



الرقم الجامعي: ١٥٩٠
الرقم الجامعي: ١٠٧١

الاسم: علاء شهدت مصطفى
الاسم: علي غازي زيني

Question 1

السيرفر

1- استيراد المكتبات الأساسية:

- socket لإنشاء اتصالات الشبكة.
- threading لإنشاء ثريدات لمعالجة اتصالات متعددة من العملاء بشكل متزامن.

2- تعريف قائمة الحسابات البنكية:

قائمة تحتوي على حسابات بنكية محددة مسبقاً، كل حساب يحتوي على الاسم، كلمة المرور، والرصيد.

3- تابع للعثور على حساب حسب الاسم وكلمة المرور:

يقوم هذا التابع بالبحث في قائمة الحسابات عن حساب يطابق الاسم وكلمة المرور المدخلين.

4- تابع للتعامل مع كل عميل:

1. التحقق من الهوية:

- يطلب من العميل إدخال اسم الحساب وكلمة المرور.
- يتم التحقق من صحة البيانات المدخلة باستخدام التابع find_account.

2. التعامل مع العمليات المصرفية:

- بعد التحقق بنجاح، يتم عرض قائمة بالخيارات المصرفية للعميل (فحص الرصيد، إيداع، سحب، إنهاء).
- يتم تنفيذ العملية المطلوبة وإرسال النتيجة إلى العميل.

```
import socket
import threading
```

```
# مسبقاً المحددة البنكية الحسابات قائمة
bank_accounts = [
```

```

{"name": "Ali", "password": "123", "balance": 5000},
{"name": "Alaa", "password": "1234", "balance": 1000},
{"name": "Isaa", "password": "12345", "balance": 7500}
]

```

المرور وكلمة الاسم حسب حساب على للعثور تابع

```

def find_account(name, password):
    for account in bank_accounts:
        if account["name"] == name and account["password"] == password:
            return account
    return None

```

عميل كل مع للتعامل تابع

```

def handle_client(client_socket, client_address):
    print(f"متصل جديد عميل {client_address}")

```

```

    authenticated = False
    account = None

```

الهوية من التحقق حلقة

```

while not authenticated:
    client_socket.send(("بمسافة مفصولين) المرور وكلمة الحساب اسم أدخل").encode())
    credentials = client_socket.recv(1024).decode().strip().split()

    if len(credentials) == 2:
        name, password = credentials
        account = find_account(name, password)
        if account:
            authenticated = True
            client_socket.send("بنجاح التحقق تم.\n".encode())
        else:
            client_socket.send("حاول .صحيحة غير المرور كلمة أو الحساب اسم\n".encode())
            client_socket.send("أخرى مرة\n".encode())
    else:
        client_socket.send("أخرى مرة حاول .صحيح غير الإدخال\n".encode())

```

العمليات حلقة

```

while True:
    client_socket.send("4: سحب، 3: إيداع، 2: الرصيد، فحص: 1) خيارًا اختر:").encode())
    option = client_socket.recv(1024).decode().strip()

    if option == "1":
        balance = account["balance"]
        client_socket.send(f"هو رصيدك: {balance:.2f}\n".encode())
    elif option == "2":
        client_socket.send("للإيداع المبلغ أدخل: ").encode())
        amount = float(client_socket.recv(1024).decode().strip())
        account["balance"] += amount
        client_socket.send(f"هو الجديد الرصيد .بنجاح الإيداع تم: {account['balance']:.2f}\n".encode())
    elif option == "3":

