



الجمهورية العربية السورية

جامعة تشرين

قسم هندسة الاتصالات والالكترونيات

السنة الخامسة

مادة برمجة الشبكات

وظيفة 1

Github: [danial-rahmoon](https://github.com/danial-rahmoon)

اعداد الطالب:

دانيال جعفر رحمون 2836

بأشراف الدكتور:

مهند عيسى

Question 1:

A. If you have two lists, L1=['HTTP','HTTPS','FTP','DNS']

L2=[80,443,21,53], convert it to generate this dictionary

d={'HTTP':80,'HTTPS':443,'FTP':21,'DNS':53 }

الكود:

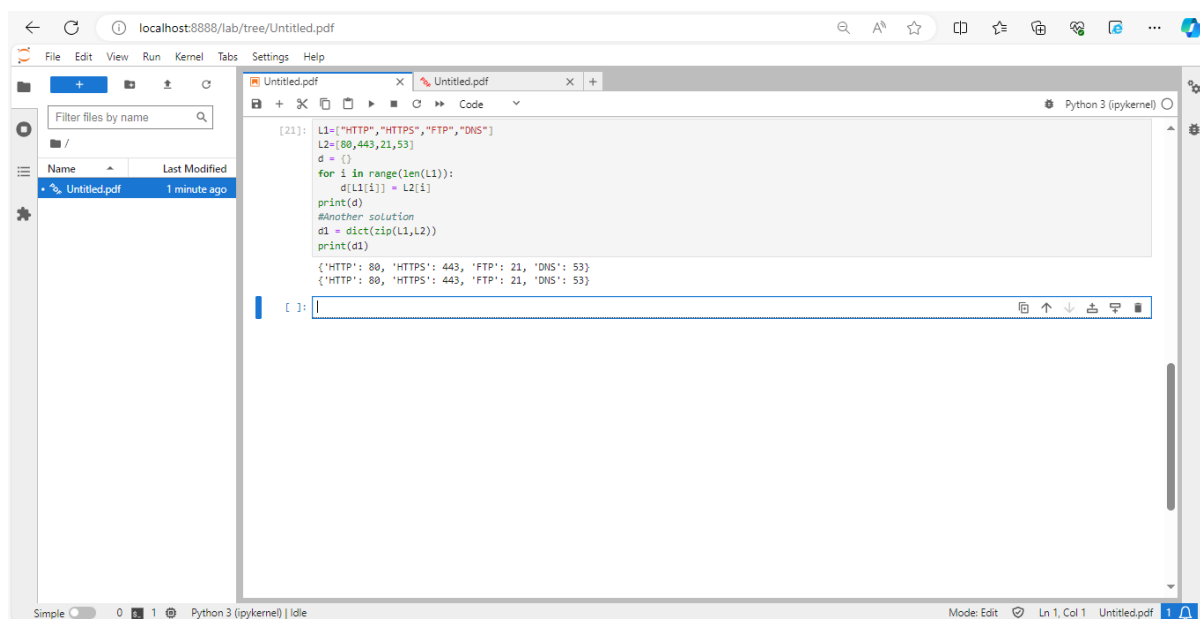
```
L1=["HTTP","HTTPS","FTP","DNS"]
L2=[80,443,21,53]
d = {}
for i in range(len(L1)):
    d[L1[i]] = L2[i]
print(d)
#Another solution
d1 = dict(zip(L1,L2))
print(d1)
```

التفسير:

تم انشاء قاموس فارغ ثم انشاء حلقة تكرارية لتعبئة القاموس حيث المفتاح سيكون من القائمة L1
وقيمة المفتاح من القائمة L2.

او باستخدام ال method الخاصة بدمج القائمتين الى قاموس وهي zip(L1,L2).

النتيجة:



```
[21]: L1=["HTTP","HTTPS","FTP","DNS"]
      L2=[80,443,21,53]
      d = {}
      for i in range(len(L1)):
          d[L1[i]] = L2[i]
      print(d)
      #Another solution
      d1 = dict(zip(L1,L2))
      print(d1)

      {'HTTP': 80, 'HTTPS': 443, 'FTP': 21, 'DNS': 53}
      {'HTTP': 80, 'HTTPS': 443, 'FTP': 21, 'DNS': 53}
```

