**Distributed Systems – CS – 6580**

Project is made in “**PYTHON**”

**Flow of Project:**

Main Server

Client

Client

Client

Logical Server

Division Server

Multiply Server

Add Server

Subtraction Server

1. Request

for addition

2. Provide

Address of Add Server

3. Provide values 4. answer

**Steps:**

1. Client send request to main server for calculation.
2. Main Server provide port number of the sub server according to client’s requirement.
3. Client send request to given port number which is provided by Main Server.
4. Sub Server calculate the equation and provide the answer at client side.

**Validations in project**

* User cannot enter invalid operator or a value. This validation is done by using regular expression.
* If division operation then second value not be 0.
* If user wants to perform not operation then only ask for the one value.

**MainServer.py**

import socket

import os

import subprocess

import re

s = socket.socket()

host = socket.gethostname()

port = 5050

s.bind((host, port))

s.listen(5)

while True:

c, addr = s.accept()

print('Got connection from', addr)

value\_op = c.recv(1024)

re\_data = value\_op.decode('utf-8')

print(re\_data)

if re\_data == '+' :

add\_port = 5051

filename = "AddServer.py"

elif re\_data == '-' :

add\_port = 5052

filename = "SubtractionServer.py"

elif re\_data == '\*' :

add\_port = 5053

filename = "MulServer.py"

elif re\_data == '/' :

add\_port = 5054

filename = "DivisionServer.py"

elif re\_data == "and" or re\_data == "or" or re\_data == "not":

add\_port = 5055

filename = "LogicalServer.py"

elif re\_data == 'exit' :

filename = ""

p.kill()

else :

add\_port = "Something goes wrong...!!:("

if filename != "" :

p = subprocess.Popen(["python", filename])

add\_port = str(add\_port)

c.sendall(add\_port.encode('utf-8'))

c.close()

**Client.py**

import socket

import sys

import re

while True:

operator = input("Enter Operator: ")

if not re.match("^[+,\*,/,-]$", operator) and operator != "and" and operator != "or" and operator != "not":

print("Enter Valid Operator.")

continue

else:

break

while True:

value\_1 = input("Enter Value1: ")

if not re.match("^\d+$", value\_1):

print("Enter Valid Number.")

continue

else:

break

if operator != "not" :

while True:

value\_2 = input("Enter Value2: ")

if not re.match("^\d+$", value\_2):

print("Enter Valid Number.")

continue

elif operator == '/' and value\_2 == "0" :

print("Enter valid value2 for division")

else:

break

s = socket.socket()

s\_request = socket.socket()

host = socket.gethostname()

port = 5050

s.connect((host, port))

s.sendall(operator.encode('utf-8'))

data = s.recv(1024)

add\_port\_num = data.decode('utf-8')

print("Port number for given operator : " + add\_port\_num)

add\_port\_num = int(add\_port\_num)

s.close()

s\_operation = socket.socket()

s\_operation\_request = socket.socket()

s\_operation.connect((host, add\_port\_num))

if operator == "and" or operator == "or":

values = str(value\_1 + "|" + value\_2 + "|" + operator)

elif operator == "not":

values = str(value\_1 + "|" + operator)

else :

values = str(value\_1 + "|" + value\_2)

s\_operation.sendall(values.encode('utf-8'))

data = s\_operation.recv(1024)

add\_port\_num = data.decode('utf-8')

print(add\_port\_num)

s\_operation.close()

s1 = socket.socket()

s1.connect((host, port))

exit\_data="exit"

s1.sendall(exit\_data.encode('utf-8'))

s1.close()

**AddServer.py**

import socket

s\_add = socket.socket()

host = socket.gethostname()

port\_add = 5051

s\_add.bind((host, port\_add))

s\_add.listen(5)

while True:

c, addr = s\_add.accept()

data\_client = c.recv(1024)

data\_client = data\_client.decode('utf-8')

split\_str = data\_client.split("|")

value1 = split\_str[0]

value2 = split\_str[1]

ans = 'The addition of given numbers is : ' + str(int(value1)+int(value2))

c.sendall(ans.encode('utf-8'))

c.close()

