



الجمهورية العربية السورية
وزارة التعليم العالي و البحث العلمي
جامعة تشرين
الهندسة الميكانيكية و الكهربائية
قسم هندسة الاتصالات و الالكترونيات

Network Programming Homework 2 وظيفة برمجة الشبكات الثانية

الاسم : حسين محمد اسماعيل

الرقم : 2708

بإشراف : د. مهدي عيسى

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 3rd-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.

حل السؤال الأول:

في البداية نقوم بتضمين المكتبات اللازمة ألا و هي:

- Socket
- Threading
- Json

من بعدها تم كتابة تابع بالاسم (handle_client) و الذي يتعامل مع المستخدمين عبر ال threads يطلب من المستخدم إدخال اسمه لأننا سوف نحتاجه في حفظ العلامة و الاسم على السيرفر في ملف العلامات.

تقوم حلقة for بالمرور على الأسئلة الموجودة ضمن ملف questions.json حيث يتم فتح الملف قبلا و تحميله ضمن متحول من النوع dictionary .

تم استعمال try و except للتعامل مع أخطاء الإرسال

يتم زيادة العلامة مع كا إجابة صحيحة.

بعد انتهاء الحلقة يتم إرسال كلمة end و التي تعلن نهاية الاختبار للمستخدم ليتم إعلان النتيجة للمستخدم و فصل ال socket بعدها.

من ثم يتم تخزين العلامة و الاسم في ملف grades.json .

بعد التابع نقوم بتعريف المتغيرات كالمضيف و المنفذ و متحول لاسم المستخدم.

ننشئ socket السيرفر و نربطها بالعنوان المطلوب .

يبدأ السيرفر بالاستماع.

يتم قبول اتصال المستخدم و إنشاء thread خاص به و طباعة ال threads النشطة.

أما لدى المستخدم يتم الآتي:

نضيف مكتبة socket و ننشئها للمستخدم، ثم نحاول الاتصال بالسيرفر و الإجابة على الأسئلة حتى تأتي كلمة end

في النهاية يتم استلام النتيجة و فصل ال socket.

```
1 import socket
2 import threading
3 import json
4
5 def handle_client(sock, addr):
6     sock.send("insert user name : ".encode())
7     user_name = sock.recv(1024).decode()
8     print(f'''+ accepting connection
9         |         address = {addr}
10        |         name = {user_name}''')
11     grade = 0
12     for i in range(1,21):
13         question = questions[str(i)][0]
14         answer = questions[str(i)][1]
15         try :
16             sock.send(f"question {i}: {question}".encode())
17             student_answer = sock.recv(1024).decode()
18             if answer == student_answer :
19                 grade += 1
20         except :
21             print("Socket Error")
22     sock.send("end".encode())
23     sock.send(f"the deserved grade is {grade} !".encode())
24     print(f"[-] tcp connection for {user_name} is closed")
25     sock.close()
26     try:
27         with open("grades.json","a") as grades:
28             grades.write(f"{user_name} : {grade}")
29     except:
30         with open("grades.json","w") as grades:
31             grades.write(f"{user_name} : {grade}")
```

```

32 user_name, host, port = "", "127.0.0.1", "54321"
33
34 server_sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
35 server_sock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
36 try :
37     server_sock.bind(('127.0.0.1', 54321))
38 except socket.error as err_msg:
39     print ("error : ", err_msg)
40     sys.exit(1)
41 server_sock.listen()
42 while True:
43     print(f"[+] tcp quiz sever running")
44     client_sock, client_addr = server_sock.accept()
45     thread = threading.Thread(target=handle_client, args=(client_sock, client_addr))
46     thread.start()
47     print(f'[ACTIVE CONNECTIONS] {threading.active_count() - 1}')
48 server_sock.close()
49

