

تعلم بايثون

باستخدام

ChatGPT

100 تمرين بلغة بايثون للمبتدئين تم حلها وشرحها باستخدام ChatGPT

ترجمة: د. علاء طعيمة



+



python™

بمه تعالى

تعلّم بايثون باستخدام ChatGPT

100 تمرين بلغة بايثون للمبتدئين تم حلها وشرحها باستخدام ChatGPT

ترجمة:

د. علاء طعيمة

مقدمة المترجم

ChatGPT هو روبوت للمحادثة أطلقته شركة OpenAI في نوفمبر 2022 كنموذج أولي للاختبار التجريبي والتعليقات. من بين العديد من الأشياء التي يمكنه القيام بها، يمكنه تعليمك لغات البرمجة. يستكشف هذا الكتاب كيف يمكن الاستفادة من طبيعة المحادثة في ChatGPT لتعلم بايثون.

في هذا الكتاب، تم استخدام ChatGPT لتجميع شروحات لـ 100 تمرين بايثون للمبتدئين تماماً. تمرين واحد في كل صفحة. باتباع شروحات التمرين، يمكن للمبتدئ في بايثون تعلم كيفية البرمجة.

لقد حاولت قدر المستطاع ان اترجم هذا الكتاب مع الشرح المناسب والكافي، ومع هذا يبقى عملاً بشرياً يحتمل النقص، فاذا كان لديك أي ملاحظات حول هذا الكتاب، فلا تتردد بمراسلتنا عبر بريدنا الالكتروني alaa.taima@qu.edu.iq.

نأمل ان يساعد هذا الكتاب كل من يريد ان يدخل في مجال لغة برمجة بايثون ومساعدة القارئ العربي على تعلم هذا المجال. اسأل الله التوفيق في هذا العمل لأثراء المحتوى العربي في مجال البرمجة بايثون. ونرجو لك الاستمتاع مع الكتاب ولا تنسونا من صالح الدعاء.

د. علاء طعيمة

كلية علوم الحاسوب وتكنولوجيا المعلومات

جامعة القادسية

العراق

مقدمة المؤلف

يُعد ChatGPT نموذجًا ثوريًا للغة كبيرة الحجم. إنه قادر على إنشاء نص حرفيًا لأي موضوع. لديها قدرات متميزة لتوليد تفسيرات الكود. يمكن أن يكون هذا بمثابة أداة ممتازة لتعليم البرمجة "حيث يتم تعلم البرمجة عن طريق البرمجة".

بعد هذه الفرضية، استخدمت ChatGPT لتجميع تفسيرات لـ 100 تمرين بايثون للمبتدئين تمامًا. تمرين واحد في كل صفحة. باتباع تفسيرات التمرين، يمكن للمبتدئ في بايثون تعلم كيفية البرمجة.

من الواضح أننا بحاجة إلى مجموعة من تمارين بايثون مناسبة تمامًا للمبتدئين. يحتوي مستودع [The Break The Ice With Python GitHub](#) على 100 سؤال بسيط من بايثون مع حلول.

بعد ذلك، باستخدام مقتطفات الكود، نطلب من ChatGPT شرحها سطرًا بسطر. للحصول على تفسيرات جيدة مناسبة للمبتدئين نستخدم الموجه التالي:

Explain me the following code snippet written in Python as explaining it to someone who doesn't know programming in Python: `<code_snippet>`
اشرح لي مقتطف الكود التالي المكتوب بلغة بايثون على النحو الموضح لشخص لا يعرف البرمجة في بايثون: `<مقتطف الكود>`
وبعد ذلك تبع مقتطف بايثون. ثم يتم أخذ الإخراج لمزيد من المعالجة.

بعد ذلك نضع جميع التفسيرات في وثيقة PDF نهائية. تمرين واحد في كل صفحة. تتبع كل صفحة نفس الهيكل الذي يحتوي على الأقسام التالية:

- التمرين N: وصف رقم التمرين N.
- الكود: حل التمرين.
- شروحات ChatGPT: شرح ChatGPT.

قم بإلقاء نظرة على المستند وقم بتنزيله من خلال النقر فوق هذا [الرابط](#).

يمكن العثور على جميع الموارد في [مستودع GitHub](#) هذا. إذا كان هذا شيئًا يعجبك وترغب في رؤية محتوى مشابه، يمكنك متابعتي على [LinkedIn](#) أو [Twitter](#). بالإضافة إلى ذلك، يمكنك الاشتراك في القائمة البريدية الخاصة بموقع isquared للحصول على تحديثات مماثلة من وقت لآخر.

Vladimir Ilievski

المحتويات

9.....	التمرين 1
10.....	التمرين 2
12.....	التمرين 3
14.....	التمرين 4
16.....	التمرين 5
18.....	التمرين 6
21.....	التمرين 7
23.....	التمرين 8
25.....	التمرين 9
27.....	التمرين 10
29.....	التمرين 11
31.....	التمرين 12
33.....	التمرين 13
35.....	التمرين 14
37.....	التمرين 15
38.....	التمرين 16
39.....	التمرين 17
41.....	التمرين 18
45.....	التمرين 19
47.....	التمرين 20
48.....	التمرين 21
50.....	التمرين 22
51.....	التمرين 23
52.....	التمرين 24
53.....	التمرين 25
55.....	التمرين 26
56.....	التمرين 27

57.....	التمرين 28
58.....	التمرين 29
59.....	التمرين 30
60.....	التمرين 31
61.....	التمرين 32
62.....	التمرين 33
63.....	التمرين 34
64.....	التمرين 35
65.....	التمرين 36
66.....	التمرين 37
67.....	التمرين 38
68.....	التمرين 39
69.....	التمرين 40
70.....	التمرين 41
71.....	التمرين 42
73.....	التمرين 43
74.....	التمرين 44
75.....	التمرين 45
76.....	التمرين 46
78.....	التمرين 47
79.....	التمرين 48
81.....	التمرين 49
83.....	التمرين 50
84.....	التمرين 51
86.....	التمرين 52
88.....	التمرين 53
90.....	التمرين 54
92.....	التمرين 55
93.....	التمرين 56

94.....	التمرين 57
95.....	التمرين 58
96.....	التمرين 59
97.....	التمرين 60
98.....	التمرين 61
100.....	التمرين 62
102.....	التمرين 63
104.....	التمرين 64
106.....	التمرين 65
107.....	التمرين 66
108.....	التمرين 67
110.....	التمرين 68
111.....	التمرين 69
112.....	التمرين 70
113.....	التمرين 71
114.....	التمرين 72
115.....	التمرين 73
117.....	التمرين 74
119.....	التمرين 75
120.....	التمرين 76
122.....	التمرين 77
124.....	التمرين 78
125.....	التمرين 79
127.....	التمرين 80
128.....	التمرين 81
129.....	التمرين 82
130.....	التمرين 83
131.....	التمرين 84
132.....	التمرين 85

133.....	التمرين 86
134.....	التمرين 87
135.....	التمرين 88
136.....	التمرين 89
138.....	التمرين 90
139.....	التمرين 91
140.....	التمرين 92
141.....	التمرين 93
142.....	التمرين 94
144.....	التمرين 95
146.....	التمرين 96
148.....	التمرين 97
150.....	التمرين 98
152.....	التمرين 99
154.....	التمرين 100

التمرين 1

الامر:

Write a program which will find all such numbers which are divisible by 7 but are not a multiple of 5, between 2000 and 3200 (both included). The numbers obtained should be printed in a comma-separated sequence on a single line.

اكتب برنامجاً سيجد كل هذه الأرقام القابلة للقسمة على 7 ولكنها ليست من مضاعفات الرقم 5، بين 2000 و 3200 (كلاهما مضمنان)، يجب طباعة الأرقام التي تم الحصول عليها في تسلسل مفصول بفاصلة على سطر واحد.

الكود:

```
for i in range(2000, 3201):
    if i % 7 == 0 and i % 5 != 0:
        print(i, end=",")
print("\b")
```

شرح ChatGPT:

This code snippet outputs all the numbers between 2000 and 3200 (inclusive) that are divisible by 7 but not divisible by 5.

The for loop iterates over the range of numbers from 2000 to 3200 and the if statement inside the loop checks if the current number *i* is divisible by 7 but not divisible by 5. If the condition is true, then the current number *i* is printed, followed by a comma.

Finally, the `\b` at the end of the code removes the last comma.

ينتج مقتطف الكود هذا جميع الأرقام بين 2000 و 3200 (ضمنياً) القابلة للقسمة على 7 ولكن لا تقبل القسمة على 5.

تتكرر حلقة for على نطاق الأرقام من 2000 إلى 3200 وتتحقق جملة if داخل الحلقة من أن الرقم الحالي *i* قابل للقسمة على 7 ولكن لا يقبل القسمة على 5. إذا كان الشرط صحيحاً، فسيتم طباعة الرقم الحالي *i* متبوعاً بفاصلة.

أخيراً، يزيل `\b` في نهاية الكود الفاصلة الأخيرة.

التمرين 2

الامر:

Write a program which can compute the factorial of a given numbers. The results should be printed in a comma-separated sequence on a single line. Suppose the following input is supplied to the program: 8 Then, the output should be: 40320.

اكتب برنامجاً يمكنه حساب مضروب أرقام معينة، ويجب طباعة النتائج في تسلسل مفصول بفاصلة على سطر واحد، افترض أن الإدخال التالي تم توفيره للبرنامج: 8 ثم يجب أن يكون الإخراج: 40320.

الكود:

```
n = int(input())
fact = 1
for i in range(1, n + 1):
    fact = fact * i
print(fact)
```

شرح ChatGPT:

This code snippet is a simple program that calculates the factorial of a given number.

The first line of code is getting an input from the user and assigns it to the variable n. The int function is used to convert the input to an integer.

The next line creates a variable called fact and assigns it the value of 1.

The following for loop starts with the variable i set to 1, and it will run as long as i is less than or equal to n + 1, and i will increment by 1 after each iteration.

In each iteration, the value of fact is multiplied by i and the result is stored in fact again, so on each iteration fact will be updated with the new value.

After the loop has finished, the final value of fact is printed using the print() function.

مقتطف الكود هذا عبارة عن برنامج بسيط يقوم بحساب مضروب رقم معين.

