

**Ex. No:11C**

**Date:22/10/2024**

**Roll No:231901030**

### **REMOTE PROCEDURE CALL FOR LIST OPERATIONS- XMLRPC**

**Aim:**

To Implement an XML RPC code for the following functions, a. No of items in a list

**Algorithm:**

**Server side:**

1. Import ``SimpleXMLRPCServer``.
2. Define list functions (``length``, ``maximum``, ``minimum``, ``to_set``, ``concatenate``).
3. Create server on ``localhost`` with port ``8000``.
4. Print "Listening on port 8000...".
5. Register functions with the server.
6. Start the server with ``serve_forever()``.
7. Server listens and responds to client requests.

## Client side:

1. Import ``xmlrpc.client`` to interact with the XML-RPC server.
2. Create a ``ServerProxy`` object to connect to the XML-RPC server at ``http://localhost:8000/``.
3. Enter a loop to repeatedly prompt the user for input to start or stop operations.
4. If the user chooses to start (option 1), prompt the user to input elements for two separate lists (``a`` and ``b``), stopping when ``-1`` or ``-2`` is entered.
5. If the user chooses to stop (option 2), exit the loop.
6. Once the lists are gathered, print the contents of both lists (``a`` and ``b``).
7. Call the registered XML-RPC functions (``list_length``, ``list_maximum``, ``list_minimum``, ``list_to_set``, ``list_concate``) via the server proxy and print the results.

## Program

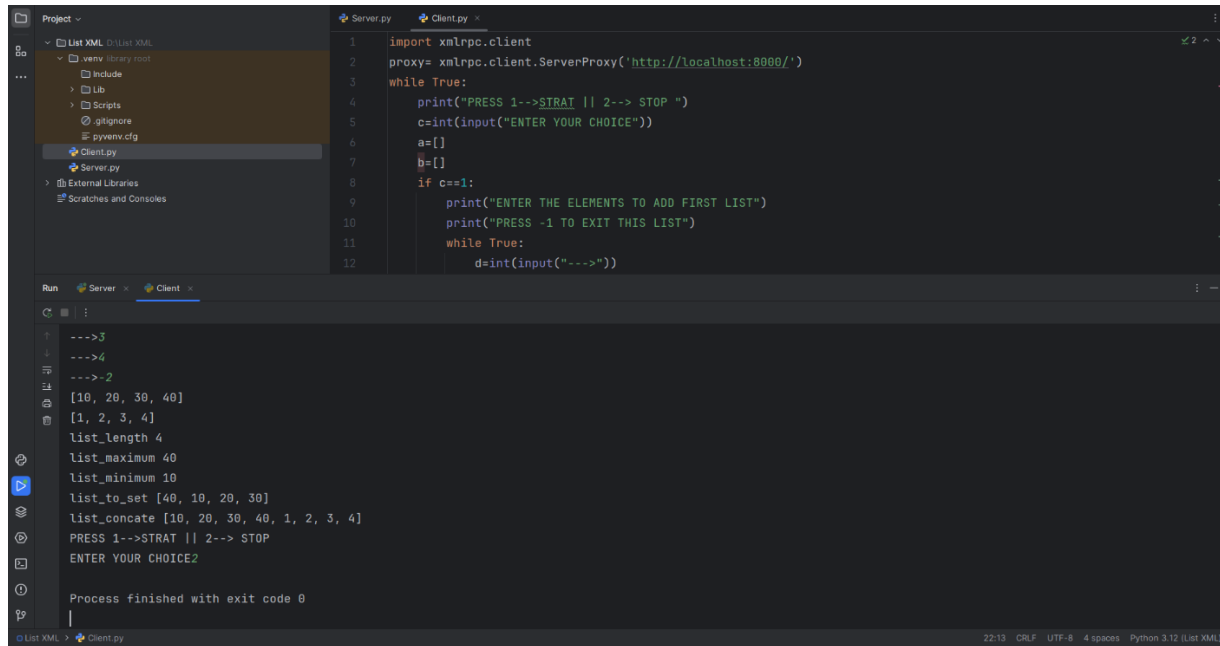
### Server Side:

```
from xmlrpc.server import SimpleXMLRPCServer
def list_length(a):
    return len(a)
def list_maximum(a):
    return max(a)
def list_minimum(a):
    return min(a)
def list_to_set(a):
    f=list(set(a))
    return f
def list_concate(a,b):
    return a+b
server = SimpleXMLRPCServer(("localhost", 8000)) print("Listening on port 8000...")
server.register_function(list_length,"list_length")
server.register_function(list_maximum, "list_maximum")
server.register_function(list_minimum, "list_minimum")
server.register_function(list_to_set, "list_to_set")
server.register_function(list_concate, "list_concate")
server.serve_forever()
```