```

```

client_socket.send("للسحب المبلغ أدخل: ".encode())
amount = float(client_socket.recv(1024).decode().strip())
if amount > account["balance"]:
    client_socket.send("كافية أموال توجد لا".encode())
else:
    account["balance"] -= amount
    client_socket.send(f"هو الجديد الرصيد .بنجاح السحب تم{amount}").encode()
{account['balance']:.2f}\n".encode())
    elif option == "4":
        client_socket.send(f"هو النهائي الرصيد{amount}").encode()
{account['balance']:.2f}\n!وداعاً\n".encode())
        break
    else:
        client_socket.send("أخرى مرة حاول .صحيح غير الخيار".encode())

client_socket.close()
print(f"الاتصال قطع تم {client_address} من العميل")

# الخادم وتشغيل إنشاء
server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
server_socket.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
server_socket.bind(('0.0.0.0', 5555))
server_socket.listen()

print("العملاء اتصالات انتظار في .بدأ الخادم")

while True:
    client_socket, client_address = server_socket.accept()
    client_thread = threading.Thread(target=handle_client,
args=(client_socket, client_address))
    client_thread.start()

```

العميل

1. استيراد مكتبة socket:

- تستخدم لإنشاء اتصال شبكة بين العميل والخادم.

2. إنشاء مقبس العميل والاتصال بالخادم:

```
client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
```

```
client_socket.connect(('localhost', 5555))
```

- إنشاء المقابس:

• socket.AF_INET يشير إلى استخدام بروتوكول IPv4.

• socket.SOCK_STREAM يشير إلى استخدام بروتوكول TCP.

- الاتصال بالخادم:

- `client_socket.connect(('localhost', 5555)):` يتصل بالخادم الذي يعمل على نفس الجهاز (localhost) على المنفذ 5555.

3. التعامل مع الاتصال باستخدام حلقة: `try - finally`

- استقبال الرسائل من الخادم:
- `message = client_socket.recv(1024).decode():` أقصى 1024 بايت.
ينتظر استقبال رسالة من الخادم بعد
- `if not message: break:` إذا لم يتم استلام أي رسالة، يتم كسر الحلقة وإنهاء الاتصال.
- `print(message, end=""):` يعرض الرسالة المستلمة بدون إضافة سطر جديد.
- إرسال المدخلات إلى الخادم:
- `if "أدخل اسم الحساب وكلمة المرور" in message or "أدخل المبلغ" in message or "اختار" in message:` خيارًا
اختار
- يتحقق من نوع الرسالة المستلمة لمعرفة ما إذا كان يجب على العميل إدخال بيانات.
- `user_input = input():` يطلب من المستخدم إدخال البيانات المطلوبة.
- `client_socket.send(user_input.encode()):` يرسل المدخلات إلى الخادم بعد تحويلها إلى صيغة بايتات.
- إغلاق الاتصال:
- `finally: client_socket.close():` يضمن إغلاق المقبس بشكل صحيح بعد انتهاء الاتصال، سواء تم بشكل طبيعي أو بسبب حدوث خطأ.

```
import socket

client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
client_socket.connect(('localhost', 5555))

try:
    while True:
        message = client_socket.recv(1024).decode()
        if not message:
            break
        print(message, end="")

        if "المبلغ أدخل" in message or "المرور وكلمة الحساب أدخل" in message or "اختار" in message:
            user_input = input()
            client_socket.send(user_input.encode())

finally:
    client_socket.close()
```

الخروج:

السيرفر بعد اتصال العملاء ال3 وخروج احدهم

```
Run: server x c1 x c2 x c3 x
C:\pythonProject\venv\Scripts\python.exe C:\pythonProject\q1\server.py
... الخادم بدأ. في انتظار اتصالات العملاء...
(62798 , '127.0.0.1') عميل جديد متصل من
(62801 , '127.0.0.1') عميل جديد متصل من
(62804 , '127.0.0.1') عميل جديد متصل من
العميل من ('127.0.0.1' , 62798) تم قطع الاتصال.
```