السطر الأول من الكود هو الحصول على مدخلات من المستخدم وتخصيصها للمتغير n . تستخدم الدالة `int` لتحويل المدخلات إلى عدد صحيح.

يُشئ السطر التالي متغيراً يسمى `fact` ويخصص له القيمة 1.

تبدأ حلقة `for` التالية بالمتغير i المعين على 1، وستعمل طالما أن i أقل من أو يساوي $n + 1$ ، وستزداد i بمقدار 1 بعد كل تكرار.

في كل تكرار، يتم ضرب قيمة `fact` بـ i ويتم تخزين النتيجة في `fact` مرة أخرى، لذلك سيتم تحديث كل تكرار `fact` بالقيمة الجديدة.

بعد انتهاء الحلقة، تتم طباعة القيمة النهائية للحقيقة باستخدام دالة `print()`.

التمرين 3

الامر:

With a given integral number n , write a program to generate a dictionary that contains $(i, i \times i)$ such that i is an integral number between 1 and n (both included). and then the program should print the dictionary. Suppose the following input is supplied to the program: 8.

باستخدام رقم متكامل محدد n ، اكتب برنامجاً لإنشاء قاموس يحتوي على $(i, i \times i)$ بحيث يكون رقمًا صحيحًا بين 1 و n (كلاهما مضمن). ومن ثم يقوم البرنامج بطباعة القاموس. افترض أن الإدخال التالي تم توفيره للبرنامج: 8.

الكود:

```
n = int(input())
ans = {}
for i in range(1, n + 1):
    ans[i] = i * i
print(ans)
```

شرح ChatGPT:

This code snippet is a simple program that creates a dictionary where the keys are the integers from 1 to n , and the values are the square of the corresponding key.

The first line of code is getting an input from the user and assigns it to the variable n . The `int` function is used to convert the input to an integer.

The next line creates an empty dictionary called `ans`.

The following `for` loop starts with the variable i set to 1, and it will run as long as i is less than or equal to $n + 1$, and i will increment by 1 after each iteration.

In each iteration, the i is used as key in the dictionary `ans` and the value is calculated as $i * i$ and assigned to that key. After the loop has finished, the final dictionary `ans` is printed using the `print()` function.

مقتطف الكود هذا عبارة عن برنامج بسيط يقوم بإنشاء قاموس حيث تكون المفاتيح هي الأعداد الصحيحة من 1 إلى n ، والقيم هي مربع المفتاح المقابل.

السطر الأول من الكود هو الحصول على مدخلات من المستخدم وتخصيصها للمتغير n . تستخدم الدالة `int` لتحويل المدخلات إلى عدد صحيح.

يشيء السطر التالي قاموساً فارغاً يسمى `ans`.

تبدأ حلقة `for` التالية بالمتغير `i` المعين على 1، وستعمل طالما أن `i` أقل من أو يساوي `n + 1`، وسأزداد `i` بمقدار 1 بعد كل تكرار.

في كل تكرار، يتم استخدام `i` كمفتاح في القاموس `ans` ويتم حساب القيمة على أنها `i * i` وتعيينها لهذا المفتاح. بعد انتهاء الحلقة، تتم طباعة القاموس النهائي باستخدام دالة `print()`.

تبدأ حلقة `for` التالية بالمتغير `i` المعين على 1، وستعمل طالما أن `i` أقل من أو يساوي `n + 1`، وستزداد `i` بمقدار 1 بعد كل تكرار.

في كل تكرار، يتم استخدام `i` كمفتاح في القاموس `ans` ويتم حساب القيمة على أنها `i * i` وتعيينها لهذا المفتاح. بعد انتهاء الحلقة، تتم طباعة القاموس النهائي باستخدام دالة `print()`.

التمرين 4

الامر:

Write a program which accepts a sequence of comma-separated numbers from console and generate a list and a tuple which contains every number. Suppose the following input is supplied to the program:

اكتب برنامجاً يقبل تسلسلاً من الأرقام المفصولة بفواصل من وحدة التحكم وقم بإنشاء قائمة وصف تحتوي على كل رقم، افترض أنه تم توفير المدخلات التالية للبرنامج:

الكود:

```
lst = input().split(",")
tpl = tuple(lst)

print(lst)
print(tpl)
```

شرح ChatGPT:

This code snippet is a simple program that takes a string of comma-separated values as input, converts it to a list and then converts it to a tuple.

The first line of code is getting an input from the user and assigns it to the variable `lst`, The `split()` method is used to convert the input string into a list of strings by splitting it at each comma.

The next line of code creates a variable called `tpl`, and it uses the `tuple()` function to convert the `lst` variable from a list to a tuple.

The last two lines of code are using the `print()` function to display the `lst` variable and the `tpl` variable on the screen.

مقتطف الكود هذا عبارة عن برنامج بسيط يأخذ سلسلة من القيم المفصولة بفواصل كمدخلات، ويحولها إلى قائمة ثم يحولها إلى صف.

السطر الأول من الكود هو الحصول على مدخلات من المستخدم وتخصيصها للمتغير `lst`، يتم استخدام طريقة `split()` لتحويل سلسلة الإدخال إلى قائمة من السلاسل عن طريق تقسيمها عند كل فاصلة.

يُنشئ السطر التالي من التعليمات البرمجية متغيراً يسمى `tpl`، ويستخدم الدالة `tuple()` لتحويل المتغير `lst` من قائمة إلى صف.

يستخدم آخر سطرين من التعليمات البرمجية دالة `print()` لعرض `lst` والمتغير `tpl` على الشاشة.

التمرين 5

الامر:

Define a class which has at least two methods:

- getString: to get a string from console input
- printString: to print the string in upper case.:

عرف فئة لها طريقتان على الأقل:

- getString: للحصول على سلسلة نصية من إدخال وحدة التحكم.
- printString: لطباعة السلسلة النصية بأحرف كبيرة:

الكود:

```
class IOstring:
    def __init__(self):
        pass

    def get_string(self):
        self.s = input()

    def print_string(self):
        print(self.s.upper())

xx = IOstring()
xx.get_string()
xx.print_string()
```

شرح ChatGPT:

This code snippet is a simple program that defines a class called `IOstring`, which has two methods: `get_string()` and `print_string()`.

The `__init__` method is the constructor of the class, it is called when an object of this class is created and does not have any specific functionality in this case, it just pass.

The `get_string()` method prompts the user to input a string and assigns it to the variable `s` as an attribute of the class.

The `print_string()` method then takes the value stored in the `s` attribute, converts it to uppercase using the `upper()` method, and prints it out.

The last three lines of code are creating an object of the IOstring class called xx, calling the get_string() method on it to set the value of the s attribute, and then calling the print_string() method to print the value of s in uppercase.

مقتطف الكود هذا هو برنامج بسيط يعرف فئة تسمى IOstring، والتي لها طريقتان: get_string() و print_string().

طريقة __init__ هي مُنشئ الفئة، ويتم استدعاؤها عندما يتم إنشاء كائن من هذه الفئة وليس له أي دالة محددة في هذه الحالة، فإنه يمر فقط.

تطالب طريقة get_string() المستخدم بإدخال سلسلة نصية وتعيينها إلى المتغير s كسمة للفئة.

تأخذ طريقة print_string() القيمة المخزنة في السمة s وتحولها إلى أحرف كبيرة باستخدام طريقة upper() وتطبعتها.

تقوم الأسطر الثلاثة الأخيرة من التعليمات البرمجية بإنشاء كائن من فئة IOstring يسمى xx، واستدعاء طريقة get_string() عليها لتعيين قيمة السمة s، ثم استدعاء طريقة print_string() لطباعة قيمة s بأحرف كبيرة.

التمرين 6

الامر:

Write a program that calculates and prints the value according to the given formula:

$Q = \text{Square root of } [(2 \cdot C \cdot D)/H]$

Following are the fixed values of C and H:

- C is 50. H is 30.

D is the variable whose values should be input to your program in a comma-separated sequence. For example Let us assume the following comma separated input sequence is given to the program:

اكتب برنامجاً يحسب القيمة ويطبعها وفقاً للصيغة المحددة:

$Q = \text{Square root of } [(2 \cdot C \cdot D)/H]$

فيما يلي القيم الثابتة لـ C و H:

- C is 50. H is 30.

D هو المتغير الذي يجب إدخال قيمه في برنامجك في تسلسل مفصولة بفواصل. على سبيل المثال، لنفترض أن تسلسل الإدخال التالي مفصول بفاصلة قد تم إعطاؤه للبرنامج:

الكود:

```
from math import sqrt

C, H = 50, 30

def calc(D):
    return sqrt((2 * C * D) / H)

print(",".join([str(int(calc(int(i)))) for i in
input().split(",")]))
```

شرح ChatGPT:

This code is a simple program that takes a string of comma-separated values as input, performs some calculations on each of the values, and then prints the result as a string of comma-separated values.

The first line of code imports the `sqrt()` function from the `math` module, which is used to calculate the square root of a number.

The next two lines define two constants, `C` and `H`, which are used in the calculations.

The following function `calc(D)` takes a parameter `D` and returns the result of the calculation $\sqrt{(2 * C * D) / H}$

The final line of code uses the `input()` function to get user input, which is expected to be a string of comma-separated values. It then uses the `split()` method to convert the input string into a list of strings by splitting it at each comma.

It then creates a list comprehension that iterates over the list of input values, calls the `calc()` function on each input value, which returns a float, and then converts the float to an int using the `int()` function. Then it uses the `join()` method to join the list of integers into a string of comma-separated values and prints it out.

هذا الكود عبارة عن برنامج بسيط يأخذ سلسلة من القيم المفصولة بفواصل كمدخلات، ويقوم ببعض العمليات الحسابية على كل من القيم، ثم يطبع النتيجة كسلسلة من القيم المفصولة بفواصل.

يستورد السطر الأول من الكود الدالة `sqrt()` من وحدة `math`، والتي تُستخدم لحساب الجذر التربيعي لرقم.

يعرف السطران التاليان ثابتين، `C` و `H`، يتم استخدامهما في العمليات الحسابية.

تأخذ السطر التالي التالي `calc(D)` المعلمة `D` وتعيد نتيجة الحساب $\sqrt{(2 * C * D) / H}$.