### Client Side:

```
import xmlrpc.client
proxy= xmlrpc.client.ServerProxy('http://localhost:8000/') while True:
    print("PRESS 1-->STRAT || 2--> STOP ")
    c=int(input("ENTER YOUR CHOICE"))
    a=[]
    b=[]
    if c==1:
        print("ENTER THE ELEMENTS TO ADD FIRST LIST") print("PRESS -1 TO EXIT THIS LIST")
        while True:
            d=int(input("--->"))
            if d== -1:
                break
            a.append(d)
        print("ENTER THE ELEMENTS TO ADD SECOND LIST") print("PRESS -2 TO EXIT THIS LIST")
        while True:
            e=int(input("--->"))
            if e== -2:
                break
            b.append(e)
        if c==2:
            break
    print(a)
    print(b)
    print("list_length",proxy.list_length(a))
    print("list_maximum",proxy.list_maximum(a))
    print("list_minimum",proxy.list_minimum(a))
    print("list_to_set",proxy.list_to_set(a))
    print("list_concate",proxy.list_concate(a,b))
```

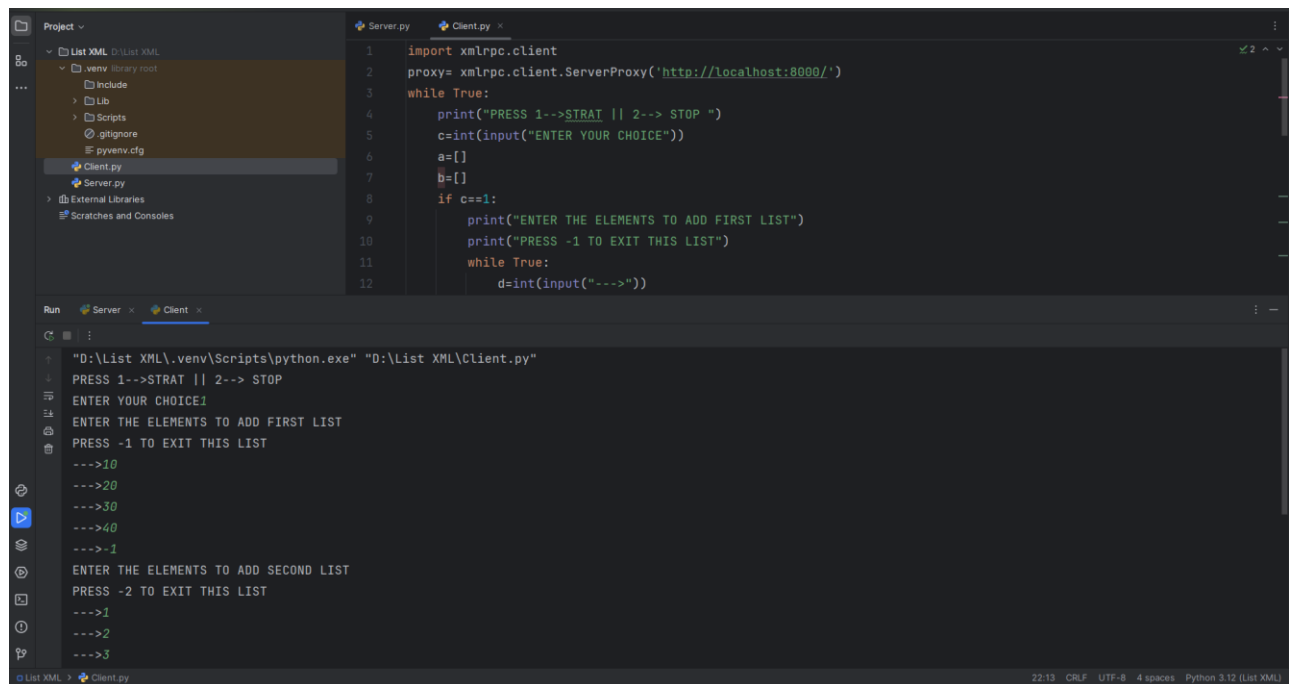
## Output:



```
1 import xmlrpc.client
2 proxy= xmlrpc.client.ServerProxy('http://localhost:8000/')
3 while True:
4     print("PRESS 1-->STRAT || 2--> STOP ")
5     c=int(input("ENTER YOUR CHOICE"))
6     a=[]
7     b=[]
8     if c==1:
9         print("ENTER THE ELEMENTS TO ADD FIRST LIST")
10        print("PRESS -1 TO EXIT THIS LIST")
11        while True:
12            d=int(input("---->"))
```