**SubtractionServer.py**

import socket

s\_add = socket.socket()

host = socket.gethostname()

port\_add = 5052

s\_add.bind((host, port\_add))

s\_add.listen(5)

while True:

c, addr = s\_add.accept()

data\_client = c.recv(1024)

data\_client = data\_client.decode('utf-8')

split\_str = data\_client.split("|")

value1 = split\_str[0]

value2 = split\_str[1]

ans = 'The subtraction of given numbers is : ' + str(int(value1)- int(value2))

c.sendall(ans.encode('utf-8'))

c.close()

**MulServer.py**

import socket

s\_add = socket.socket()

host = socket.gethostname()

port\_add = 5053

s\_add.bind((host, port\_add))

s\_add.listen(5)

while True:

c, addr = s\_add.accept()

data\_client = c.recv(1024)

data\_client = data\_client.decode('utf-8')

split\_str = data\_client.split("|")

value1 = split\_str[0]

value2 = split\_str[1]

ans = 'The multiplication of given numbers is : ' + str(int(value1)\*int(value2))

c.sendall(ans.encode('utf-8'))

c.close()

**DivisionServer.py**

import socket

s\_add = socket.socket()

host = socket.gethostname()

port\_add = 5054

s\_add.bind((host, port\_add))

s\_add.listen(5)

while True:

c, addr = s\_add.accept()

data\_client = c.recv(1024)

data\_client = data\_client.decode('utf-8')

split\_str = data\_client.split("|")

value1 = split\_str[0]

value2 = split\_str[1]

ans = 'The division of given numbers is : ' + str(int(value1)/int(value2))

c.sendall(ans.encode('utf-8'))

c.close()

**LogicalServer.py**

import socket

s\_add = socket.socket()

host = socket.gethostname()

port\_add = 5055

s\_add.bind((host, port\_add))

s\_add.listen(5)

while True:

c, addr = s\_add.accept()

data\_client = c.recv(1024)

data\_client = data\_client.decode('utf-8')

split\_str = data\_client.split("|")

value1 = split\_str[0]

if len(split\_str) == 2 :

operator = split\_str[1]

else :

value2 = split\_str[1]

operator = split\_str[2]

if operator == "and" :

answer = int(value1) & int(value2)

if operator == "or" :

answer = int(value1) | int(value2)

if operator == "not" :

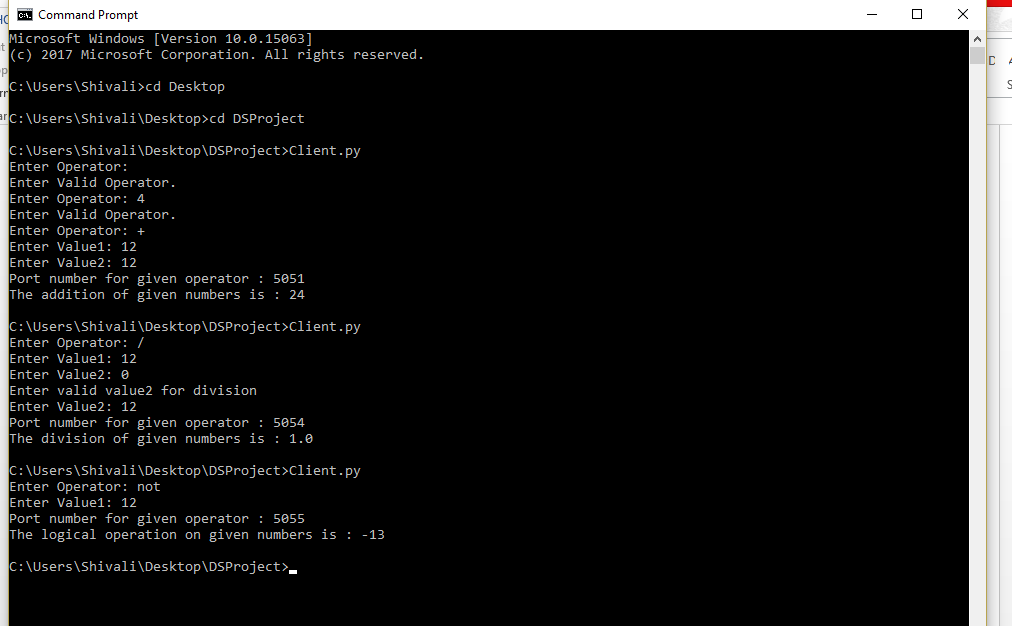
answer = ~int(value1)

ans = 'The logical operation on given numbers is : ' + str(answer)

c.sendall(ans.encode('utf-8'))

c.close()

**OUTPUTS:**

****