يستخدم السطر الأخير من الكود دالة `input()` للحصول على مدخلات المستخدم، والتي من المتوقع أن تكون سلسلة من القيم المفصولة بفواصل. ثم تستخدم طريقة `split()` لتحويل سلسلة الإدخال إلى قائمة سلاسل بتقسيمها عند كل فاصلة.

يقوم بعد ذلك بإنشاء list comprehension التي تتكرر عبر قائمة قيم الإدخال، وتستدعي الدالة `calc()` على كل قيمة إدخال، والتي تُرجع عددًا عشريًا، ثم تقوم بتحويلها إلى عدد صحيح باستخدام الدالة `int()`. ثم يستخدم الطريقة `join()` للانضمام إلى قائمة الأعداد الصحيحة في سلسلة من القيم المفصولة بفاصلة ويطبعها.

التمرين 7

الامر:

Write a program which takes 2 digits, X, Y as input and generates a 2-dimensional array. The element value in the i-th row and j-th column of the array should be i_j .

اكتب برنامجًا يقبل سلسلة من الكلمات مفصولة بفاصلة كمدخلات ويطبع الكلمات في تسلسل مفصولة بفاصلة بعد فرزها أبجديًا:

الكود:

```
x, y = map(int, input().split(", "))
lst = [[i * j for j in range(y)] for i in range(x)]
print(lst)
```

شرح ChatGPT:

This code is a simple program that takes two integers as input, separated by a comma and creates a list of lists (also known as a 2D list) by using a nested list comprehension.

The first line of code uses the `input()` function to get user input, which is expected to be two integers separated by a comma. It then uses the `split()` method to separate the input string at the comma and the `map()` function to convert the resulting two strings into integers and assigns them to the variables `x`, `y`.

The second line of code creates a list of lists called `lst` using a nested list comprehension. The outer list comprehension iterates over the range of `x`, and the inner list comprehension iterates over the range of `y`, and it creates a new list of integers where each integer is the product of the outer loop variable and the inner loop variable.

The last line of code prints out the list of lists.

هذا الكود عبارة عن برنامج بسيط يأخذ رقمين صحيحين كمدخلات، مفصولة بفاصلة وينشئ قائمة من القوائم (تُعرف أيضًا بالقائمة ثنائية الأبعاد) باستخدام List Comprehension متداخلة.

يستخدم السطر الأول من الكود دالة `input()` للحصول على مدخلات المستخدم، والتي من المتوقع أن تكون عددين صحيحين مفصولين بفاصلة. ثم تستخدم طريقة `split()` لفصل سلسلة الإدخال عند

الفاصلة والدالة `map()` لتحويل السلسلتين الناتجتين إلى أعداد صحيحة وتخصيصهما للمتغيرات `x` و `y`.

التمرين 8

الامر:

Write a program that accepts a comma separated sequence of words as input and prints the words in a comma-separated sequence after sorting them alphabetically.

اكتب برنامجاً يقبل سلسلة من الكلمات مفصولة بفاصلة كمدخلات ويطبع الكلمات في تسلسل مفصولة بفاصلة بعد فرزها أبجدياً:

الكود:

```
lst = input().split(",")
lst.sort()
print(",".join(lst))
```

شرح ChatGPT:

This code is a simple program that takes a string of comma-separated values as input, sorts the values in alphabetical or numerical order, and then prints the sorted values as a string of comma-separated values.

The first line of code uses the `input()` function to get user input, which is expected to be a string of comma-separated values, and the `split()` method to convert the input string into a list of strings by splitting it at each comma. The second line of code sorts the list of strings using the `sort()` method. This sorts the list in alphabetical order for strings and numerical order for numbers. The final line of code uses the `join()` method to join the list of strings into a string of comma-separated values, and then prints it out.

هذا الكود عبارة عن برنامج بسيط يأخذ سلسلة من القيم المفصولة بفواصل كمدخلات، ويفرز القيم بترتيب أبجدي أو رقمي، ثم يطبع القيم التي تم فرزها كسلسلة من القيم المفصولة بفواصل.

يستخدم السطر الأول من الكود دالة `input()` للحصول على مدخلات المستخدم، والتي من المتوقع أن تكون سلسلة من القيم المفصولة بفواصل، وطريقة `split()` لتحويل سلسلة الإدخال إلى قائمة من السلاسل عن طريق تقسيمها عند كل فاصلة.

يفرز السطر الثاني من التعليمات البرمجية قائمة السلاسل النصية باستخدام طريقة `sort()`. يقوم هذا بفرز القائمة بالترتيب الأبجدي للسلاسل والترتيب الرقمي للأرقام.

يستخدم السطر الأخير من الكود طريقة `join()` لربط قائمة السلاسل في سلسلة من القيم المفصولة بفاصلة ، ثم يطبعها.

التمرين 9

الامر:

Write a program that accepts sequence of lines as input and prints the lines after making all characters in the sentence capitalized.

اكتب برنامجاً يقبل تسلسل الأسطر كمدخلات ويطبع الأسطر بعد جعل جميع الأحرف في الجملة كبيرة:

الكود:

```
def user_input():
    while True:
        s = input()
        if not s:
            return
        yield s

for line in map(str.upper, user_input()):
    print(line)
```

شرح ChatGPT:

This code defines a function `user_input()` that takes no arguments and continuously prompts the user for input until they provide an empty string (by just pressing enter). Each input string provided by the user is then passed to the `str.upper()` method to convert the string to uppercase and then passed to the `print()` function to print it out. The `map()` function applies `str.upper()` method to each input string.

The `while` loop in the function `user_input()` uses the `input()` function to continuously prompt the user for input and assigns the input string to the variable `s`.

The `if not s:` line checks whether the input string is empty or not. If it is empty, the `return` statement is executed, which terminates the function and the loop stops. If the input string is not empty, it is passed to the `yield` statement, which returns the input string to the calling code and continues to execute the next iteration of the loop.

The `for` loop uses the `user_input()` function to get user input and the `map()` function to call the `str.upper()` method on each input string to convert it to uppercase and the `print()` function to print it out.

يعرّف هذا الكود دالة `user_input()` لا تأخذ أي وسيطات وتطلب باستمرار من المستخدم إدخالها حتى توفر سلسلة فارغة (فقط بالضغط على `enter`). يتم بعد ذلك تمرير كل سلسلة إدخال يقدمها المستخدم إلى طريقة `str.upper()` لتحويل السلسلة إلى أحرف كبيرة ثم يتم تمريرها إلى دالة `print()` لطباعتها. تطبق الدالة `map()` طريقة `str.upper()` على كل سلسلة إدخال.

تستخدم حلقة `while` في الدالة `user_input()` دالة `input()` لمطالبة المستخدم باستمرار بالإدخال وتعيين سلسلة الإدخال إلى المتغير `s`.

يتحقق سطر `if not s:` مما إذا كانت سلسلة الإدخال فارغة أم لا. إذا كانت فارغة، يتم تنفيذ تعليمة الإرجاع `return`، والتي تنهي الدالة وتتوقف الحلقة. إذا لم تكن سلسلة الإدخال فارغة، يتم تمريرها إلى جملة `yield`، والذي يعيد سلسلة الإدخال إلى كود الاستدعاء ويستمر في تنفيذ التكرار التالي للحلقة.

تستخدم حلقة `for` دالة `user_input()` للحصول على مدخلات المستخدم ودالة `map()` لاستدعاء طريقة `str.upper()` على كل سلسلة إدخال لتحويلها إلى أحرف كبيرة ودالة `print()` لطباعتها.

التمرين 10

الامر:

Write a program that accepts a sequence of whitespace separated words as input and prints the words after removing all duplicate words and sorting them alphanumerically.

اكتب برنامجًا يقبل سلسلة من الكلمات المفصولة بمسافات بيضاء كمدخلات ويطبع الكلمات بعد إزالة جميع الكلمات المكررة وفرزها أبجديًا رقميًا.

الكود:

```
word = sorted(
    list(set(input().split()))
)
print(" ".join(word))
```

شرح ChatGPT:

This code is a simple program that takes a string of words as input, sorts them in alphabetical order and removes any duplicates, and then prints out the unique and sorted words as a string.

The first line of code uses the `input()` function to get user input, which is expected to be a string of words, and the `split()` method to convert the input string into a list of words by splitting it at each space.

Then it converts the list of words into a set with the `set()` method, which removes any duplicates. Then the set is converted back to a list with `list()` method.

The second line sorts the list of words using the `sorted()` method. This sorts the list in alphabetical order.

The final line of code uses the `join()` method to join the list of words into a string of words separated by space, and then prints it out.

هذا الكود عبارة عن برنامج بسيط يأخذ سلسلة من الكلمات كمدخلات، ويصنفها بترتيب أبجدي ويزيل أي تكرارات، ثم يطبع الكلمات الفريدة والمصنفة كسلسلة.

يستخدم السطر الأول من الكود دالة `input()` للحصول على مدخلات المستخدم، والتي من المتوقع أن تكون سلسلة من الكلمات ، وطريقة `split()` لتحويل سلسلة الإدخال إلى قائمة كلمات عن طريق تقسيمها في كل مسافة.

ثم يقوم بتحويل قائمة الكلمات إلى مجموعة باستخدام طريقة `set()` التي تزيل أي تكرارات. ثم يتم تحويل المجموعة مرة أخرى إلى قائمة بطريقة `list()`.

يفرز السطر الثاني قائمة الكلمات باستخدام طريقة `sorted()`. يقوم هذا بفرز القائمة بالترتيب الأبجدي.

يستخدم السطر الأخير من الكود طريقة `join()` لربط قائمة الكلمات في سلسلة من الكلمات مفصولة بمسافة، ثم يطبعها.

التمرين 11

الامر:

Write a program which accepts a sequence of comma separated 4 digit binary numbers as its input and then check whether they are divisible by 5 or not. The numbers that are divisible by 5 are to be printed in a comma separated sequence.

اكتب برنامجًا يقبل سلسلة من الأرقام الثنائية المكونة من 4 أرقام مفصولة بفاصلة كمدخلات، ثم تحقق مما إذا كانت قابلة للقسمة على 5 أم لا. الأرقام التي يمكن القسمة على 5 يجب طباعتها في تسلسل مفصول بفاصلة.

الكود:

```
data = input().split(",")
data = list(
    filter(lambda i: int(i, 2) % 5 == 0, data)
) # lambda is an operator that helps to write function of one line
print(",".join(data))
```

شرح ChatGPT:

This code takes a string of comma-separated binary numbers as input, filters out the numbers that are not divisible by 5 when converted to decimal, and then prints the remaining numbers as a comma-separated string.