B. Write a Python program that calculates the factorial of a given number entered by user.

الكود:

```
def factorial(num):
    if num == 0:
        return 1
    else:
        fact = 1
        for i in range(1, num + 1):
            fact = fact * i
        return fact

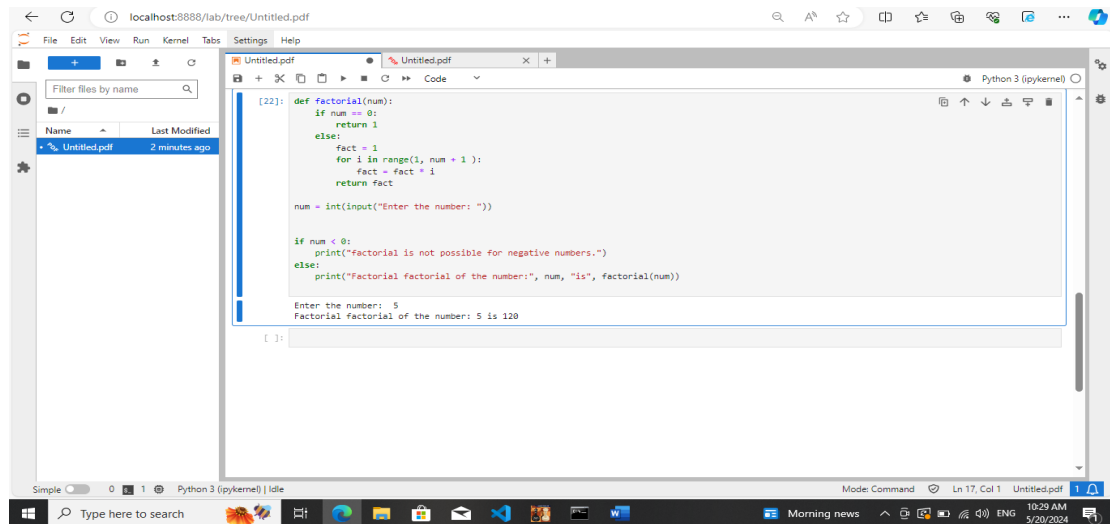
num = int(input("Enter the number: "))

if num < 0:
    print("factorial is not possible for negative numbers.")
else:
    print("Factorial factorial of the number:", num, "is", factorial(num))
```

التفسير:

1. لا يقبل الكود عدد سلبى.
2. إذا تم ادخال العدد 0 يعيد 1.
3. أي عدد طبيعي يدخل يتم إيجاد العامل المضروب من 1 حتى ذلك العدد.
4. ان العبارة (num + 1): هنا اضيف رقم 1 لأنه بالأصل المجال لا يأخذ العدد الأكبر.

النتيجة:



C. L=['Network' , 'Bio' , 'Programming' , 'Physics' , 'Music'] In this exercise, you will implement a Python program that reads the items of the previous list and identifies the items that starts with 'B' letter, then prin

الكود:

```
L = ['Network', 'Bio', 'Programming', 'Physics', 'Music']

for item in L:

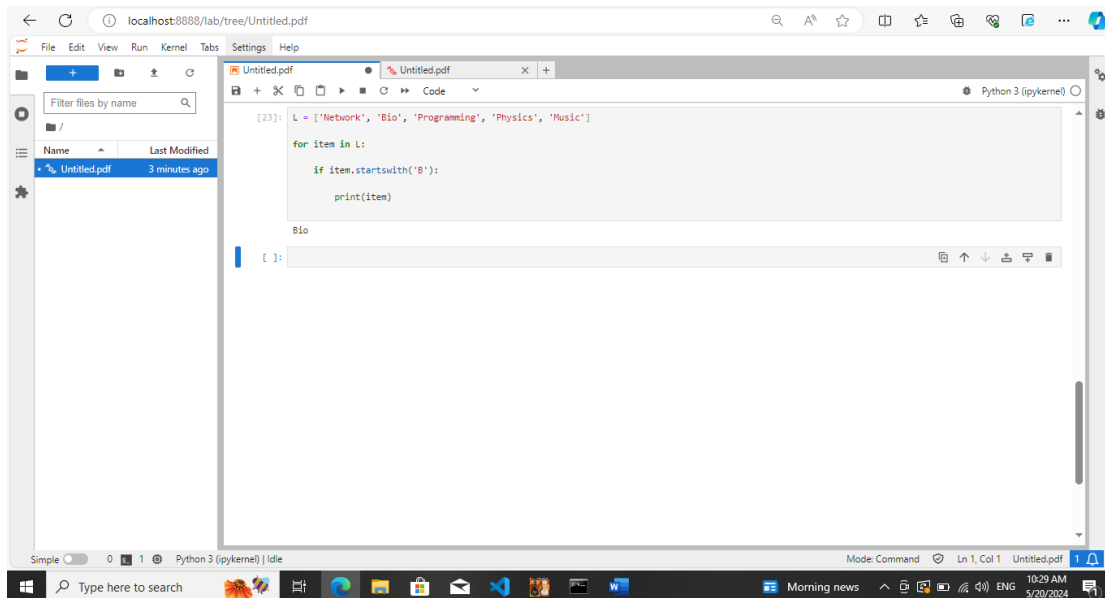
    if item.startswith('B'):

        print(item)
```

