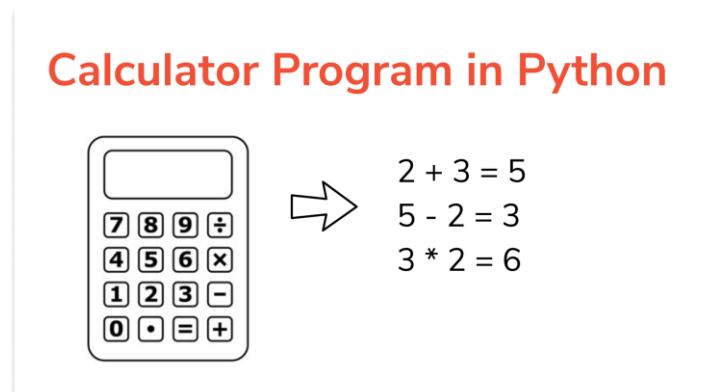


I. Pengenalan Simple Calculator Python Socket Programming



Simple Calculator menggunakan Socket Programming adalah sebuah aplikasi yang memungkinkan pengguna untuk mengirimkan ekspresi matematika ke server melalui koneksi socket, kemudian server akan menghitung ekspresi tersebut dan mengirimkan hasil perhitungan kembali ke pengguna.

Melalui implementasi Socket Programming, aplikasi Simple Calculator memungkinkan pengguna untuk menggunakan kalkulator yang berada di server melalui jaringan. Ini memungkinkan pengguna untuk menggunakan kalkulator dari jarak jauh dan memanfaatkan sumber daya server untuk melakukan perhitungan matematika kompleks.

II. Cara Kerja Simple Calculator Python Socket Programming

Proses kerja aplikasi Simple Calculator menggunakan Socket Programming sebagai berikut:

- **Server Setup:** Server diatur dengan membuat socket server menggunakan modul socket dari Python. Socket server ini akan melakukan binding ke alamat IP dan port tertentu, kemudian akan mulai mendengarkan koneksi dari klien.
- **Koneksi Klien:** Ketika seorang klien ingin menggunakan kalkulator, dia membuat koneksi dengan server menggunakan socket client. Klien dan server akan berkomunikasi melalui koneksi socket yang terbentuk.
- **Mengirim Ekspresi:** Setelah koneksi terbentuk, pengguna (klien) akan memasukkan ekspresi matematika yang ingin dihitung ke dalam aplikasi kalkulator. Ekspresi ini akan dikirimkan melalui koneksi socket kepada server.
- **Menerima dan Menghitung Ekspresi:** Server menerima ekspresi matematika dari klien melalui koneksi socket. Setelah menerima ekspresi, server akan melakukan perhitungan matematika terhadap ekspresi tersebut menggunakan fungsi perhitungan yang telah ditentukan.

- Mengirim Hasil Perhitungan: Setelah melakukan perhitungan, server akan mengirimkan hasil perhitungan kembali ke klien melalui koneksi socket yang terhubung. Klien akan menerima hasil perhitungan ini.
- Terus Menerima dan Mengirim Ekspresi: Aplikasi akan terus menerima ekspresi dari klien dan mengirimkan hasil perhitungan kembali. Pengguna dapat mengulangi langkah ini dengan memasukkan ekspresi baru.
- Disconnect: Jika pengguna ingin keluar dari aplikasi, pengguna dapat mengirimkan pesan khusus atau memberikan perintah tertentu untuk memutus koneksi dengan server. Server kemudian akan mengidentifikasi permintaan ini dan menutup koneksi dengan klien.

III. Pembuatan Simple Calculator Python Socket Programming

Berikut adalah contoh sederhana program Pembuatan Simple Calculator Python Socket Programming menggunakan Python:

Server:

```
import socket

# Inisialisasi socket TCP
sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

# Bind socket ke alamat dan port tertentu
server_address = ('localhost', 5000)
sock.bind(server_address)

# Menunggu koneksi masuk
sock.listen(1)
print('Menunggu koneksi dari klien...')

while True:
    # Menerima koneksi dari klien
    client_socket, client_address = sock.accept()
    print('Terhubung dengan klien:', client_address)

    # Menerima pesan dari klien
    data = client_socket.recv(1024).decode()
    print('Pesan diterima dari klien:', data)

    # Memisahkan operator dan operand
    operator, operand1, operand2 = data.split()

    # Melakukan perhitungan matematika
    result = None
    if operator == '+':
```

```

        result = int(operand1) + int(operand2)
    elif operator == '-':
        result = int(operand1) - int(operand2)
    elif operator == '*':
        result = int(operand1) * int(operand2)
    elif operator == '/':
        result = int(operand1) / int(operand2)

    # Mengirim pesan balasan berisi hasil perhitungan ke klien
    response = str(result)
    client_socket.send(response.encode())

    # Menutup koneksi dengan klien
    client_socket.close()

```

Client :

```

import socket

# Inisialisasi socket TCP
sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

# Menghubungkan ke server
server_address = ('localhost', 5000)
sock.connect(server_address)

# Meminta input dari pengguna
operator = input('Masukkan operator (+, -, *, /): ')
operand1 = input('Masukkan operand pertama: ')
operand2 = input('Masukkan operand kedua: ')

# Menggabungkan operator dan operand menjadi satu string
message = operator + ' ' + operand1 + ' ' + operand2

# Mengirim pesan ke server
sock.send(message.encode())

# Menerima pesan balasan dari server
response = sock.recv(1024).decode()
print('Hasil perhitungan:', response)

# Menutup koneksi dengan server
sock.close()

```

IV. Tugas dan Latihan

1. Membuat laporan percobaan praktikum dan beri Analisa Hasil Percobaan tadi yang sudah dibuat Simple Calculator Python Socket Programming
2. Ubah kodingan agar client dapat mengirimkan operasi matematika dalam 1 kali input contoh “1 + 1” result = 2, dan client dapat mengirimkan operasi matematika secara terus menerus.