```

server.py

```

1  import socket
2
3  csock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
4  csock.settimeout(5)
5  try:
6      csock.connect(('127.0.0.1', 54321))
7      while True:
8          servermsg = csock.recv(1024).decode()
9          if servermsg == "end":
10             break
11             print(servermsg)
12             csock.sendall(input("").encode())
13 except socket.error as err_msg:
14     print("error msg :", err_msg)
15
16 finally:
17     grade = csock.recv(1024).decode()
18     print(grade)
19     csock.close()

```

client1.py & client2.py

بعد التشغيل

```
PS C:\Users\hus0a\Desktop> & 'C:\Users\hus0a\AppData\Local\Programs\Python\Python310\python.exe' 'c:\Users\hus0a\.vscode\extensions\ms-pythonFiles\lib\python\debugpy\adapter\..\..\debugpy\launcher' '62503' '--' 'c:\Users\hus0a\Desktop\server_multi.py'
[+] tcp quiz sever running
[ACTIVE CONNECTIONS] 1
[+] tcp quiz sever running
[+] accepting connection
    address = ('127.0.0.1', 62613)
    name = husayn
[ACTIVE CONNECTIONS] 2
[+] tcp quiz sever running
[+] accepting connection
    address = ('127.0.0.1', 62894)
    name = hazem
[-] tcp connection for hazem is closed
[-] tcp connection for husayn is closed
[]
```

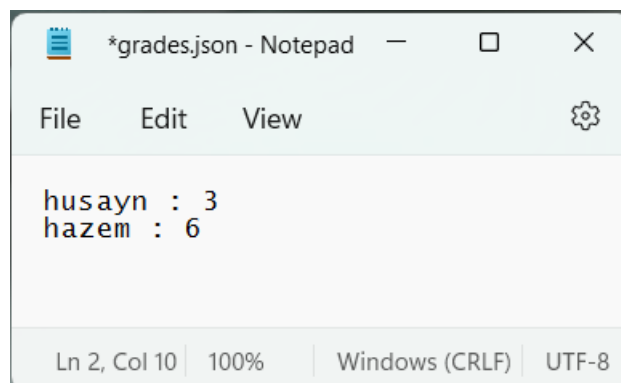
server

```
-
question 18: can we devide by zero?
2
question 19: 0 celecius is what in fehrenhite?
2
question 20: 7*2?
14
the deserved grade is 3 !
PS C:\Users\hus0a\Desktop> []
```

client1

```
←
question 20: 7*2?
2
the deserved grade is 6 !
PS C:\Users\hus0a\Desktop> []
```

client2



grades.json

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:

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

الحل:

الفكرة إنشاء موقع لعرض تصاميمي الشخصية و استضافته على flask server على جهازه المحلي و كذلك تم إضافة ميزة للتمييز بين صور القطط و الكلاب كونها تصب في فكرة مشروع التخرج الخاص بي. تم إنشاء البيئة الافتراضية في مجلد اسمه web و تسميتها venv من خلال التالي

```
mkdir Web
```

```
python -m venv venv
```

تفعيل البيئة الافتراضية:

```
venv\Scripts\activate
```

تم إنشاء ملف داخل venv بالاسم app

و إنشاء الملفات الآتية:

- Static
- Templates
- Webpage.py

في الملف static يتم تخزين الملفات الثابتة كالصور و ملفات bootstrap و المودل الخاص بالتعرف على القطط و الكلاب بالاسم catordog.h5

في الملف templates يتم تخزين صفحات الويب بصيغة html

Base.html يتم تخزين قطع html المكررة بين الصفحات مثل شريط التنقل و تضمين bootstrap و ال footer لتجنب كتابتها اكثر من مرة و تضمينها بالصفحات الأخرى عن طريق الوراثة.

