

الاسم: لما مالك حيدر الرقم الجامعي: 2795 GitHub :Lama0934

## First network programming homework

### Question1:

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 }

```
L1 = ['HTTP', 'HTTPS', 'FTP', 'DNS']
```

```
L2 = [80, 443, 21, 53]
```

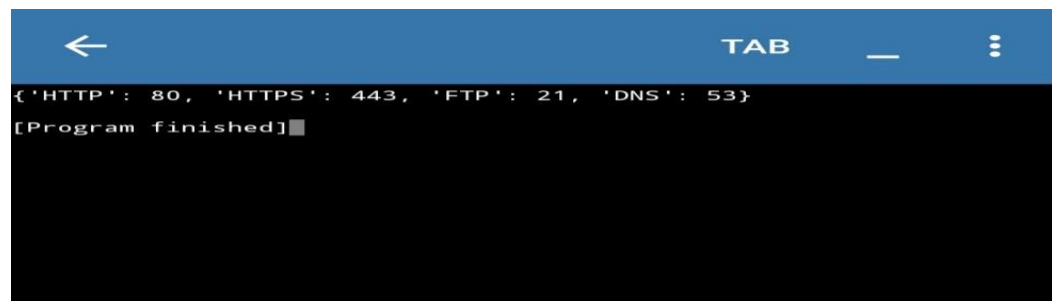
```
d = dict(zip(L1, L2))
```

```
print(d)
```

الشرح:

استخدمنا zip() لدمج القائمتين حيث تقوم بإنشاء tuples تحتوي على العناصر ثم حولنا النتيجة إلى قاموس باستخدام dict() حيث عناصر القائمة الأولى المفاتيح وعناصر القائمة الثانية هي القيم، وبالتالي الكود يقوم بإنشاء قاموس يربط أسماء بروتوكولات الإنترنت بأرقام المنافذ المقابلة لها.

النتيجة:



```
← TAB _ ⋮
{'HTTP': 80, 'HTTPS': 443, 'FTP': 21, 'DNS': 53}
[Program finished]
```

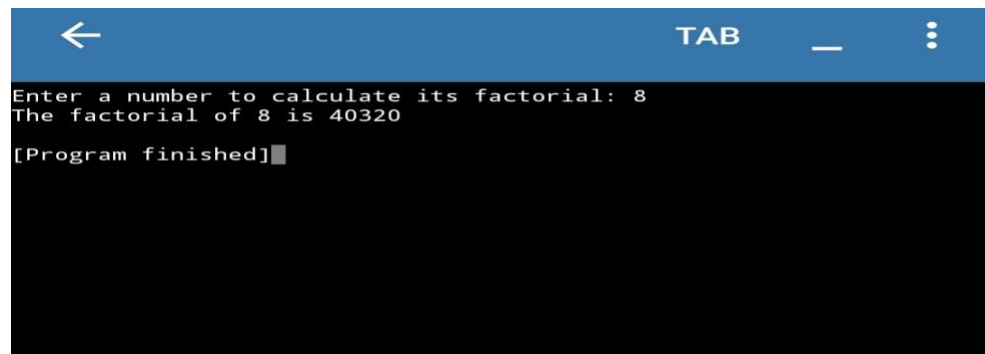
**B-** Write a Python program that calculates the factorial of a given number entered by user.

```
def calculate_factorial(num):  
    if num == 0:  
        return 1  
    else:  
        return num * calculate_factorial(num - 1)  
  
num = int(input("Enter a number to calculate its factorial: "))  
factorial = calculate_factorial(num)  
print(f"The factorial of {num} is {factorial}")
```

**الشرح:**

يطلب من المستخدم إدخال رقم عند تشغيل البرنامج، وبمجرد إدخاله سيقوم بحساب مضروب ذلك الرقم وطباعة النتيجة، وفي حال الرقم يساوي الصفر فإنه بعيد قيمة 1 (لأن مضروب الصفر يساوي 1).

**النتيجة:**



```
Enter a number to calculate its factorial: 8  
The factorial of 8 is 40320  
[Program finished]
```

**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 print it on screen. Tips: using loop, 'len ()' , startswith() methods.

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

```
for item in L:
```

```
    if item.startswith('B'):
```

```
        print(item)
```

الشرح:

نستخدم حلقة for لتكرار عناصر القائمة، واستخدمنا if لاختبار إذا كان العنصر يبدأ بحرف b إذا تحقق الشرط نطبع العنصر.

