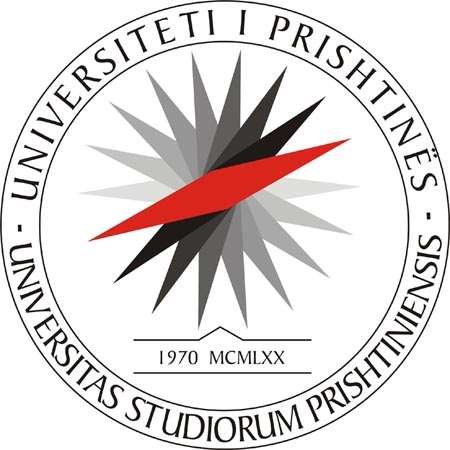
Universiteti i Prishtines “Hasan Prishtina”

Fakulteti i Inxhinierisë Elektrike dhe Kompjuterike



RAPORT

**Programimi me soketa**

Projekti në lëndën Rrjeta Kompjuterike

Gjuha programuese: Python

Vegla : Pycharm

Sistemi operativ : Ëindoës 10

Kandidati: Mentor:  
Shpend Jahiri Mr.Ass.Haxhi Lajqi

170714100074

18.04.2019 ,Prishtinë

Përmbajtje

[Hyrje 3](#_Toc6484280)

[Metoda IPADRESA 3](#_Toc6484281)

[Metoda NUMRIIPORTIT 4](#_Toc6484282)

[Metoda BASHKETINGELLORET 4](#_Toc6484283)

[Metoda KOHA 5](#_Toc6484284)

[Metoda EMRIIKOMPJUTERIT 5](#_Toc6484285)

[Metoda PRINTIMI 6](#_Toc6484286)

[Metoda LOJA 6](#_Toc6484287)

[Metoda FIBONACCI 7](#_Toc6484288)

[Metoda KONVERTIMI 8](#_Toc6484289)

[Metoda DITA(shtesë) 9](#_Toc6484290)

[Metoda KALKULATORI(shtesë) 10](#_Toc6484291)

[Protokolli TCP dhe UDP 11](#_Toc6484292)

[Threads 11](#_Toc6484293)

[Klienti 12](#_Toc6484294)

[Serveri 12](#_Toc6484295)

[Përmbyllje 13](#_Toc6484296)

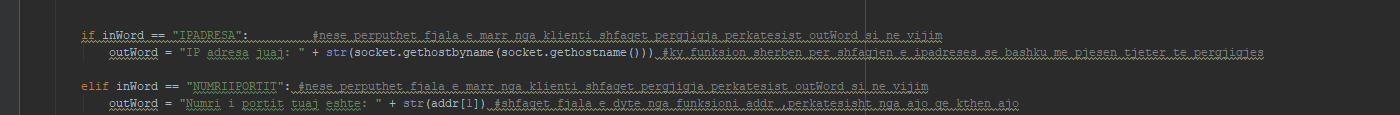
## Hyrje

Ky është projekti i parë nga lënda Rrjeta Kompjuterike që ka të bëjë me programimin me soketa si dhe përdorimin e protokolleve TCP dhe UDP. Meqë rast une kam bërë dizajnimin ,implementimin dhe testimin e programit server-klient në versionet TCP dhe UDP. Projekti përmban këto kërkesa (metoda): IPADRESA, NUMRIIPORTIT, BASHKETINGELLORET, PRINTIMI, EMRIIKOMPJUTERIT, KOHA, LOJA, KONVERTIMI ,FIBONACCI si dhe metodat shtesë KALKULATORI dhe DITA te cilat mund te dërgohen nga klienti tek serveri. Serveri përgjigjet me një mesazh i cili është specifik për secilën kërkesë (metodë)!

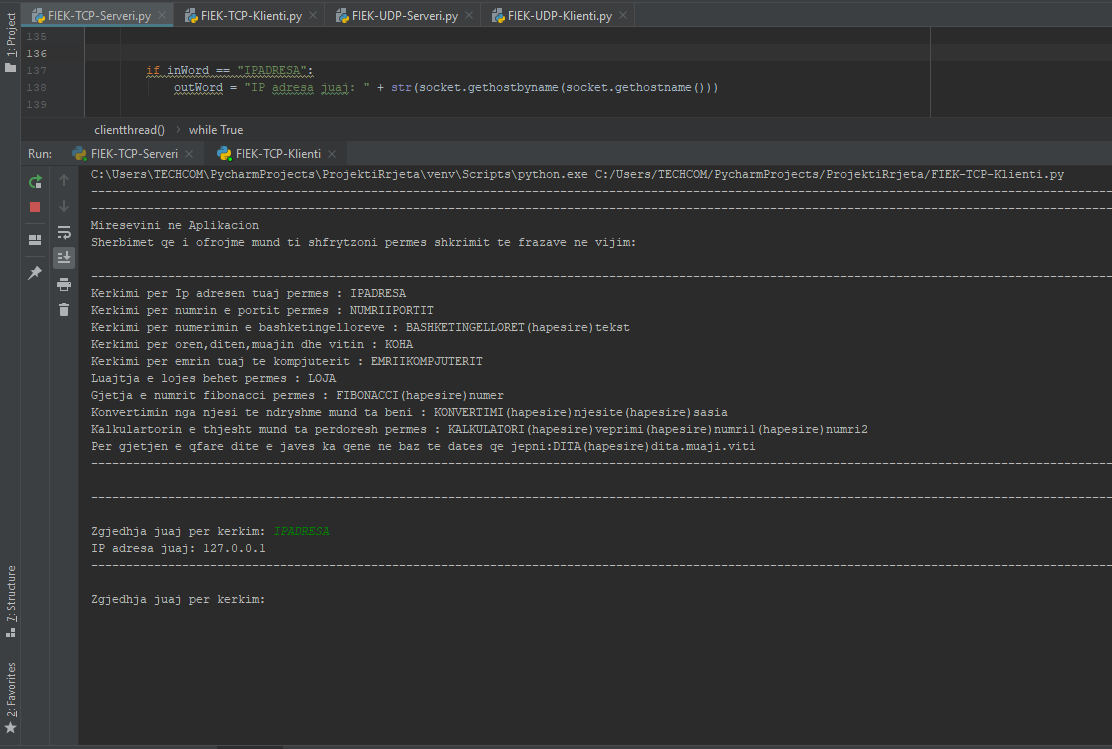
## Metoda IPADRESA

Metodë përmes së cilës klienti qoftë në TCP qoftë në UDP shkruan IPADRESA dhe serveri ia kthen si përgjigje ip adresën e tij. Këtë metodë e kam bërë përmes funksioneve socket.gethostbyname(socket.gethostname()).

