

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»
Факультет инфокоммуникационных технологий

ОТЧЕТ
О ЛАБОРАТОРНОЙ РАБОТЕ № 1
по теме: Работа с сокетами
по дисциплине: Web Программирование

Специальность:
09.03.03 Мобильные и сетевые технологии

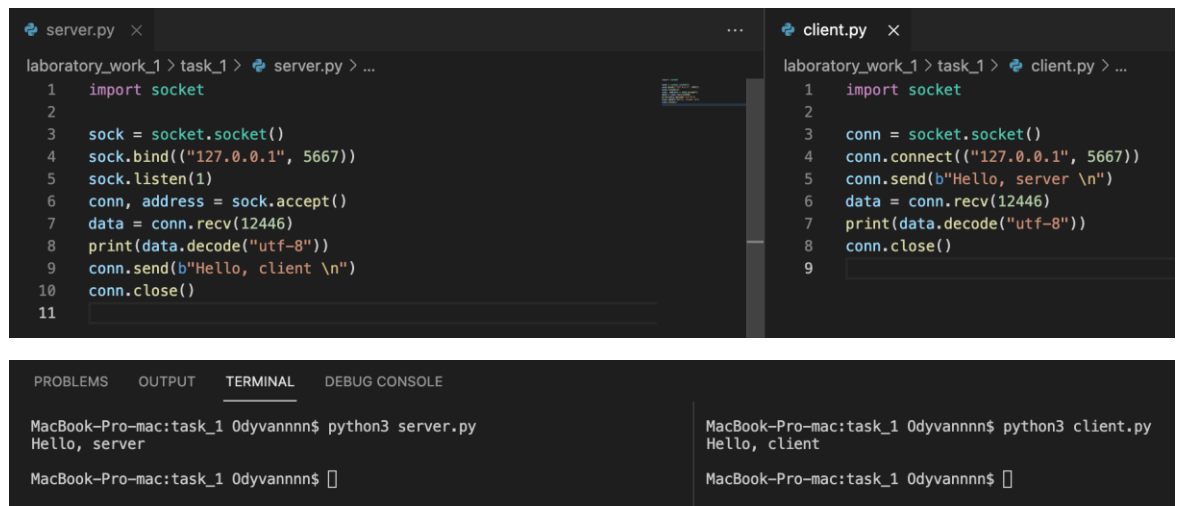
Выполнил:
студент группы К33402
Поляков Андрей.

Санкт-Петербург 2021/2022

Цель работы: овладеть практическими навыками и умениями реализации web-сервисов и использования сокетов.

Практическое задание:

1. Реализовать клиентскую и серверную часть приложения. Клиент отправляет серверу сообщение «Hello, server». Сообщение должно отразиться на стороне сервера. Сервер в ответ отправляет клиенту сообщение «Hello, client». Сообщение должно отобразиться у клиента.



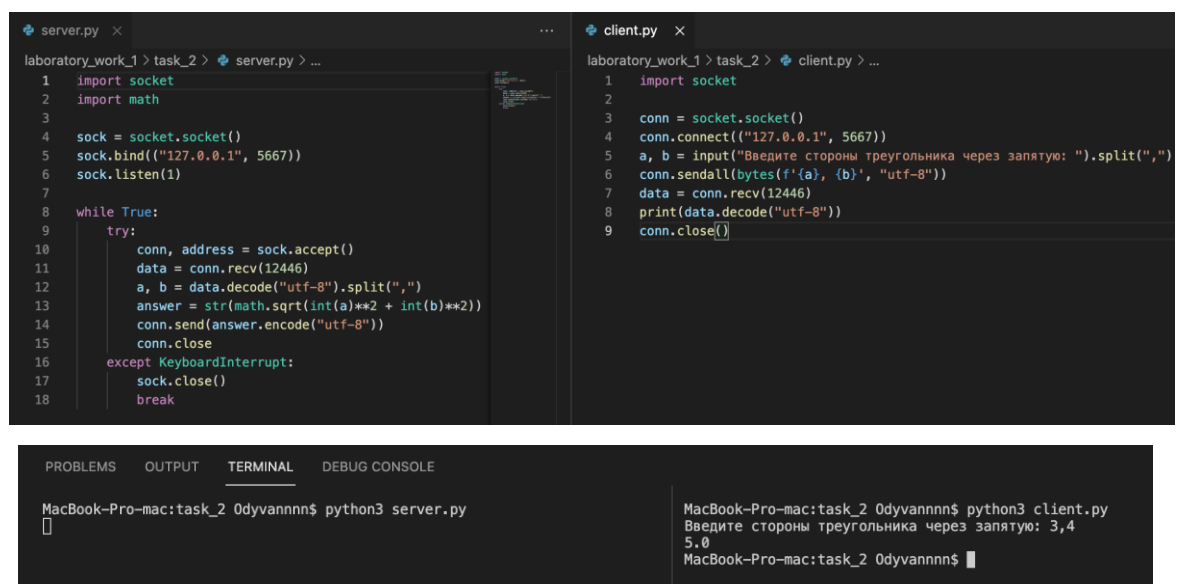
```
server.py
1 import socket
2
3 sock = socket.socket()
4 sock.bind(("127.0.0.1", 5667))
5 sock.listen(1)
6 conn, address = sock.accept()
7 data = conn.recv(12446)
8 print(data.decode("utf-8"))
9 conn.send(b"Hello, client \n")
10 conn.close()
11

client.py
1 import socket
2
3 conn = socket.socket()
4 conn.connect(("127.0.0.1", 5667))
5 conn.send(b"Hello, server \n")
6 data = conn.recv(12446)
7 print(data.decode("utf-8"))
8 conn.close()
9

Terminal:
MacBook-Pro-mac:task_1 Odyvannnn$ python3 server.py
Hello, server
MacBook-Pro-mac:task_1 Odyvannnn$

MacBook-Pro-mac:task_1 Odyvannnn$ python3 client.py
Hello, client
MacBook-Pro-mac:task_1 Odyvannnn$
```