النتيجة:



```
← TAB _ ⋮
Bio
[Program finished]
```

**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}
```

```
d = {x: x+1 for x in range(11)}
```

```
print(d)
```

الشرح:

Range(11) يولد تسلسل بالأرقام من 0 إلى 10، يقوم الكود بإنشاء قاموساً يربط المفتاح بالقيمة ، سيتم طباعة القاموس بالمفاتيح التي تتراوح من 0 إلى 10 حيث يتم ربط كل مفتاح بقيمته المقابلة مع زيادة واحد.

**النتيجة:**



```
← TAB _ ⋮
{0: 1, 1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 6: 7, 7: 8, 8: 9, 9: 10, 10: 1
1}
[Program finished]
```

## 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.

```
def binary_to_decimal(binary):
```

```
    decimal = 0
```

```
    power = 0
```

```
    for digit in binary[::-1]:
```

```
        if digit == '1':
```

```
            decimal += 2 ** power
```

```
        elif digit != '0':
```

```

        return None

    power += 1

    return decimal

while True:

    binary_number = input("الرجاء إدخال رقم ثنائي: ")

    if not all(char in '01' for char in binary_number):

        print("رقم ثنائي غير صالح. يرجى إدخال رقم ثنائي يتكون من 0 و 1 فقط.")

    else:

        decimal_number = binary_to_decimal(binary_number)

        if decimal_number is not None:

            print(f"الرقم العشري المكافئ لـ {binary_number} هو: {decimal_number}")

            break

        else:

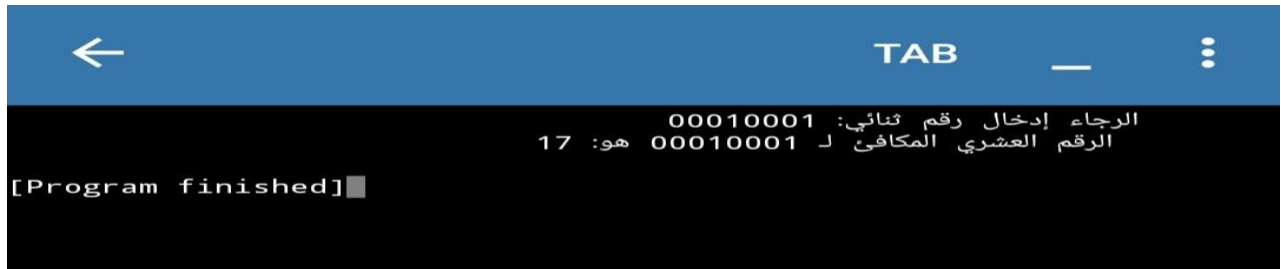
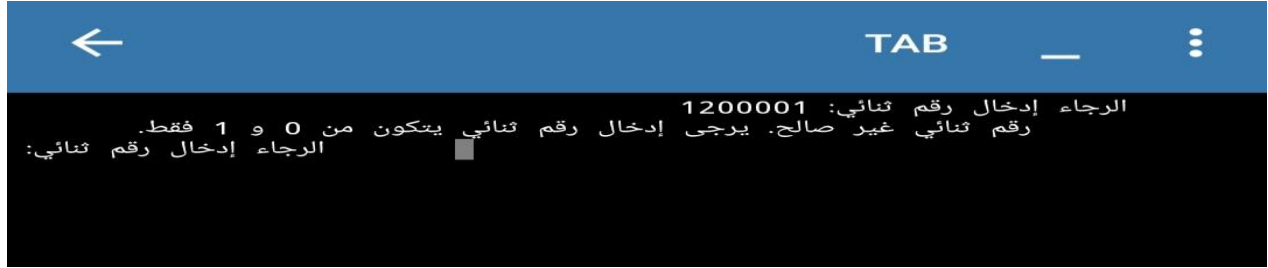
            print("رقم ثنائي غير صالح. يرجى إدخال رقم ثنائي صحيح.")

```

### الشرح:

يقوم المستخدم بإدخال رقم ثنائي، يتم قراءة الرقم من قبل المستخدم والتأكد أنه ثنائي (يحتوي على 0 و 1 فقط)، إذا كان صالحاً يتم تحويله إلى رقم عشري في حال لم يكن ثنائي يعرض رسالة خطأ.

النتيجة:



### 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.

