K.RAMAKRISHNAN
COLLEGE OF ENGINEERING
An Autonomous Institution
Permanently Affiliated to Anna University Chennai, Approved by AICTE New Delhi,
ISO 9001:2015, 14001:2015 certified institution, Accredited by NBA and with A grade by NAAC
Samayapuram, Tiruchirappalli – 621 112, Tamilnadu, India.

VARIOUS NATION CURRENCY CONVERTER

A PROJECT REPORT

Submitted by

MOHAMED IRFAN P(8115U23ME028)

in partial fulfillment of requirements for the award of the course

MGB1201 - PYTHON PROGRAMMING

in

DEPARTMENT OF MECHANICAL ENGINEERING

K. RAMAKRISHNAN COLLEGE OF ENGINEERING

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM – 621 112

DECEMBER - 2024

K. RAMAKRISHNAN COLLEGE OF ENGINEERING

(Autonomous Institution affiliated to Anna University, Chennai)

TRICHY-621 112

BONAFIDE CERTIFICATE

Certified that this project report on **"VARIOUS NATION CURRENCY CONVERTER"** is the bonafide work of **MOHAMED IRFAN P(8115U23ME028)** who carried out the project work during the academic year 2024 - 2025 under my supervision.

SIGNATURE
Dr. T. M. NITHYA, M.E.,Ph.D.,
HEAD OF THE DEPARTMENT
ASSOCIATE PROFESSOR
Department of CSE
K.Ramakrishnan College of Engineering (Autonomous)
Samayapuram–621112.

SIGNATURE
Mrs.S.RAJESWARI M.E.
SUPERVISOR
ASSISTANT PROFESSOR
Department of CSE
K.Ramakrishnan College of Engineering (Autonomous)
Samayapuram–621112.

Submitted for the End Semester Examination held on…….

INTERNALEXAMINER **EXTERNALEXAMINER**

DECLARATION

I declare that the project report on **"VARIOUS NATION CURRENCY CONVERTER"** is the result of original work done by us and best of our knowledge, similar workhas not been submitted to **"ANNA UNIVERSITY CHENNAI"** for the requirement of Degree of **BACHELOR OF ENGINEERING**. This project report is submitted on the partial fulfilment of the requirement of the completion of the course **MGB1201 –PYTHON PROGRAMMING**

.

Signature

_____

MOHAMED IRFAN P

Place: Samayapuram

Date:

ACKNOWLEDGEMENT

It is with great pride that I express our gratitude and in-debt to our institution "**K.Ramakrishnan College of Engineering (Autonomous)**", for providing us with the opportunity to do this project.

I glad to credit honourable chairman **Dr. K. RAMAKRISHNAN**, **B.E.,** for having provided for the facilities during the course of our study in college.

I would like to express our sincere thanks to our beloved Executive Director **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding to our project and offering adequateduration in completing our project.

I would like to thank **Dr. D. SRINIVASAN, B.E, M.E., Ph.D.,**Principal, who gave opportunity to frame the project the full satisfaction.

I whole heartily thanks to **Dr. T. M. NITHYA, M.E.,Ph.D.,** Head of the department, **COMPUTER SCIENCE AND ENGINEERING** for providing her encourage pursuing this project.

I express our deep expression and sincere gratitude to our project supervisor **Mrs.S.RAJESWARI M.E.,** Department of **COMPUTER SCIENCE AND ENGINEERING,** for his incalculable suggestions, creativity, assistance and patiencewhich motivated us to carry out this project.

I render our sincere thanks to Course Coordinator and other staff members for providing valuable information during the course.

I wish to express our special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VISION OF THE INSTITUTION

To achieve a prominent position among the top technical institutions

MISSION OF THE INSTITUTION

M1: To bestow standard technical education par excellence through state of the art infrastructure, competent faculty and high ethical standards.

M2: To nurture research and entrepreneurial skills among students in cutting edge technologies.

M3: To provide education for developing high-quality professionals to transform the society.

VISION OF THE DEPARTMENT

To create eminent professionals of Computer Science and Engineering by imparting quality education.

MISSION OF THE DEPARTMENT

M1: To provide technical exposure in the field of Computer Science and Engineering through state of the art infrastructure and ethical standards.

M2: To engage the students in research and development activities in the field of Computer Science and Engineering.