التفسير:

1. في هذا الكود يتم تكرار حلقة للبحث عن كلمة تبدأ بحرف B في القائمة L.
2. يتم استخدام الدالة startswith('B') حيث ما بين القوسين يتم البحث عنه في بداية كل جملة.

النتيجة:



D. Using Dictionary comprehension, Generate this dictionary

`d={0:1,1:2,2:3,3:4,4:5,5:6,6:7,7:8,8:9,9:10,10:11}`

الكود:

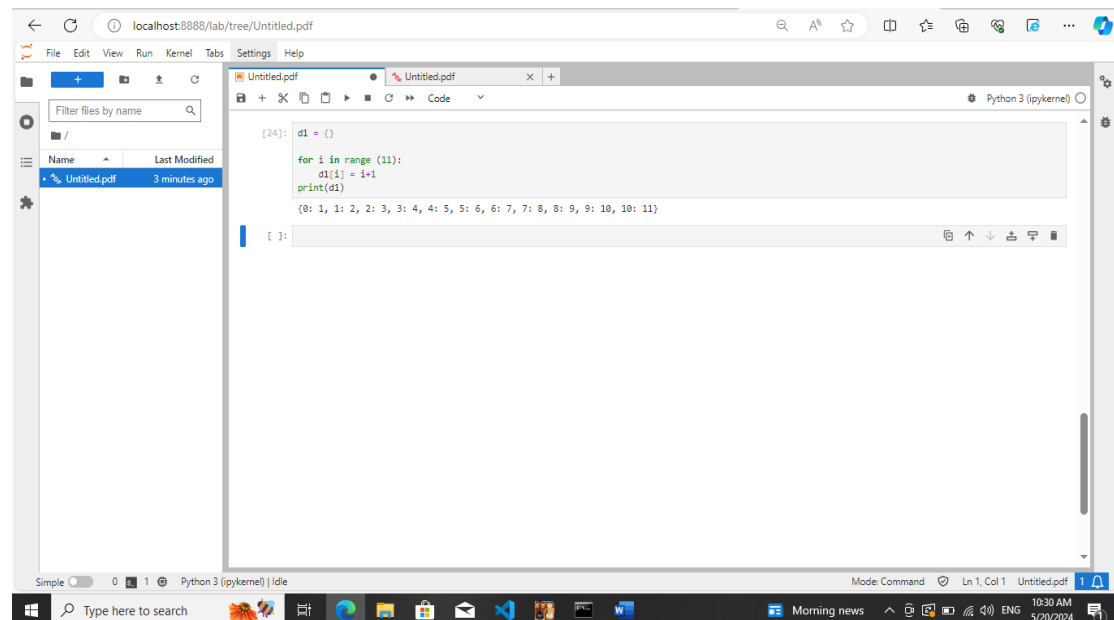
```
d1 = {}

for i in range (11):
    d1[i] = i+1
print(d1)
```

التفسير:

1. تم تعريف قاموس فارغ.
2. تم انشاء حلقة تكرار لتخزين العناصر في القاموس كمفتاح وقيمة.

النتيجة:



Question 2:

Convert from Binary to Decimal Write a Python program that converts a Binary number into its equivalent Decimal number. The program should start reading the binary number from the user. Then the decimal equivalent number must be calculated. Finally, the program must display the equivalent decimal number on the screen. Tips: solve input errors.

الكود:

```
def binary_to_decimal(binary_num):  
    if not set(binary_num).issubset('01'):  
        return "This is not a valid binary number."  
    else:  
        decimal_num = int(binary_num, 2)  
        return decimal_num  
  
binary_num = input("Enter the binary number: ")  
print("The equivalent decimal number is: ", binary_to_decimal(binary_num))
```

التفسير:

1. تم تعريف دالة تقوم بتحويل العدد الثنائي الى عشري حيث أي عدد يرجع لها انه غير ثنائي ستطبع انه العدد غير ثنائي.
2. يطلب من المستخدم ادخال الرقم.
3. بعدها يتم استدعاء الدالة وتحويل العدد الى ثنائي.
4. تم تجربة الرقم 000110111101101 حيث ارجعت القيمة 3565.