Index.html الصفحة الرئيسية و التي يتم فيها عرض الأعمال

About.html معلومات حول الأعمال

Contact.html معلومات التواصل

Predict.py للتمييز بين القطط و الكلاب

في الملف webpage.py يتم كتابة كود البايثون لتشغيله على سيرفر فلاسك

```
1 from flask import Flask, render_template, url_for, request, jsonify
2 import tensorflow as tf
3 import numpy as np
4 from PIL import Image
5 app = Flask(__name__)
6 model = tf.keras.models.load_model("static/catordog.h5")
7 classes = ['cat', 'dog']
8
9 @app.route('/')
10 def index():
11     return render_template("index.html")
12
13 @app.route('/about')
14 def about():
15     return render_template("about.html")
16
17 @app.route('/conatcts')
18 def contacts():
19     return render_template("contacts.html")
20
21 @app.route('/upload', methods = ['POST', 'GET'])
22 def predict():
23     if request.method == 'GET':
24         return render_template("predict.html")
25     else:
26         file = request.files['file']
27         image = Image.open(file)
28         resized_image = image.resize((256,256))
29         image = np.array(resized_image)
30         image = image.astype('float32')/255.0
31         image = np.expand_dims(image, axis=0)
```

```

32
33     prediction = model.predict(image)
34     if prediction < 0.5:
35         predicted_class = "\U0001F431"
36     else :
37         predicted_class = "\U0001F436"
38     print("predict = ",predicted_class)
39     return render_template("result.html",result=predicted_class)
40
41
42 if __name__ == "__main__":
43     app.run(debug= True)

```

webpage.py

المكتبات :

- Tensorflow
- Numpy
- PIL
- Flask
- Render_template
- url_for
- request
- jsonify

tensorflow للتعلم العميق باستخدام بايثون

numpy للتعامل مع المصفوفات الخاصة قسم التعلم العميق أثناء معالجة الصورة قبيل تمريرها إلى المودل

PIL للتعامل مع الملفات المختلفة و هنا للصور.

url_for للحصول على عناوين الملفات المختلفة ضمن السيرفر

Request للتعامل مع الاتصالات المختلفة

Jsonify لتحويل النتيجة إلى صيغة json.

في البداية نعمل instance من فلاسك و هو app و نقوم بتحميل المودل catordog.h5 و ما يليه هو عبارة عن تعليمات التوجيه routing و توابعها .

التابع predict :

تابع له نوعين من الاتصال GET و POST :

GET يستخدم لتحميل صفحة الويب إلى المتصفح أما POST فيستخدم عندما يرسل المستخدم صورة إلى السيرفر لمعالجتها.

حيث تتم معالجة أولية للصورة كي تناسب المودل ومن ثم يتم تمريرها للمودل لكي يقوم بالتنبؤ يتم إرسال النتيجة إلى result.html لتعرض للمستخدم بشكل جميل و منسق.

تم استعمال الوراثة بين صفحات .html

سوف نستعرض بعض الأكواد كمثال ثم نستعرض الصفحة الأساسية:

```
<> base.html X
templates > <> base.html
27      <a class="nav-link" href="{{ url_for('ab
28      </li>
29      <li class="nav-item">
30      |   <a class="nav-link" href="{{ url_for('co
31      </li>
32      <li class="nav-item">
33      |   <a class="nav-link" href="{{ url_for('pr
34      </li>
35      </ul>
36      </div>
37      </div>
38      </nav>
39      {% block content%}{% endblock %}
40      <script src="{{ url_for('static' filename='js/b
41      </body>
```

Snapshot from base.html

تعليلة block تتيح لنا هذه التعليلة ترك مكان فارغ في ال base.html يتم ملؤها بمختلف محتويات الصفحات.

<> result.html X

templates > <> result.html

```
1 {% extends 'base.html' %}
2 {% block content %}
3 <div class="container-fluid" align="center">
4     <h1>Result</h1>
5     <h1>The uploaded image is a {{ result }}</h1>
6 </div>
7 {% endblock %}
```

result.html

البلوك الذي يعبأ مكان الفراغ الذي تركناه في base.html

تعليقات الوراثة يمكن ملاحظة تعليمة extends

H.Ismail Designs Home about contacts predict

Create a powerful brand identity with our professional logo design services

Your logo is the face of your business



STACIE JO
DESIGNS

DESAFE

simple & elegant

logo designs

Why do you need a logo?

