Malla Reddy Engineering College
(UGC Autonomous Institution, Approved by AICTE, & Affiliated to JNTUH).
Accredited by NAAC with 'A++' Grade (III Cycle),
Maisammaguda (H), Medchal-Malkajgiri District, Secunderabad, Telangana –500100,
www.mrec.ac.in

B-Tech / ll-year -l semester
A Project Report
on

CURRENCY CONVERTER BY JAVA

2025-2026

JAVA PROGRAMMING

Submitted by
Name : G.Nagendra
Roll no.24J41A1277
Class &sec :IT-A
Under the Guidence of
PUSHPA MAM

🎯 Objective:

To convert an amount from one currency to another (e.g., USD → INR, EUR → GBP, etc.) based on predefined or live exchange rates

🧩 Project Type Options:

Console-based (Simple) – using fixed conversion rates (ideal for beginners)

API-based (Advanced) – fetch live rates using an API like ExchangeRate API

Below, I'll show both versions

🟢 1. Console-Based Currency Converter (Beginner Level)

✅ Features:

User inputs amount

User selects source and target currency

Program calculates and displays converted amount

🧮 Example Conversion Rates

From To Rate →

INR83.20 INR → USD0.012

EUR → INR89.50

GBP → INR102.75

INR → EUR0.011

INR → GBP 0.0097

SOURCE CODE:

```
import java.util.Scanner;


public class CurrencyConverter {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);


        System.out.println("================================");
        System.out.println("     CURRENCY CONVERTER (JAVA) ");
        System.out.println("================================");


        System.out.println("Available Currencies:");
        System.out.println("1. USD (US Dollar)");
        System.out.println("2. INR (Indian Rupee)");
        System.out.println("3. EUR (Euro)");
        System.out.println("4. GBP (British Pound)");


        System.out.print("\nEnter the source currency code (e.g., USD): ");
        String from = sc.next().toUpperCase();


        System.out.print("Enter the target currency code (e.g., INR): ");
        String to = sc.next().toUpperCase();


        System.out.print("Enter the amount to convert: ");
        double amount = sc.nextDouble();


        double rate = getExchangeRate(from, to);
```

```
        if(rate == 0) {
            System.out.println("❌ Conversion rate not available for selected currencies!");
        }else {
            double converted = amount * rate;
            System.out.printf("\n✅ %.2f %s = %.2f %s\n", amount, from, converted, to);
        }

        System.out.println("\nThank you for using the Currency Converter!");
        sc.close();
    }

    //Method to get predefined conversion rates
    public static double getExchangeRate(String from, String to) {
        if (from.equals("USD") && to.equals("INR")) return 83.20;
        if (from.equals("INR") && to.equals("USD")) return 0.012;
        if (from.equals("EUR") && to.equals("INR")) return 89.50;
        if (from.equals("INR") && to.equals("EUR")) return 0.011;
        if (from.equals("GBP") && to.equals("INR")) return 102.75;
        if (from.equals("INR") && to.equals("GBP")) return 0.0097;
        return 0;
    }
}
```

OUTPUT

==========

CURRENCY CONVERTER (JAVA)

==========

Available Currencies:

1. USD (US Dollar)
2. INR (Indian Rupee)
3. EUR (Euro)
4. GBP (British Pound)

✅ 100.00 USD = 8320.00 INR

Thank you for using the Currency Converter!

how it will work :

1. **User Input:**
The user enters the amount of money and selects the source and target currencies (for example, USD to INR).

2. **Fetch Exchange Rate:**
The program retrieves the current exchange rate between the selected currencies — either from predefined values or from an online API.

3. **Perform Calculation:**
The converter multiplies the entered amount by the exchange rate to get the converted value.

4. **Display Result:**
The converted amount is shown to the user, clearly indicating both currencies.

5. **Optional Enhancements:**
Features like currency swapping, saving conversion history, or fetching live rates can be added for better functionality.

Conclusion:

The Currency Converter project helps users easily convert amounts from one currency to another using predefined or live exchange rates.

It demonstrates the practical use of arithmetic operations and conditional logic in Java.

By automating currency conversion, it saves time and reduces manual calculation errors.

The project also provides a clear understanding of how exchange rates work globally.

It can be further enhanced by adding live data integration and a user-friendly GUI.

Overall, this project is a simple yet effective application of Java programming in the financial domain.

It strengthens both logical thinking and real-world problem-solving skills.