النتيجة:

The screenshot shows a Jupyter Notebook running on a local host. The notebook contains a Python function named `binary_to_decimal` that takes a binary string as input and returns its decimal equivalent. The function uses `int(binary_num, 2)` for conversion. Below the function definition, there is an input prompt and a print statement. The output shows the binary number `000110111101101` being converted to the decimal number `3565`.

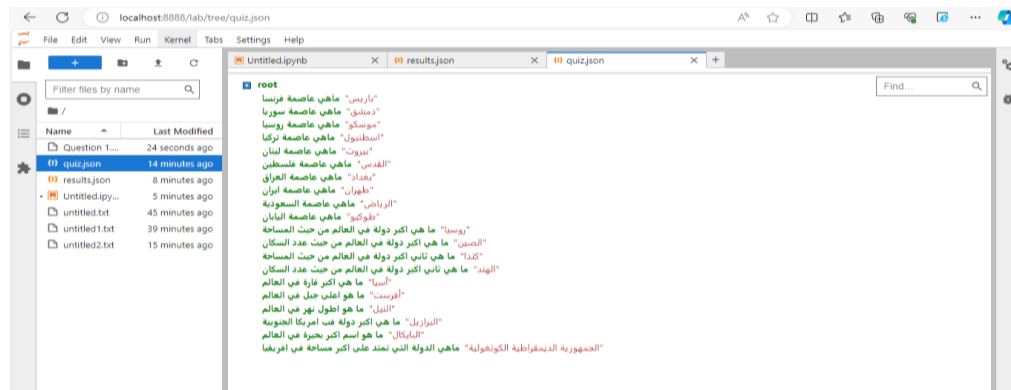
```
[29]: def binary_to_decimal(binary_num):  
      if not set(binary_num).issubset('01'):  
          return "This is not a valid binary number."  
      else:  
          decimal_num = int(binary_num, 2)  
          return decimal_num  
  
      binary_num = input("Enter the binary number: ")  
      print("The equivalent decimal number is: ", binary_to_decimal(binary_num))  
  
      Enter the binary number: 000110111101101  
      The equivalent decimal number is: 3565  
  
[ ]:
```

Question 3:

Working with Files” Quiz Program” Type python quiz program that takes a text or json or csv file as input for (20 (Questions, Answers)). It asks the questions and finally computes and prints user results and store user name and result in separate file csv or json file

الكود:

1. تم كتابة 20 سؤالاً مع جواباً في ملف وحفظ بامتداد json حيث البيانات على هيئة قاموس فكان السؤال هو مفتاح القاموس والجواب هو قيمة المفتاح:



2.

- استيراد المكتبة json التي تتعامل مع ملفات بامتداد json.

```
import json

with open('quiz.json', 'r', encoding='utf-8') as f:
    quiz = json.load(f)

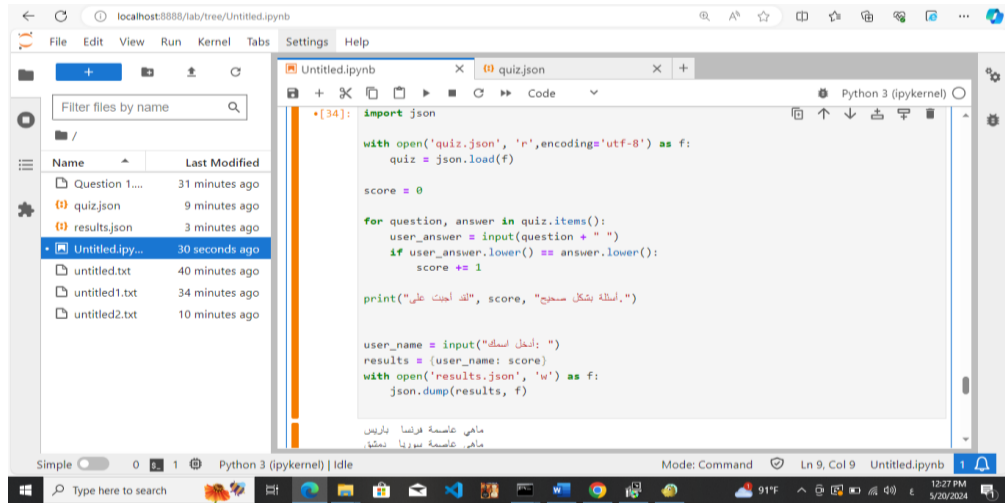
score = 0

for question, answer in quiz.items():
    user_answer = input(question + " ")
    if user_answer.lower() == answer.lower():
        score += 1

print("أسئلة بشكل صحيح", score, "لقد أجبت على")

user_name = input("أدخل اسمك: ")
results = {user_name: score}
with open('results.json', 'w') as f:
    json.dump(results, f)
```