العميل Ali

```
أدخل اسم الحساب وكلمة المرور (مفصولين بمسافة): Ali 123
تم التحقق بنجاح.
اختر خيارًا (1: فحص الرصيد، 2: إيداع، 3: سحب، 4: إنهاء): 2
أدخل المبلغ للإيداع: 4000
تم الإيداع بنجاح. الرصيد الجديد هو: 9000.00
اختر خيارًا (1: فحص الرصيد، 2: إيداع، 3: سحب، 4: إنهاء): 3
أدخل المبلغ للسحب: 1500
تم السحب بنجاح. الرصيد الجديد هو: 7500.00
اختر خيارًا (1: فحص الرصيد، 2: إيداع، 3: سحب، 4: إنهاء): 4
الرصيد النهائي هو: 7500.00
وداعًا!

Process finished with exit code 0
```

العميل Alaa

```
أدخل اسم الحساب وكلمة المرور (مفصولين بمسافة): Alaa 1234
تم التحقق بنجاح.
اختر خيارًا (1: فحص الرصيد، 2: إيداع، 3: سحب، 4: إنهاء): 2
أدخل المبلغ للإيداع: 10000
تم الإيداع بنجاح. الرصيد الجديد هو: 11000.00
اختر خيارًا (1: فحص الرصيد، 2: إيداع، 3: سحب، 4: إنهاء): 1
رصيدك هو: 11000.00
اختر خيارًا (1: فحص الرصيد، 2: إيداع، 3: سحب، 4: إنهاء): |
```

اختيارات التصميم:

1. وحدة Socket: استخدمنا الوحدة المدمجة socket في بايثون لإنشاء الخادم والعميل بنظام TCP بدون الاعتماد على أي حزم خارجية، مما يضمن الحل خفيف الوزن وسهل النشر.
2. تعدد المهام: استخدمنا وحدة threading لمعالجة اتصالات متعددة من العملاء في نفس الوقت. يتم إدارة كل اتصال عميل بواسطة ثريد منفصل، مما يسمح بالتفاعلات المتزامنة مع عدة عملاء.
3. التحقق من الهوية: يحتفظ الخادم بقائمة من الحسابات المحددة مسبقًا مع كلمات المرور والأرصدة. يجب على العملاء التحقق من هويتهم باستخدام تفاصيل الحساب قبل إجراء أي معاملات.
4. العمليات المصرفية: يدعم الخادم العمليات المصرفية الأساسية:
 - فحص الرصيد: يسترجع ويرسل الرصيد الحالي للعميل.
 - إيداع: يضيف مبلغًا محددًا إلى رصيد العميل.
 - سحب: يخصم مبلغًا محددًا من رصيد العميل إذا كانت الأموال كافية.
5. إشعار الرصيد النهائي في نهاية الجلسة، يرسل الخادم الرصيد النهائي للعميل قبل قطع الاتصال.

التحديات التي واجهتنا:

1. إدارة التزامن: معالجة اتصالات متعددة من العملاء في نفس الوقت تتطلب تصميمًا دقيقًا لتجنب ظروف السباق، خاصة عند تحديث أرصدة الحسابات. استخدمنا تعدد المهام ساعد في إدارة الاتصالات المتزامنة بكفاءة.
2. التعامل مع الأخطاء والتحقق من الإدخال: ضمان التعامل الجيد مع الأخطاء والتحقق من الإدخالات كان تحديًا. يجب على الخادم التعامل مع الإدخالات غير الصحيحة بلباقة وتقديم ردود واضحة للعميل.
3. اعتبارات الأمان: بينما تعمل آلية التحقق الأساسية لهذا المشروع، إلا أن الطرق الأكثر أمانًا (مثل التشفير وتخزين كلمات المرور بشكل آمن) ستكون ضرورية لحماية المعلومات الحساسة في سيناريوهات العالم الحقيقي.

