

Client Calcolatrice 1.0 : calcoClient1.py (versione base)

22 lines (20 sloc) | 706 Bytes

```
1  import socket
2  import json
3
4  HOST="127.0.0.1"
5  PORT=65432
6
7  with socket.socket(socket.AF_INET,socket.SOCK_STREAM) as s:
8      s.connect((HOST,PORT))
9      while True:
10         primoNumero=input("Inserisci il primo numero. exit() per uscire")
11         if primoNumero=="exit()":
12             break
13         primoNumero=float(primoNumero)
14         operazione=input("Inserisci l'operazione (+,-,*,/,%)")
15         secondoNumero=float(input("Inserisci il secondo numero"))
16         messaggio={'primoNumero':primoNumero,
17                   'operazione':operazione,
18                   'secondoNumero':secondoNumero}
19         messaggio=json.dumps(messaggio) # Trasformiamo l'oggetto in una stringa
20         s.sendall(messaggio.encode("UTF-8"))
21         data=s.recv(1024)
22         print("Risultato: ",data.decode())
```

Server calcolatrice 1.0 : calcoServer1.py (versione base)

42 lines (40 sloc) | 1.09 KB

```
1  # server.py
2  import socket
3  import json
4
5  HOST = '127.0.0.1'
6  PORT=65432
7
8  with socket.socket(socket.AF_INET,socket.SOCK_STREAM) as s:
9      s.bind((HOST,PORT))
10     s.listen()
11     print("[*] In ascolto su %s:%d"%(HOST,PORT))
12     clientsocket, address=s.accept()
13     with clientsocket as cs:
14         print("Connessione da ",address)
15         while True:
16             data=cs.recv(1024)
17             # if len(data)==0:
18             if not data:
19                 break
20             data=data.decode()
21             data=json.loads(data)
22             primoNumero=data['primoNumero']
23             operazione=data['operazione']
24             secondoNumero=data['secondoNumero']
25             ris=""
26             if operazione=="+":
27                 ris=primoNumero+secondoNumero
28             elif operazione=="-":
29                 ris=primoNumero-secondoNumero
30             elif operazione=="*":
31                 ris=primoNumero*secondoNumero
32             elif operazione=="/":
33                 if secondoNumero==0:
34                     ris="Non puoi dividere per 0"
35                 else:
36                     ris=primoNumero/secondoNumero
37             elif operazione=="%":
38                 ris=primoNumero%secondoNumero
39             else:
40                 ris="Operazione non riconosciuta"
41             ris=str(ris)
42             cs.sendall(ris.encode("UTF-8"))
```

Client Calcolatrice 2.0 : calcoClient2.py (versione con funzioni)

```
calcoClientFunc.py > ...
1  #!/usr/bin/env python3
2
3  import socket
4
5  SERVER_ADDRESS = '127.0.0.1'
6  SERVER_PORT = 22224
7
8  def invia_comandi(sock_service):
9  >  while True: ...
36
37
38  def connessione_server(address,port):
39      sock_service = socket.socket()
40      sock_service.connect((SERVER_ADDRESS, SERVER_PORT))
41      print("Connesso a " + str((SERVER_ADDRESS, SERVER_PORT)))
42      invia_comandi(sock_service)
43
44  if __name__=='__main__':
45      connessione_server(SERVER_ADDRESS,SERVER_PORT)
```

Server calcolatrice 2.0 : calcoServer2.py (versione con funzioni e thread)

```
calcoServerFunc.py > ...
1  import socket
2  from threading import Thread
3
4
5  SERVER_ADDRESS = '127.0.0.1'
6  SERVER_PORT = 22224
7
8  def ricevi_comandi(sock_service,addr_client):
9      print("avviato")
10 > while True:--
37
38      sock_service.close()
39
40
41
42  def ricevi_conessioni(sock_listen):
43      while True:
44          sock_service, addr_client = sock_listen.accept()
45          print("\nConnessione ricevuta da " + str(addr_client))
46          print("\nCreo un thread per servire le richieste ")
47          try:
48              Thread(target=ricevi_comandi, args=(sock_service,addr_client)).start()
49          except:
50              print("il thread non si avvia")
51              sock_listen.close()
52
53
54  def avvia_server(indirizzo,porta):
55      sock_listen = socket.socket()
56      sock_listen.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
57      sock_listen.bind((SERVER_ADDRESS, SERVER_PORT))
58      sock_listen.listen(5)
59      print("Server in ascolto su %s." % str((SERVER_ADDRESS, SERVER_PORT)))
60      ricevi_conessioni(sock_listen)
61
62  if __name__ == '__main__':
63      avvia_server(SERVER_ADDRESS,SERVER_PORT)
```