- تم تشغيل الملف quiz.json بوضع القراءة 'r' ونمط الترميز 'utf-8'.
- يتم قراءة الملف وتخزين البيانات كقاموس في المتغير quiz.
- يتم انشاء المتغير score لتتبع عدد الأسئلة التي أجاب عليها المستخدم بشكل صحيح.
- يتم تنفيذ الحلقة for تتكرر على كل سؤال وجواب في القاموس quiz.
- في كل تكرار يتم طرح السؤال على المستخدم وينتظر البرنامج حتى الإجابة.
- بالنهاية يطلب من المستخدم ادخال اسمه ويتم انشاء قاموسا جديدا يحوي على اسم المستخدم وعدد النقاط التي حصل عليها.



```
[34]: import json

with open('quiz.json', 'r', encoding='utf-8') as f:
    quiz = json.load(f)

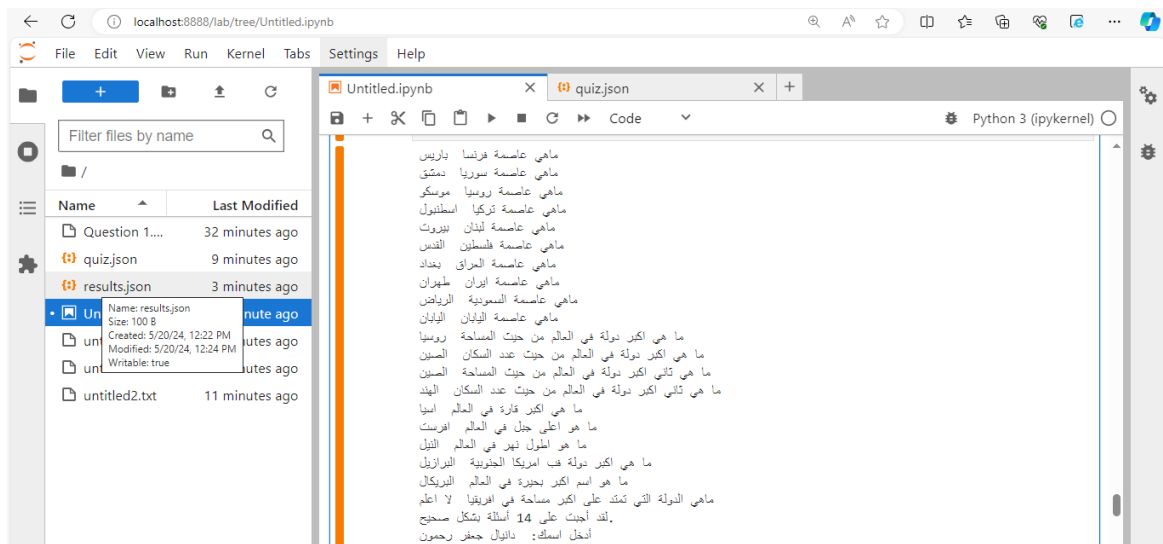
score = 0

for question, answer in quiz.items():
    user_answer = input(question + " ")
    if user_answer.lower() == answer.lower():
        score += 1

print("لقد أجبت على", score, "أسئلة بشكل صحيح.")

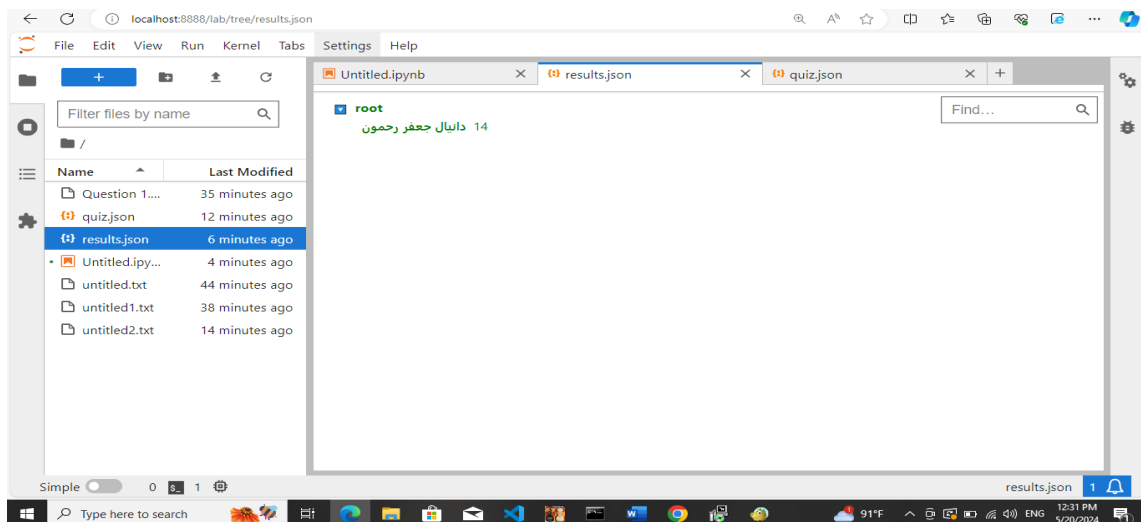
user_name = input("أدخل اسمك: ")
results = (user_name, score)
with open('results.json', 'w') as f:
    json.dump(results, f)
```