The first line of code uses the `input()` function to get user input, which is expected to be a string of comma-separated binary numbers. It then uses the `split()` method to convert the input string into a list of binary numbers by splitting it at each comma.

The next line of code uses the `filter()` function to filter out the binary numbers that are not divisible by 5 when converted to decimal. `filter()` function applies the lambda function to each element in the list, the lambda function takes one argument `i`, converts it to decimal using `int(i, 2)` and check whether it's divisible by 5 or not, if it's true it keeps the element otherwise it filters out the element.

Finally, the `join()` method is used to join the filtered list of binary numbers into a string of comma-separated binary numbers and then the string is printed out.

يأخذ هذا الكود سلسلة من الأرقام الثنائية مفصولة بفواصل كإدخال، ويصفي الأرقام غير القابلة للقسم على 5 عند تحويلها إلى رقم عشري، ثم يطبع الأرقام المتبقية كسلسلة مفصولة بفواصل.

يستخدم السطر الأول من الكود دالة `input()` للحصول على مدخلات المستخدم، والتي من المتوقع أن تكون سلسلة من الأرقام الثنائية مفصولة بفواصل. ثم تستخدم طريقة `split()` لتحويل سلسلة الإدخال إلى قائمة من الأرقام الثنائية بتقسيمها عند كل فاصلة.

يستخدم السطر التالي من التعليمات البرمجية دالة `filter()` لتصفية الأرقام الثنائية التي لا تقبل القسمة على 5 عند تحويلها إلى رقم عشري. تقوم دالة `filter()` بتطبيق دالة `lambda` على كل عنصر في القائمة، وتأخذ دالة `lambda` وسيطة واحدة `i`، وتحويلها إلى رقم عشري باستخدام `int(i)` (2) وتحقق مما إذا كانت قابلة للقسم على 5 أم لا، وإذا كان هذا صحيحًا، فستحتفظ بالعنصر وإلا فإنها تقوم بتصفية العنصر.

أخيرًا، يتم استخدام طريقة `join()` للانضمام إلى قائمة الأرقام الثنائية التي تمت تصفيتها في سلسلة من الأرقام الثنائية المفصولة بفواصل ثم يتم طباعة السلسلة.

التمرين 12

الامر:

Write a program, which will find all such numbers between 1000 and 3000 (both included) such that each digit of the number is an even number. The numbers obtained should be printed in a comma-separated sequence on a single line.

اكتب برنامجًا سيجد كل هذه الأرقام بين 1000 و 3000 (كلاهما مضمنين) بحيث يكون كل رقم من الرقم عددًا زوجيًا، ويجب طباعة الأرقام التي تم الحصول عليها في تسلسل مفصول بفاصلة على سطر واحد.

الكود:

```
lst = [str(i) for i in range(1000, 3001)]
lst = list(
    filter(lambda i: all(ord(j) % 2 == 0 for j in i), lst)
) # using lambda to define function inside filter function
print(", ".join(lst))
```

شرح ChatGPT:

This code generates a list of numbers between 1000 and 3000 and filters out the numbers that have any odd-numbered digits in them.

The first line of code uses a list comprehension to generate a list of numbers between 1000 and 3000 and converts them to strings.

The next line of code uses the `filter()` function to filter out the numbers that have any odd-numbered digits in them. `filter()` function applies the lambda function to each element in the list, the lambda function takes one argument `i`, and for each digit `j` in `i`, it checks whether the ASCII value of `j` is even using the `ord()` function and the modulus operator `%`. If all digits are even, the lambda function returns `True`, and the number is kept in the list, otherwise it filters out the number.

Finally, the `join()` method is used to join the filtered list of numbers into a string of comma-separated numbers and then the string is printed out.

يشيء هذا الكود قائمة بالأرقام بين 1000 و 3000 ويصفي الأرقام التي تحتوي على أي أرقام فردية.

يستخدم السطر الأول من التعليمات البرمجية List Comprehension لإنشاء قائمة من الأرقام بين 1000 و 3000 وتحويلها إلى سلاسل نصية.

يستخدم السطر التالي من الكود دالة `filter()` لتصفية الأرقام التي تحتوي على أي أرقام فردية. تقوم دالة `filter()` بتطبيق دالة `lambda` على كل عنصر في القائمة، وتأخذ دالة `lambda` وسيطة واحدة `i`، ولكل رقم `i` في `i`، فإنها تتحقق مما إذا كانت قيمة ASCII لـ `i` تستخدم حتى الدالة `ord()` وعامل المعامل `%`. إذا كانت جميع الأرقام زوجية، فإن دالة `lambda` ترجع `True`، ويتم الاحتفاظ بالرقم في القائمة، وإلا فإنها تقوم بتصفية الرقم.

أخيراً، يتم استخدام طريقة `join()` للانضمام إلى قائمة الأرقام التي تمت تصفيتها في سلسلة من الأرقام المفصولة بفواصل ثم يتم طباعة السلسلة.

التمرين 13

الامر:

Write a program that accepts a sentence and calculate the number of letters and digits.

اكتب برنامجاً يقبل جملة واحسب عدد الأحرف والأرقام.

الكود:

```
word = input()
letter, digit = 0, 0

for i in word:
    if i.isalpha(): # returns True if alphabet
        letter += 1
    elif i.isnumeric(): # returns True if numeric
        digit += 1
print(
    f"LETTERS {letter}\n{digits}"
) # two different types of formatting method is shown in both
solution
```

شرح ChatGPT:

This code takes an input string, and then it iterates over each character in the string.

For each character, it checks if the character is an alphabet using the `isalpha()` method. If it is an alphabet, the variable `letter` is incremented by 1.

Then, it checks if the character is a digit using the `isnumeric()` method. If it is a digit, the variable `digit` is incremented by 1.

Finally, it prints the count of letters and digits in the input string. `f"LETTERS {letter}\n{digits}"` is used to format the output string where `letter` and `digit` are the variables that contain the count of letters and digits respectively in the input word.

يأخذ هذا الكود سلسلة إدخال، ثم يتكرر على كل حرف في السلسلة.

لكل حرف، يتحقق مما إذا كان الحرف أبجدياً باستخدام طريقة `isalpha()`. إذا كانت أبجدية، فإن الحرف المتغير يزداد بمقدار 1.

ثم يتحقق مما إذا كان الحرف رقمًا باستخدام طريقة `isnumeric()`. إذا كان رقمًا، فسيتم زيادة الرقم المتغير بمقدار 1.

أخيرًا، يقوم بطباعة عدد الأحرف والأرقام في سلسلة الإدخال. يتم استخدام `f"LETTERS {letter}\n{digits}"` لتنسيق سلسلة الإخراج حيث يمثل الحرف والرقم المتغيرات التي تحتوي على عدد الأحرف والأرقام على التوالي في كلمة الإدخال.

التمرين 14

الامر:

Write a program that accepts a sentence and calculate the number of upper-case letters and lower-case letters.

اكتب برنامجاً يقبل جملة واحسب عدد الأحرف الكبيرة والصغيرة.

الكود:

```
string = input("Enter the sentence")
upper = 0
lower = 0
for x in string:
    if x.isupper() == True:
        upper += 1
    if x.islower() == True:
        lower += 1

print("UPPER CASE: ", upper)
print("LOWER CASE: ", lower)
```

شرح ChatGPT:

This code takes an input string, and then it iterates over each character in the string.

For each character, it checks if the character is uppercase using the `isupper()` method. If it is uppercase, the variable `upper` is incremented by 1.

Then, it checks if the character is lowercase using the `islower()` method. If it is lowercase, the variable `lower` is incremented by 1.

Finally, it prints the count of uppercase and lowercase characters in the input string.

يأخذ هذا الكود سلسلة إدخال، ثم يتكرر على كل حرف في السلسلة.

لكل حرف، يتحقق مما إذا كان الحرف كبيراً باستخدام طريقة `isupper()`. إذا كانت الأحرف كبيرة، تتم زيادة المتغير `upper` بمقدار 1.

ثم يتحقق مما إذا كان الحرف صغيراً باستخدام طريقة `islower()`. إذا كانت صغيرة، فإن المتغير `lower` يزداد بمقدار 1.

أخيراً، يطبع عدد الأحرف الكبيرة والصغيرة في سلسلة الإدخال.

التمرين 15

الامر:

Write a program that computes the value of $a+aa+aaa+aaaa$ with a given digit as the value of a .

اكتب برنامجاً يحسب قيمة $a + aa + aaa + aaaa$ برقم معطى كقيمة a .

الكود:

```
a = input()
total, tmp = 0, str() # initialing an integer and empty string

for i in range(4):
    tmp += a # concatenating 'a' to 'tmp'
    total += int(tmp) # converting string type to integer type

print(total)
```

شرح ChatGPT:

This code takes an input, and then assigns it to a variable a .

It then initializes two variables, `total` and `tmp`, with the values of 0 and empty string respectively.

The code then uses a for loop that iterates 4 times. In each iteration, it concatenates the variable a to the variable `tmp` and then converts it to an integer and adds it to the variable `total`.

At the end, it prints the total.

يأخذ هذا الكود مدخلاً، ثم يعينه إلى متغير a .

ثم يقوم بتهيئة متغيرين، إجمالي و `tmp`، بقيم 0 وسلسلة فارغة على التوالي.

يستخدم الكود بعد ذلك حلقة `for` تتكرر 4 مرات. في كل تكرار، يربط المتغير a بالمتغير `tmp` ثم يحوله إلى عدد صحيح ويضيفه إلى إجمالي المتغير.

في النهاية، تطبع المجموع.

التمرين 16

الامر:

Use a list comprehension to square each odd number in a list. The list is input by a sequence of comma-separated numbers. > Suppose the following input is supplied to the program.

استخدم List Comprehension لترتيب كل رقم فردي في القائمة. يتم إدخال القائمة من خلال تسلسل من الأرقام المفصولة بفواصل <. افترض أنه تم توفير المدخلات التالية للبرنامج.

الكود:

```
lst = [str(int(i) ** 2) for i in input().split(",") if int(i) % 2]
print(", ".join(lst))
```

شرح ChatGPT:

This code takes an input, which is expected to be a comma-separated list of numbers, converts it into a list of strings, then it applies a list comprehension on it.

