

Санкт-Петербургский национальный исследовательский университет
информационных технологий, механики и оптики

Лабораторная работа № 1

«Работа с сокетами»

Выполнил: Евдокимов Владислав Борисович

Группа: К33421

Преподаватель: Говоров Антон Игоревич

Санкт-Петербург
2022

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

Оборудование: компьютерный класс.

Программное обеспечение: Python 2.7-3.6, библиотеки

Python: sys, socket

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.
- 2) Реализовать клиентскую и серверную часть приложения. Клиент запрашивает у сервера выполнение математической операции, параметры, которые вводятся с клавиатуры. Сервер обрабатывает полученные данные и возвращает результат клиенту. Варианты:
 - a) Теорема Пифагора
 - b) Решение квадратного уравнения.
 - c) Поиск площади трапеции.
 - d) Поиск площади параллелограмма
- 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.
- 4) Реализовать двухпользовательский или многопользовательский чат. Реализация многопользовательского чата позволяет получить максимальное количество баллов.

ВЫПОЛНЕНИЕ РАБОТЫ

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.

Серверная часть:

```
import socket

sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
sock.bind(('', 6666))
sock.listen(1)

clientsocket, addr = sock.accept()
msg = clientsocket.recv(1024)
print(msg.decode("utf-8"))
clientsocket.send(bytes("Hello, client!!!!", "utf-8"))

clientsocket.close()
```

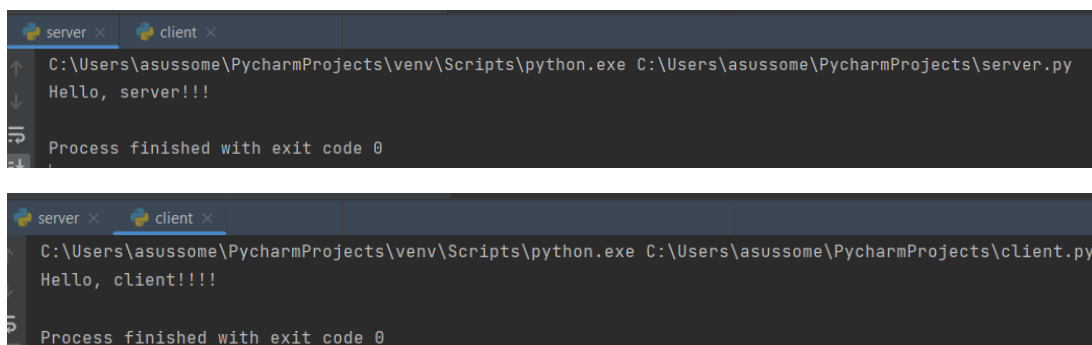
Клиентская часть:

```
import socket

sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
sock.connect(('localhost', 6666))

sock.send(bytes("Hello, server!!!", "utf-8"))
msg = sock.recv(1024)
print(msg.decode("utf-8"))
```

Док-во работы:



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

Вариант: Поиск площади параллелограмма

Клиентская часть:

```
import socket

sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
sock.connect(('localhost', 5675))

while True:
    try:
        data = sock.recv(1024)
        print(data.decode('utf-8'))
        sock.send(input().encode('utf-8'))
    except KeyboardInterrupt:
        sock.close()
        break

sock.close()
```

Серверная часть:

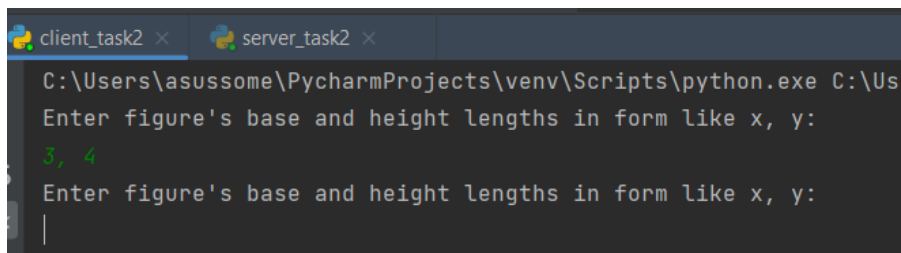
```
import socket

sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
sock.bind(('localhost', 5675))
sock.listen(1)
conn, addr = sock.accept()

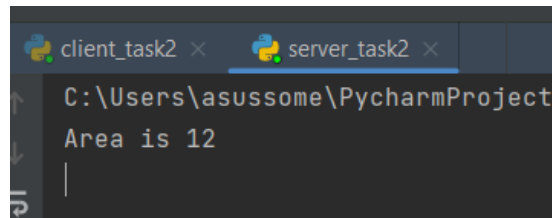
while True:
    conn.send("Enter figure's base and height lengths in form like x, y:".encode('utf-8'))
    try:
        data = conn.recv(1024)
        a, h = data.decode('utf-8').split(' ')
        s = (int(a) * int(h))
        print('Area is', s)
    except KeyboardInterrupt:
        conn.close()
        break

sock.close()
```