3. النتيجة بعد الإجابة على الأسئلة كاملة.



```
ماهي عاصمة فرنسا باريس
ماهي عاصمة سوريا دمشق
ماهي عاصمة روسيا موسكو
ماهي عاصمة تركيا انطونول
ماهي عاصمة لبنان بيروت
ماهي عاصمة فلسطين القدس
ماهي عاصمة العراق بغداد
ماهي عاصمة ايران طهران
ماهي عاصمة السعودية الرياض
ماهي عاصمة اليابان اليابان
ما هي اكبر دولة في العالم من حيث المساحة روسيا
ما هي اكبر دولة في العالم من حيث عدد السكان الصين
ما هي ثاني اكبر دولة في العالم من حيث المساحة الصين
ما هي ثاني اكبر دولة في العالم من حيث عدد السكان الهند
ما هي اكبر قارة في العالم اسيا
ما هو اعلى جبل في العالم افريت
ما هو الطول نهر في العالم النيل
ما هي اكبر دولة في أمريكا الجنوبية البرازيل
ما هو اسم اكبر بحيرة في العالم البوركينا
ماهي الدولة التي تمتد على اكبر مساحة في افريقيا لا اعلم
لقد أجبت على 14 أسئلة بشكل صحيح
أدخل اسمك: دانيال جعفر رحمون
```

ملف ال results الناتج عن تنفيذ الكود السابق حيث هذا الملف أيضا بامتداد json ويحوي بداخله على اسم المستخدم ونتيجته على الإجابة عن الأسئلة.



```
root
دانيال جعفر رحمون 14
```

Question 4:

1. Define a class BankAccount with the following attributes and methods: Attributes:

account_number (string), account_holder (string), balance (float, initialized to 0.0)

Methods: deposit(amount), withdraw(amount) , get_balance()

- Create an instance of BankAccount, - Perform a deposit of \$1000, - Perform a withdrawal of \$500. - Print the current balance after each operation.