The list comprehension iterates over each element of the input list, checks if it is odd by checking if the integer representation of the element modulo 2 is non-zero. If it is odd, it takes the square of the element and converts it back to a string before appending it to the new list.

At the end, it uses the join method to join the elements of the new list with commas and prints it.

يأخذ هذا الكود مُدخلاً، يُتوقع أن يكون قائمة أرقام مفصولة بفواصل، ويحولها إلى قائمة سلاسل نصية، ثم يطبق عليها List Comprehension.

يتكرر List Comprehension على كل عنصر من عناصر قائمة الإدخال، ويتحقق مما إذا كان فردياً عن طريق التحقق مما إذا كان تمثيل الأعداد الصحيحة لباقي القسمة على 2 غير صفري. إذا كان الأمر فردياً، فإنه يأخذ مربع العنصر ويحوله مرة أخرى إلى سلسلة قبل إلحاقه بالقائمة الجديدة.

في النهاية، تستخدم طريقة join لربط عناصر القائمة الجديدة بفواصل وطباعتها.

التمرين 17

الامر:

Write a program that computes the net amount of a bank account based a transaction log from console input. The transaction log format is shown as following:

اكتب برنامجًا يحسب المبلغ الصافي للحساب المصرفي استنادًا إلى سجل المعاملات من إدخال وحدة التحكم. يظهر تنسيق سجل المعاملات على النحو التالي:

D 100

W 200

D means deposit while W means withdrawal.

D تعني الإيداع بينما W تعني السحب.

الكود:

```
lines = []
while True:
    loopInput = input()
    if loopInput == "done":
        break
    else:
        lines.append(loopInput)

lst = list(int(i[2:]) if i[0] == "D" else -int(i[2:]) for i in
lines)
print(sum(lst))
```

شرح ChatGPT:

This code snippet first creates an empty list called `lines`.

It then enters a `while` loop that continues until the user inputs "done". For each iteration of the loop, the script takes in user input and checks if it is "done". If it is not, it appends the input to the `lines` list.

Once the loop breaks, the script creates a new list called `lst` using list comprehension. It iterates through each element of `lines` and checks if the first character of the element is "D". If it is, it adds the element as is to the new list, otherwise it adds the element as a negative number.

Finally, the script calculates the sum of the elements in the `lst` list and prints it.

ينشئ مقتطف الكود هذا أولاً قائمة فارغة تسمى `lines`.

ثم يدخل في حلقة `while` التي تستمر حتى يُدخل المستخدم `"done"`. لكل تكرار للحلقة، يأخذ السكريبت مدخلات المستخدم ويتحقق مما إذا كان `"done"`. إذا لم يكن كذلك، فإنه يقوم بإلحاق الإدخال بقائمة الأسطر.

بمجرد أن تنكسر الحلقة، يقوم السكريبت بإنشاء قائمة جديدة تسمى `lst` باستخدام `List Comprehension`. يتكرر خلال كل عنصر من عناصر السطور ويتحقق مما إذا كان الحرف الأول للعنصر هو `"D"`. إذا كان الأمر كذلك، فإنه يضيف العنصر كما هو إلى القائمة الجديدة، وإلا فإنه يضيف العنصر كرقم سالب.

أخيراً، يحسب السكريبت مجموع العناصر في القائمة الأولى ويطبعها.

التمرين 18

الامر:

A website requires the users to input username and password to register. Write a program to check the validity of password input by users. Following are the criteria for checking the password:

- At least 1 letter between [a-z]
- At least 1 number between [0-9]
- At least 1 letter between [A-Z]
- At least 1 character from [!@#\$%&]
- Minimum length of transaction password: 6
- Maximum length of transaction password: 12

Your program should accept a sequence of comma separated passwords and will check them according to the above criteria. Passwords that match the criteria are to be printed, each separated by a comma:

يتطلب موقع الويب من المستخدمين إدخال اسم المستخدم وكلمة المرور للتسجيل. اكتب برنامجاً للتحقق من صحة إدخال كلمة المرور من قبل المستخدمين. فيما يلي معايير التحقق من كلمة المرور:

- حرف واحد على الأقل بين [a-z]
- رقم واحد على الأقل بين [0-9]
- حرف واحد على الأقل بين [A-Z]
- حرف واحد على الأقل من [!@#\$%&]
- الحد الأدنى لطول كلمة المرور للمعاملة: 6
- الحد الأقصى لطول كلمة المرور للمعاملة: 12

يجب أن يقبل برنامجك سلسلة من كلمات المرور مفصولة بفواصل وسيقوم بفحصها وفقاً للمعايير المذكورة أعلاه. يجب طباعة كلمات المرور التي تطابق المعايير، كل منها مفصول بفاصلة.

الكود:

```
def is_low(x): # Returns True if the string has a lowercase
    for i in x:
        if "a" <= i and i <= "z":
            return True
```

```
return False

def is_up(x): # Returns True if the string has a uppercase
    for i in x:
        if "A" <= i and i <= "Z":
            return True
    return False

def is_num(x): # Returns True if the string has a numeric digit
    for i in x:
        if "0" <= i and i <= "9":
            return True
    return False

def is_other(x): # Returns True if the string has any "$#@"
    for i in x:
        if i == "$" or i == "#" or i == "@":
            return True
    return False

s = input().split(",")
lst = []

for i in s:
    length = len(i)
    if (
        6 <= length
        and length <= 12
        and is_low(i)
        and is_up(i)
        and is_num(i)
        and is_other(i)
    ): # Checks if all the requirments are fulfilled
        lst.append(i)

print(",".join(lst))
```

شرح ChatGPT:

This script takes input of a string of comma-separated values, and checks each value against certain conditions. The conditions are:

1. The length of the string is between 6 and 12 characters
2. The string contains at least one lowercase letter
3. The string contains at least one uppercase letter
4. The string contains at least one numeric digit

5. The string contains at least one of the special characters "\$#@".

The script uses 4 separate functions (`is_low`, `is_up`, `is_num`, `is_other`) to check if each string meets each of these requirements.

If a value meets all these conditions, it is added to a list. Finally, the script prints all the values in the list, joined by commas.

The function `is_low(x)` checks if the input string `x` has any lowercase letters by iterating through each character in the string and checking if it falls within the range of lowercase letters. It returns `True` if at least one lowercase letter is found, and `False` otherwise.

The function `is_up(x)` returns `True` if the input string `x` contains at least one uppercase letter, and `False` otherwise. It iterates through each character in the input string, and checks if it is within the range of uppercase letters using the comparison operator `<=`. If it finds an uppercase letter, it immediately returns `True`. If it reaches the end of the input string without finding an uppercase letter, it returns `False`.

The function `is_num(x)` takes in a single argument `x` which is a string. The function then uses a `for` loop to iterate through each character in the string `x`. Within the `for` loop, there is an `if` statement that checks if the current character being iterated through is a numeric digit (i.e. between 0 and 9). If the current character is a numeric digit, the function immediately returns `True`. If the `for` loop completes without finding a numeric digit, the function returns `False` at the end. In other words, this function checks if the input string contains any numeric digits, and returns `True` if it does and `False` if it does not.

The function `is_other(x)` takes in a single input, `x`. The function then iterates through each character in the input `x` using a `for` loop. Inside the loop, the function checks if the current character is equal to any of the special characters `$`, `#`, or `@`. If any of these characters are present in the input, the function immediately returns `True`. If the loop completes without finding any of these characters, the function returns `False` indicating that none of the special characters were found in the input.

يأخذ هذا السكريبت مدخلات من سلسلة من القيم المفصولة بفواصل، ويتحقق من كل قيمة مقابل شروط معينة. الشروط هي:

- يتراوح طول السلسلة بين 6 و 12 حرفاً.

- تحتوي السلسلة على حرف صغير واحد على الأقل.
- تحتوي السلسلة على حرف كبير واحد على الأقل.
- تحتوي السلسلة على رقم رقمي واحد على الأقل.
- تحتوي السلسلة على واحد على الأقل من الأحرف الخاصة "\$#@".

يستخدم السكريبت 4 دوال منفصلة (`is_low`, `is_up`, `is_num`, `is_other`) للتحقق مما إذا كانت كل سلسلة تفي بكل من هذه المتطلبات.

إذا كانت القيمة تفي بكل هذه الشروط، تتم إضافتها إلى القائمة. أخيراً، يقوم السكريبت بطباعة جميع القيم الموجودة في القائمة، مع ربطها بفواصل.

تتحقق الدالة `is_low(x)` مما إذا كانت سلسلة الإدخال `x` تحتوي على أي أحرف صغيرة عن طريق التكرار خلال كل حرف في السلسلة والتحقق مما إذا كانت تقع ضمن نطاق الأحرف الصغيرة. تقوم بإرجاع `True` إذا تم العثور على حرف صغير واحد على الأقل، و `False` خلاف ذلك.

ترجع الدالة `is_up(x)` `True` إذا كانت سلسلة الإدخال `x` تحتوي على حرف كبير واحد على الأقل، وتعيد `False` بخلاف ذلك. يتكرر من خلال كل حرف في سلسلة الإدخال، ويتحقق مما إذا كان ضمن نطاق الأحرف الكبيرة باستخدام عامل المقارنة `<=`. إذا عثر على حرف كبير، فسيتم إرجاع `True` على الفور. إذا وصلت إلى نهاية سلسلة الإدخال دون العثور على حرف كبير، فإنها ترجع `False`.

تأخذ الدالة `is_num(x)` وسيطة واحدة `x` وهي سلسلة نصية. تستخدم الدالة بعد ذلك حلقة `for` للتكرار خلال كل حرف في السلسلة `x`. داخل حلقة `for`، توجد عبارة `if` التي تتحقق مما إذا كان الحرف الحالي الذي يتم التكرار خلاله هو رقم رقمي (أي بين 0 و 9). إذا كان الحرف الحالي عبارة عن رقم رقمي، تقوم الدالة بإرجاع `True` على الفور. إذا اكتملت حلقة `for` بدون العثور على رقم رقمي، ترجع الدالة `False` في النهاية. بمعنى آخر، تتحقق هذه الدالة مما إذا كانت سلسلة الإدخال تحتوي على أي أرقام رقمية، وتعيد `True` إذا كانت كذلك وتعيد `False` إذا لم تكن كذلك.

