friendly Interface:** Easy to use, even for beginners.
- **Educational Purpose:** Good project for understanding Java GUI, event handling, and basic logic.
- **Offline Use:** Works without an internet connection (since it uses predefined rates).
- **Lightweight Application:** No heavy installations or dependencies required.
- **Extendable:** Can be enhanced to support real-time exchange rates or additional currencies.

10.Future Enhancements

Although the current version of the Currency Converter is functional and efficient for basic use, there is potential for several enhancements to improve its capabilities and user experience:

- **Real-Time Exchange Rates:** Integrate the application with a live currency exchange API (such as OpenExchangeRates, Fixer.io, or ExchangeRate-API) to fetch real-time conversion rates.
- **More Currencies:** Extend support to a wider range of global currencies beyond the current five.
- **Conversion History:** Store and display a log of past conversions for user reference.
- **Enhanced UI Design:** Improve the visual aesthetics using advanced styling and layout managers.
- **Mobile App Version:** Develop Android or iOS versions using platforms like Kotlin, Flutter, or React Native.

11.Conclusion

The **Currency Converter using Java** is a simple yet effective application that serves the basic purpose of converting one currency into another using predefined exchange rates. The project demonstrates a clear understanding of:

- Java programming fundamentals
- GUI development using Swing components
- Event handling and user input validation

It is an excellent beginner-level project that lays the foundation for developing more complex applications. With a few enhancements, this currency converter can evolve into a robust, real-world financial tool capable of assisting users with up-to-date currency conversion.