

Name: _____ يزن غسان معروف _____, Number: _____ 2870 _____, Submitted To
GitHub: _____

Question 1: Python Basics?

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 }

Answer:

```
yazan1 = ['HTTP', 'HTTPS', 'FTP', 'DNS']
```

```
yazan2 = [80, 443, 21, 53]
```

```
y = dict(zip(yazan1, yazan2))
```

```
print(y)
```

output:

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

شرح الكود: لقد قمت بإنشاء قائمة تدعى yazan1 تحوي على أربعة عناصر وإنشأت قائمة ثانية تدعى yazan2 تحوي أربعة عناصر واستخدمت zip() لإنشاء زوج من القيم من القائمتين حيث العنصر الأول من القائمة الأولى يقابل العنصر الأول من القائمة الثانية ثم نمرر الزوج الى dict() لإنشاء قاموس جديد يدعى y

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

Answer:

```
yazan = int(input("Enter a number: "))
```

يطلب من المستخدم إدخال رقم ويخزنه في المتغير yazan

```
def factorial(n):
```

```
    if n == 0 or n == 1:
```

```
        return 1
```

```
    else:
```

```
        return n * factorial(n-1)
```

إذا تحقق الشرط السابق، تقوم الوظيفة بإرجاع القيمة 1 لأن (factorial) لـ 0 و 1 هي 1

إذا لم يتحقق الشرط السابق تقوم بإرجاع حاصل ضرب $n * \text{factorial}(n-1)$

```
result = factorial(yazan)
```

هنا نستدعي factorial مع القيمة المخزنة في yazan ونخزن النتيجة في result

```
print(f"The factorial of {yazan} is {result}")
```

نستخدم سلسلة f-string لطباعة قيمة yazan والنتيجة المخزنة في result بشكل منسق

outbut:

```
Enter a number: 5
The factorial of 5 is 120
```

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.

Answer:

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

انشأنا قائمة تدعى a تحوي خمس عناصر من النوع string

for item in a:

حلقة for والتي ستكرر التنفيذ لكل عنصر في a والمتغير item سيتعين لكل عنصر في a اثناء التنفيذ

if item.startswith('B'):

يقوم بفحص النص الموجود في item اذا يبدأ بحرف B باستخدام startswith() فإذا بدأ item ب B يكون الشرط صحيح

```
print(item)
```

إذا كان الشرط صحيحاً فسيتم طباعة قيمة item

outbut:

Bio

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}`

Answer:

`z = {i: i+1 for i in range(11)}`

يستخدم تعبير القاموس المتضمن (dictionary comprehension) لإنشاء قاموس جديد يُسمى z وهو طريقة مختصرة لإنشاء قاموس باستخدام حلقة for

داخل التعبير: i يمثل المفتاح وسيتم إنشاؤه من نطاق range(11) ينتج الاعداد من 1 الى 10 والقيمة المرتبطة بكل مفتاح i ستكون i+1 لذلك سيكون لدينا القاموس z المكون من 11 مفتاح-قيمة حيث المفاتيح الاعداد من 0 الى 10 والقيم الاعداد من 1 الى 11

`print(z)`

outbut:

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

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.

Answer:

`def binary_to_decimal(binary_str):`

تعريف الدالة binary_to_decimal والتي تأخذ بارمتر واحد وهو binary_str والتي من المتوقع أن تكون سلسلة نصية تمثل رقم ثنائي

try:

هذه الكتلة تحتوي على المنطق الرئيسي للدالة

for D in binary_str:

هذا الحلقة تكرر على كل حرف في السلسلة النصية `binary_str`

if D not in ['0', '1']:

هذا الجزء يتحقق من أن الحرف الحالي ليس 0 أو 1 مما يشير إلى رقم ثنائي غير صالح

raise ValueError("Invalid binary number. Please enter a valid binary number.")