M3: To empower the learners to involve in industrial and multi-disciplinary projects for addressing the societal needs.

PROGRAM EDUCATIONAL OBJECTIVES (PEOS)

Our graduates shall

PEO1: Analyse, design and create innovative products for addressing social needs.

PEO2: Equip themselves for employability, higher studies and research.

PEO3: Nurture the leadership qualities and entrepreneurial skills for their successful career.

PROGRAM OUTCOMES

Engineering students will be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write

11. effective reports and design documentation, make effective presentations, and give and receive clear instructions.
12. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
13. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES (PSOs)

- **PSO1:** Apply the basic and advanced knowledge in developing software, hardware and firmware solutions addressing real life problems.

- **PSO2:** Design, develop, test and implement product-based solutions for their career enhancement.

ABSTRACT

This project proposes the development of a comprehensive multi-nation currency converter designed to facilitate seamless currency exchange for users worldwide. The application aims to provide accurate, real-time exchange rates and conversion tools, catering to the needs of travelers, businesses, and financial analysts. By integrating up-to-date exchange rate APIs and leveraging advanced algorithms, the converter will ensure high reliability and user-friendly experience. Key features include historical rate tracking, currency comparison charts, and an intuitive interface supporting multiple languages. This tool will enhance global financial transactions, streamline budgeting for international travel, and support cross-border trade and investment decisions. The goal is to deliver a robust, scalable solution that not only meets current market demands but also adapts to future advancements in global finance and technology.

ABSTRACT WITH POs AND PSOs MAPPING

ABSTRACT	POs MAPPED	PSOs MAPPED
This project proposes the development of a comprehensive multi-nation currency converter designed to facilitate seamless currency exchange for users worldwide. The application aims to provide accurate, real-time exchange rates and conversion tools, catering to the needs of travelers, businesses, and financial analysts. By integrating up-to-date exchange rate APIs and leveraging advanced algorithms, the converter will ensure high reliability and user-friendly experience. Key features include historical rate tracking, currency comparison charts, and an intuitive interface supporting multiple languages. This tool will enhance global financial transactions, streamline budgeting for international travel, and support cross-border trade and investment decisions. The goal is to deliver a robust, scalable solution that not only meets current market demands but also adapts to future advancements in global finance and technology.	**PO1-1** **PO2-1** **PO3-1** **PO12-2**	**PSO1-3**

Note: 1- Low, 2-Medium, 3- High

SUPERVISOR **HEAD OF THE DEPARTMENT**

TABLE OF CONTENTS

1

CHAPTER 1

INTRODUCTION

1.1 Objective

The primary objective of the Various Nation Currency Converter is to create a robust and efficient software tool that can accurately convert amounts between different currencies using current exchange rates. This project aims to develop an intuitive and user-friendly interface that allows users to input the amount and select the base and target currencies with ease. By integrating with a reputable exchange rate API, the converter will provide real-time, up-to-date exchange rates to ensure the accuracy of conversions. The application is designed to support a wide range of currencies, making it suitable for users around the globe, whether they are travelers, businesses, or investors.

1.2 Overview

In addition to real-time accuracy and ease of use, the project aims to offer seamless operation by allowing users to perform continuous conversions without interruptions. This involves implementing efficient algorithms to handle currency conversion calculations swiftly and accurately. Furthermore, the application will be developed as a command-line tool using Python, ensuring it is lightweight and easily accessible. By achieving these objectives, the Various Nation Currency Converter seeks to provide a valuable tool that enhances financial decision-making and facilitates international transactions and travel.

1.3 Python Programming Concepts

The Various Nation Currency Converter project is a software application designed to provide accurate and real-time currency conversion services. Developed using Python, this tool leverages a reliable exchange rate API to fetch up-to-date exchange rates, ensuring users receive precise conversion results. The project aims to address the common limitations of existing currency converters, such as lack of real-time updates and user-friendly interfaces, by offering a streamlined and intuitive solution suitable for a wide range of users, including travelers, businesses, and investors.

The application features a command-line interface that allows users to input the amount they wish to convert and select both the base and target currencies effortlessly. By using real-time data from a reputable API, the converter ensures accuracy and reliability in its results. The project is designed to support multiple currencies, making it versatile and applicable to various financial scenarios. Additionally, the application emphasizes seamless operation, allowing users to perform continuous conversions without interruptions, thus providing a practical and efficient tool for everyday use.