الكود:

```
class BankAccount:
    def __init__(self, number, holder):
        self.number = number
        self.holder = holder
        self._balance = 0.0

    def add(self, amount):
        self._balance += amount
        print(f"الرصيد الحالي هو ${amount} تم إضافة ${self._balance}.")

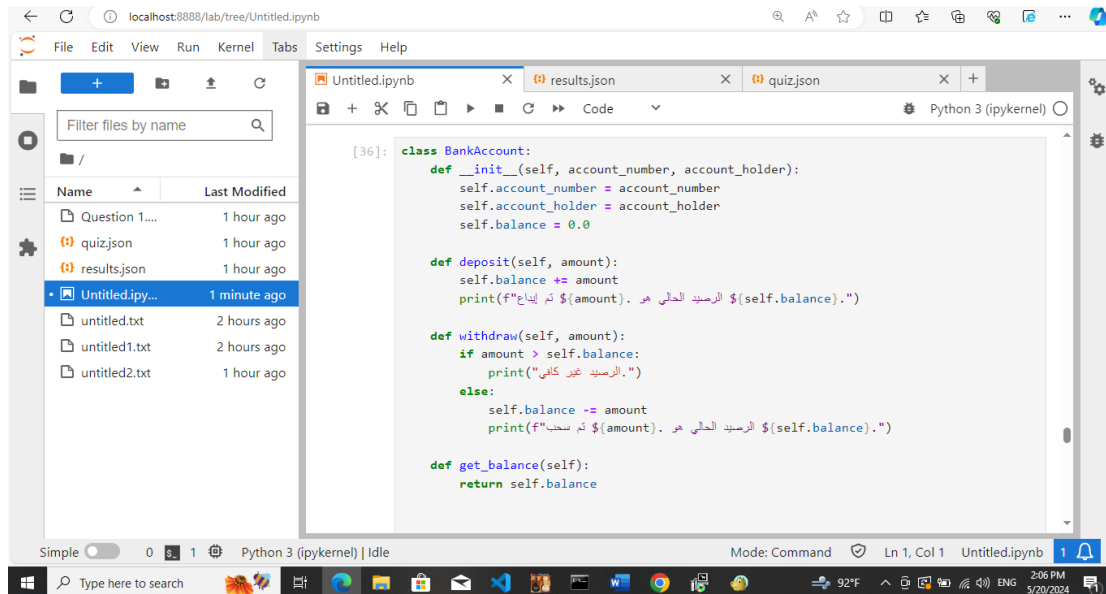
    def remove(self, amount):
        if amount > self._balance:
            print("الرصيد غير كافي.")
        else:
            self._balance -= amount
            print(f"الرصيد الحالي هو ${amount} تم إزالة ${self._balance}.")

    def current_balance(self):
        return self._balance
```

التفسير:

- تم تعريف كلاس هو BankAccount.
- الدالة __init__: تستدعي تلقائياً عند انشاء كائن جديد من الكلاس وتأخذ هذه الدالة ثلاثة مدخلات هي self (يشير الى الكائن ذاته). Number (الذي يمثل رقم الحساب). Holder (الذي يمثل صاحب الحساب).

- يتم تعيين قيم المدخلات الى خصائص الكائن.
- يتم تعيين القيمة الابتدائية للرصيد في الحساب الى 0.0.
- الدالتين add و remove لإضافة وإزالة مبلغ من الرصيد في الحساب.
- الدالة current_balance تعيد الرصيد الحالي في الحساب.



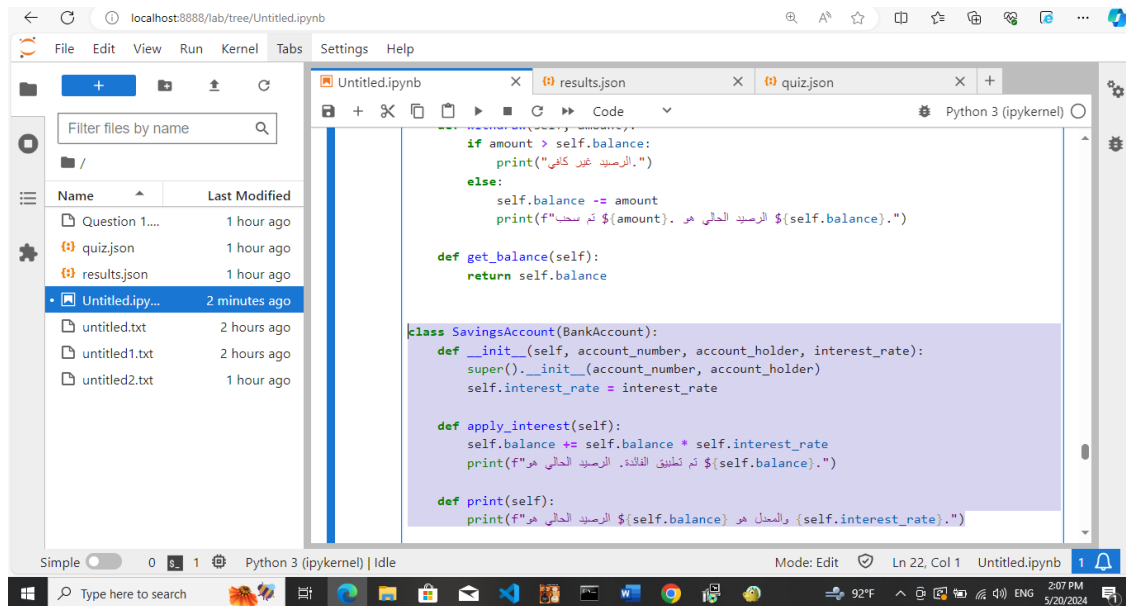
2. - Define a subclass SavingsAccount that inherits from BankAccount and adds interest_rate Attribute and apply_interest() method that Applies interest to the balance based on the interest rate. And Override print() method to print the current balance and rate.

