Syrian Arab Republic

Lattakia - Tishreen University

Department of Communication and electrical engineering

5th , Network Programming : Homework No2

الجمهورية العربية السورية

اللاذقية ـ جامعـــة تشريـــن

كلية الهندسة الكهربائية والميكانيكية

قسم هندسة الاتصالات والالكترونيات

السنة الخامسة: وظيفة2 برمجة شبكات

Name:_ __________, Number:_ ___________, Submitted To GitHub:_ _____

Second Network Programming Homework

Question 1: TCP Server/Client Quiz App with Multi-threading?

As an improvement to previous first homework, build a TCP server and client quiz application using Python. The server should handle multiple client connections simultaneously using multi-threading. The application should allow clients to connect, participate in a quiz, and receive their quiz scores upon completion.

Requirements:

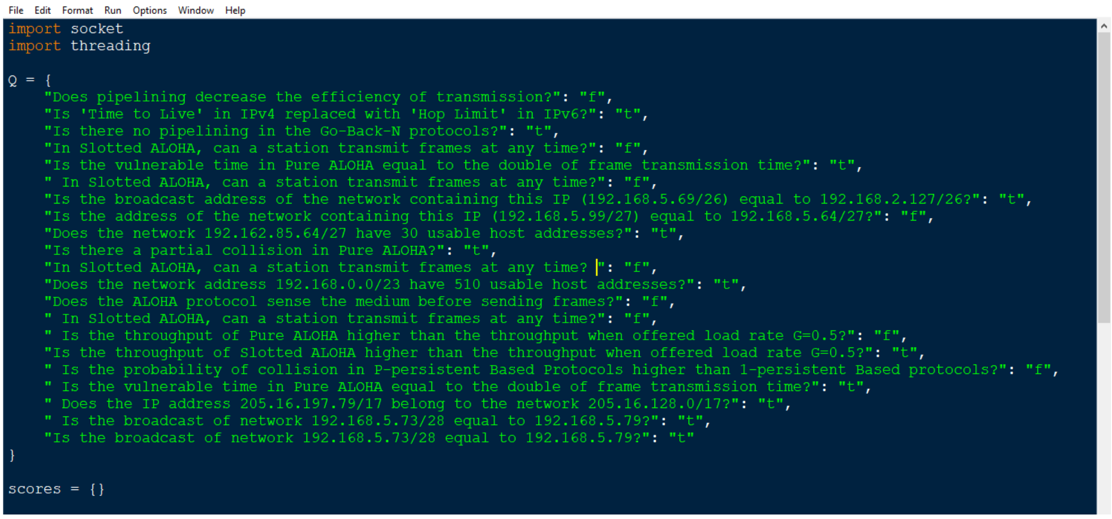
A. The server should be able to handle multiple client connections concurrently.
B. The quiz should consist of a set of pre-defined questions stored on the server.
C. Each client should connect to the server and receive the quiz questions.
D. Clients should send their answers to the server.
E. The server should keep track of the scores for each client.
F. At the end of the quiz, the server should send the final scores to each client.

Guidelines:

- Use Python's socket module “don’t use 3thd-party packages”.
- Implement multi-threading to handle multiple client connections concurrently.
- Store the quiz questions and correct answers on the server side.

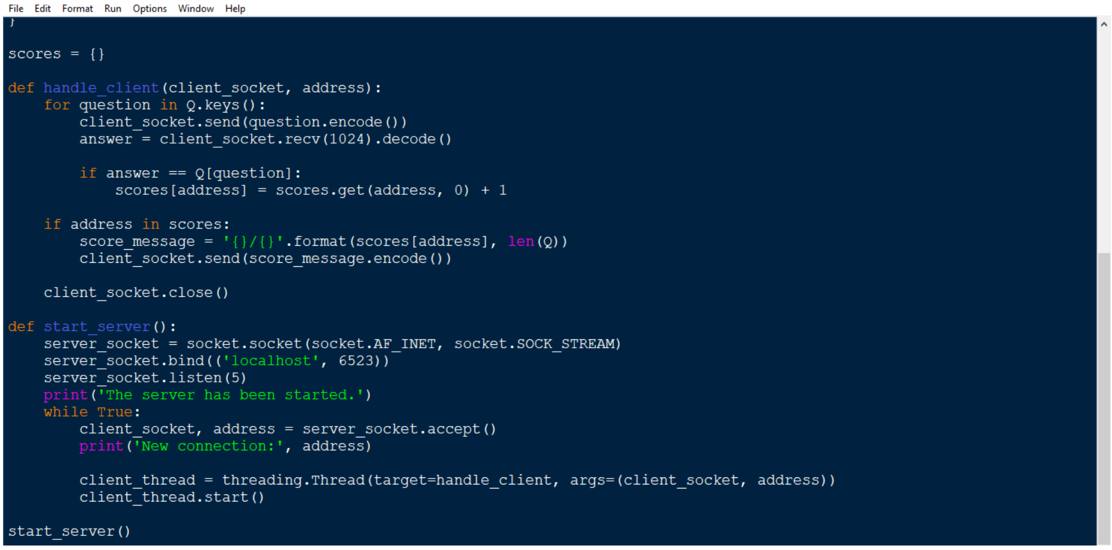
Notes:

- Write brief report describing the design choices you made and any challenges faced during implementation.
- You can make a TCP Server/Client of your choice, such as Bank ATM, Chat application, or any other appropriate application that fulfil all requirements.



```python
import socket
import threading

Q = {
    "Does pipelining decrease the efficiency of transmission?": "f",
    "Is 'Time to Live' in IPv4 replaced with 'Hop Limit' in IPv6?": "t",
    "Is there no pipelining in the Go-Back-N protocols?": "t",
    "In Slotted ALOHA, can a station transmit frames at any time?": "f",
    "Is the vulnerable time in Pure ALOHA equal to the double of frame transmission time?": "t",
    " In Slotted ALOHA, can a station transmit frames at any time?": "f",
    "Is the broadcast address of the network containing this IP (192.168.5.69/26) equal to 192.168.2.127/26?": "t",
    "Is the address of the network containing this IP (192.168.5.99/27) equal to 192.168.5.64/27?": "f",
    "Does the network 192.162.85.64/27 have 30 usable host addresses?": "t",
    "Is there a partial collision in Pure ALOHA?": "t",
    "In Slotted ALOHA, can a station transmit frames at any time? ": "f",
    "Does the network address 192.168.0.0/23 have 510 usable host addresses?": "t",
    "Does the ALOHA protocol sense the medium before sending frames?": "f",
    " In Slotted ALOHA, can a station transmit frames at any time?": "f",
    " Is the throughput of Pure ALOHA higher than the throughput when offered load rate G=0.5?": "f",
    "Is the throughput of Slotted ALOHA higher than the throughput when offered load rate G=0.5?": "t",
    " Is the probability of collision in P-persistent Based Protocols higher than 1-persistent Based protocols?": "f",
    " Is the vulnerable time in Pure ALOHA equal to the double of frame transmission time?": "t",
    " Does the IP address 205.16.197.79/17 belong to the network 205.16.128.0/17?": "t",
    " Is the broadcast of network 192.168.5.73/28 equal to 192.168.5.79?": "t",
    "Is the broadcast of network 192.168.5.73/28 equal to 192.168.5.79?": "t"
}

scores = {}
```

Name:_ ___________, Number:_ ____________, Submitted To GitHub:_ ______



```
}

scores = {}

def handle_client(client_socket, address):
    for question in Q.keys():
        client_socket.send(question.encode())
        answer = client_socket.recv(1024).decode()

        if answer == Q[question]:
            scores[address] = scores.get(address, 0) + 1

    if address in scores:
        score_message = '{}/{}'.format(scores[address], len(Q))
        client_socket.send(score_message.encode())

    client_socket.close()

def start_server():
    server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    server_socket.bind(('localhost', 6523))
    server_socket.listen(5)
    print('The server has been started.')
    while True:
        client_socket, address = server_socket.accept()
        print('New connection:', address)

        client_thread = threading.Thread(target=handle_client, args=(client_socket, address))
        client_thread.start()

start_server()
```

يتم استيراد مكتبات Socket و Threading.

حتوي المتغير Q على مجموعة من الأسئلة والإجابات المقابلة لها في شكل قاموس.

المتغير scores يستخدم لتتبع نتائج العملاء وحصر النقاط التي يحصلون عليها.

تعريف الدالة handle_client تقوم بالتواصل مع عميل معين. تقوم الدالة بإرسال الأسئلة إلى العميل واستقبال إجاباته، ثم تقوم بتحديد ما إذا كانت الإجابة صحيحة أم لا وتحسب النقاط المحصلة. في النهاية، تقوم الدالة بإرسال نتيجة العميل إليه.

دالة start_server تستخدم لبدء سيرفر الاستماع. تتم معالجة اتصال العميل الجديد عبر إنشاء خيط جديد يستدعي الدالة handle_client لكل عميل جديد يتصل به.

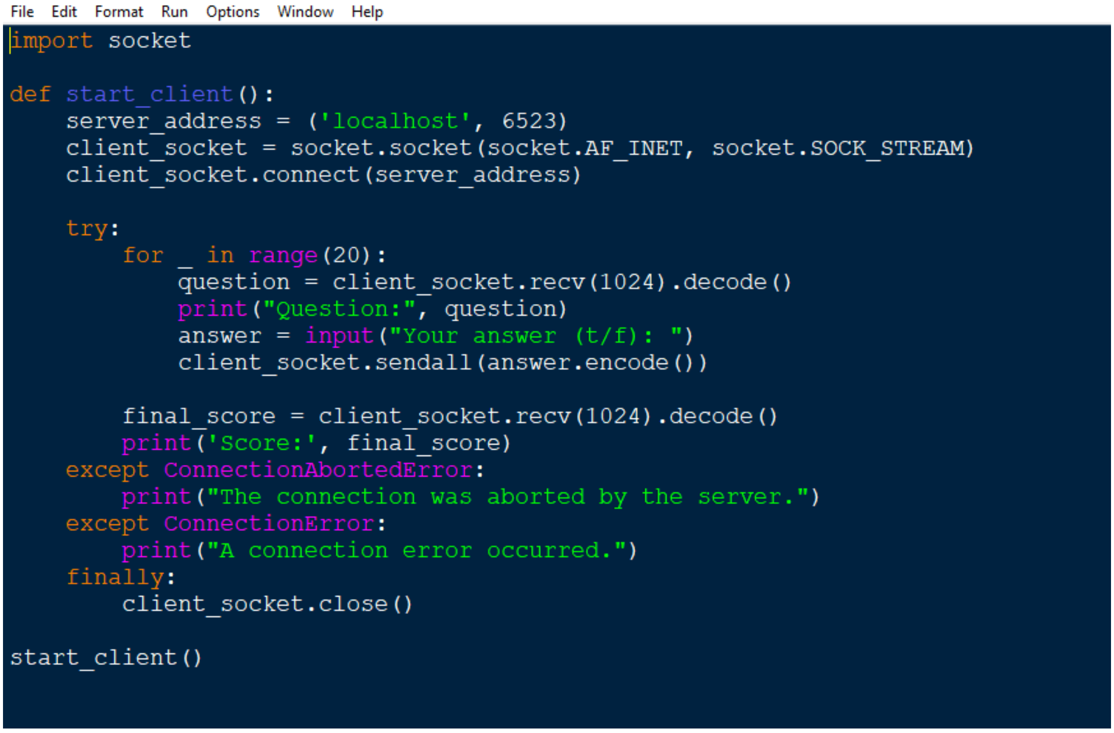
يتم ربط سيرفر Socket بعنوان 'localhost 'IP ومنفذ ٦٥٢٣.