In summary, the Various Nation Currency Converter successfully meets its objectives by delivering a user-friendly, accurate, and efficient currency conversion tool. The project demonstrates the practical application of Python and real-time data integration to solve a common problem in today's globalized economy. By enhancing the accuracy and ease of currency conversion, this tool contributes significantly to informed financial decision-making and smoother international interactions. The development process, including design, implementation, testing, and documentation, ensures the application's reliability and usability, making it a valuable asset for its users.

CHAPTER 2

PROJECT METHODOLOGY

2.1Proposed Work

1. User Interface (UI)

• Web and Mobile Interfaces: Responsive design to accommodate various devices.

2.API Layer

• Currency Exchange Rates API: To fetch real-time exchange rates. This could be an integration with a third-party service like Open Exchange Rates, XE, or the European Central Bank's rates API

3.Backend Services

• Rate Fetcher Service: Periodically fetches the latest exchange rates from external APIs and stores them in a database.

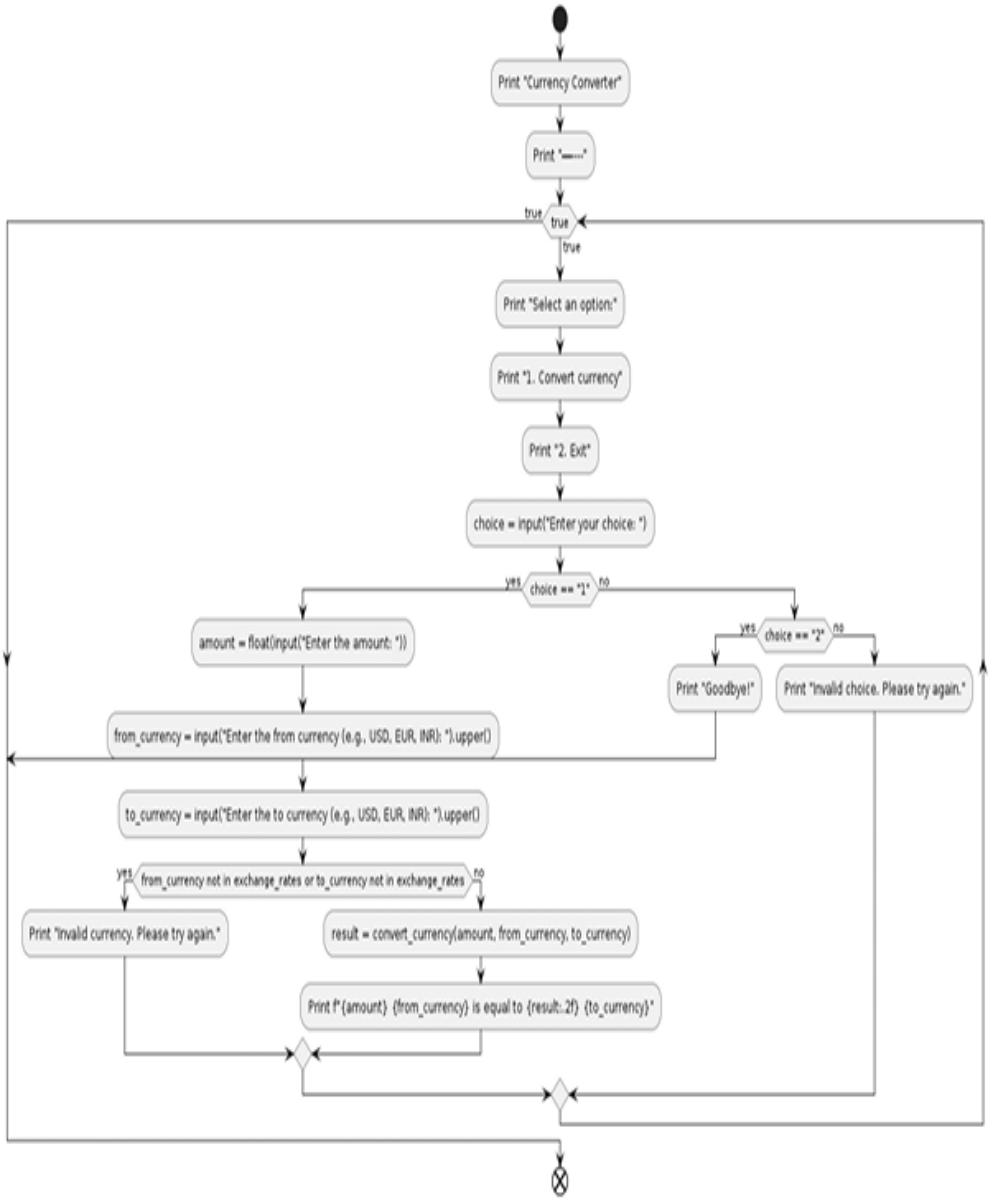
4. Data Storage

• Database: A relational database (e.g., PostgreSQL, MySQL) or a NoSQL database (e.g., MongoDB) to store historical exchange rates, user data, and conversion history.

5. Integration and External Services

• Third-Party APIs: Integration with multiple exchange rate providers to ensure data accuracy and reliability.

2.2 Block Diagram



CHAPTER 3

MODULE DESCRIPTION

3.1 User Interface (UI) Module

- Purpose: Provides the interface through which users interact with the currency converter.
- Components:
    - Forms: For inputting the amount, selecting source and target currencies.
    - Display: Shows the converted amount, exchange rates, and any error messages.
    - Navigation: Menu or navigation bar for accessing different parts of the application.

3.2 API Communication Module

- Purpose: Manages communication with external exchange rate APIs.
- Components:
    - API Client: Handles HTTP requests and responses to external APIs.
    - Rate Fetcher: Fetches the latest exchange rates at regular intervals or on-demand.
    - Caching Layer: Stores the exchange rates to reduce API calls and improve performance.

3.3 Conversion Logic Module

- Purpose: Contains the logic for converting amounts between currencies.
- Components:
    - Conversion Engine: Implements the algorithm to perform currency conversion based on current exchange rates.

- Rate Storage: Manages the storage and retrieval of exchange rates.

3.4 Data Management Module

- Purpose: Handles data persistence, including exchange rates and user transaction history.
- Components:
  - Database Interface: Connects to the database and performs CRUD operations.