```
import json
```

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

```
    question_with_user = json.load(f)
```

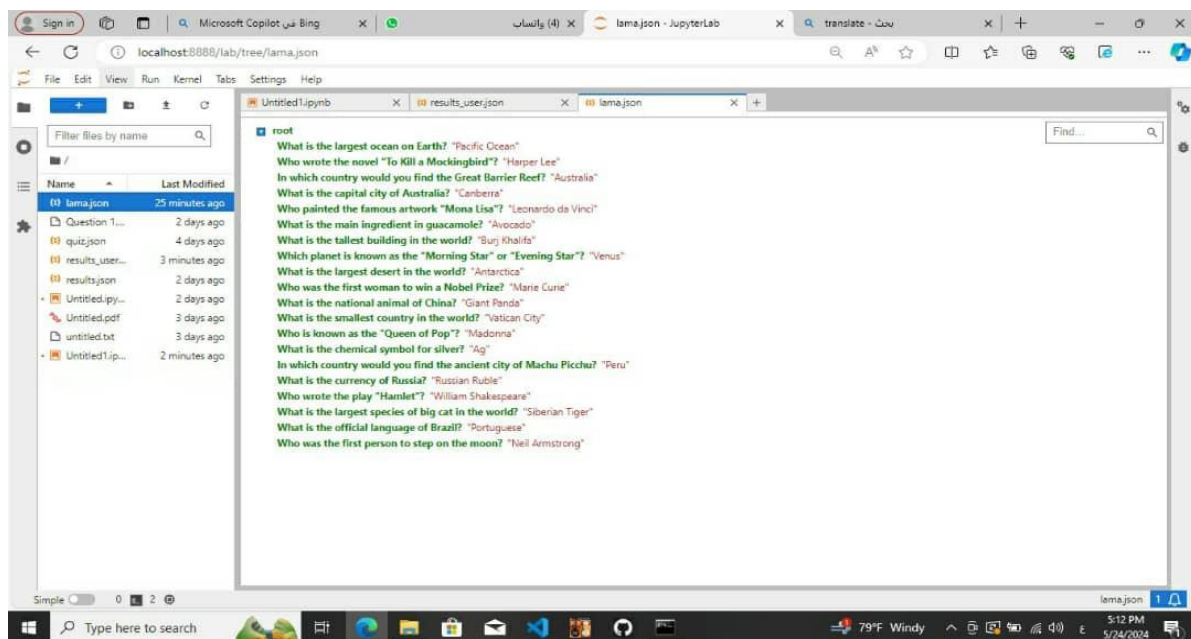
```
score = 0
```

```
for question, answer in question_with_user.items:()
```

```
user_answer = input(question + " ")
if user_answer.lower() == answer.lower():
    score += 1
```