Run Server Client

--->3  
--->4  
--->-2  
[10, 20, 30, 40]  
[1, 2, 3, 4]  
list\_length 4  
list\_maximum 40  
list\_minimum 10  
list\_to\_set [40, 10, 20, 30]  
list\_concat [10, 20, 30, 40, 1, 2, 3, 4]  
PRESS 1-->STRAT || 2--> STOP  
ENTER YOUR CHOICE2  
Process finished with exit code 0



```
1 import xmlrpc.client
2 proxy= xmlrpc.client.ServerProxy('http://localhost:8000/')
3 while True:
4     print("PRESS 1-->STRAT || 2--> STOP ")
5     c=int(input("ENTER YOUR CHOICE"))
6     a=[]
7     b=[]
8     if c==1:
9         print("ENTER THE ELEMENTS TO ADD FIRST LIST")
10        print("PRESS -1 TO EXIT THIS LIST")
11        while True:
12            d=int(input("---->"))
```

Run Server Client

"D:\List XML\.venv\Scripts\python.exe" "D:\List XML\Client.py"  
PRESS 1-->STRAT || 2--> STOP  
ENTER YOUR CHOICE1  
ENTER THE ELEMENTS TO ADD FIRST LIST  
PRESS -1 TO EXIT THIS LIST  
--->10  
--->20  
--->30  
--->40  
--->-1  
ENTER THE ELEMENTS TO ADD SECOND LIST  
PRESS -2 TO EXIT THIS LIST  
--->1  
--->2  
--->3

The screenshot displays a code editor with a project structure on the left and a Python script in the center. The project structure shows a folder named 'List XML' containing a '.venv' directory and a 'Scripts' folder. The 'Scripts' folder contains 'pyvenv.cfg', 'Client.py', and 'Server.py'. The 'Server.py' file is open in the editor, showing the following code:

```
1 from xmlrpc.server import SimpleXMLRPCServer
2 def list_length(a): 1 usage
3     return len(a)
4 def list_maximum(a): 1 usage
5     return max(a)
6 def list_minimum(a): 1 usage
7     return min(a)
8 def list_to_set(a): 1 usage
9     f=list(set(a))
10    return f
11 def list_concat(a,b): 1 usage
12    return a+b
13 server = SimpleXMLRPCServer(("localhost", 8000))
14 print("Listening on port 8000...")
15 server.register_function(list_length, name="list_length")
16 server.register_function(list_maximum, name="list_maximum")
17 server.register_function(list_minimum, name="list_minimum")
18
19 server.register_function(list_to_set, name="list_to_set")
20 server.register_function(list_concat, name="list_concat")
21 ..
```

The console output shows the server running and listening on port 8000. It receives five POST requests from 127.0.0.1, all returning a 200 status code. The requests are as follows:

IP Address	Timestamp	Method	Protocol	Status
127.0.0.1	[07/Nov/2024 08:30:12]	POST	HTTP/1.1	200
127.0.0.1	[07/Nov/2024 08:30:14]	POST	HTTP/1.1	200
127.0.0.1	[07/Nov/2024 08:30:17]	POST	HTTP/1.1	200
127.0.0.1	[07/Nov/2024 08:30:19]	POST	HTTP/1.1	200
127.0.0.1	[07/Nov/2024 08:30:21]	POST	HTTP/1.1	200

## Result:

Thus, the list operations using Remote Procedure Call was executed.