Question 2:

يبدأ الكود بتعريف متغير app باستخدام كلاس Flask وتمرير name إليه كمعرف لتطبيقنا. ثم يتم تعريف ثلاثة توابع مختلفة باستخدام التوجيهات route لتحديد الصفحات المختلفة في التطبيق. يتم استدعاء تابع render_template في كل تابع لنقوم بتحميل ملف HTML المرتبط بكل صفحة. يتحقق التعبير if name == 'main': من أن التطبيق يعمل فقط عندما يتم تشغيله مباشرة من نفس المكان ويقوم بتشغيل التطبيق باستخدام تابع run الموجود في Flask.

```
from flask import Flask, render_template

app = Flask(__name__)

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

@app.route('/page2')
def page2():
    return render_template('page2.html')

@app.route('/page3')
def page3():
    return render_template('page3.html')

if __name__ == '__main__':
    app.run()
```

الصفحة الأولى:

هي ملف HTML يتم استخدامه في تطبيق Flask ويتضمن بعض العناصر المختلفة لتنسيق الصفحة ويحتوي الملف على

العناصر التالية:

<DOCTYPE html!>: هذا يحدد نوع المستند كـ HTML5.

<html> و <html/>: هذا يحدد بداية ونهاية الصفحة HTML.

<head> و <head/>: هذا يحدد بداية ونهاية ترويسة الصفحة وتحتوي على عناصر التعريف مثل عنوان الصفحة ونمط

.CSS

<title> و </title>: يتم وضع عنوان الصفحة بين هذه العناصر.

<link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">: يتم استخدام هذا العنصر

لتضمين ملف نمط CSS الذي يحتوي على تنسيقات المظهر للموقع. يتم تضمين الملف بواسطة استخدام url_for() لتوليد عنوان URL المناسب للملف.

<link rel="stylesheet">

<href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css">: يتم استخدام

هذا العنصر لتضمين ملف نمط CSS الذي يحتوي على تنسيقات Bootstrap للموقع. يتم تضمين الملف من Bootstrap CDN.

<body> و </body>: هذا يحدد بداية ونهاية جسم الصفحة ويحتوي على عناصر HTML الفعلية التي يتم عرضها.

<div class="container"> و </div>: هذا يحدد بداية ونهاية عنصر div الذي يحتوي على العناصر الأخرى في الصفحة.

<div class="row"> و </div>: هذا يحدد بداية ونهاية عنصر div آخر يستخدم لإنشاء صفوف في الصفحة.

<div class="col-md-6"> و </div>: هذا يحدد بداية ونهاية عنصر div يتم استخدامه لتحديد عمود في الصفحة. تم استخدام هذا المصفوفة في صفحة الرئيسية لعرض صورة ونص.

: يتم استخدام هذا العنصر لعرض

صورة على الصفحة. يتم تضمين الملف بواسطة استخدام url_for لتوليد عنوان URL المناسب للملف.

<h1> و </h1>: يتم استخدام هذا العنصر لإنشاء عنوان رئيسي.

<p> و </p>: يتم استخدام هذا العنصر لإنشاء فقرات.

`` و ``: يستخدم هذا العنصر لإنشاء رابط لصفحة أخرى في التطبيق. يتم استخدام `url_for` لتوليد عنوان URL المناسب للصفحة. يتم استخدام هذا العنصر لإنشاء رابطين لصفحتين أخريين.

```
<!DOCTYPE html>
<html>
<head>
    <title>Home</title>
    <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.cs
s">
</head>
<body>
    <div class="container">
        <div class="row">
            <div class="col-md-6">
                
            </div>
            <div class="col-md-6">
                <h1>الثاني السؤال</h1>
                <p>
والاتصالات والشبكات التواصل مثل المهمة، المواضيع من العديد الاتصالات مجال يتضمن
هذا ويعد .الكثير وغيرها الحديثة، والتكنولوجيا والإنترنت واللاسلكية، السلكية
من العديد تعتمد حيث اليومية، حياتنا على تؤثر التي المجالات أهم من المجال
وتقنياتها الاتصالات على والخدمات والمهام الأعمال
                </p>
                <p><a href="{{ url_for('page2') }}">برمجة</a></p>
                <p><a href="{{ url_for('page3') }}">خليوي</a></p>
            </div>
        </div>
        <ul>
            <li>
                <strong>مصطفى شهدت علاء</strong>:الاسم</li>
            <li>
                <strong>1590</strong>:الجامعي الرقم</li>
            <li>
                <strong>زيني غازي علي</strong>:الاسم</li>
            <li>
                <strong>1071</strong>:الجامعي الرقم</li>
        </ul>
    </div>
</body>
</html>
```