عند تشغيل البرنامج، يتم طباعة رسالة "The server has been started." للإشارة إلى بدء تشغيل السيرفر.

عندما يتصل عميل جديد، يتم طباعة رسالة "New connection:" مع عنوان العميل.

ثم يتم إنشاء خيط جديد للتعامل مع العميل ويتم تشغيل الخيط باستخدام الدالة handle_client.

السيرفر يستمر في الاستماع للعملاء الجدد وتكرار نفس العملية لكل عميل جديد يتصل به.



```
import socket

def start_client():
    server_address = ('localhost', 6523)
    client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    client_socket.connect(server_address)

    try:
        for _ in range(20):
            question = client_socket.recv(1024).decode()
            print("Question:", question)
            answer = input("Your answer (t/f): ")
            client_socket.sendall(answer.encode())

        final_score = client_socket.recv(1024).decode()
        print('Score:', final_score)
    except ConnectionAbortedError:
        print("The connection was aborted by the server.")
    except ConnectionError:
        print("A connection error occurred.")
    finally:
        client_socket.close()

start_client()
```

استيراد مكتبة Socket.

تعريف الدالة start_client يقوم بإنشاء عميل والاتصال بسيرفر Socket بعنوان IP 'localhost' ومنفذ ٦٥٢٣.

يتم استقبال الأسئلة من السيرفر وطباعتها للعميل. العميل يدخل إجابته على السؤال (صح أم خطأ) وترسل الإجابة إلى السيرفر.

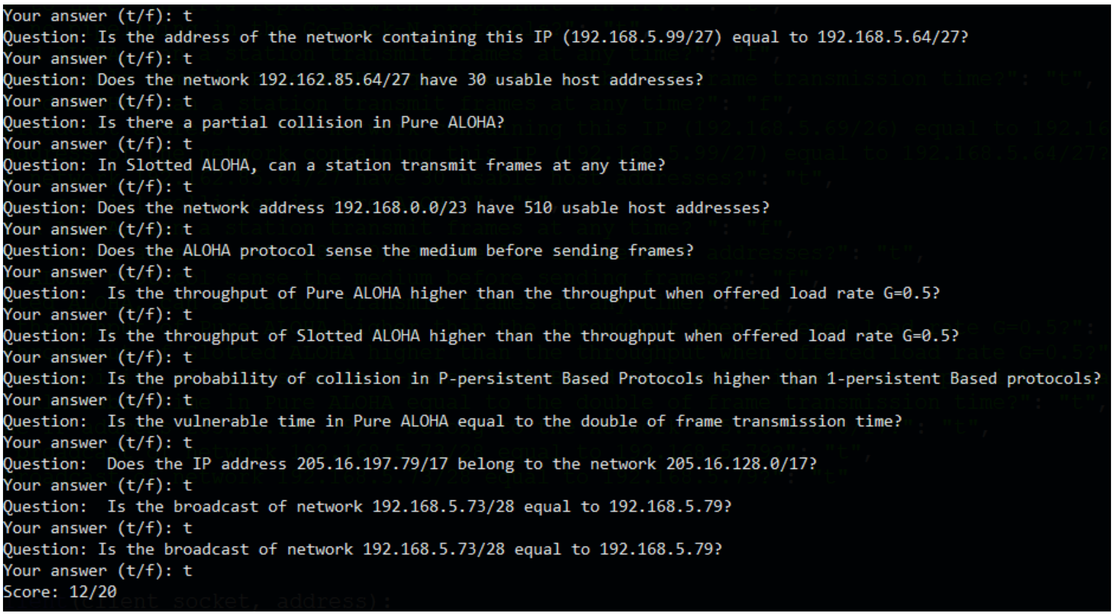
تتكرر العملية لمجموعة من ٢٠ سؤال.

في النهاية، يتم استقبال النتيجة النهائية من السيرفر وطباعتها للعميل.

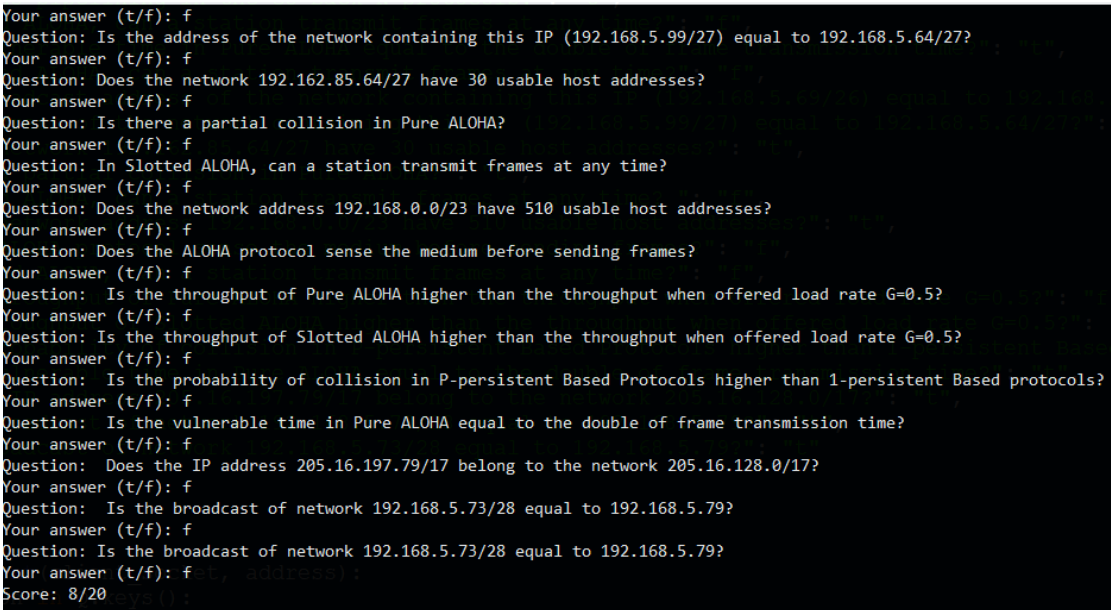
يتم التعامل مع بعض الأخطاء المحتملة أثناء التواصل مع السيرفر مثل انقطاع الاتصال أو حدوث خطأ في الاتصال.