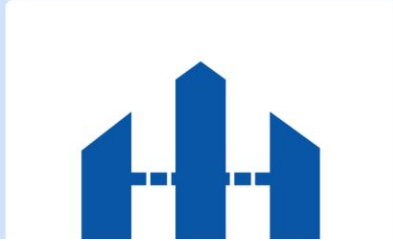
Logo design is a crucial aspect of branding for any business or organization. A well-designed logo can instantly communicate a company's values, personality, and unique selling proposition to its audience.

A logo should be simple, memorable, and visually appealing, with a design that is both timeless and relevant to the brand. It should also be versatile, able to be used across various mediums and sizes without losing its impact.

A good logo designer will have a deep understanding of color theory, typography, and composition, and will work closely with the client to ensure that the final design accurately reflects the brand's identity and goals. Ultimately, a strong logo can help to establish a brand's credibility, build trust with customers, and differentiate it from competitors in a crowded marketplace.

Why do you need a business card?

Designing a business card is an essential part of creating a professional image for your business. A well-designed business card serves as a tangible representation of your brand and helps to establish credibility and legitimacy with potential clients or customers. It's a marketing tool that can help to make a memorable first impression and set you apart from competitors. A business card is also a convenient way to share your contact information, making it easy for people to get in touch with you after a meeting or networking event. A good design will showcase your brand's personality and values, and can be customized to fit your specific business needs. Overall, investing in a well-designed business card is a smart and effective way to promote your business and leave a lasting impression on potential customers.



 H.Ismail Designs [Home](#) [about](#) [contacts](#) [predict](#)

About us

Name:

Husayn Ismail

gender:

Male

age:

23 years old

studies:

communications and electronics engineering at Tishreen university

About the service

Welcome to our logo design service! We specialize in creating unique and memorable logos that will help your business stand out from the competition. Our team of talented designers is passionate about creating custom logos that capture the essence of your brand and communicate your values and mission to your target audience.

We understand that a logo is the face of your company, and we take pride in designing logos that represent the personality and identity of your brand. Whether you need a new logo from scratch or want to update your current one, we are here to help you create a logo that will leave a lasting impression on your customers.

Our logo design process is collaborative and transparent. We work closely with our clients to understand their vision and goals and provide multiple design concepts to choose from. We then refine the chosen concept until it meets your exact specifications and expectations. Our goal is to create a logo that you will love and that will help your business succeed.

At our logo design service, we believe that every business deserves a great logo, regardless of their size or budget. That's why we offer affordable and flexible pricing options to fit your needs. We are committed to delivering high-quality logos that will help your business grow and thrive.

Thank you for considering our logo design service. We look forward to working with you to create a logo that will help your business achieve its goals.