Принцип работы:



```
client_task2 x server_task2 x
C:\Users\asussome\PycharmProjects\venv\Scripts\python.exe C:\Us
Enter figure's base and height lengths in form like x, y:
3, 4
Enter figure's base and height lengths in form like x, y:
|
```



```
client_task2 x server_task2 x
C:\Users\asussome\PycharmProject
Area is 12
|
```

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.

Клиент:

```
import socket

sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
sock.bind(('localhost', 6060))
sock.listen(1)

conn, addr = sock.accept()
conn.sendall(b"HTTP/1.0 200 OK\nContent-Type: text/html\n\n" + open("index.html", "rb").read())
conn.close()
```

Сервер:

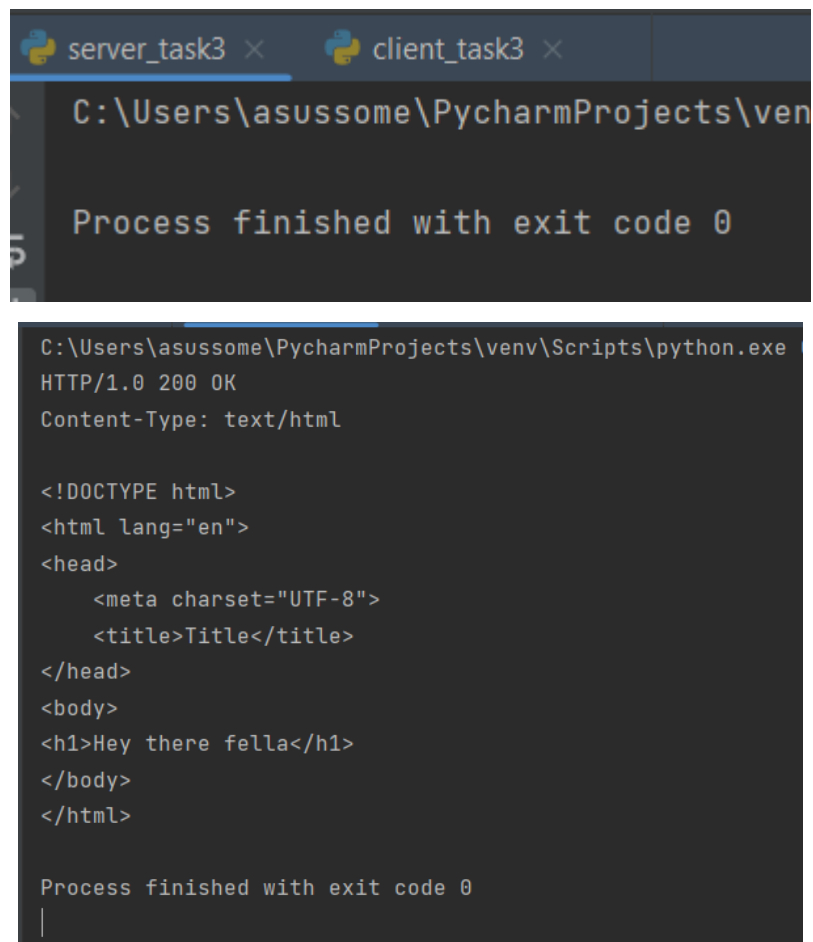
```
import socket

sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
sock.connect(('localhost', 6060))

while True:
    data = sock.recv(1024)
    if not data:
        break
    print(data.decode("utf-8"))

sock.close()
```

Работа:



```
server_task3 x client_task3 x
C:\Users\asussome\PycharmProjects\venv
Process finished with exit code 0

C:\Users\asussome\PycharmProjects\venv\Scripts\python.exe
HTTP/1.0 200 OK
Content-Type: text/html

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
<body>
<h1>Hey there fella</h1>
</body>
</html>

Process finished with exit code 0
|
```

4. Реализовать двухпользовательский или многопользовательский чат. Реализация многопользовательского чата позволяет получить максимальное количество баллов.