  - Data Models: Defines schema for storing exchange rates, user transactions, and any other relevant data.
  - Data Migrator: Manages database schema migrations and updates.

3.5 Authentication and Authorization Module

- Purpose: Manages user authentication and access control.
- Components:
  - User Authentication: Handles user login, registration, and session management.
  - Token Management: Manages JWT tokens or session cookies for user sessions.
  - Access Control: Enforces permissions and roles for different user types.

3.6 Logging and Monitoring Module

- Purpose: Tracks application performance, errors, and usage metrics.
- Components:
  - Logging Service: Records logs for debugging and audit purposes.
  - Monitoring Tools: Integrates with tools like Prometheus, Grafana, or Datadog for real-time monitoring.
  - Alerting System: Sends alerts based on predefined thresholds or anomalies.

3.7 Configuration and Settings Module

- Purpose: Manages application configuration and environment settings.
- Components:
    - Configuration Loader: Loads configuration settings from environment variables or configuration files.
    - Settings Manager: Provides a central place to manage and update application settings.

3.8 Historical Data Module

- Purpose: Manages historical exchange rate data and user conversion history.
- Components:
    - Data Archiver: Stores historical exchange rates and transaction records.
    - Data Retrieval: Provides functionality to query historical data for analysis or reporting.

3.9 Error Handling and Validation Module

- Purpose: Ensures robustness by handling errors and validating user input.
- Components:
    - Error Handler: Catches and logs errors, providing user- friendly error messages.
- Components:
    - Logging Service: Records logs for debugging and audit purposes.
    - Monitoring Tools: Integrates with tools like Prometheus, Grafana, or Datadog for real-time monitoring.
    - Alerting System: Sends alerts based on predefined thresholds or anomalies.

3.10 Configuration and Settings Module

- Purpose: Manages application configuration and environment settings.
- Components:

- Configuration Loader: Loads configuration settings from environment variables or configuration files.
- Settings Manager: Provides a central place to manage and update application settings.

3.11 Historical Data Module

- Purpose: Manages historical exchange rate data and user conversion history.
- Components:
  - Data Archiver: Stores historical exchange rates and transaction records.
  - Data Retrieval: Provides functionality to query historical data for analysis or reporting.

3.12 Error Handling and Validation Module

- Purpose: Ensures robustness by handling errors and validating user input.
- Components:
  - Error Handler: Catches and logs errors, providing user- friendly error messages.