عند انتهاء التواصل مع السيرفر، يتم إغلاق الاتصال بالسيرفر.

عند تشغيل البرنامج، يتم تنفيذ الدالة start_client ليتصل العميل بالسيرفر ويشارك في المسابقة.



```
Your answer (t/f): t
Question: Is the address of the network containing this IP (192.168.5.99/27) equal to 192.168.5.64/27?
Your answer (t/f): t
Question: Does the network 192.162.85.64/27 have 30 usable host addresses?
Your answer (t/f): t
Question: Is there a partial collision in Pure ALOHA?
Your answer (t/f): t
Question: In Slotted ALOHA, can a station transmit frames at any time?
Your answer (t/f): t
Question: Does the network address 192.168.0.0/23 have 510 usable host addresses?
Your answer (t/f): t
Question: Does the ALOHA protocol sense the medium before sending frames?
Your answer (t/f): t
Question:  Is the throughput of Pure ALOHA higher than the throughput when offered load rate G=0.5?
Your answer (t/f): t
Question: Is the throughput of Slotted ALOHA higher than the throughput when offered load rate G=0.5?
Your answer (t/f): t
Question:  Is the probability of collision in P-persistent Based Protocols higher than 1-persistent Based protocols?
Your answer (t/f): t
Question:  Is the vulnerable time in Pure ALOHA equal to the double of frame transmission time?
Your answer (t/f): t
Question:  Does the IP address 205.16.197.79/17 belong to the network 205.16.128.0/17?
Your answer (t/f): t
Question:  Is the broadcast of network 192.168.5.73/28 equal to 192.168.5.79?
Your answer (t/f): t
Question: Is the broadcast of network 192.168.5.73/28 equal to 192.168.5.79?
Your answer (t/f): t
Score: 12/20
```



```
Your answer (t/f): f
Question: Is the address of the network containing this IP (192.168.5.99/27) equal to 192.168.5.64/27?
Your answer (t/f): f
Question: Does the network 192.162.85.64/27 have 30 usable host addresses?
Your answer (t/f): f
Question: Is there a partial collision in Pure ALOHA?
Your answer (t/f): f
Question: In Slotted ALOHA, can a station transmit frames at any time?
Your answer (t/f): f
Question: Does the network address 192.168.0.0/23 have 510 usable host addresses?
Your answer (t/f): f
Question: Does the ALOHA protocol sense the medium before sending frames?
Your answer (t/f): f
Question:  Is the throughput of Pure ALOHA higher than the throughput when offered load rate G=0.5?
Your answer (t/f): f
Question: Is the throughput of Slotted ALOHA higher than the throughput when offered load rate G=0.5?
Your answer (t/f): f
Question:  Is the probability of collision in P-persistent Based Protocols higher than 1-persistent Based protocols?
Your answer (t/f): f
Question:  Is the vulnerable time in Pure ALOHA equal to the double of frame transmission time?
Your answer (t/f): f
Question:  Does the IP address 205.16.197.79/17 belong to the network 205.16.128.0/17?
Your answer (t/f): f
Question:  Is the broadcast of network 192.168.5.73/28 equal to 192.168.5.79?
Your answer (t/f): f
Question: Is the broadcast of network 192.168.5.73/28 equal to 192.168.5.79?
Your answer (t/f): f
Score: 8/20
```