Kodi :



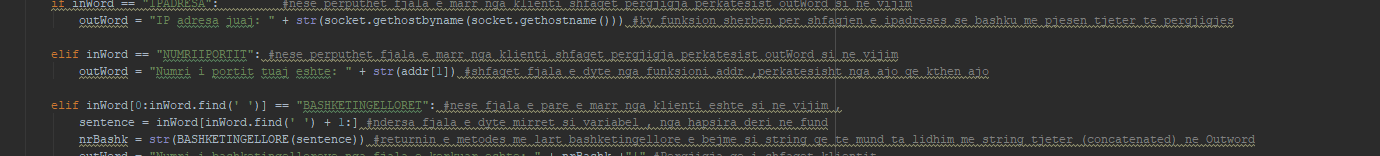
Testimi :

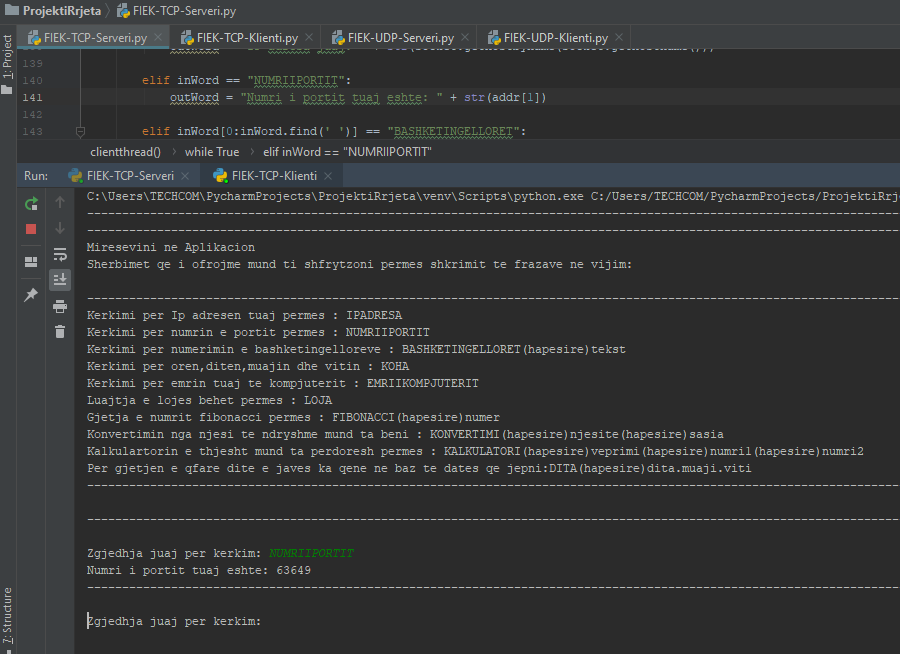


## Metoda NUMRIIPORTIT

Metodë kjo e cila mundëson që klientit ti kthehet numri i portit të tij. Për kryerjen e këtij funksioni përdoret addr[1] meqë rast mirret fjala e dyt nga variabla addr.

Kodi:

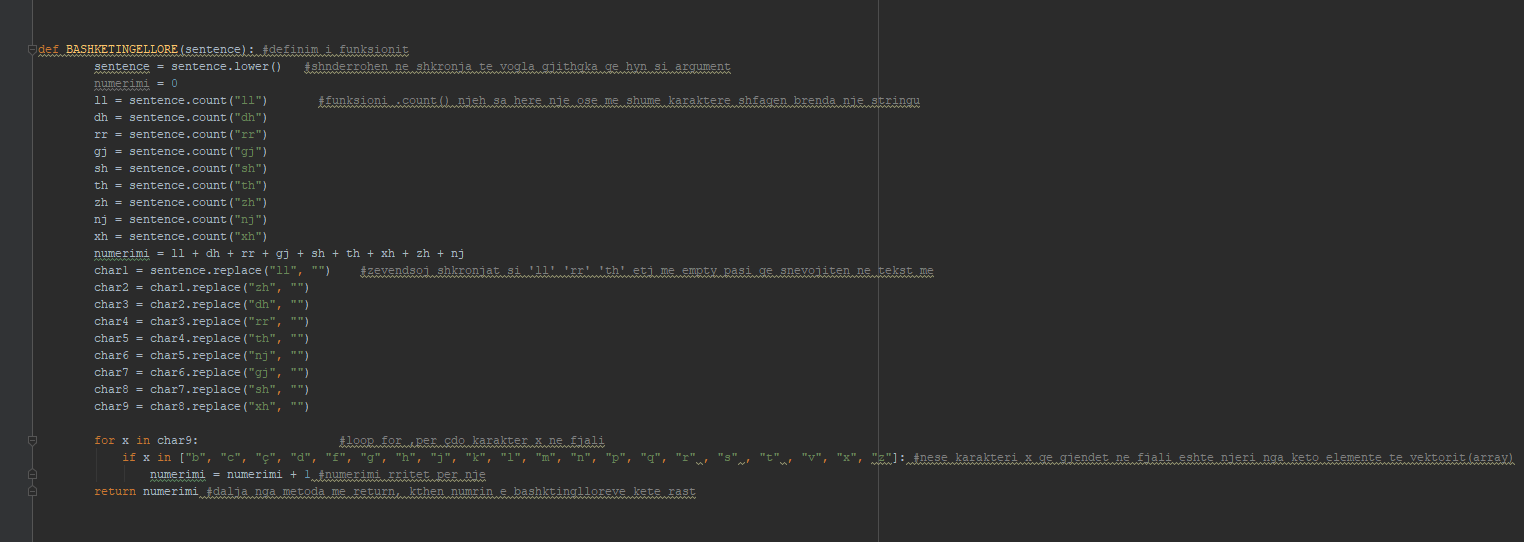


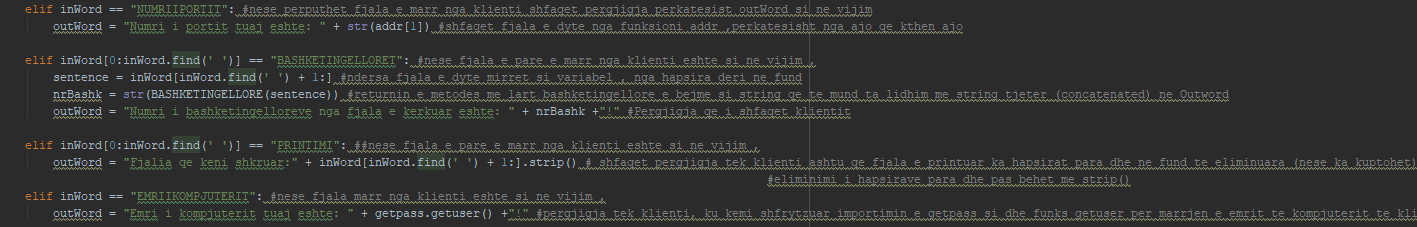
Testimi:  