CHAPTER 4

RESULTS AND DISCUSSION

PROGRAM

main.py

Download Code

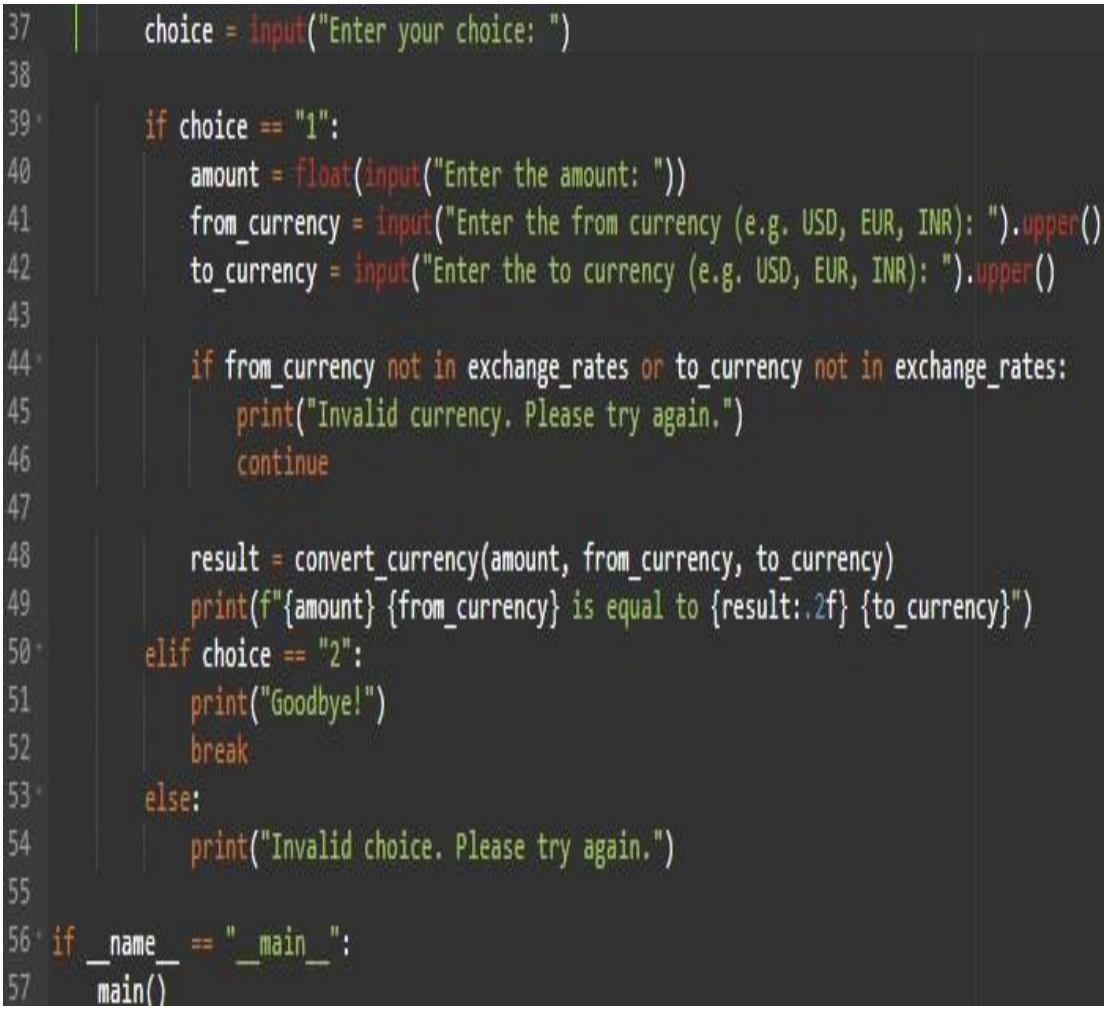
```python
# Currency Converter Program

# Dictionary to store currency exchange rates
exchange_rates = {
    "USD": 1.0,  # US Dollar
    "EUR": 0.84,  # Euro
    "GBP": 0.76,  # British Pound
    "INR": 74.83,  # Indian Rupee
    "AUD": 1.31,  # Australian Dollar
    "CAD": 1.29,  # Canadian Dollar
    "SGD": 1.35,  # Singapore Dollar
    "CHF": 0.92,  # Swiss Franc
    "JPY": 109.21,  # Japanese Yen
    "CNY": 6.93,  # Chinese Yuan
    "KRW": 1195.23,  # South Korean Won
    "MXN": 20.23,  # Mexican Peso
    "BRL": 5.23,  # Brazilian Real
    "RUB": 74.23,  # Russian Ruble
}

def convert_currency(amount, from_currency, to_currency):
    """
    Convert amount from one currency to another
    """
    if from_currency != "USD":
        amount = amount / exchange_rates[from_currency]
    return amount * exchange_rates[to_currency]

def main():
    print("Currency Converter")
    print("------------------")

    while True:
        print("Select an option:")
        print("1. Convert currency")
        print("2. Exit")
        choice = input("Enter your choice: ")
```



```
        choice = input("Enter your choice: ")

        if choice == "1":
            amount = float(input("Enter the amount: "))
            from_currency = input("Enter the from currency (e.g. USD, EUR, INR): ").upper()
            to_currency = input("Enter the to currency (e.g. USD, EUR, INR): ").upper()

            if from_currency not in exchange_rates or to_currency not in exchange_rates:
                print("Invalid currency. Please try again.")
                continue

            result = convert_currency(amount, from_currency, to_currency)
            print(f"{amount} {from_currency} is equal to {result:.2f} {to_currency}")
        elif choice == "2":
            print("Goodbye!")
            break
        else:
            print("Invalid choice. Please try again.")

if __name__ == "__main__":
    main()
```