Cat or Dog classifier



Upload Image

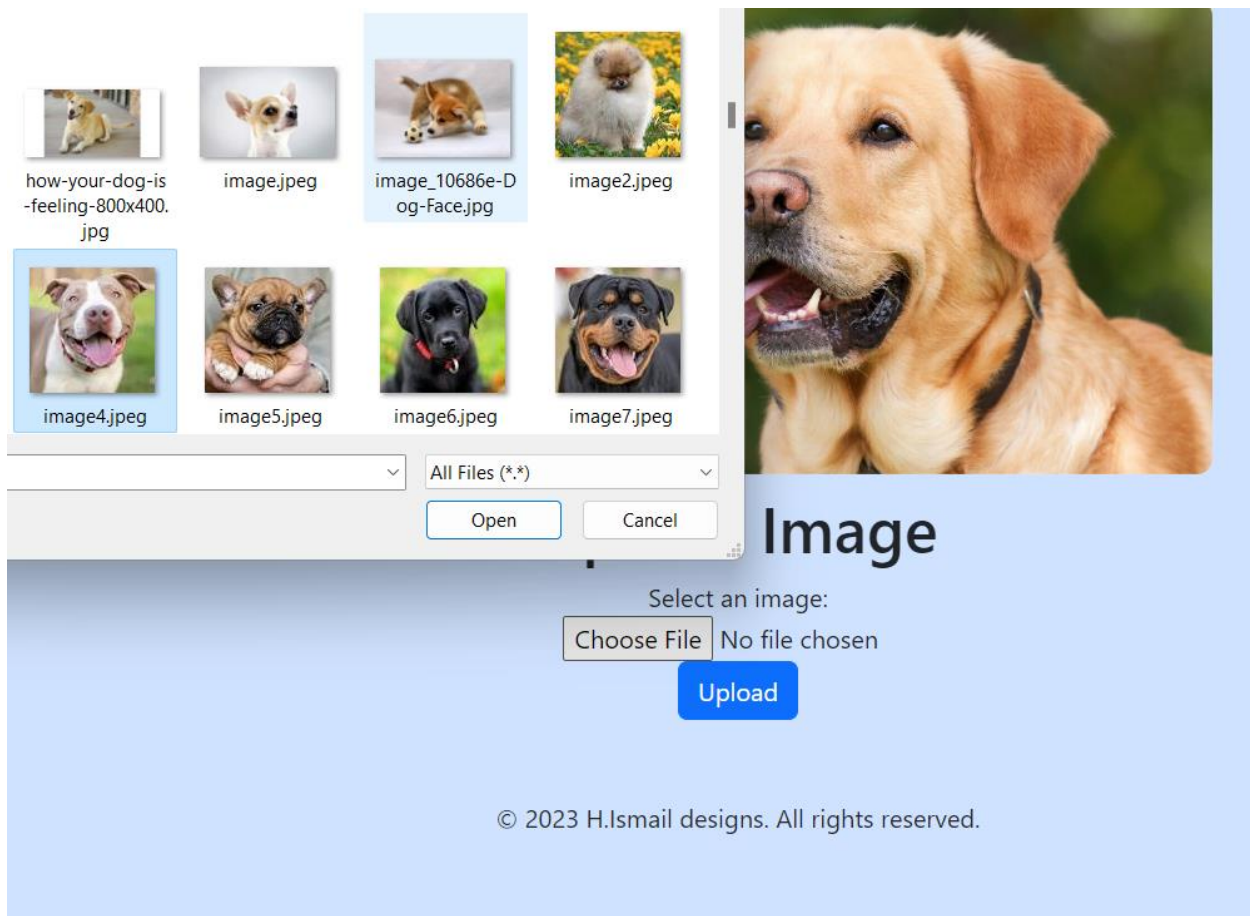
Select an image:

Choose File

No file chosen

Upload

© 2023 H.Ismail designs. All rights reserved.

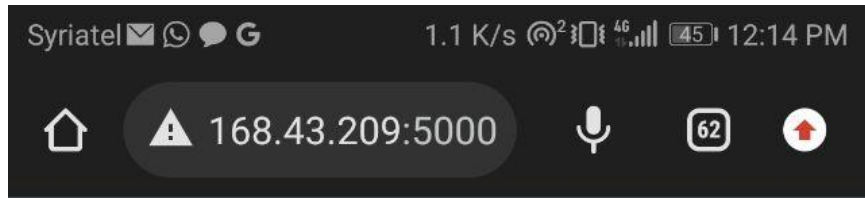


predict

Result

The uploaded image is a 🐶

© 2023 H.Ismail designs. All rights reserved.



 H.Ismail Designs



Create a powerful
brand identity with our
professional logo
design services

Your logo is the face of your business





انتهت الوظيفة