الكود:

```
class SavingsAccount(BankAccount):
    def __init__(self, number, holder, rate):
        super().__init__(number, holder)
        self.rate = rate

    def interest(self):
        self._balance += self._balance * self.rate
        print(f"${self._balance}. تم تطبيق الفائدة. الرصيد الحالي هو")

    def display(self):
        print(f"والمعدل هو ${self._balance} الرصيد الحالي هو {self.rate}")
```



التفسير:

1. يتم تعريف كلاس savingsAccount الذي يرث من الكلاس BankAccount.
2. الدالة __init__: تستدعي تلقائياً عند انشاء كائن جديد من هذا الكلاس حيث تأخذ أربع مدخلات هي: self, number, holder, rate
3. الخاصية () super تقوم باستدعاء الدالة من الكلاس الاب وتعطيها قيم الدخلات number, holder الى خصائص الكائن.
4. الدالة interest () تحسب الفائدة.
5. الدالة display () تطبع الرصيد الحالي في الحساب ومعدل الفائدة.

3. - Create an instance of SavingsAccount , and call apply_interest() and print() functions

الكود:

```
acc = BankAccount("000000", "jaffer")
acc.add(1000)
acc.remove(500)

savings_acc = SavingsAccount("002836", "danial" , 0.05)
savings_acc.interest()
savings_acc.display()
```

التفسير:

1. المتغير acc هو عبارة عن كائن منشأ من الكلاس BankAccount برقم حساب 000000 وصاحب الحساب هو (jaffer).
2. يتم إضافة 1000 الى رصيد Jaffer.
3. ثم يتم إزالة 500 من رصيد Jaffer.
4. المتغير savings_acc هو عبارة عن كائن منشأ من الكلاس savingsAccount برقم حساب 002836 ومعدل فائدة (0.05) وصاحب الحساب هو (danial).
5. يتم استدعاء الدالة interest على الكائن savings_acc وتحسب الفائدة على الرصيد ونضيفها الى الرصيد.
6. يتم استدعاء الدالة display على الكائن savings_acc وتطبع الرصيد الحالي في الحساب ومعدل الفائدة.

```

class BankAccount:
    def __init__(self, number, holder, rate):
        self.number = number
        self.holder = holder
        self.rate = rate

    def add(self, amount):
        self.balance += amount
        print(f"تم إضافة {amount} الى الرصيد الحالي هو {self.balance}.")

    def remove(self, amount):
        self.balance -= amount
        print(f"تم إزالة {amount} من الرصيد الحالي هو {self.balance}.")

    def interest(self):
        self.balance += self.balance * self.rate
        print(f"تم تطبيق الفائدة. الرصيد الحالي هو {self.balance}.")

    def display(self):
        print(f"الرصيد الحالي هو {self.balance} والمعدل هو {self.rate}.")

acc = BankAccount("000000", "jaffer")
acc.add(1000)
acc.remove(500)

savings_acc = SavingsAccount("002836", "danial", 0.05)
savings_acc.interest()
savings_acc.display()

1000.05 تم إضافة 1000$ الى الرصيد الحالي هو 1000.05
500.05 تم إزالة 500$ من الرصيد الحالي هو 500.05
تم تطبيق الفائدة. الرصيد الحالي هو 500.05
0.05 الرصيد الحالي هو 500.05 والمعدل هو 0.05

```

النتيجة:

```

def interest(self):
    self.balance += self.balance * self.rate
    print(f"تم تطبيق الفائدة. الرصيد الحالي هو {self.balance}.")

def display(self):
    print(f"الرصيد الحالي هو {self.balance} والمعدل هو {self.rate}.")

acc = BankAccount("123456", "ahmad")
acc.add(1000)
acc.remove(500)

savings_acc = SavingsAccount("789012", "danial", 0.05)
savings_acc.interest()
savings_acc.display()

1000.05 تم إضافة 1000$ الى الرصيد الحالي هو 1000.05
500.05 تم إزالة 500$ من الرصيد الحالي هو 500.05
تم تطبيق الفائدة. الرصيد الحالي هو 500.05
0.05 الرصيد الحالي هو 500.05 والمعدل هو 0.05

```