## Metoda BASHKETINGELLORET

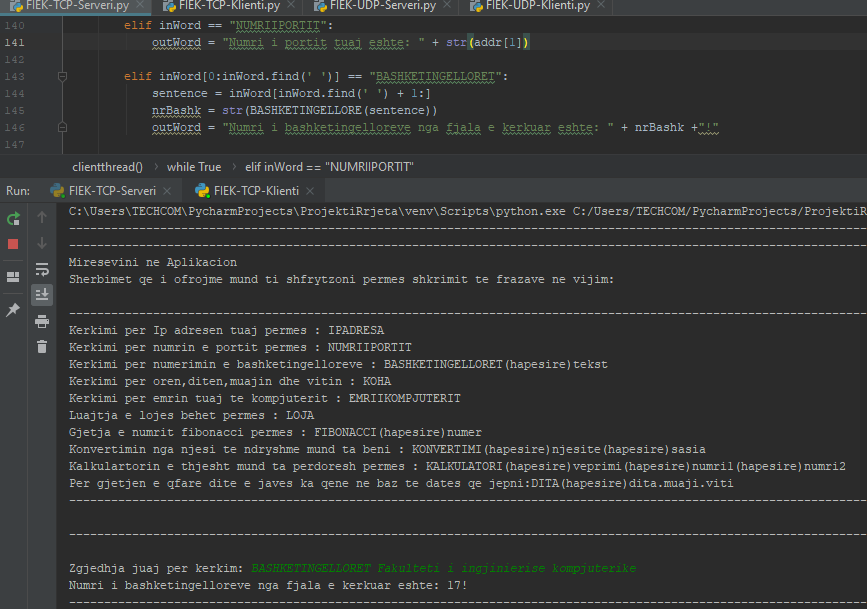
Metodë që bën numërimin e bashkëtingelloreve nga teksti që klienti shkruan. Numërimi bëhet si në alfabetin shqip duke përfshirë si bashkëtingëllore edhe shkronjat që përbëhen nga dy karaktere si ‘dh’,’sh’ etj.Kjo metodë nga ana e klientit kërkohet ashtu që shënojnë: BASHKETINGELLORET (hapësirë) tekst

Kodi:





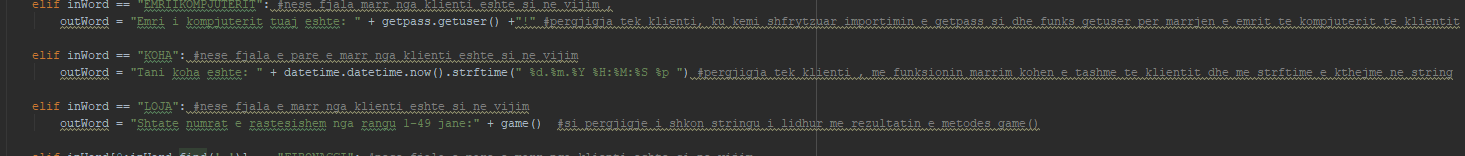
Testimi:



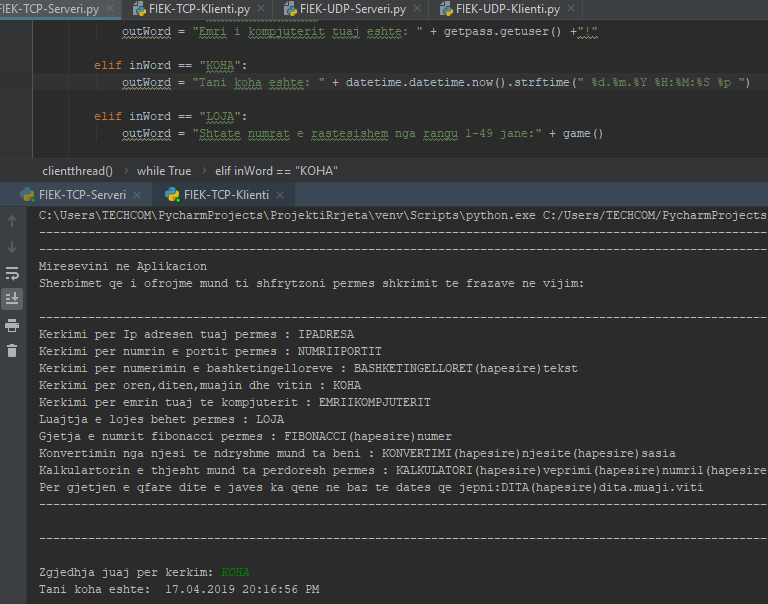
## Metoda KOHA

Metodë e cila kthen kohën e saktë të klientit kur ai e kërkon.Fillimisht importohet datetime pastaj përmes funksionit datetime.datetime.noë.strftime().