Код для всех пользователей (см. ниже):

```

import socket
from threading import Thread

class Client:
    def __init__(self, port):
        self._socket = socket.socket(family=socket.AF_INET,
                                     type=socket.SOCK_STREAM,
                                     proto=0)
        self._socket.connect(('127.0.0.1', port))

    def get(self):
        while True:
            try:
                data = self._socket.recv(1024)
                print(data.decode('utf-8'))
            except OSError:
                exit()

    def send(self):
        while True:
            message = input()
            if message == 'покинуть':
                self._socket.send(bytes(f'{message}', 'utf-8'))
                print('Connection closed')
                self._socket.close()
                break
            self._socket.send(bytes(f'{message}', 'utf-8'))

```

```

if __name__ == '__main__':
    clint = Client(53330)
    th_1, th_2 = Thread(target=clint.send), Thread(target=clint.get)
    th_1.start(), th_2.start()

```

Серверная часть (см. ниже):

```

import asyncio
import socket
connections = []
async def handle_client(client, address):
    request = None
    while request != 'покинуть':
        request = (await loop.sock_recv(client, 255)).decode('utf8')
        response = f'User{address}: ' + str(request)
        if request == 'покинуть':
            response = f'User {address} left the chat'
        for client_ in connections:
            if client_ == client:
                continue
            await loop.sock_sendall(client_, response.encode('utf8'))
    connections.remove(client)
    print(f'Client: {address} disconnected')
    client.close()

async def run_server():
    while True:
        client, address = await loop.sock_accept(server)
        if client not in connections:
            connections.append(client)
            print(f'Client {address} connected to the chat!')
            loop.create_task(handle_client(client, address))
server = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
server.bind(('localhost', 53330))
server.listen(8)
server.setblocking(False)

loop = asyncio.get_event_loop()
loop.run_until_complete(run_server())

```


Док-во работы:

```
server4 x client4_3 x client4_2 x client4_1 x
C:\Users\asussome\PycharmProjects\venv\Scripts\python.exe C:\Users\
C:\Users\asussome\PycharmProjects\server4.py:37: DeprecationWarning
    loop = asyncio.get_event_loop()
Client ('127.0.0.1', 53179) connected to the chat!
Client ('127.0.0.1', 53180) connected to the chat!
Client ('127.0.0.1', 53181) connected to the chat!
Client: ('127.0.0.1', 53181) disconnected
```

```
server4 x client4_3 x client4_2 x client4_1 x
C:\Users\asussome\PycharmProjects\venv\Scripts\python.exe C:\Users\asussome\
привет всем))))))
User('127.0.0.1', 53180): держи +15 рублей
User('127.0.0.1', 53179): когда ты там говоришь ты зарегистрировался?
пнх
покинуть
Connection closed

Process finished with exit code 0
```

```
server4 x client4_3 x client4_2 x client4_1 x
C:\Users\asussome\PycharmProjects\venv\Scripts\python.exe C:\Users\asussome\
User('127.0.0.1', 53181): привет всем))))))
держи +15 рублей
User('127.0.0.1', 53179): когда ты там говоришь ты зарегистрировался?
User('127.0.0.1', 53181): пнх
User ('127.0.0.1', 53181) left the chat
```

```
server4 x client4_3 x client4_2 x client4_1 x
C:\Users\asussome\PycharmProjects\venv\Scripts\python.exe
User('127.0.0.1', 53181): привет всем))))))
User('127.0.0.1', 53180): держи +15 рублей
когда ты там говоришь ты зарегистрировался?
User('127.0.0.1', 53181): пнх
User ('127.0.0.1', 53181) left the chat
```

Вывод: я обуздал основы и принцип работы с сокетами.