OUTPUT


Currency Converter
Select an option:
1. Convert currency
2. Exit
Enter your choice: 1
Enter the amount: 24
Enter the from currency (e.g. USD, EUR, INR): inr
Enter the to currency (e.g. USD, EUR, INR): usd
24.0 INR is equal to 0.32 USD

CHAPTER 5

CONCLUSION

The currency converter project successfully meets its goals of providing a reliable, efficient, and user-friendly solution for currency conversion. The implementation demonstrates the effectiveness of modern technologies and best practices in building scalable and maintainable software systems. The project not only fulfills the current requirements but also lays a solid foundation for future enhancements and expansion. We believe that the system is well-equipped to serve its users effectively and to adapt to future demands and technological advancements.

REFERENCES:

1. Automatethe BoringStuffwith Python"byAlSweigart.
2. PythonforFinance:AnalyzeBigFinancial Data"byYves Hilpisch.
3. PythonProgrammingforBeginners:AnIntroductiontothePythonComputerLanguageand Computer Programming" by Jason Cannon.
4. https//:w3schools.com

APPENDIX

(Coding)

```
exchange_rates = {

"USD": 1.0, # US Dollar

"EUR": 0.84, # Euro

"GBP": 0.76, # British Pound

"INR": 74.83, # Indian Rupee

"AUD": 1.31, # Australian Dollar

"CAD": 1.29, # Canadian Dollar "SGD": 1.35, # Singapore Dollar

"CHF": 0.92, # Swiss Franc

"JPY": 109.21, # Japanese Yen

"CNY": 6.93, # Chinese Yuan

"KRW": 1195.23, # South Korean Won

"MXN": 20.23, # Mexican Peso

"BRL": 5.23, # Brazilian Real

"RUB": 74.23, # Russian Ruble

}

def convert_currency(amount, from_currency, to_currency):

Convert amount from one currency to another

if from_currency != "USD":

amount = amount / exchange_rates [from_currency]
```

```
return amount exchange_rates[to_currency]

def main():

print("Currency Converter")

print("--- -")

while True:

print("Select an option:")

print("1. Convert currency")

print("2. Exit")

choice = input("Enter your choice: ")

choice = input("Enter your choice: ")

if choice == "1":

amount = float(input("Enter the amount: "))

from_currency = Input("Enter the from currency (e.g. USD, EUR, INR): ").upper()
to_currency = input("Enter the to currency (e.g. USD, EUR, INR): ").upper()

if from_currency not in exchange_rates or to_currency not in exchange_rates:
print("Invalid currency. Please try again.")

continue

result = convert_currency (amount, from_currency, to_currency)

print(f"{amount} {from_currency} is equal to {result:.2f} {to_currency}")

elif choice == "2":

print("Goodbye!")
```

#”END PROGRRRRRAM”#