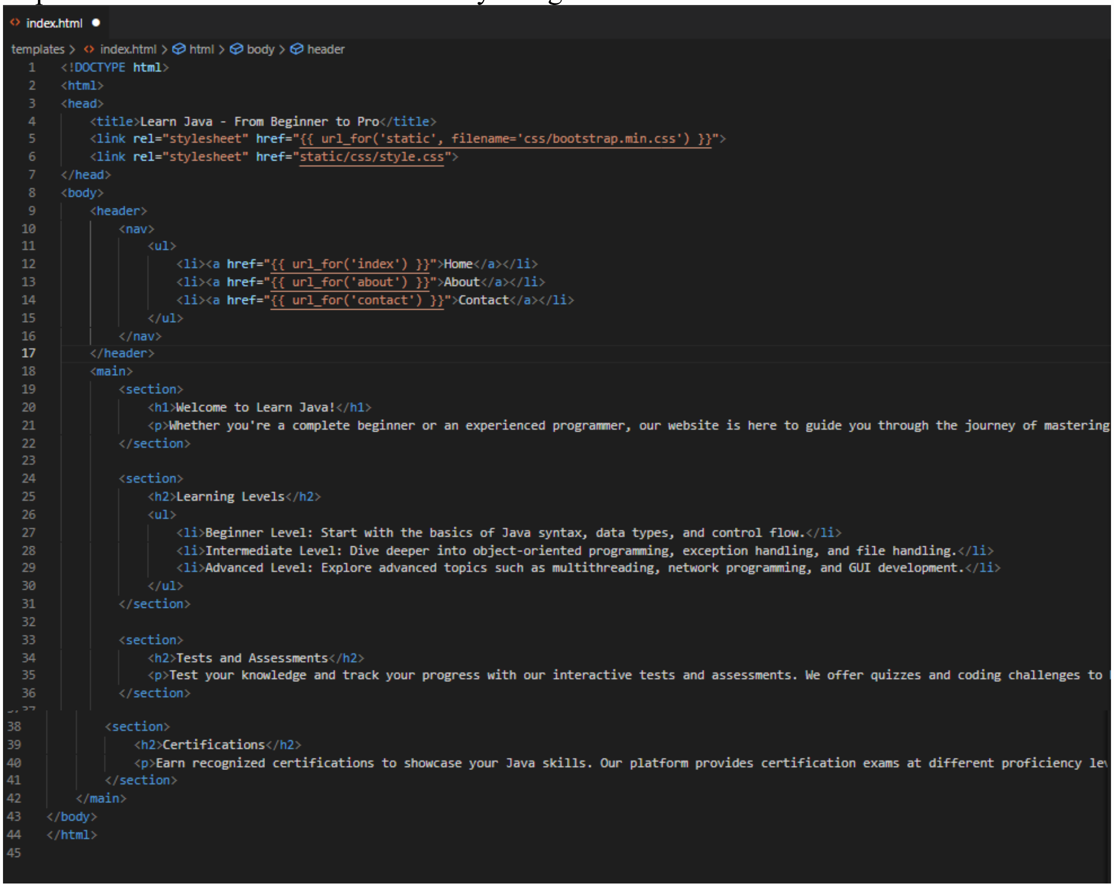
Name:_ __________, Number:_ ___________, Submitted To GitHub:_ _____

Question 2: Simple Website with Python Flask Framework

Create a simple website with multiple pages using Flask, HTML, CSS, and Bootstrap. The website should demonstrate your understanding of web design principles.

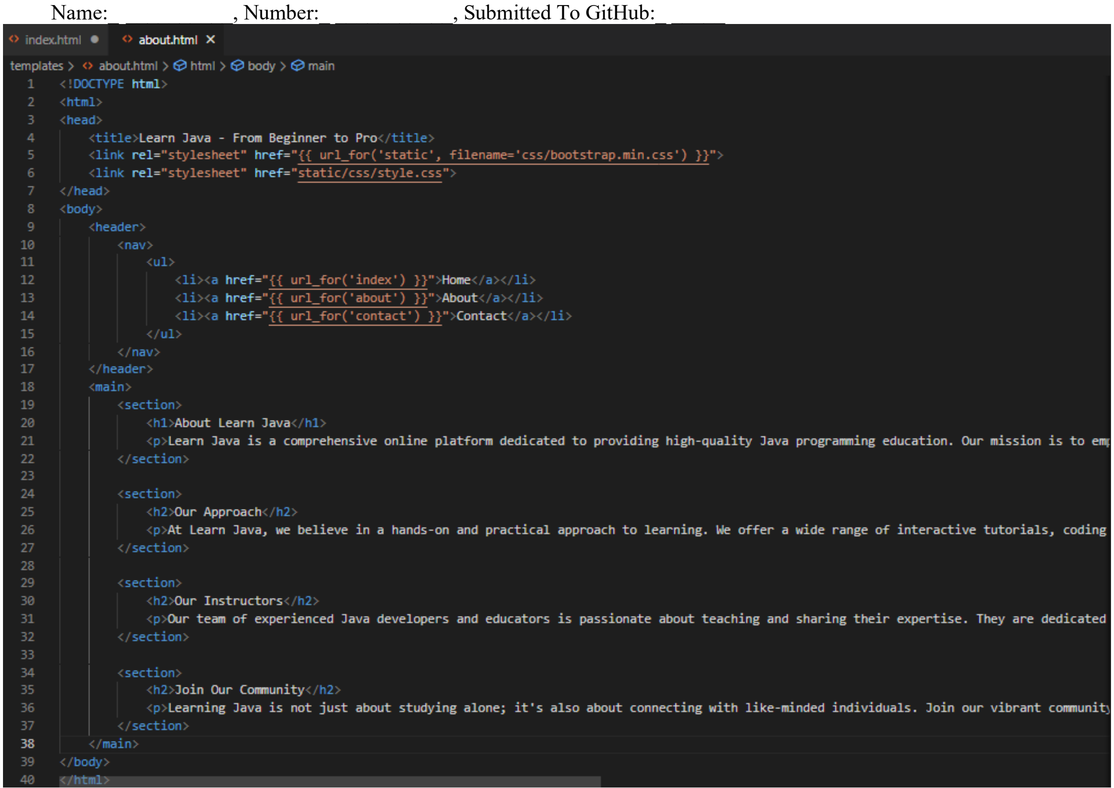
Requirements:

A. Set up a local web server using XAMPP, IIS, or Python's built-in server (using Flask).
B. Apply CSS and Bootstrap to style the website and make it visually appealing.
C. Ensure that the website is responsive and displays correctly on different screen sizes.
D. Implement basic server-side functionality using Flask to handle website features.

