**PRACTICAL NO 1**

**AIM:** Define a simple services like Converting Rs into Dollars and call it from different platforms like JAVA and.NET

**SOLUTION:**

**Currency\_converter.py**

from flask import Flask, request, jsonify

app = Flask(\_\_name\_\_)

conversion\_rate = 0.012

@app.route('/convert', methods=['GET'])

def convert\_currency():

    inr = request.args.get('inr')

    if not inr:

        return jsonify({"error": "INR amount missing"}), 400

    try:

        inr\_value = float(inr)

        return jsonify({"INR": inr\_value, "USD": inr\_value \* conversion\_rate})

    except ValueError:

        return jsonify({"error": "Invalid INR amount"}), 400

if \_\_name\_\_ == '\_\_main\_\_':

    app.run(debug=True)

**CurrencyConverterClient.java**

import java.io.BufferedReader;

import java.io.InputStreamReader;

import java.net.HttpURLConnection;

import java.net.URL;

import java.util.Scanner;

import java.io.\*;

import java.net.\*;

public class CurrencyConverterClient {

    public static void main(String[] args) {

        try (Scanner scanner = new Scanner(System.in)) {

            System.out.print("Enter amount in INR: ");

            String inrAmount = scanner.nextLine();

            URL url = new URL("http://127.0.0.1:5000/convert?inr=" + inrAmount);

            HttpURLConnection conn = (HttpURLConnection) url.openConnection();

            conn.setRequestMethod("GET");

            conn.setRequestProperty("Accept", "application/json");

            if (conn.getResponseCode() != 200) throw new RuntimeException("Failed : HTTP error code : " + conn.getResponseCode());

            try (BufferedReader br = new BufferedReader(new InputStreamReader(conn.getInputStream()))) {

                br.lines().forEach(System.out::println);

            }

            conn.disconnect();

        } catch (Exception e) {

            e.printStackTrace();

        }

    }

}

**OUTPUT:**