2. Реализовать клиентскую и серверную часть приложения. Клиент запрашивает у сервера выполнение математической операции (Теорема Пифагора), параметры, которые вводятся с клавиатуры. Сервер обрабатывает полученные данные и возвращает результат клиенту.



```
server.py
1 import socket
2 import math
3
4 sock = socket.socket()
5 sock.bind(("127.0.0.1", 5667))
6 sock.listen(1)
7
8 while True:
9     try:
10         conn, address = sock.accept()
11         data = conn.recv(12446)
12         a, b = data.decode("utf-8").split(",")
13         answer = str(math.sqrt(int(a)**2 + int(b)**2))
14         conn.send(answer.encode("utf-8"))
15         conn.close()
16     except KeyboardInterrupt:
17         sock.close()
18         break

client.py
1 import socket
2
3 conn = socket.socket()
4 conn.connect(("127.0.0.1", 5667))
5 a, b = input("Введите стороны треугольника через запятую: ").split(",")
6 conn.sendall(bytes(f'{a}, {b}', "utf-8"))
7 data = conn.recv(12446)
8 print(data.decode("utf-8"))
9 conn.close()

Terminal:
MacBook-Pro-mac:task_2 Odyvannnn$ python3 server.py

MacBook-Pro-mac:task_2 Odyvannnn$ python3 client.py
Введите стороны треугольника через запятую: 3,4
5.0
MacBook-Pro-mac:task_2 Odyvannnn$
```

3. Реализовать серверную часть приложения. Клиент подключается к серверу. В ответ клиент получает http-сообщение, содержащее html-страницу, которую сервер подгружает из файла index.html.

```
server.py  ×
laboratory_work_1 > task_3 > server.py > ...
1  import socket
2  import math
3
4  sock = socket.socket()
5  sock.bind(("127.0.0.1", 5667))
6  sock.listen(1)
7
8  while True:
9      try:
10         conn, address = sock.accept()
11         with open("index.html", "r") as file:
12             response_type = "HTTP/1.0 200 OK\n"
13             headers = "content-type: text/html\n\n"
14             body = file.read()
15             response = response_type + headers + body
16             conn.send(response.encode("utf-8"))
17             conn.close
18     except KeyboardInterrupt:
19         sock.close()
20         break

client.py  ×
laboratory_work_1 > task_3 > client.py > ...
1  import socket
2
3  conn = socket.socket()
4  conn.connect(("127.0.0.1", 5667))
5  data = conn.recv(12446)
6  print(data.decode("utf-8"))
7  conn.close()
```

```
client.py  <> index.html  ×
laboratory_work_1 > task_3 > <> index.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3      <head>
4          <meta charset="utf-8">
5          <title>
6              Hi!
7          </title>
8      </head>
9      <body>
10     </body>
11 </html>
```

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE
MacBook-Pro-mac:task_3 Odyvannnn$ python3 server.py
␣

MacBook-Pro-mac:task_3 Odyvannnn$ python3 client.py
HTTP/1.0 200 OK
content-type: text/html

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>
      Hi!
    </title>
  </head>
  <body>
  </body>
</html>
MacBook-Pro-mac:task_3 Odyvannnn$
```

4. Реализовать многопользовательский чат.

```
client1.py ×
laboratory_work_1 > task_4 > client1.py > ...
1  import socket
2  import threading
3
4  def read_msg():
5      while True:
6          data = conn.recv(1024)
7          print(data.decode("utf-8"))
8  conn = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
9  server = "127.0.0.1", 5667
10 nickname = input("Write your nickname: ")
11 conn.sendto((nickname+" Connected to server").encode("utf-8"), server)
12 stream = threading.Thread(target=read_msg)
13 stream.start()
14 while True:
15     msg = input()
16     conn.sendto(("["+nickname+"]" + msg).encode("utf-8"), server)
17
```

```
laboratory_work_1 > task_4 > server.py > ...
1  import socket
2
3  conn = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
4  conn.bind(("127.0.0.1", 5667))
5  client=[]
6  print("Server started")
7
8  while True:
9      try:
10         data, address = conn.recvfrom(1024)
11         if address not in client:
12             client.append(address)
13         for clients in client:
14             if clients == address:
15                 continue
16             conn.sendto(data, clients)
17     except KeyboardInterrupt:
18         conn.close()
19         break
```

PROBLEMS	OUTPUT	TERMINAL	DEBUG CONSOLE
	MacBook-Pro-mac:task_4 OdyvanNN\$ p ython3 server.py Server started █	MacBook-Pro-mac:task_4 OdyvanNN\$ py thon3 client1.py Write your nickname: Sam Jane Connected to server Ann Connected to server Hello everybody! [Jane]Hi! [Ann]HEYHEY!!! █	MacBook-Pro-mac:task_4 OdyvanNN\$ py thon3 client2.py Write your nickname: Jane Ann Connected to server [Sam>Hello everybody! Hi! [Ann]HEYHEY!!! █
	MacBook-Pro-mac:task_4 OdyvanNN\$ p ython3 client3.py Write your nickname: Ann [Sam>Hello everybody! [Jane]Hi! HEYHEY!!! █		