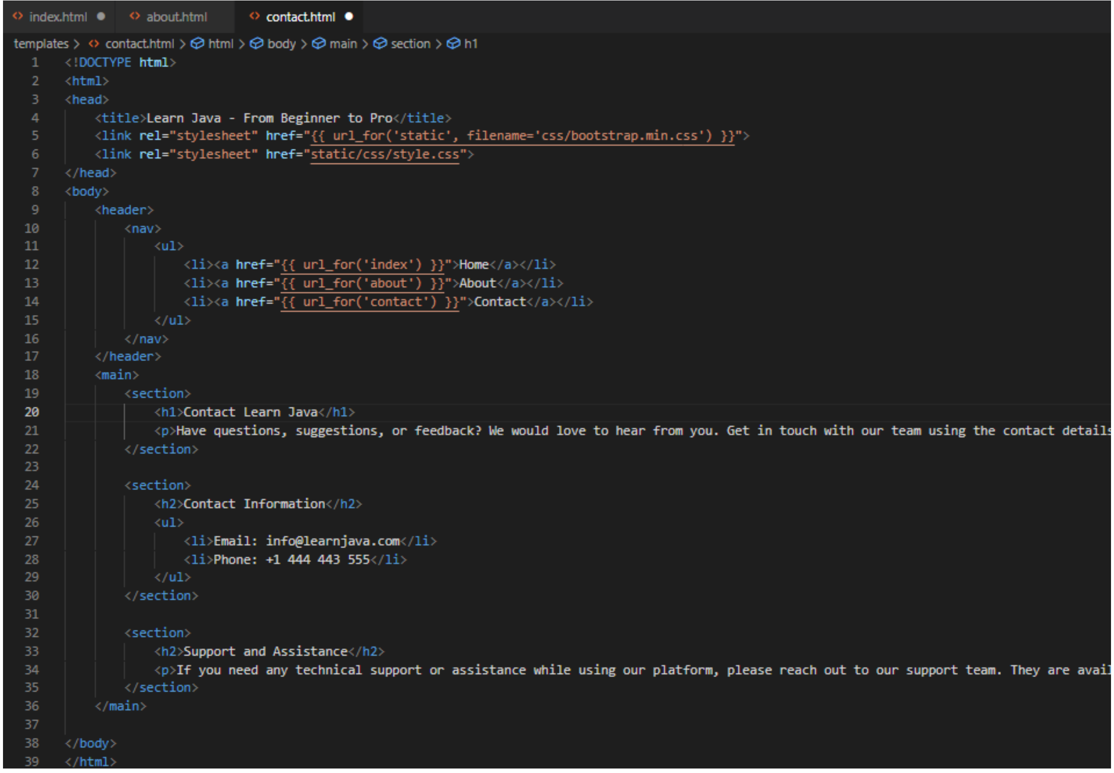


```html
<!DOCTYPE html>
<html>
<head>
    <title>Learn Java - From Beginner to Pro</title>
    <link rel="stylesheet" href="{{ url_for('static', filename='css/bootstrap.min.css') }}">
    <link rel="stylesheet" href="static/css/style.css">
</head>
<body>
    <header>
        <nav>
            <ul>
                <li><a href="{{ url_for('index') }}">Home</a></li>
                <li><a href="{{ url_for('about') }}">About</a></li>
                <li><a href="{{ url_for('contact') }}">Contact</a></li>
            </ul>
        </nav>
    </header>
    <main>
        <section>
            <h1>Welcome to Learn Java!</h1>
            <p>Whether you're a complete beginner or an experienced programmer, our website is here to guide you through the journey of mastering
        </section>

        <section>
            <h2>Learning Levels</h2>
            <ul>
                <li>Beginner Level: Start with the basics of Java syntax, data types, and control flow.</li>
                <li>Intermediate Level: Dive deeper into object-oriented programming, exception handling, and file handling.</li>
                <li>Advanced Level: Explore advanced topics such as multithreading, network programming, and GUI development.</li>
            </ul>
        </section>

        <section>
            <h2>Tests and Assessments</h2>
            <p>Test your knowledge and track your progress with our interactive tests and assessments. We offer quizzes and coding challenges to
        </section>

        <section>
            <h2>Certifications</h2>
            <p>Earn recognized certifications to showcase your Java skills. Our platform provides certification exams at different proficiency lev
        </section>
    </main>
</body>
</html>
```

