Project Report: Currency Converter in Java

1. Introduction

The objective of this project is to develop a simple Currency Converter application using Java. The application allows users to convert an amount from one currency to another using predefined exchange rates. This project demonstrates fundamental Java programming concepts such as user input handling, control structures, and basic arithmetic operations.

---

2. Project Objectives

- To create a console-based currency converter in Java.
- To enable conversion between USD, EUR, and INR.
- To demonstrate the use of Java's Scanner class for user input.
- To apply conditional logic for currency conversion.
- To provide a foundation for future enhancements such as real-time rate fetching and GUI integration.

---

3. System Analysis

3.1 Existing System

Currency conversion is commonly done using online tools or mobile apps. However, these often require internet connectivity and may not be available as simple desktop applications.

3.2 Proposed System

The proposed system is a lightweight Java application that runs on any machine with Java installed. It provides quick currency conversion without the need for an internet connection.

---

4. System Design

4.1 Architecture

- Input: User enters the amount, source currency, and target currency.
- Processing: The application checks the currency combination and applies the corresponding conversion rate.
- Output: The converted amount is displayed to the user.

4.2 Flowchart

text

```
Start
  |
  v
Prompt user for amount, source currency, and target currency
  |
  v
Check if currency combination is valid
  |
  v
If valid:
    Apply conversion rate and display result
Else:
    Display error message
  |
  v
End
```

4.3 User Interface

- Console-based, text prompts for input and output.

---

5. Implementation

5.1 Tools & Technologies

- Programming Language: Java
- IDE: Any Java IDE (Eclipse, IntelliJ IDEA, NetBeans, etc.)
- JDK Version: 8 or above

5.2 Code Overview

java

```java
import java.util.Scanner;

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

        System.out.print("Enter the amount: ");
        double amount = scanner.nextDouble();

        System.out.print("Enter the currency to convert from (USD, EUR, INR): ");
        String fromCurrency = scanner.next().toUpperCase();

        System.out.print("Enter the currency to convert to (USD, EUR, INR): ");
        String toCurrency = scanner.next().toUpperCase();

        double conversionRate = 0;

        if (fromCurrency.equals("USD") && toCurrency.equals("EUR")) {
            conversionRate = 0.92;
        } else if (fromCurrency.equals("USD") && toCurrency.equals("INR")) {
            conversionRate = 82.75;
        } else if (fromCurrency.equals("EUR") && toCurrency.equals("USD")) {
            conversionRate = 1.08;
        } else if (fromCurrency.equals("EUR") &&
toCurrency.equals("INR")) {
            conversionRate = 89.65;
        } else if (fromCurrency.equals("INR") &&
toCurrency.equals("USD")) {
            conversionRate = 0.012;
        } else if (fromCurrency.equals("INR") &&
toCurrency.equals("EUR")) {
            conversionRate = 0.011;
        } else {
            System.out.println("Invalid currency combination.");
            return;
        }

        double convertedAmount = amount * conversionRate;
        System.out.println("Converted amount: " +
convertedAmount + " " + toCurrency);
    }
}
```

6. Testing

- Test Case 1: Convert 100 USD to INR → Output: 8275.0 INR
- Test Case 2: Convert 50 EUR to USD → Output: 54.0 USD
- Test Case 3: Invalid currency input → Output: "Invalid currency combination."

7. Limitations

- Only three currencies supported (USD, EUR, INR).
- Exchange rates are hardcoded and may not reflect real-time values.
- No error handling for non-numeric input.
- Console-based interface only.

8. Future Enhancements

- Integrate with live currency exchange APIs for real-time rates.
- Add more currencies.
- Implement a graphical user interface (GUI) using Swing or JavaFX.
- Add input validation and exception handling.
- Allow for batch conversions and history tracking.

---

9. Conclusion

The Java Currency Converter project demonstrates the basics of Java programming, user interaction, and control structures. It provides a foundation for more advanced applications and can be extended with real-time data and a graphical interface for enhanced usability.

---

10. References

- Java Documentation
- Currency Exchange Rates (Wikipedia)
- Java Scanner Class

Prepared by:
[Your Name]
Date: June 21, 2025