إذا تم العثور على حرف غير صالح يتم رفع `ValueError` مع رسالة خطأ

deci = 0

نقوم بتهيئة متغير `deci` إلى 0 والذي سيخزن القيمة العشرية المكافئة للرقم الثنائي

for i, bit in enumerate(reversed(binary_str)):

هذه الحلقة تكرر على كل بت في الرقم الثنائي، بدءاً من البت الأيمن

deci += int(bit) * 2 ** i

هذا السطر يحسب القيمة العشرية للبت الحالي عن طريق ضرب قيمته (0 أو 1) في 2 مرفوعاً إلى قوة ويضيفه إلى المتغير `deci`

return deci

بعد الحلقة، هذا السطر يرجع القيمة العشرية النهائية

```
except ValueError as w:
```

```
    print(w)
```

```
    return None
```

إذا تم رفع `ValueError` اثناء كتلة `try` فسوف تلتقطه كتلة `except` وتطبع رسالة خطأ وترجع `None`

```
binary_number = input("Enter a binary number: ")
```

يطلب من المستخدم إدخال رقم ثنائي

```
decimal_number = binary_to_decimal(binary_number)
```

دالة `binary_to_decimal` بالرقم الثنائي المقدم من المستخدم ويخزن النتيجة في المتغير `decimal_number`

```
if decimal_number is not None:
```

التحقق من إرجاع رقم عشري صالح

```
    print(f"The equivalent decimal number is: {decimal_number}")
```

إذا كان صالح يطبع النتيجة

outbut:

```
Enter a binary number: 1111
The equivalent decimal number is: 15
```

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

Answer:

```
import json
```

```
import csv
```

```
def calculate_score(user_answers, correct_answers):
```

```
    score = 0
```

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

```
        if answer == correct_answers.get(question):
```

```
            score += 1
```

```
    return score
```

```
def save_results(user_name, user_answers, score, file_format):
```

```
    file_name = f'{user_name}_quiz_results.{file_format}'
```

```
    if file_format == 'json':
```

```
        with open(file_name, 'w') as json_file:
```

```
            json.dump({'username': user_name, 'result': f'{score}/20', 'Answers':  
user_answers}, json_file)
```

```
            print('results saved in JSON')
```

```
    elif file_format == 'csv':
```

```
        with open(file_name, 'w', newline='') as csv_file:
```

```
            csv_writer = csv.writer(csv_file)
```

```

        csv_writer.writerow(['Question', 'User Answer'])
    for question, answer in user_answers.items():
        csv_writer.writerow([question, answer])
        csv_writer.writerow(['Score', f'{score}/20'])
    print('results saved in CSV')
else:
    print('file format not available. do not save it.')

user_name = input(' your name: ')

file_name = input(' the file name : ')

with open(file_name, 'r') as file:
    inf = file.read()

questions_and_answers = json.loads(inf)

user_answers = {}
for question in questions_and_answers.keys():
    user_answer = input(f'Your answer for "{question}": ')
    user_answers[question] = user_answer

score = calculate_score(user_answers, questions_and_answers)
print(f'Your score is: {score}/20')

```

```
file_format = input('Choose the file format to save the results (JSON or CSV): ')
```

```
save_results(user_name, user_answers, score, file_format)
```

output:

```
your name: yazan
the file name : questions.txt
Your answer for "Among marine creatures, scientists have classified one of them
as the most intelligent. What is its name? ": dolphin
Your answer for "Philosophy has been widely used and researched since ancient ti
mes. So what does this word mean? ": It means a person who loves wisdom
Your answer for "who is the Prince of poets? ": Ahmad Shawqi
Your answer for "Who is the poet of the two countries? ": Khalil Mutran
Your answer for "How many teeth does a cat have? ": 40 years
Your answer for "In which city is Okaz Market located? ": taif city
Your answer for "Who is the poet of poets? ": Imru' al-Qais bin Hajar
Your answer for "Which science is the oldest in human history? ": astronomy
Your answer for "What is the largest island in the Mediterranean? ": Sicily Isla
nd
Your answer for "What is the second largest country in the world in terms of are
a? ": canada
Your answer for "What is the system of government followed in Japan? ": Imperial
rule
Your answer for "What is the smallest country in the world? ": the vatican
Your answer for "What color is an avocado? ": green
Your answer for "What is the capital of Lebanon? ": Beirut
Your answer for "What is the largest country in the Arabian Gulf in terms of are
a? ": Kingdom of Saudi Arabia
Your answer for "What is the fastest means of transportation available at the mo
ment? ": the plane
Your answer for "How many colors of the spectrum does a rainbow make up? ": seve
n colors
Your answer for "What is the name of the largest lake on planet Earth? ": Caspia
n Lake
Your answer for "What is the name of the Islamic state that made Istanbul its ca
pital? ": Ottoman Empire
Your answer for "What is the smallest continent in the world by area?": australi
a
Your score is: 18/20
Choose the file format to save the results (JSON or CSV): csv
results saved in CSV
```

شرح الكود:

في البداية قمنا باستيراد مكتبتَي json و csv لمعالجة البيانات بتنسيقات json و csv ثم عند تشغيل البرنامج يقوم بالطلب من المستخدم اسمه و اسم الملف الذي يحوي على الأسئلة والأجوبة

باستخدام `with open(file_name, 'r') as file` : يقوم بفتح ملف باسم `file_name` بطريقة قراءة ('r'). باستخدام `with` لضمان أن الملف سيتم إغلاقه تلقائيًا عند الانتهاء من استخدامه.