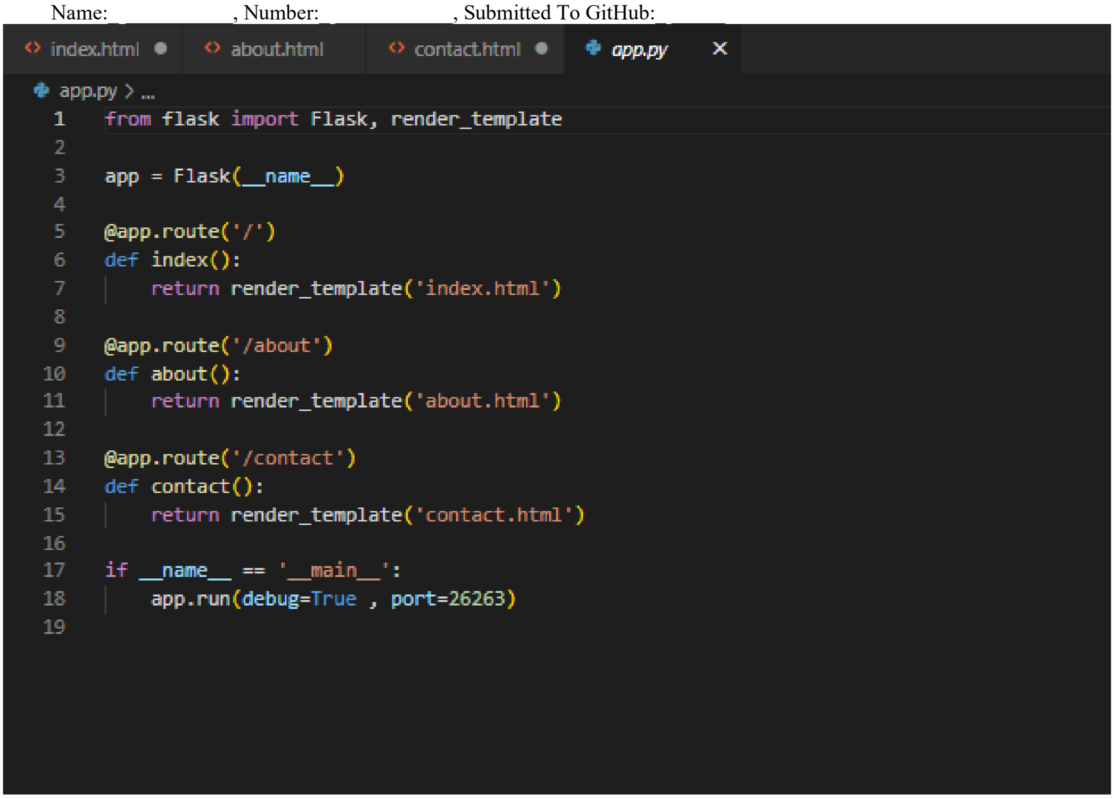
Name: ____________, Number: ____________, Submitted To GitHub: ______



```html
<!DOCTYPE html>
<html>
<head>
    <title>Learn Java - From Beginner to Pro</title>
    <link rel="stylesheet" href="{{ url_for('static', filename='css/bootstrap.min.css') }}">
    <link rel="stylesheet" href="static/css/style.css">
</head>
<body>
    <header>
        <nav>
            <ul>
                <li><a href="{{ url_for('index') }}">Home</a></li>
                <li><a href="{{ url_for('about') }}">About</a></li>
                <li><a href="{{ url_for('contact') }}">Contact</a></li>
            </ul>
        </nav>
    </header>
    <main>
        <section>
            <h1>About Learn Java</h1>
            <p>Learn Java is a comprehensive online platform dedicated to providing high-quality Java programming education. Our mission is to emp
        </section>

        <section>
            <h2>Our Approach</h2>
            <p>At Learn Java, we believe in a hands-on and practical approach to learning. We offer a wide range of interactive tutorials, coding
        </section>

        <section>
            <h2>Our Instructors</h2>
            <p>Our team of experienced Java developers and educators is passionate about teaching and sharing their expertise. They are dedicated
        </section>

        <section>
            <h2>Join Our Community</h2>
            <p>Learning Java is not just about studying alone; it's also about connecting with like-minded individuals. Join our vibrant community
        </section>
    </main>
</body>
</html>
```

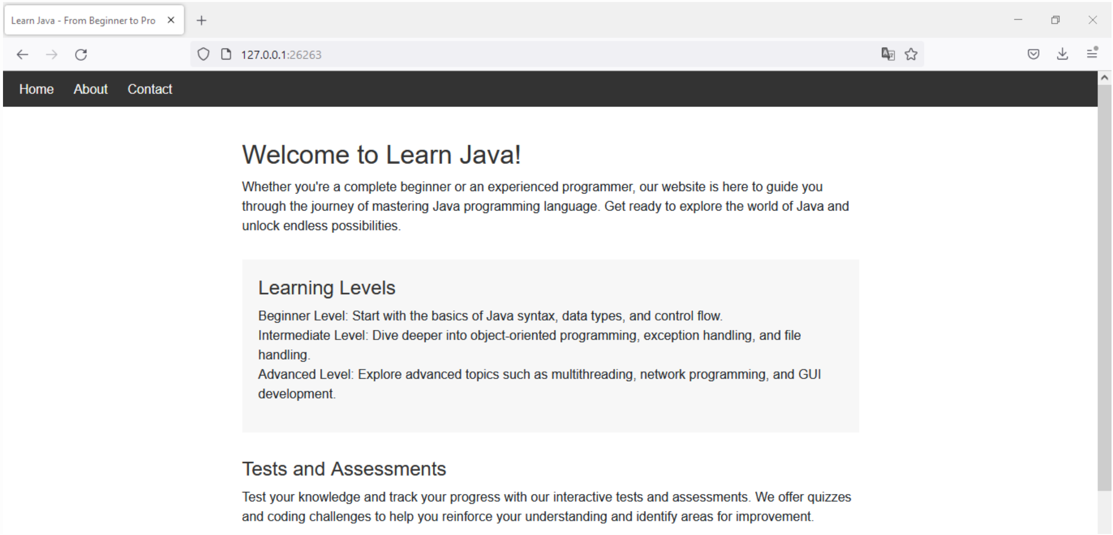

index.html ●
about.html
contact.html ●
templates > contact.html > html > body > main > section > h1

```html
<!DOCTYPE html>
<html>
<head>
    <title>Learn Java - From Beginner to Pro</title>
    <link rel="stylesheet" href="{{ url_for('static', filename='css/bootstrap.min.css') }}">
    <link rel="stylesheet" href="static/css/style.css">
</head>
<body>
    <header>
        <nav>
            <ul>
                <li><a href="{{ url_for('index') }}">Home</a></li>
                <li><a href="{{ url_for('about') }}">About</a></li>
                <li><a href="{{ url_for('contact') }}">Contact</a></li>
            </ul>
        </nav>
    </header>
    <main>
        <section>
            <h1>Contact Learn Java</h1>
            <p>Have questions, suggestions, or feedback? We would love to hear from you. Get in touch with our team using the contact details
        </section>

        <section>
            <h2>Contact Information</h2>
            <ul>
                <li>Email: info@learnjava.com</li>
                <li>Phone: +1 444 443 555</li>
            </ul>
        </section>

        <section>
            <h2>Support and Assistance</h2>
            <p>If you need any technical support or assistance while using our platform, please reach out to our support team. They are avai
        </section>
    </main>

</body>
</html>
```



```python
from flask import Flask, render_template

app = Flask(__name__)

@app.route('/')
def index():
    return render_template('index.html')

@app.route('/about')
def about():
    return render_template('about.html')

@app.route('/contact')
def contact():
    return render_template('contact.html')

if __name__ == '__main__':
    app.run(debug=True , port=26263)

```

نشغل الكود السابق بعد ذلك نتوجه للمتصفح ونضع به العنوان 127.0.0.1:26263



Welcome to Learn Java!

