

NAME:M.KALAIYARASI

ROLL NO:231901503

DATE:18/10/24

EXP NO:11B

ARITHMETIC OPERATION USING RPC

AIM:

To develop a simple calculator using XMLRPC.

ALGORITHM:

Server.py

1. Import XMLRPCServer package
2. Define functions for addition, subtraction, multiplication, division and modulus
3. Initialize simple XMLRPCServer with IP address (or localhost) and port number
4. Register the functions add, sub, mul, div and mod with the server
5. Handle the request
6. Close the connection.

Client.py

1. Import XMLRPC Client package
2. Define functions for addition, subtraction, multiplication, division and modulus
3. Initialize simple XMLRPC Client with Server IP address (or localhost) and port number
4. Get two numbers a and b for arithmetic operations
5. Call add() function and print the result
6. Call sub() function and print the result

7. Call mul() function and print the result
8. Call div() function and print the result
9. Call mod() function and print the result
10. Close the connection

Sample Code for Arithmetic operations using RPC XML

RPC PROGRAM- SERVER SIDE:

```
from xmlrpc.server import SimpleXMLRPCServer

# Define a function def is_even(n): return n % 2 ==
0 def add(a,b): return a+b def sub(a,b): return a-b
def factorial(n): factorial=1 for i in range(1,n+1):
factorial = factorial*i return factorial def
multiply(x, y): return x * y def divide(x, y): return
x // y # Create server server =
SimpleXMLRPCServer(("localhost", 8000))
print("Listening on port 8000...") # Register a
function under a different name
server.register_function(is_even, "is_even")
server.register_function(add, "add")
server.register_function(sub, "sub")
server.register_function(factorial, "factorial")
#server.register_function(factorial, "factorial")
server.register_function(multiply, 'multiply')
server.register_function(divide, 'divide') # Run the
```

server's main loop server.serve_forever()

```
XML RPC PROGRAM- SERVER SIDE.py × XML RPC PROGRAM- CLIENT SIDE.py
1  from xmlrpc.server import SimpleXMLRPCServer
2  # Define a function
3  def is_even(n): 1 usage
4      return n % 2 == 0
5  def add(a,b): 1 usage
6      return a+b
7  def sub(a,b): 1 usage
8      return a-b
9  def factorial(n): 1 usage
10     factorial=1
11     for i in range(1,n+1):
12         factorial = factorial*i
13     return factorial
14 def multiply(x, y): 1 usage
15     return x * y
16 def divide(x, y): 1 usage
17     return x // y
18 # Create server
19 server = SimpleXMLRPCServer(("localhost", 8000))
20 print("Listening on port 8000...")
21 # Register a function under a different name
22 server.register_function(is_even, name="is_even")
23 server.register_function(add, name="add")
24 server.register_function(sub, name="sub")
25 server.register_function(factorial, name="factorial")
26 #server.register_function(factorial,"factorial")
27 server.register_function(multiply, name='multiply')
28 server.register_function(divide, name='divide')
29 # Run the server's main loop
30 server.serve_forever()
31
```

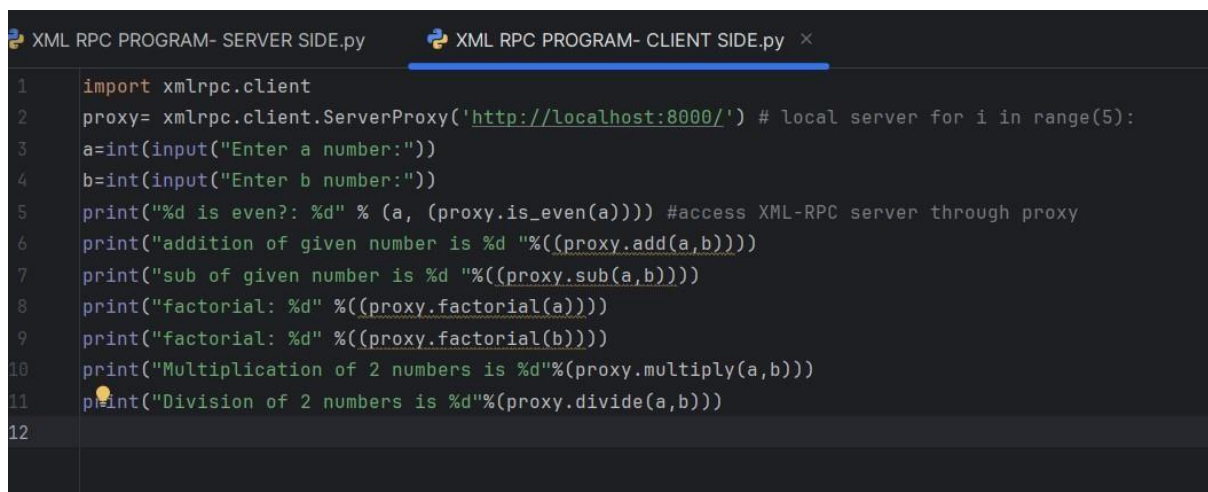
XML RPC PROGRAM- CLIENT SIDE:

```
import xmlrpc.client
```

```
proxy= xmlrpc.client.ServerProxy('http://localhost:8000/') # local server for i in range(5):
```

```
a=int(input("Enter a number:"))
```