تأخذ الدالة `is_num(x)` في إدخال واحد، `x`. تقوم الدالة بعد ذلك بالتكرار خلال كل حرف في الإدخال `x` باستخدام حلقة `for`. داخل الحلقة، تتحقق الدالة مما إذا كان الحرف الحالي يساوي أي من الأحرف الخاصة '\$' أو '#' أو '@'. في حالة وجود أي من هذه الأحرف في الإدخال، تقوم الدالة بإرجاع `True` على الفور. إذا اكتملت الحلقة دون العثور على أي من هذه الأحرف، ترجع الدالة `False` مشيرة إلى عدم العثور على أي من الأحرف الخاصة في الإدخال.

التمرين 19

الامر:

You are required to write a program to sort the (name, age, score) tuples by ascending order where name is string, age and score are numbers. The tuples are input by console. The sort criteria is:

1. Sort based on name
2. Then sort based on age
3. Then sort by score

أنت مطالب بكتابة برنامج لفرز مجموعات (الاسم والعمر والنتيجة) بترتيب تصاعدي حيث يكون الاسم عبارة عن سلسلة نصية والعمر والنتيجة هي أرقام. يتم إدخال المجموعات بواسطة وحدة التحكم. معايير الفرز هي:

- فرز على أساس الاسم.
- ثم قم بالفرز على أساس العمر.
- ثم قم بالفرز حسب النتيجة.

الكود:

```
lst = []
while True:
    s = input().split(",")
    if not s[0]: # breaks for blank input
        break
    lst.append(tuple(s))

lst.sort(
    key=lambda x: (x[0], x[1], x[2])
) # here key is defined by lambda and the data is sorted by
element priority 0>1>2 in ascending order
print(lst)
```

شرح ChatGPT:

This code snippet is a Python program that prompts the user for input of strings separated by commas.

It creates an empty list called `lst` and enters a `while` loop. Within the `while` loop, it takes user input, splits it into a list of strings by the commas,

and assigns it to the variable `s`. If the first element of `s` is blank, the loop breaks. Otherwise, it appends a tuple of the elements of `s` to `lst`. After the loop, the program sorts `lst` by the first, second, and third elements of each tuple, and then prints `lst`.

مقتطف الكود هذا هو برنامج بايثون الذي يطالب المستخدم بإدخال سلاسل مفصولة بفواصل.

يقوم بإنشاء قائمة فارغة تسمى `lst` ويدخل في حلقة `while`. داخل حلقة `while`، يأخذ مدخلات المستخدم، ويقسمها إلى قائمة من السلاسل بواسطة الفواصل، ويخصصها للمتغير `s`. إذا كان العنصر الأول من `s` فارغاً، تنقطع الحلقة. وإلا فإنه يُلحق مجموعة من عناصر `s` بـ `lst`.

بعد الحلقة، يقوم البرنامج بفرز العنصر الأول والثاني والثالث من كل مجموعة، ثم يقوم بطباعة `lst`.

التمرين 20

الامر:

Define a class with a generator which can iterate the numbers, which are divisible by 7, between a given range 0 and n.

عرف فئة بمولد يمكنه تكرار الأرقام القابلة للقسمة على 7 بين نطاق معين 0 و n.

الكود:

```
class MyGen:
    def by_seven(self, n):
        for i in range(0, int(n / 7) + 1):
            yield i * 7

for i in MyGen().by_seven(int(input("Please enter a number... "))):
    print(i)
```

شرح ChatGPT:

This code defines a class called MyGen that has a method called by_seven. The method takes in a number n as an argument and uses a for loop to iterate over a range of numbers that are multiples of 7 and less than or equal to n. The method uses the yield keyword to return each multiple of 7 one at a time as the for loop iterates through the range.

After the class is defined, the code prompts the user to enter a number, and then it creates an instance of the MyGen class and calls the by_seven method with the user-provided number as an argument. The code then uses a for loop to print out each number generated by the by_seven method one at a time.

يعرّف هذا الكود فئة تسمى MyGen لها طريقة تسمى by_seven. تأخذ الطريقة الرقم n كوسيلة وتستخدم حلقة for للتكرار على نطاق من الأرقام التي تكون مضاعفات 7 وأقل من أو تساوي n. تستخدم الطريقة الكلمة الأساسية yield لإرجاع كل مضاعف لـ 7 واحداً في كل مرة حيث تتكرر حلقة for عبر النطاق.

بعد تحديد الفئة، يطالب الكود المستخدم بإدخال رقم، ثم يقوم بإنشاء مثيل لفئة MyGen ويستدعي طريقة by_seven مع الرقم الذي يوفره المستخدم كوسيلة. يستخدم الكود بعد ذلك حلقة for لطباعة كل رقم تم إنشاؤه بواسطة طريقة by_seven واحداً تلو الآخر.

التمرين 21

الامر:

A robot moves in a plane starting from the original point (0,0). The robot can move toward UP, DOWN, LEFT and RIGHT with a given steps. The trace of robot movement is shown as the following:

- UP 5
- DOWN 3
- LEFT 3
- RIGHT 2

The numbers after the direction are steps. Please write a program to compute the distance from current position after a sequence of movement and original point. If the distance is a float, then just print the nearest integer. Example: If the following tuples are given as input to the program:

يتحرك الروبوت في مستوى يبدأ من النقطة الأصلية (0,0). يمكن للروبوت أن يتحرك لأعلى ولأسفل ولليسار ولليمين بخطوات معينة. يظهر أثر حركة الروبوت على النحو التالي:

- أعلى 5.
- أسفل 3.
- يسار 3.
- يمين 2.

الأرقام التي تلي الاتجاه هي خطوات. يرجى كتابة برنامج لحساب المسافة من الموقع الحالي بعد سلسلة من الحركة والنقطة الأصلية. إذا كانت المسافة عددًا عشريًا، فما عليك سوى طباعة أقرب عدد صحيح. مثال: إذا تم إعطاء المجموعات التالية كمدخلات للبرنامج:

الكود:

```
import math

x, y = 0, 0
while True:
    s = input().split()
    if not s:
        break
```



```

if s[0] == "UP": # s[0] indicates command
    x -= int(s[1]) # s[1] indicates unit of move
if s[0] == "DOWN":
    x += int(s[1])
if s[0] == "LEFT":
    y -= int(s[1])
if s[0] == "RIGHT":
    y += int(s[1])
    # N**P means N^P
dist = round(
    math.sqrt(x ** 2 + y ** 2)
) # euclidean distance = square root of (x^2+y^2) and rounding it
to nearest integer
print(dist)

```

شرح ChatGPT:

This code snippet is a Python program that calculates the distance from the origin $(0,0)$ to a point on the x-y plane.

The program takes input as a series of commands (UP, DOWN, LEFT, RIGHT) and an integer value indicating the number of units to move in that direction. The input is repeatedly taken in a while loop until a blank input is given.

The program keeps track of the x and y coordinates as the input commands are executed and then calculates the distance from the origin to the final point using the Pythagorean theorem (the square root of $x^2 + y^2$). The calculated distance is rounded off and then printed.

مقتطف الكود هذا هو برنامج بايثون الذي يحسب المسافة من الأصل $(0,0)$ إلى نقطة على المستوى x-y.

يأخذ البرنامج المدخلات كسلسلة من الأوامر (UP، DOWN، LEFT، RIGHT) وقيمة عددية تشير إلى عدد الوحدات للتحرك في هذا الاتجاه. يتم أخذ الإدخال بشكل متكرر في حلقة while حتى يتم إعطاء إدخال فارغ.

يتتبع البرنامج إحداثيات x و y حيث يتم تنفيذ أوامر الإدخال ثم يحسب المسافة من نقطة الأصل إلى النقطة النهائية باستخدام نظرية فيثاغورس (الجذر التربيعي لـ $x^2 + y^2$). يتم تقريب المسافة المحسوبة ثم طباعتها.

التمرين 22

الامر:

Write a program to compute the frequency of the words from the input. The output should output after sorting the key alphanumerically:

اكتب برنامجًا لحساب تكرار الكلمات من الإدخال. يجب أن يتم إخراج الإخراج بعد فرز المفتاح أبجديًا رقميًا:

الكود:

```
from pprint import pprint

p = input().split()
pprint({i: p.count(i) for i in p})
```

شرح ChatGPT:

This code snippet in Python is used for counting the occurrences of each element in a given list and printing the results in a dictionary format.

The input is taken as a string of elements separated by a space, which is then split into a list of elements using the `split` method. The `pprint` module is imported to format the output as a dictionary. A dictionary comprehension is used to iterate over the list of elements and count the occurrences of each element. The resulting dictionary is then printed using the `pprint` function.

يتم استخدام مقتطف الكود هذا في بايثون لحساب تكرارات كل عنصر في قائمة معينة وطباعة النتائج بتنسيق قاموس.

يتم أخذ الإدخال كسلسلة من العناصر مفصولة بمسافة، والتي يتم تقسيمها بعد ذلك إلى قائمة من العناصر باستخدام طريقة `split`. يتم استيراد وحدة `pprint` لتنسيق الإخراج كقاموس. يتم استخدام `dictionary comprehension` للتكرار على قائمة العناصر وإحصاء تكرارات كل عنصر. ثم تتم طباعة القاموس الناتج باستخدام دالة `pprint`.

التمرين 23

الامر:

Write a method which can calculate square value of number:

اكتب طريقة يمكنها حساب القيمة التربيعية للعدد:

الكود:

```
n = int(input())  
print(n ** 2)
```

[شرح ChatGPT:](#)

This code snippet prompts the user to input an integer, and then raises that integer to the power of 2 (n^2) and prints the result. For example, if the user inputs the number 5, the program would calculate 5^2 and print 25.

يطلب مقتطف الكود هذا المستخدم بإدخال عدد صحيح، ثم يرفع هذا العدد الصحيح إلى أس 2 (n^2) ويطبع النتيجة. على سبيل المثال، إذا أدخل المستخدم الرقم 5، فسيحسب البرنامج 5^2 ويطبع 25.