Kodi:



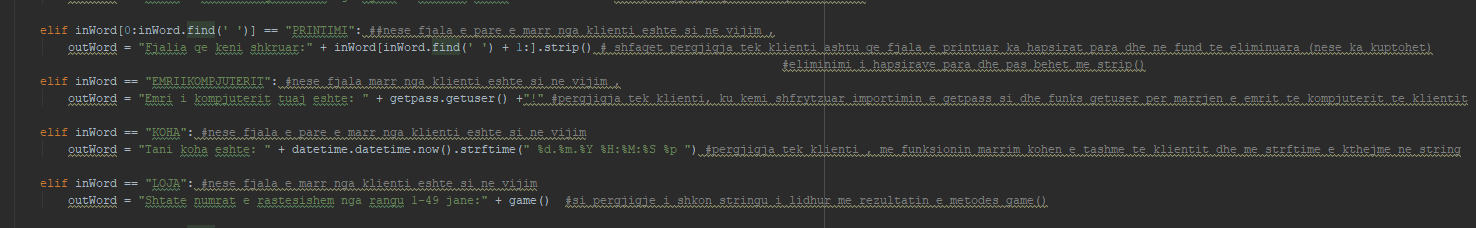
Testimi:



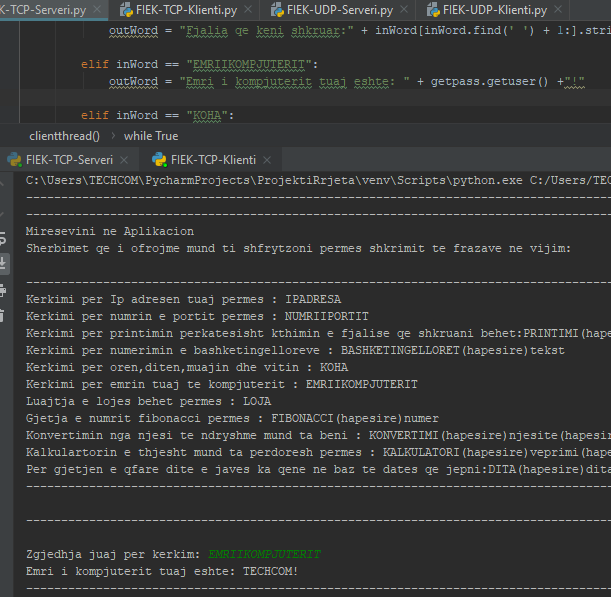
## Metoda EMRIIKOMPJUTERIT

Metodë që klienti mund ta përdorë për njohjen e shpejtë të emrit të kompjuterit të tij.Kjo mundësohet përmes funksioni getpass.getuser()

Kodi:



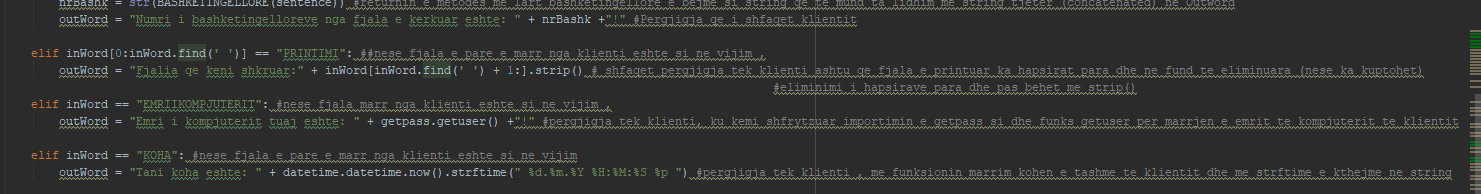
Testimi:



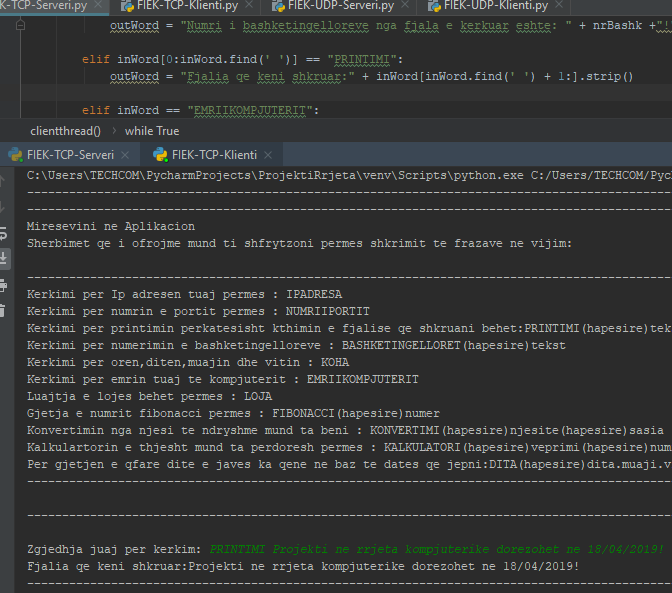
## Metoda PRINTIMI

Metoda printimi shërben që klientit ti kthejë të printuar fjalët që ai vet i ka shënuar. Në këtë metodë për largimin e hapësirave në fillim dhe fund kam përdorë funksionin .strip() .

Kodi:



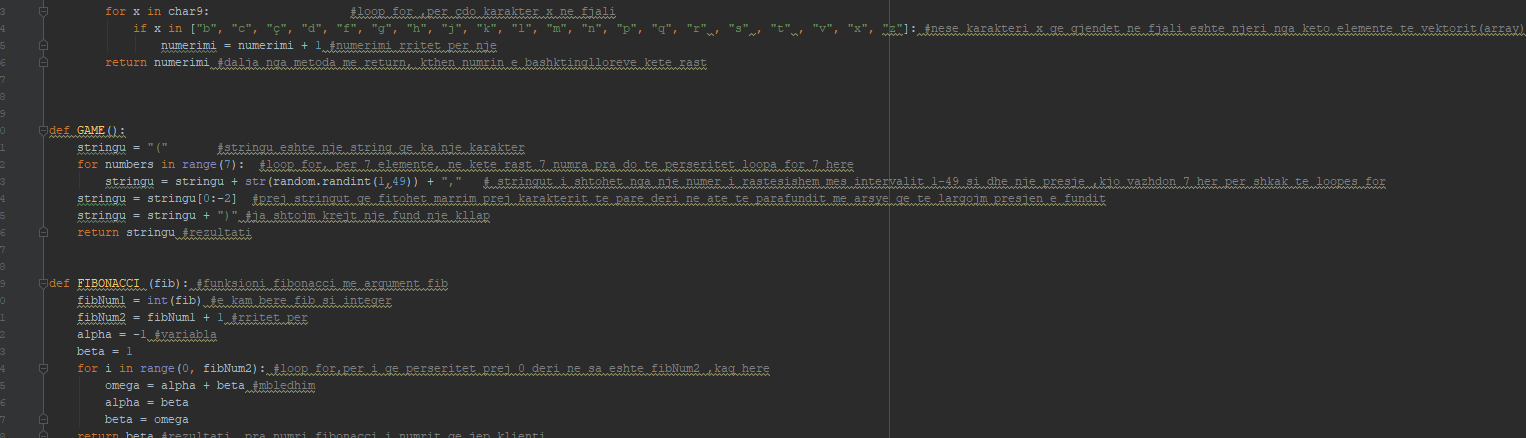
Testimi:

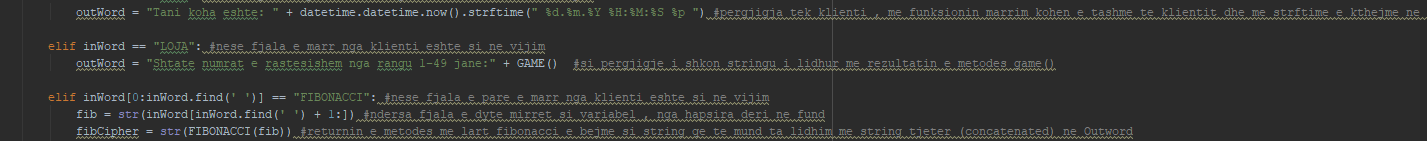


## Metoda LOJA

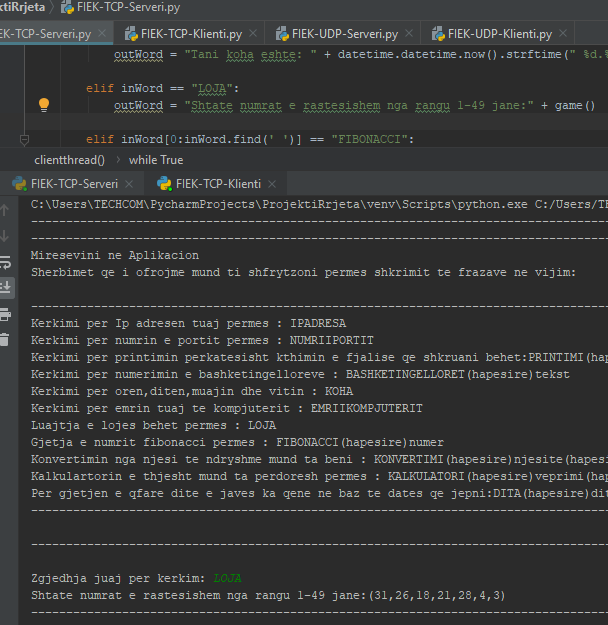
Kjo metodë është një lojë që përmes importimit te random zgjedh 7 numra të rastëshishëm në intervalin nga 1 deri 49.

Kodi:





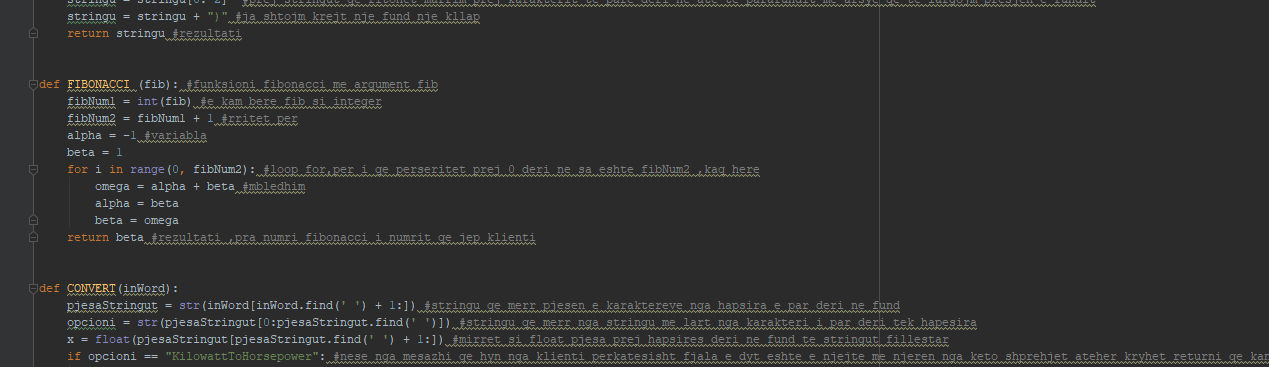
Testimi:

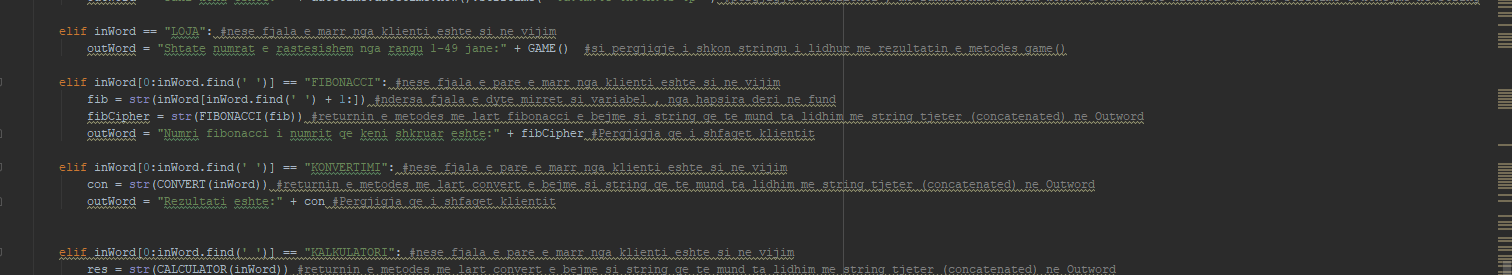


## Metoda FIBONACCI

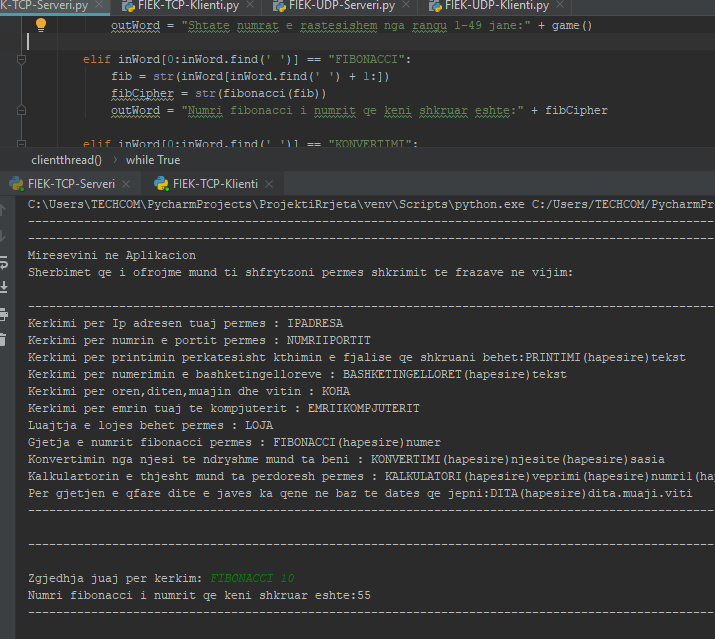
Metodë që shërben për gjetjen e një numri sipas algoritmit Fibonacci për një numër të shënuar nga vetë klienti .