`inf = file.read()`: يقوم بقراءة كامل محتوى الملف وتخزينه في المتغير `inf`.

`questions_and_answers = json.loads(inf)`: هذا يقوم بتحويل محتوى الملف المخزن في `inf` من تنسيق JSON إلى قاموس Python باستخدام الدالة `json.loads()`. النتيجة تُخزن في المتغير `questions_and_answers`.

بعد هذه الخطوات، `questions_and_answers` سيكون قاموس Python يحتوي على الأسئلة والإجابات المحفوظة في الملف.

تم إنشاء قاموس فارغ بالاسم `user_answers` لتخزين إجابات المستخدم ثم حلقة `for` تبدأ في التكرار على جميع مفاتيح القاموس `questions_and_answers`، والتي تمثل الأسئلة ثم يقوم بالطلب من المستخدم إدخال إجابته على السؤال الحالي ثم يقوم بتخزين إجابة المستخدم في القاموس `user_answers` تحت المفتاح الذي يمثل السؤال وفي نهاية هذه الحلقة، `user_answers` سيكون قاموس يحتوي على إجابات المستخدم لجميع الأسئلة

يقوم بإرسال قاموسي `user_answers` و `questions_and_answers` إلى `calculate_score()` والتي تقوم بحساب درجة المستخدم. النتيجة يتم تخزينها في المتغير `score` قيمته البدائية 0 عن طريق حلقة `for` التي تختبر كل إجابة فإذا كانت صحيحة تقوم بزيادة `score` بمقدار 1 ثم تعيد قيمة `score` ويقوم بطباعة قيمة `{score}/20`

ثم يقوم بالطلب من المستخدم اختيار صيغة الملف المراد ان تظهر النتائج به إما `json` أو `csv`

ثم يقوم التابع `save_results` باختبار صيغة الملف المراد الحفظ به إذا كانت من النوع `json` أو `csv` يقوم بحفظ اسم المستخدم وإجابات المستخدم والنتيجة وصيغة الملف ويقوم بطباعة تم الحفظ بالصيغة التي ادخلها المستخدم وإذا ادخل المستخدم صيغة منافية ل `json` أو `csv` يقوم بطباعة الصيغة غير متوفرة لا تحفظ

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

Answer:

```
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):
```

```
        self.balance += amount
```

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

```
        if self.balance >= amount:
```

```
            self.balance -= amount
```

```
        else:
```

```
            print("Insufficient funds.")
```

```
    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):
        self.balance += self.balance * self.interest_rate

    def __str__(self):
        return f"Account Holder: {self.account_holder}\nAccount Balance:
        ${self.balance:.2f}\nInterest Rate: {self.interest_rate*100:.2f}%"

bank_account = BankAccount("124578963", "Ahmad Marouf")

bank_account.deposit(1000)
print(f"Current Balance: ${bank_account.get_balance():.2f}")

bank_account.withdraw(500)
print(f"Current Balance: ${bank_account.get_balance():.2f}")

savings_account = SavingsAccount("222333444", "Yazan Marouf", 0.05)

savings_account.apply_interest()
print(savings_account)
```

outbut:

```
Current Balance: $1000.00
Current Balance: $500.00
Account Holder: Yazan Marouf
Account Balance: $0.00
Interest Rate: 5.00%
```

شرح الكود:

BankAccount:

تحتوي على معلومات الحساب البنكي مثل رقم الحساب واسم صاحبه والرصيد.
توفر طرق لإيداع وسحب الأموال وعرض الرصيد الحالي.

SavingsAccount:

ترث من BankAccount وتُضيف معدل الفائدة كخاصية.
توفر طريقة لتطبيق الفائدة على الرصيد.
تعرف تمثيلاً نصياً مخصصاً للكائن.

لقد قمنا بإنشاء كائنين BankAccount و SavingsAccount

تم استخدام طرق الإيداع والسحب والفائدة على الحسابات.
تم عرض تفاصيل الحسابات باستخدام الطرق المناسبة.