شكل الصفحة:



السؤال الثاني

يتضمن مجال الاتصالات العديد من المواضيع المهمة، مثل التواصل والشبكات والاتصالات السلكية واللاسلكية، والإنترنت والتكنولوجيا الحديثة، وغيرها الكثير. ويعد هذا المجال من أهم المجالات التي تؤثر على حياتنا اليومية، حيث تعتمد العديد من الأعمال والمهام والخدمات على الاتصالات وتقنياتها.

برمجة

خليوي

- الاسم: علاء شهدت مصطفى
الرقم الجامعي: 1590
- الاسم: علي غازي زيني
الرقم الجامعي: 1071

الصفحة الثانية:

```
<!DOCTYPE html>
<html>
<head>
  <title>Page 2</title>
  <link rel="stylesheet" href="{ url_for('static', filename='style.css')
  }}">
  <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.cs
s">
</head>
<body>
  <div class="container">
    <div class="row">
      <div class="col-md-6">
        
      </div>
      <div class="col-md-6">
</div>
</div>
<h1>برمجة شبكات</h1>
<p>مقرر البرمجة بالبايثون هو مقرر دراسي يهدف إلى تعليم الطلاب
أساسيات برمجة الحاسوب باستخدام لغة البايثون. وتعتبر لغة البايثون من بين أكثر
اللغات شيوعاً في عالم البرمجة حالياً، وتستخدم في العديد من المجالات الحيوية مثل
التحليل البياني والذكاء الاصطناعي والتطبيقات الحاسوبية والمواقع الإلكترونية
والألعاب.</div>
</p>
</div>
</div>
```

```
</body>
</html>
```

شكل الصفحة:



برمجة شبكات

مقرر البرمجة بالبايثون هو مقرر دراسي يهدف إلى تعليم الطلاب أساسيات برمجة الحاسوب باستخدام لغة البايثون. وتعتبر لغة البايثون من بين أكثر اللغات شيوعاً في عالم البرمجة حالياً، وتستخدم في العديد من المجالات الحيوية مثل التحليل البياني والذكاء الاصطناعي والتطبيقات الحاسوبية والمواقع الإلكترونية والألعاب.

الصفحة الثالثة:

```
<!DOCTYPE html>
<html>
<head>
  <title>Page 3</title>
  <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
  <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.cs
s">
</head>
<body>
  <div class="container">
    <div class="row">
      <div class="col-md.6">
        
      </div>
      <div class="col-md-6">
</h1>خليوي<h1>
  <p>مقرر الخليوي هو مقرر دراسي يهدف إلى تعليم الطلاب حول تقنية
الاتصالات الخلوية وكيفية عمل الشبكات اللاسلكية. يتعلق الأمر بتعليم الطلاب
الأساسيات المتعلقة بالاتصالات الخلوية، بما في ذلك الأجهزة المستخدمة في الاتصالات
```

الخلوية وتقنيات الاتصال المستخدمة وشبكات الاتصال الخلوية وتقنيات الاتصال
اللاسلكية</p>
</div>
</div>
</body>
</html>

شكل الصفحة:



خليوي

مقرر الخليوي هو مقرر دراسي يهدف إلى تعليم الطلاب حول تقنية الاتصالات الخلوية وكيفية عمل الشبكات اللاسلكية. يتعلق الأمر بتعليم الطلاب الأساسيات المتعلقة بالاتصالات الخلوية، بما في ذلك الأجهزة المستخدمة في الاتصالات الخلوية وتقنيات الاتصال المستخدمة وشبكات الاتصال الخلوية وتقنيات الاتصال اللاسلكية.