Kodi:





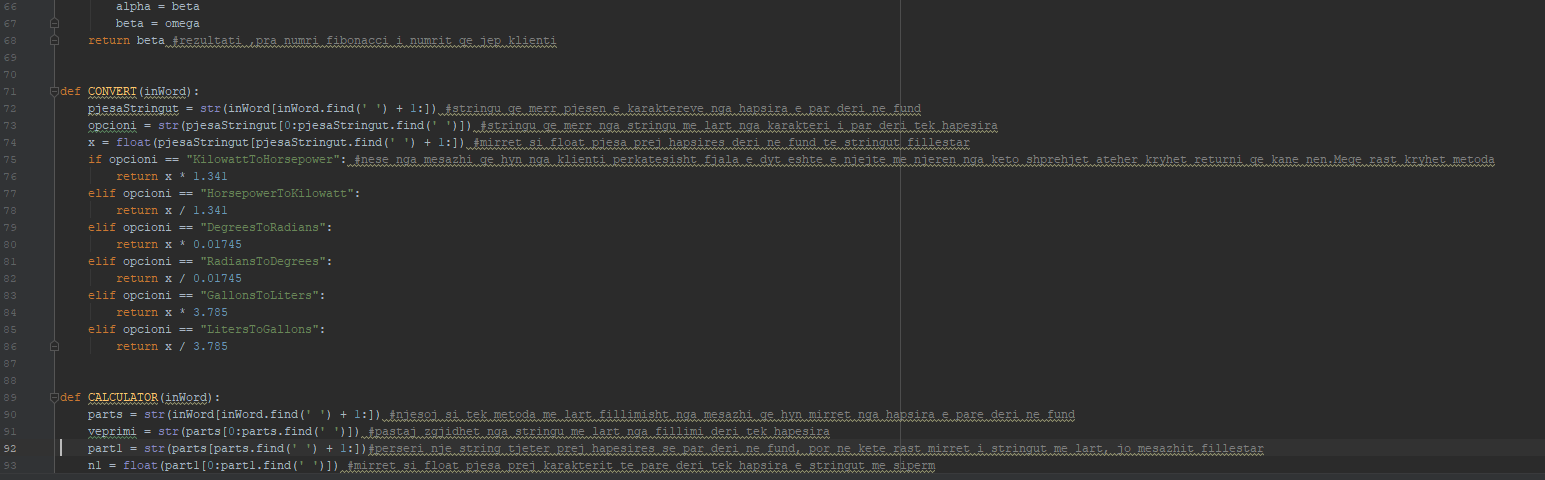
Testimi:

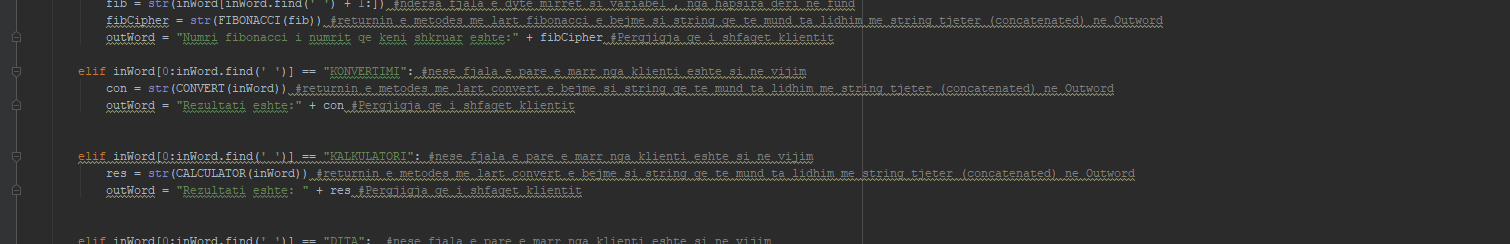


## Metoda KONVERTIMI

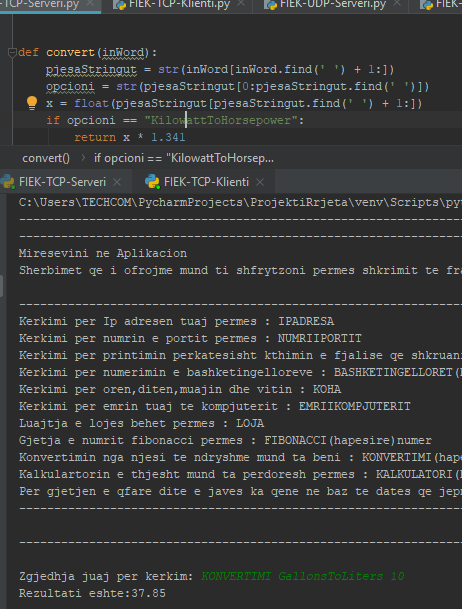
Kjo metodë shërben për konvertimin prej një njësie në njësitë tjera. Kështu klienti e shënon metodën ,opsionin prej cilës njësi në cilën dëshiron dhe në fund numrin apo sasinë. Mundësitë për konvertime janë: KiloëattToHorsepower , GallonsToLiters, HorsepoëerToKiloëatt, DegreesToRadians, RadiansToDegrees, LitersToGallons.