```
print("You answered", score, "Questions correctly")
```

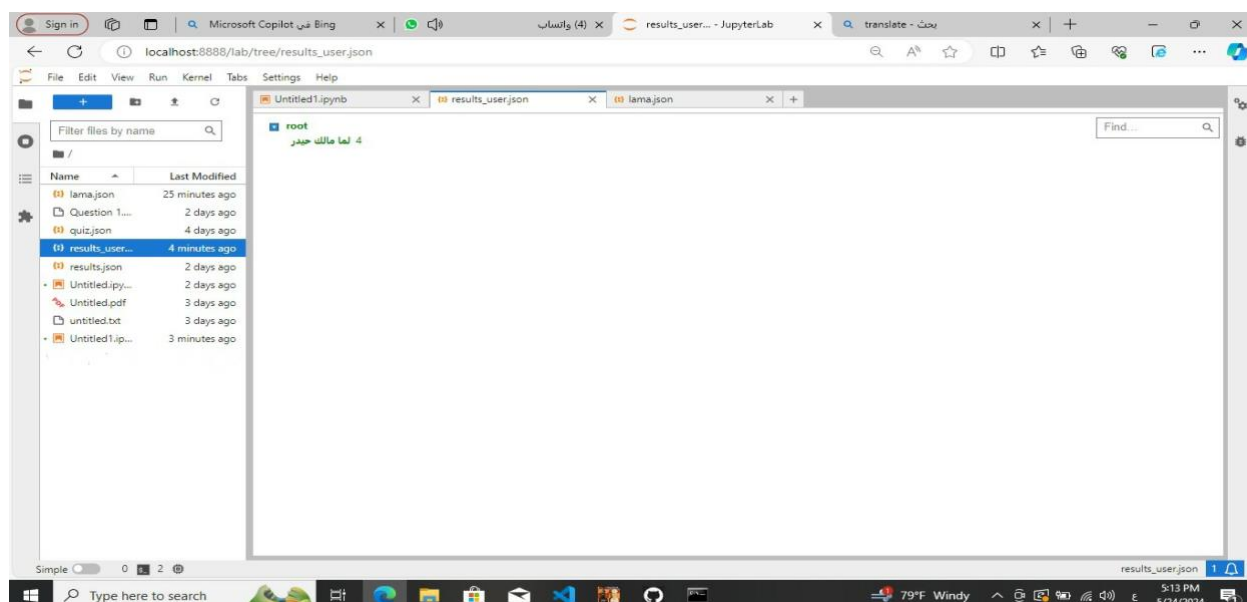
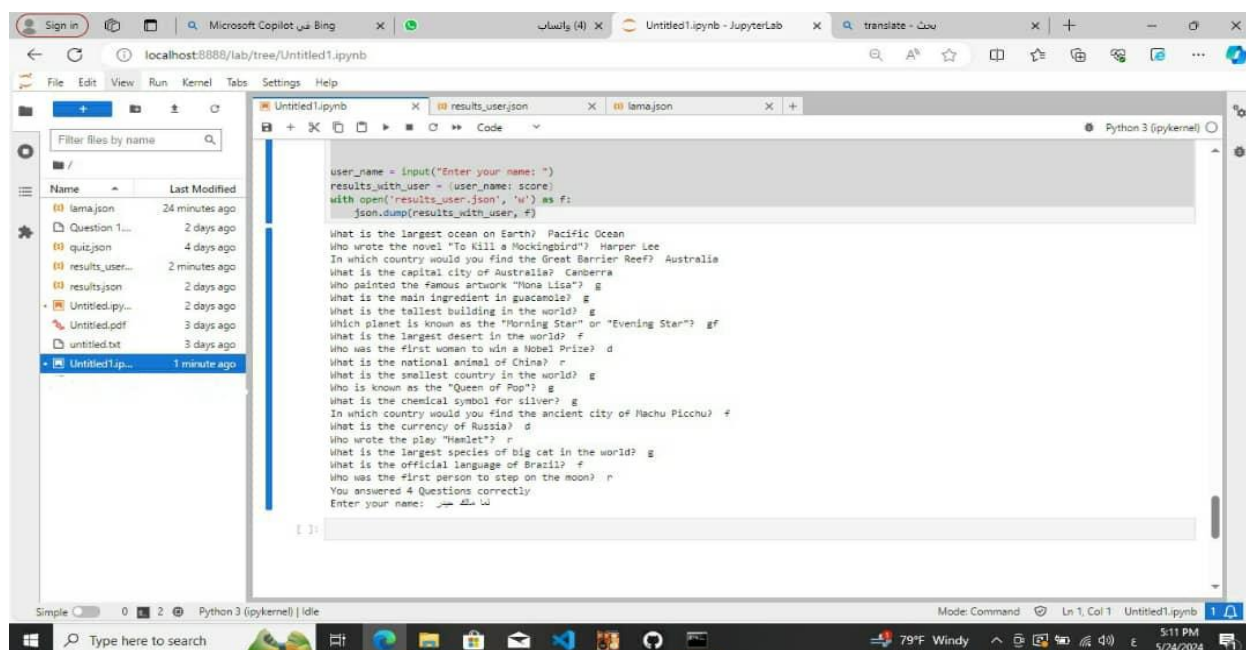
```
user_name = input("Enter your name: ")
results_with_user = {user_name: score}
with open('results_user.json', 'w') as f:
    json.dump(results_with_user, f)
```



الشرح:

يقوم الكود بفتح ملف json الذي يحتوي على أسئلة وإجابات، ويطلب من المستخدم الإجابة  
ويقوم بحساب الإجابات الصحيحة ثم يسأل المستخدم عن اسمه ويحفظ نتيجة الاختبار في ملف  
json جديد

النتيجة:





#### Question 4: Object-Oriented Programming - Bank Class

: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 -

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

Create an instance of SavingsAccount , and call apply\_interest() and - .print() functions

```
class BankAccount:
```

```
    def __init__(self, account_number, account_holder):
```

```
        self.account_number = account_number
```

```
        self.account_holder = account_holder
```

```
        self.balance = 0.0
```

```
    def deposit(self, amount):
```

```
        if amount > 0:
```

```
        self.balance += amount
        print(f"Deposited ${amount}. New balance: ${self.balance}")
    else:
        print("Invalid deposit amount.")
```