Whether you're a complete beginner or an experienced programmer, our website is here to guide you through the journey of mastering Java programming language. Get ready to explore the world of Java and unlock endless possibilities.

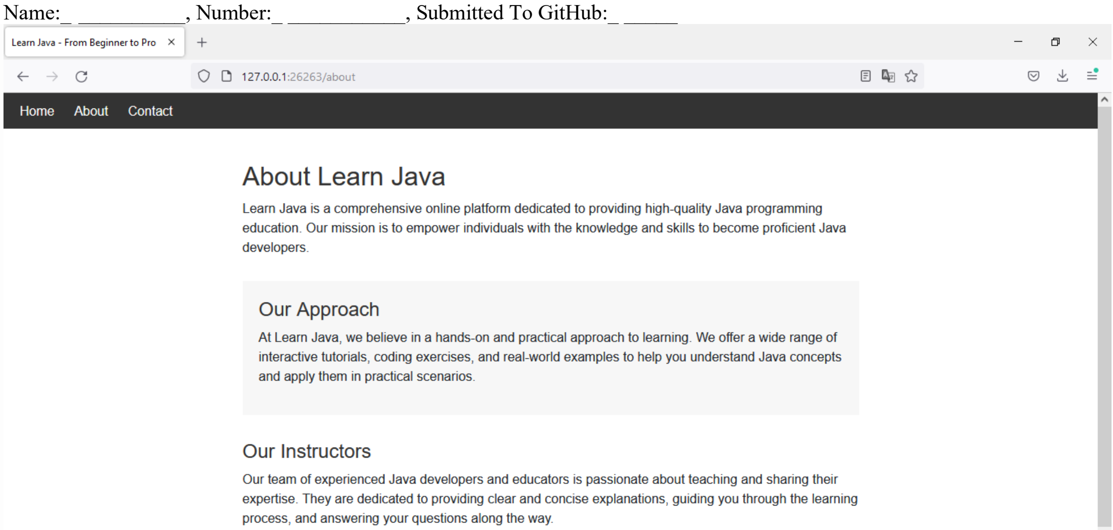
Learning Levels

Beginner Level: Start with the basics of Java syntax, data types, and control flow.
Intermediate Level: Dive deeper into object-oriented programming, exception handling, and file handling.
Advanced Level: Explore advanced topics such as multithreading, network programming, and GUI development.

Tests and Assessments

Test your knowledge and track your progress with our interactive tests and assessments. We offer quizzes and coding challenges to help you reinforce your understanding and identify areas for improvement.

Name:_ __________, Number:_ ___________, Submitted To GitHub:_ _____



About Learn Java

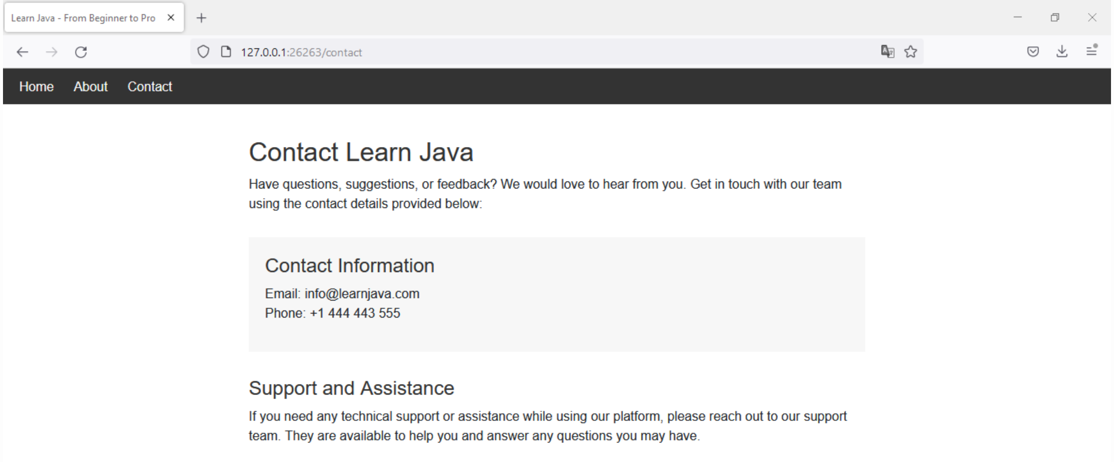
Learn Java is a comprehensive online platform dedicated to providing high-quality Java programming education. Our mission is to empower individuals with the knowledge and skills to become proficient Java developers.

Our Approach

At Learn Java, we believe in a hands-on and practical approach to learning. We offer a wide range of interactive tutorials, coding exercises, and real-world examples to help you understand Java concepts and apply them in practical scenarios.

Our Instructors

Our team of experienced Java developers and educators is passionate about teaching and sharing their expertise. They are dedicated to providing clear and concise explanations, guiding you through the learning process, and answering your questions along the way.



Contact Learn Java

Have questions, suggestions, or feedback? We would love to hear from you. Get in touch with our team using the contact details provided below:

Contact Information

Email: info@learnjava.com
Phone: +1 444 443 555

Support and Assistance

If you need any technical support or assistance while using our platform, please reach out to our support team. They are available to help you and answer any questions you may have.

Name:_ __________, Number:_ ___________, Submitted To GitHub:_ _____