Kodi:





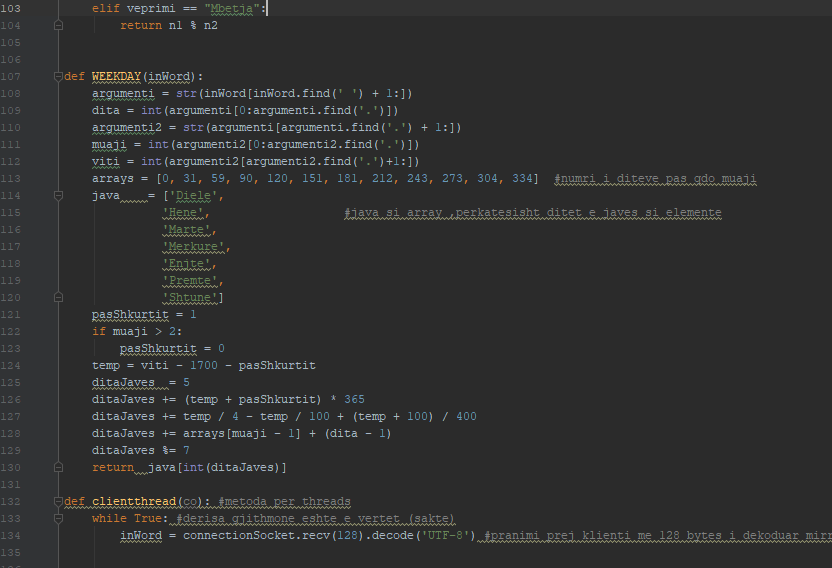
Testimi:

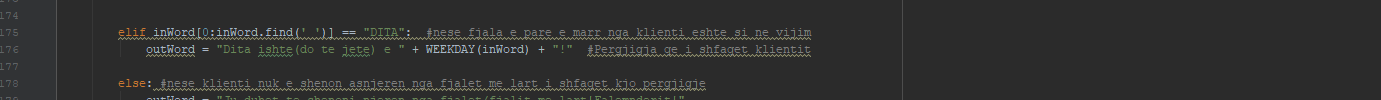


## Metoda DITA(shtesë)

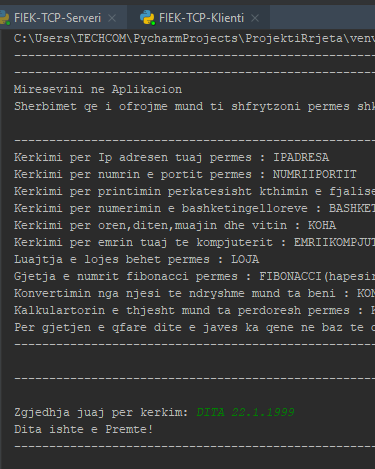
Metodë e cila kërkon nga klienti të shënojë një ditë përmes numrave p.sh (01.02.2001) qoftë datë e kaluar apo e ardhme dhe server me anë të algoritmit e kthenë se çfarë dite e javës ka qenë.

Kodi:





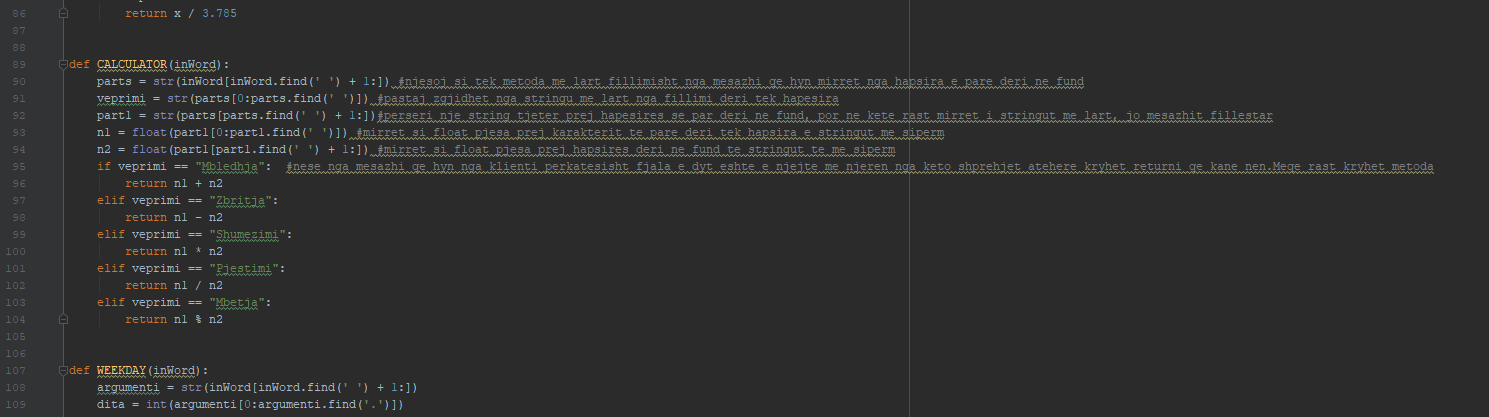
Testimi:

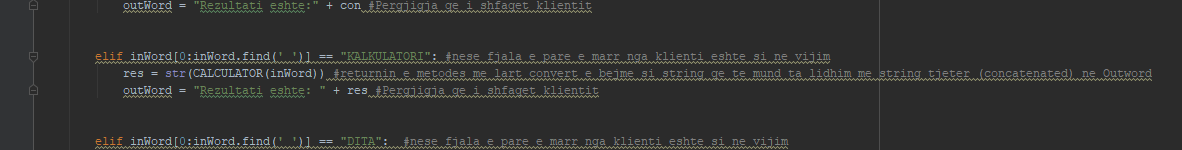


## Metoda KALKULATORI(shtesë)

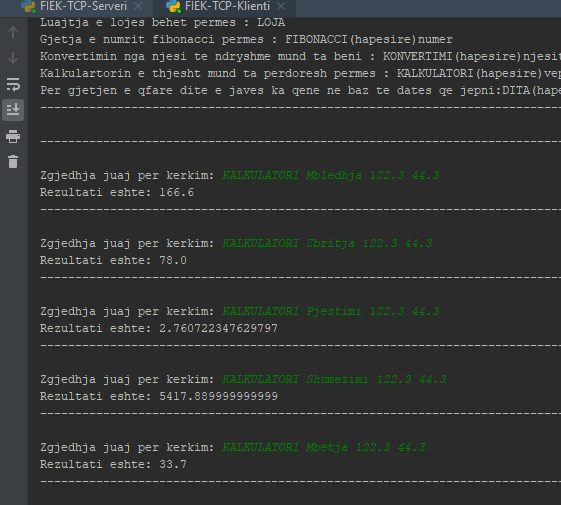
Kjo metodë shërben si kalkulator i thjeshtë me dy numra dhe një operatorë (+, -, /, \*, %) . Veprimet që mund ti zgjedh klienti janë: Mbledhja, Zbritja, Shumezimi, Pjestimi dhe Mbetja. Si numra mund të shënohen edhe numrat me presje dhjetore !