التمرين 24

الامر:

Please write a program to print some Python built-in functions documents, such as `abs()`, `int()`, `raw_input()`. And add document for your own function:

يرجى كتابة برنامج لطباعة بعض مستندات الدوال المضمنة في بايثون، مثل `abs()` ، `int()` ، `raw_input()` وإضافة وثيقة للدالتك الخاصة:

الكود:

```
def pow(n, p):
    """
    param n: This is any integer number
    param p: This is power over n
    return: n to the power p = n^p
    """
    return n ** p

print(pow(3, 4))
print(pow.__doc__)
```

شرح ChatGPT:

This is a Python function definition for a function called `pow` that takes in two parameters, `n` and `p`.

The function calculates and returns the result of raising the value of `n` to the power of `p` (`n` to the power of `p` = n^p). The function also has a `docstring` that describes the purpose and parameters of the function.

The last two lines of the code snippet are calling the `pow` function with the values 3 and 4 as arguments, which will return $3^4 = 81$ and also prints the `docstring` of the function.

هذا تعريف دالة بايثون لدالة تسمى `pow` تأخذ معلمتين، `n` و `p`.
تحتسب الدالة وترجع نتيجة رفع قيمة `n` إلى أس `p` (`n` إلى أس `p` = n^p). تحتوي الدالة أيضاً على سلسلة `docstring` التي تصف الغرض من الدالة ومعلوماتها.
يستدعي آخر سطرين من مقتطف الكود الدالة `pow` مع القيمتين 3 و 4 كوسيطات، والتي ستعيد $3^4 = 81$ وتطبع أيضاً سلسلة `docstring` كدالة.

التمرين 25

الامر:

Define a class, which have a class parameter and have a same instance parameter:

عرف فئة، لها معلمة فئة ولها نفس معامل الحالة:

الكود:

```
class Car:
    name = "Car"
    def __init__(self, name=None):
        self.name = name

honda = Car("Honda")
print(f"{Car.name} name is {honda.name}")

toyota = Car()
toyota.name = "Toyota"
print(f"{Car.name} name is {toyota.name}")
```

شرح ChatGPT:

This code defines a class called Car with a class variable name which is set to "Car" by default. When creating a new object of the Car class, you can pass in a custom name as an argument, otherwise it will use the default "Car" value. The first instance of the class is created and assigned to the variable honda. The class variable name is overridden with the value "Honda" that was passed in during the object's creation. Then a message is printed, displaying the class variable's value of "Car" and the object variable's value of "Honda".

The second instance of the class is created and assigned to the variable toyota. In this case, the name of the object is not passed during the object creation, so the default value of "Car" is used. Later, the object variable's value is overridden with "Toyota" and a message is printed, displaying the class variable's value of "Car" and the object variable's value of "Toyota".

يعرّف هذا الكود فئة تسمى Car مع متغير فئة name يتم تعيينه افتراضياً على "Car". عند إنشاء كائن جديد لفئة Car، يمكنك تمرير اسم مخصص كوسيلة، وإلا فسيستخدم قيمة "Car" الافتراضية.

يتم إنشاء أول مثيل للفئة وتعيينه إلى المتغير honda . تم تجاوز اسم متغير الفئة بالقيمة "Honda" التي تم تمريرها أثناء إنشاء الكائن. ثم يتم طباعة رسالة تعرض قيمة متغير الفئة "Car" وقيمة متغير الكائن "Honda".

يتم إنشاء المثل الثاني للفئة وتعيينه إلى المتغير toyota. في هذه الحالة، لا يتم تمرير اسم الكائن أثناء إنشاء الكائن، لذلك يتم استخدام القيمة الافتراضية لـ "Car". في وقت لاحق، تم تجاوز قيمة متغير الكائن بـ "Toyota" ويتم طباعة رسالة تعرض قيمة متغير الفئة "Car" وقيمة متغير الكائن "Toyota".

التمرين 26

الامر:

Define a function which can compute the sum of two numbers:

عرف دالة يمكنها حساب مجموع رقمين:

الكود:

```
sum = lambda n1, n2: n1 + n2 # here lambda is use to define little
function as sum
print(sum(1, 2))
```

شرح ChatGPT:

This code creates an anonymous function called sum that takes in two parameters, n1 and n2, and returns their sum. The function is then immediately called with the arguments 1 and 2, and the result of that function call (3) is printed.

يشي هذا الكود دالة مجهولة تسمى sum التي تأخذ معلمتين، n1 و n2، وتعيد مجموعهما. يتم استدعاء الدالة فوراً باستخدام الوسيطين 1 و 2، وتتم طباعة نتيجة استدعاء الدالة (3).

التمرين 27

الامر:

Define a function that can convert a integer into a string and print it in console:

عرف دالة يمكنها تحويل عدد صحيح إلى سلسلة نصية وطباعتها في وحدة التحكم:

الكود:

```
conv = lambda x: str(x)
n = conv(10)
print(n)
print(type(n)) # checks the type of the variable
```

شرح ChatGPT:

This code defines a lambda function named conv, which takes in a single argument x and returns the string representation of that argument.

It then assigns the value of the function applied to the integer 10 to a variable n. It then prints out the value of n and its data type using the print and type function respectively. It will output:

```
10
<class 'str'>
```

يعرّف هذا الكود دالة lambda المسماة conv، والتي تأخذ وسيطة واحدة x وتعيد تمثيل السلسلة لهذه الوسيطة.

ثم تقوم بتعيين قيمة الدالة المطبقة على العدد الصحيح 10 إلى متغير n. ثم يقوم بطباعة قيمة n ونوع البيانات الخاص به باستخدام دالة الطباعة والكتابة على التوالي. سيخرج:

```
10
<class 'str'>
```


التمرين 28

الامر:

Define a function that can receive two integer numbers in string form and compute their sum and then print it in console:

عرف دالة يمكنها تلقي رقمين صحيحين في شكل سلسلة وحساب مجموعهما ثم طباعته في وحدة التحكم:

الكود:

```
sum = lambda s1, s2: int(s1) + int(s2)
print(sum("10", "45")) # 55
```

شرح ChatGPT:

This code defines a lambda function called `sum`, which takes in two parameters `s1` and `s2` and converts them from strings to integers using the `int()` function.

Then it performs the sum of the integers and returns the result. Then, the code calls the `sum` function with the arguments "10" and "45" and prints the result, which is 55.

يعرف هذا الكود دالة lambda تسمى `sum`، والتي تأخذ معلمتين `s1` و `s2` وتحولهما من سلاسل نصية إلى أعداد صحيحة باستخدام دالة `int()`.

ثم ينفذ مجموع الأعداد الصحيحة ويعيد النتيجة. بعد ذلك، يستدعي الكود دالة `sum` مع الوسيطين "10" و "45" ويطبع النتيجة، وهي 55.

التمرين 29

الامر:

Define a function that can accept two strings as input and concatenate them and then print it in console:

عرف دالة يمكنها قبول سلسلتين كمدخلات وربطهما ثم طباعتها في وحدة التحكم:

الكود:

```
sum = lambda s1, s2: s1 + s2
print(sum("10", "45")) # 1045
```

شرح ChatGPT:

This code defines a lambda function named `sum` that takes in two arguments, `s1` and `s2`, and returns their concatenation as a string. Then it calls this lambda function with the arguments "10" and "45" and prints the result, which is "1045".

يعرّف هذا الكود دالة lambda المسماة `sum` التي تأخذ وسيطتين، `s1` و `s2`، وترجع تسلسلهما كسلسلة. ثم تستدعي هذه الدالة lambda بالعاملين "10" و "45" وتطبع النتيجة، وهي "1045".

التمرين 30

الامر:

Define a function that can accept two strings as input and print the string with maximum length in console. If two strings have the same length, then the function should print all strings line by line:

عرف دالة يمكنها قبول سلسلتين كمدخلات وطباعة السلسلة بأقصى طول في وحدة التحكم. إذا كانت هناك سلسلتان لهما نفس الطول، فيجب على الدالة طباعة كل السلاسل سطرًا بسطر:

الكود:

```
func = (
    lambda a, b: print(max((a, b), key=len))
    if len(a) != len(b)
    else print(a + "\n" + b)
)
```

شرح ChatGPT:

This code defines a lambda function `func` that takes in two arguments `a` and `b`.

The function then checks if the lengths of `a` and `b` are not equal. If they are not, it prints the argument with the larger length using the `max` function and the `len` key. If the lengths are equal, it prints both `a` and `b` on separate lines using string concatenation. Note that this function definition doesn't execute it.

You would need to call the function by passing two parameters.

يعرف هذا الكود دالة `func lambda` التي تأخذ وسيطتين `a` و `b`.

تتحقق الدالة بعد ذلك من عدم تساوي أطوال كل من `a` و `b`. إذا لم تكن كذلك، فإنها تطبع الوسيطة بطول أكبر باستخدام دالة `max` ومفتاح `len`. إذا كانت الأطوال متساوية، فإنها تطبع كلا من `a` و `b` على أسطر منفصلة باستخدام دمج السلسلة. لاحظ أن تعريف الدالة هذا لا ينفذها.

ستحتاج إلى استدعاء الدالة عن طريق تمرير معلمتين.

التمرين 31

الامر:

Define a function which can print a dictionary where the keys are numbers between 1 and 20 (both included) and the values are square of keys:

عرف دالة يمكنها طباعة قاموس حيث تكون المفاتيح أرقامًا بين 1 و 20 (كلاهما مضمن) والقيم مربعة من المفاتيح:

الكود:

```
def printDict():  
    dict = {i: i ** 2 for i in range(1, 21)}  
    print(dict)  
  
printDict()
```

شرح ChatGPT:

This code defines a function `printDict()`. When called, the function creates a dictionary, `dict`, with keys being numbers from 1 to 20, and values being the square of the corresponding key. It then prints the dictionary to the console.

يعرّف هذا الكود دالة `printDict()`. عند الاستدعاء، تنشئ الدالة قاموسًا، حيث تكون المفاتيح عبارة عن أرقام من 1 إلى 20، والقيم هي مربع المفتاح المقابل. ثم يقوم بطباعة القاموس على وحدة التحكم.

التمرين 32

الامر:

Define a function which can generate a dictionary where the keys are numbers between 1 and 20 (both included) and the values are square of keys. The function should just print the keys only:

عرف دالة يمكنها إنشاء قاموس حيث تكون المفاتيح أرقامًا بين 1 و 20 (كلاهما مضمن) والقيم مربعة من المفاتيح. يجب أن تقوم الدالة فقط بطباعة المفاتيح فقط:

الكود:

```
def printDict():
    dict = {i: i ** 2 for i in range(1, 21)}
    print(dict.keys()) # print keys of a dictionary

printDict()
```

شرح ChatGPT:

This code defines a function called `printDict()`. Inside the function, it creates a dictionary called `dict` with keys being integers from 1 to 20, and values being the square of the corresponding key.