```
def withdraw(self, amount):
    if amount > 0 and amount <= self.balance:
        self.balance -= amount
        print(f"Withdrew ${amount}. New balance: ${self.balance}")
    else:
        print("Invalid withdrawal amount or insufficient balance.")
```

```
def get_balance(self):
    return self.balance
```

```
class SavingsAccount(BankAccount):
    def __init__(self, account_number, account_holder, interest_rate):
        super().__init__(account_number, account_holder)
        self.interest_rate = interest_rate

    def apply_interest(self):
        interest_amount = self.balance * (self.interest_rate / 100)
```

```
        self.balance += interest_amount

        print(f"Applied interest at rate {self.interest_rate}%. New balance:
        ${self.balance}")

    def print(self):
        print(f"Account Holder: {self.account_holder}")
        print(f"Account Number: {self.account_number}")
        print(f"Balance: ${self.balance}")
        print(f"Interest Rate: {self.interest_rate}%")

#Create an instance of SavingsAccount
savings_account = SavingsAccount("987654321", "lama", 2.5)

#Perform a deposit of $2000
savings_account.deposit(2000)

#Apply interest to the balance based on the interest rate
savings_account.apply_interest()

#Print the current balance and rate
savings_account.print()
```

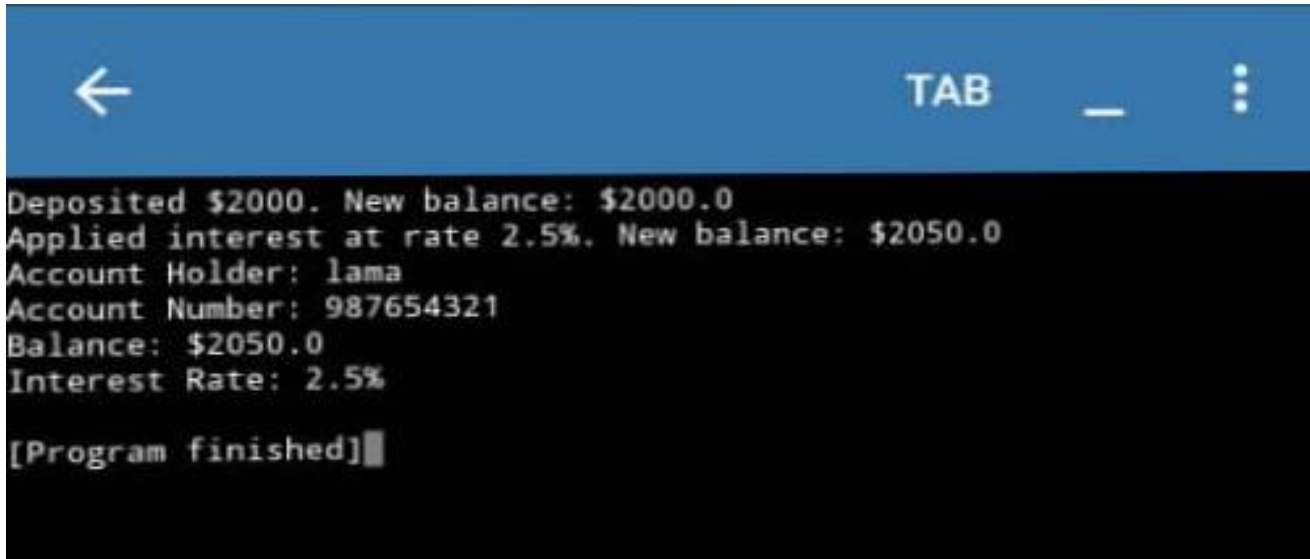
الشرح:

يقوم بتعريف savingsaccount هو حساب مصرفي، تحتوي على ( سحب أموال وإيداع أموال وتطبيق فائدة وطباعة تفاصيل الحساب)

ثم ينشئ my\_savings\_account ( إيداع أموال وسحب أموال ثم يطبق الفائدة وأخيراً طباعة تفاصيل الحساب).

ونحصل على تفاصيل الحساب بما فيها رقم الحساب واسم صاحب الحساب والرصيد الحالي ومعدل الفائدة.

**النتيجة:**



```
Deposited $2000. New balance: $2000.0
Applied interest at rate 2.5%. New balance: $2050.0
Account Holder: lama
Account Number: 987654321
Balance: $2050.0
Interest Rate: 2.5%
[Program finished]
```