Kodi:





Testimi:



## Protokolli TCP dhe UDP

Të gjitha metodat më sipër janë të shkruara dhe të testuara me TCP. Por që kodi është i njëjtë edhe për protokollin UDP përveç se ndërron emri i variablës me të cilën është pranuar dhe dërguar e dhëna nga serveri në klient .Si dhe TCP përmban shprehjen në vijim:

*serverSocket = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)*

ndërsa UDP përmban :

*serversocket = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)*

pra UDP me datagram.

Që domethënë gjithçka është e njëjtë për atë arsye nuk kam future printscreen të kodeve dhe testimeve.

## Threads

Poashtu kam bërë implementimin e threads me anë të metodës si në vijim:

Fillimisht kam bërë importimin në fillim të serverit me anë të :

*From \_threads import*

Pastaj:

*Def clientthread(co):*

Si dhe shkruarjen e rreshtit më poshtë për fillimin e tyre nën ëhile 1:

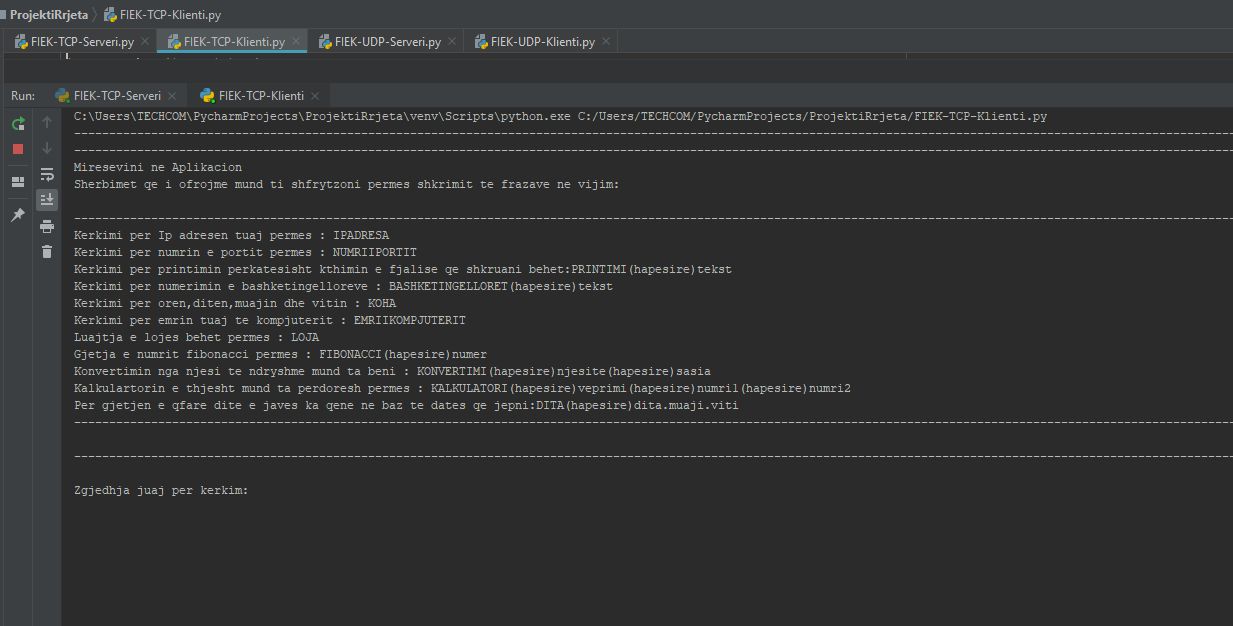
*start\_neë\_thread(clientthread,(connectionSocket,))*

Implementimin e threads e kam bërë vetëm në TCP !

Implementimi i threads ka qenë i nevojshëm pasi që vet threads janë procese më të lehta që nuk iu nevojitet shumë memory dhe janë më të lira se proceset në përgjithësi.

## Klienti

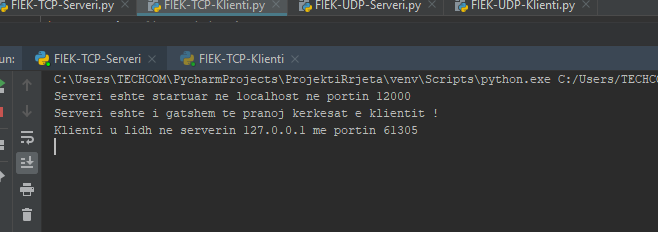
Pamja e klientit si aplikacion ii mi është kësisoj:



Kjo pamje është e njëjtë si për TCP ashtu edhe për UDP.Vetëm se dallimi është se te klienti në TCP ofrohet mundësia e jepjes së kërkesave vazhdimisht deri në një error të mundshëm ndërsa në UDP vetëm njëherë dhe bëhet ndalja e serverit.

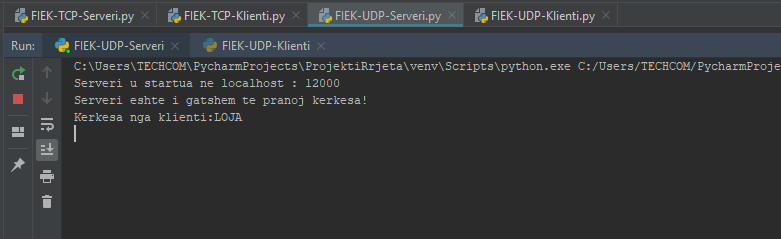
## Serveri

Pamja e serverit është kësisoj në TCP:



Në TCP serveri nuk tregon se çfarë ka shkruar klienti si dhe siç e ceka më lartë punon gjithmonë.

Kurse për UDP server duket kështu:



## Përmbyllje

Si fund dëshiroj të theksoj se të gjitha metodat jane funksionale si dhe mund të bëhet lidhja me klient në të njëjtën IP adresë.

Poashtu mendoj se ky projekt ka qenë shumë i dobishëm për shtimin e njohurive rreth programimit në soketa .