

```

1  #server.py
2  import socket
3  import random
4  import time
5
6
7  print("\nThe IP adress is by defult set to 'localhost' on the server
   side")
8  Host = "localhost"
9
10 Clients = []
11 recv_message = []
12
13 try:
14
15     #port = 4242
16     print("\nType in a preferred port number here in the command
   line, to start running the Chat room on this port: ( Eks: 4242 ) ")
17     port = int(input("Inputt port number here: "))
18
19 except Exception as e:
20     time.sleep(1)
21     print("\nERROR! - A port number can only contain numbers \n(Run
   the server again)\n")
22
23 def server_broadcast(Client_message, client_index):
24     if client_index:
25         for bots_msg in Clients: # den clienten vi er ved i start
26
27             if bots_msg != client_index: #Hvis meldingen som har
   blitt motatt ikke befinner seg i de andre botne
28
29                 bots_msg.send(Client_message.encode()) #sender ut
   meldingen vi mottok fra clienten vi er på nå til de andre clientene
30
31             else: #boten som sendte sin melding vil ikke for denne
   tilbake, men serveren vil ha motatt den
32
33                 bots_msg.send("This clients response has been sendt
   to the server\n".encode()) #motatt melding
34
35             else: # broadcaster reseterende meldinger som er motatt i
   serveren tilbake til alle clienter
36                 for bots_msg in Clients:
37                     bots_msg.send(Client_message.encode())
38
39 def server():
40     global Clients
41     global recv_message

```

```

42     global bots
43
44     #dette er greit
45     action_event = random.choice(["hug", "play", "eat", "shop", "yell",
46                                   ", "steal", "fight"])
47
48     Socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM) #
49     # generell kode for å sette opp Socket
50     Socket.bind((Host, port)) # binder socket til adresse - localhost
51     # og valfrlitt port nummer
52     Socket.listen() # vente på kobling til clienter - da 4 boter
53
54     print("\nThe TCP server chat rom is operating on port number {}".format(port))
55     print("Now the server is waiting to connect to the clients")
56
57     print("\n(REMEMBER TO START THE CLIENTS NOW!!!) - NB! Start each new client in paralell:")
58     print("0/4 bots has connected to the chat room at this point \n")
59
60     while True:
61
62         # Connection = Socket.accept() # går ikke
63         # address = Socket.accept() x går ikke
64         Connection, address = Socket.accept() # gikk fordi socket
65         # tar imot '2 inputs'
66         Clients.append(Connection) # legger til koblingen som vi
67         # mottok fra socket til å finne ut hvor hvilken client som blir koblet
68         # til serveren
69
70         client_names = Connection.recv(500).decode() # Får tilsendt
71         # det definerte 'navnet som ble satt for botten' fra client siden
72         print("{} - {} - {} / 4 bots connected ".format(client_names,
73                   address, len(Clients))) #printer ut bot_navn, (ip adresse og port
74         # number), samt rekkefølgen av boter som blir koblet til først og sist
75
76
77         connected_bots = "\nBot: {} has been sucessfully connected\n
78 {}/4 bots have been connected soo far\n \nwaiting for the rest of the
79 bots to be connected before starting the chat room....".format(
80 client_names, len(Clients))
81
82         Connection.send(connected_bots.encode())
83
84         if len(Clients) == 4: # if all bots are connected
85
86             print("\n{} /4 bots have connected successfully".format(
87 len(Clients))) # får et nummer på hvor mange clienter/boter vi har

```

```

76         print("The chat rom will noe proceed to run with {}/4
bots fully connected\n".format(len(Clients)))
77
78         time.sleep(1)
79         server_broadcast("\nBatman: Do you want to {}?".format(
action_event), "") #sending to every bot that is connected
80
81         print("A new suggestion!")
82         print("Batman sugessted: Do you want to {}? \n".format(
action_event))
83
84
85
86         bot_index = 0 # starter på index 0 of forsetter videre
for hver bot slik at vi mottar alle meldingene fra den som connected
først til sist
87
88         for bots in range(4): #loop over alle 4 bots
89             #bot_index = 0 # printer kun ut svar med bot1 som
sender selv om vi får meldinene til de andre botene
90             if bot_index > 3:
91                 time.sleep(2)
92                 bot_index = 0 # stopper loopen etter at vi har
fått loppet over alle 4 boter
93                 time.sleep(1.2)
94
95             for C in range(len(Clients)): # definer client
nummer med en index av C
96                 Client_msg = (Clients[C].recv(500).decode()) # get
message from alle clients basert på hvilken bot som sender melding
først og decoder dete fra bites til leselig tekst
97                 recv_message.append(Client_msg) # vi bruker her
index C for å holde styr på hvilken bot det er som sender melding
først til sist
98                 time.sleep(0.7)
99
100                 print(recv_message[bot_index]) #printer alle
meldingene fra botene i serveren etter at de har blitt motatt
101                 server_broadcast(recv_message[bot_index], Clients[
bot_index]) # Broadcastet alle meldingene som har blitt motatt av
serveren tilbake til clientene med untakk av den som sendte
meldingen
102                 time.sleep(0.3)
103                 recv_message.clear() # clearer den motatte meldingen
i for loopen fra den første som sendte meldingen bot1/index0 å går
videre til å bli neste bot
104                 bot_index+=1 # øker indeksen for å definere hvilken
bot sin melding vi mottar for hver itterasjon
105

```

```
106
107         Socket.close() #avslutter socket
108
109         end = str(input("\nThe clients have automatically been  
disconnected\nInput ( 'end' ) into the terminal to end the Chat rom  
: ")) #Avslutter chat rommet fra serveren sin side
110
111         if end == "end": #avslutte serveren
112             time.sleep(1)
113             print("\nThe chat room has been disconnected  
sucesfully")
114             exit()
115
116         else:
117             print("\nERROR! you did not disconnect from the  
server in the proper way!!!")
118             print("REMEMBER TO TYPE 'end' into the terminal and  
press enter to end the Chat room in the 'correct way' next time\n")
119             exit()
120
121
122
123 server()
124
```