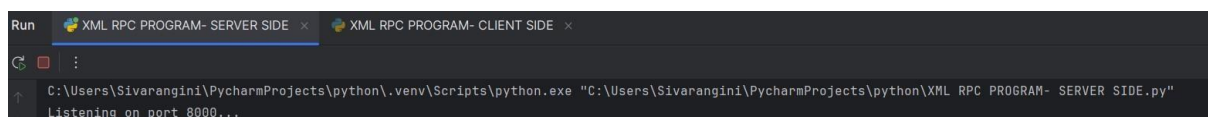
```
b=int(input("Enter b number:")) print("%d is even?: %d" % (a, (proxy.is_even(a)))) #access XML-RPC server through proxy print("addition of given number is %d "%((proxy.add(a,b)))) print("sub of given number is %d "%((proxy.sub(a,b)))) print("factorial: %d" %((proxy.factorial(a)))) print("factorial: %d" %((proxy.factorial(b)))) print("Multiplication of 2 numbers is %d" %(proxy.multiply(a,b))) print("Division of 2 numbers is %d" %(proxy.divide(a,b)))
```



The screenshot shows a code editor with two tabs: "XML RPC PROGRAM- SERVER SIDE.py" and "XML RPC PROGRAM- CLIENT SIDE.py". The client-side code is visible, showing the same logic as the text above, including importing xmlrpc.client, creating a ServerProxy, and using it to call various methods like is_even, add, sub, factorial, multiply, and divide. The code is numbered from 1 to 12.

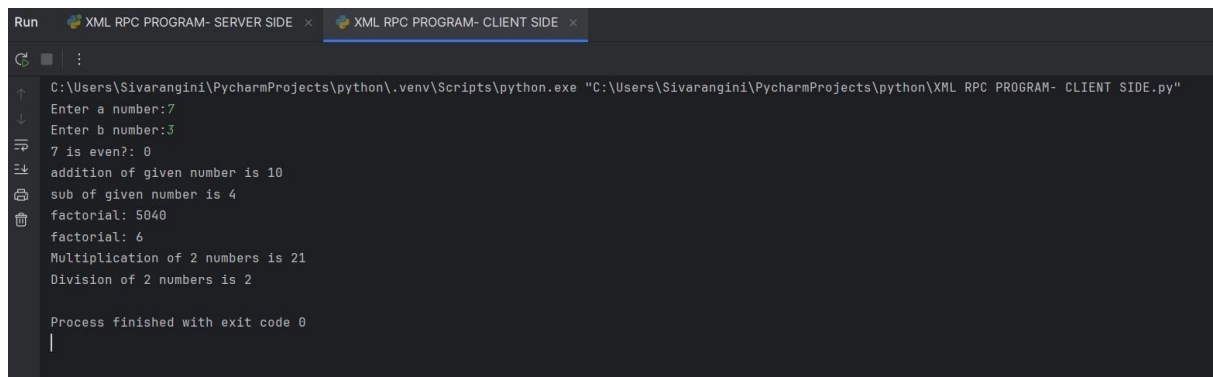
OUTPUT:

For server:



The screenshot shows a terminal window with the command prompt. The output indicates that the server is listening on port 8000. The command executed is: C:\Users\Sivarangini\PycharmProjects\python\venv\Scripts\python.exe "C:\Users\Sivarangini\PycharmProjects\python\XML RPC PROGRAM- SERVER SIDE.py".

For client:



```
Run XML RPC PROGRAM- SERVER SIDE XML RPC PROGRAM- CLIENT SIDE
C:\Users\Sivarangini\PycharmProjects\python\.venv\Scripts\python.exe "C:\Users\Sivarangini\PycharmProjects\python\XML RPC PROGRAM- CLIENT SIDE.py"
Enter a number:7
Enter b number:3
7 is even?: 0
addition of given number is 10
sub of given number is 4
factorial: 5040
factorial: 6
Multiplication of 2 numbers is 21
Division of 2 numbers is 2

Process finished with exit code 0
```

RESULT:

A simple calculator was designed using XMLRPC.