Then it calls the `keys()` method on the `dict` object, which returns a view object of all the keys in the dictionary, and prints it. When the function is called, it will print all the keys in the dictionary (1, 2, 3, 4, ..., 20) without values.

يعرف هذا الكود دالة تسمى `printDict()`. داخل الدالة، يُنشئ قاموساً يُدعى `dict` مع كون المفاتيح أعداداً صحيحة من 1 إلى 20، والقيم هي مربع المفتاح المقابل.

ثم تستدعي طريقة `keys()` الموجودة في كائن `dict`، والتي تُعيد كائن عرض لجميع المفاتيح الموجودة في القاموس وتطبعه. عندما يتم استدعاء الدالة، ستقوم بطباعة جميع المفاتيح الموجودة في القاموس (1, 2, 3, 4, ..., 20) بدون قيم.

التمرين 33

الامر:

Define a function which can generate and print a list where the values are square of numbers between 1 and 20 (both included):

عرف دالة يمكنها إنشاء وطباعة قائمة حيث تكون القيم مربعة من الأرقام بين 1 و 20 (كلاهما مضمن):

الكود:

```
def printList():
    lst = [i ** 2 for i in range(1, 21)]
    print(lst)

printList()
```

شرح ChatGPT:

This code defines a function called `printList()` that creates a list of the squares of the numbers from 1 to 20 (inclusive). It then prints the list to the console.

When the function is called (by the last line `printList()`), it will execute the instructions inside it and print the list of the squares of the numbers from 1 to 20 to the console.

يعرف هذا الكود دالة تسمى `printList()` التي تنشئ قائمة بمربعات الأرقام من 1 إلى 20 (ضمنياً). ثم يقوم بطباعة القائمة إلى وحدة التحكم.

عندما يتم استدعاء الدالة (بواسطة آخر سطر `printList()`)، فإنها ستقوم بتنفيذ التعليمات الموجودة بداخلها وطباعة قائمة مربعات الأرقام من 1 إلى 20 إلى وحدة التحكم.

التمرين 34

الامر:

Define a function which can generate a list where the values are square of numbers between 1 and 20 (both included). Then the function needs to print the first 5 elements in the list:

عرف دالة يمكنها إنشاء قائمة حيث تكون القيم مربعة للأرقام بين 1 و 20 (كلاهما مضمن). ثم تحتاج الدالة إلى طباعة العناصر الخمسة الأولى في القائمة:

الكود:

```
func = lambda: ([i ** 2 for i in range(1, 21)][:5])
print(*(func()), sep="\n")
```

شرح ChatGPT:

This code defines a lambda function `func` that returns a list of the first 5 squares of the integers from 1 to 20.

The lambda function is immediately called and the result is unpacked and passed as separate arguments to the `print` function, which then separates the items in the list with newlines and prints them. The output will be a list of the first five square of numbers 1 to 20.

يعرّف هذا الكود دالة `func lambda` التي تُرجع قائمة المربعات الخمسة الأولى من الأعداد الصحيحة من 1 إلى 20.

يتم استدعاء دالة `lambda` على الفور ويتم تفريغ النتيجة وتميرها كوسائط منفصلة لدالة `print`، والتي تفصل بعد ذلك العناصر الموجودة في القائمة بأسطر جديدة وتطبعها. سيكون الإخراج عبارة عن قائمة بالمربعات الخمسة الأولى من الأرقام من 1 إلى 20.

التمرين 35

الامر:

Define a function which can generate a list where the values are square of numbers between 1 and 20 (both included). Then the function needs to print the last 5 elements in the list:

عرف دالة يمكنها إنشاء قائمة حيث تكون القيم مربعة للأرقام بين 1 و 20 (كلاهما مضمن). ثم تحتاج الدالة إلى طباعة آخر 5 عناصر في القائمة:

الكود:

```
def printList():
    lst = [i ** 2 for i in range(1, 21)]
    for i in range(19, 14, -1):
        print(lst[i])

printList()
```

شرح ChatGPT:

This function creates a list of the squares of the numbers from 1 to 20 using a list comprehension. Then, it uses a for loop to iterate through the list in reverse order, starting from the 19th index and going to the 14th index (with a step of -1, to count backwards). For each iteration, the function prints the value of the corresponding element in the list. So it will print the squares of the numbers 15 to 20 in descending order.

تشى هذه الدالة قائمة بمربعات الأرقام من 1 إلى 20 باستخدام list comprehension. بعد ذلك، تستخدم حلقة for للتكرار خلال القائمة بترتيب عكسي، بدءاً من الفهرس التاسع عشر والانتقال إلى الفهرس الرابع عشر (بخطوة -1، للعد تنازلي). لكل تكرار، تطبع الدالة قيمة العنصر المقابل في القائمة. لذلك ستطبع مربعات الأرقام من 15 إلى 20 بترتيب تنازلي.

التمرين 36

الامر:

Define a function which can generate a list where the values are square of numbers between 1 and 20 (both included). Then the function needs to print all values except the first 5 elements in the list:

عرف دالة يمكنها إنشاء قائمة حيث تكون القيم مربعة للأرقام بين 1 و 20 (كلاهما مضمن). ثم تحتاج الدالة إلى طباعة جميع القيم باستثناء العناصر الخمسة الأولى في القائمة:

الكود:

```
def printList():
    lst = [i ** 2 for i in range(1, 21)]
    for i in range(5, 20):
        print(lst[i])

printList()
```

شرح ChatGPT:

This code defines a function called `printList()`. Inside the function, it creates a list called `lst` that contains the squares of the numbers from 1 to 20. Then it uses a `for` loop to iterate through a range of numbers from 5 to 20 (not including 20), and for each iteration, it prints the element of the `lst` list at that index.

So it will print the elements of the list that are at index 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, and 19. Essentially, it is printing the last 15 elements of the list.

يعرف هذا الكود دالة تسمى `printList()`. داخل الدالة، تقوم بإنشاء قائمة تسمى `lst` تحتوي على مربعات الأرقام من 1 إلى 20. ثم تستخدم حلقة `for` للتكرار خلال نطاق من الأرقام من 5 إلى 20 (لا تتضمن 20)، ولكل تكرار، فإنها تطبع عنصر القائمة الأولى في ذلك الفهرس.

لذلك ستطبع عناصر القائمة الموجودة في الفهرس 5 و 6 و 7 و 8 و 9 و 10 و 11 و 12 و 13 و 14 و 15 و 16 و 17 و 18 و 19. بشكل أساسي، تطبع آخر 15 عنصرًا من القائمة.

التمرين 37

الامر:

Define a function which can generate and print a tuple where the value are square of numbers between 1 and 20 (both included):

عرف دالة يمكنها إنشاء صف وطباعته حيث تكون القيمة مربعة للأرقام بين 1 و 20 (كلاهما مضمن):

الكود:

```
def square_of_numbers():
    return tuple(i ** 2 for i in range(1, 21))

print(square_of_numbers())
```

شرح ChatGPT:

This code defines a function `square_of_numbers()` that creates and returns a tuple containing the squares of numbers from 1 to 20 (inclusive).

The function uses a generator expression to create the tuple, which is more memory efficient than creating a list and then converting it to a tuple. The function doesn't take any parameters. When the function is called and the tuple is returned, it is then printed to the console.

يعرف هذا الكود دالة `square_of_numbers()` التي تنشئ وتعيد صف يحتوي على مربعات الأرقام من 1 إلى 20 (ضمنياً).

تستخدم الدالة تعبيراً منشئاً لإنشاء الصف، وهي أكثر كفاءة في استخدام الذاكرة من إنشاء قائمة ثم تحويلها إلى صف. لا تأخذ الدالة أي معلمات. عندما يتم استدعاء الدالة وإعادة الصف، يتم طباعتها بعد ذلك على وحدة التحكم.

التمرين 38

الامر:

With a given tuple (1,2,3,4,5,6,7,8,9,10), write a program to print the first half values in one line and the last half values in one line:

باستخدام صف معين (1,2,3,4,5,6,7,8,9,10)، اكتب برنامجاً لطباعة قيم النصف الأول في سطر واحد وقيم النصف الأخيرة في سطر واحد:

الكود:

```
tup = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
lt = int(len(tup) / 2)
print(tup[:lt], tup[lt:])
```

شرح ChatGPT:

The code creates a tuple `tup` containing the numbers 1 to 10. Then it creates a variable `lt` that is the integer division of the length of the tuple by 2.

Finally, it prints out the first half of the tuple by slicing it from the start to the index of `lt` and the second half of the tuple by slicing it from the index of `lt` to the end.

The final output would be (1, 2, 3, 4, 5) (6, 7, 8, 9, 10).

يقوم الكود بإنشاء صف `tup` تحتوي على الأرقام من 1 إلى 10. ثم يقوم بإنشاء متغير `lt` وهو القسمة الصحيحة لطول الصف على 2.

أخيراً، يقوم بطباعة النصف الأول من الصف عن طريق تقسيمها من البداية إلى فهرس `lt` والنصف الثاني من الصف عن طريق تشريحها من فهرس `lt` إلى النهاية.

سيكون الناتج النهائي (1، 2، 3، 4، 5) (6، 7، 8، 9، 10).

التمرين 39

الامر:

Write a program to generate and print another tuple whose values are even numbers in the given tuple (1,2,3,4,5,6,7,8,9,10):

اكتب برنامجًا لتوليد وطباعة صف آخر تكون قيمه أرقامًا زوجية في الصف المعطى (1,2,3,4,5,6,7,8,9,10):

الكود:

```
tp1 = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
tp11 = tuple(i for i in tp1 if i % 2 == 0)
print(tp11)
```

شرح ChatGPT:

This code creates a tuple called `tp1` that contains the integers 1 through 10. It then creates a new tuple called `tp11` that contains only the even numbers from `tp1` using a list comprehension.

It iterates over each element in `tp1` and checks if it is divisible by 2 using the modulo operator. If it is, it adds it to the new tuple `tp11`.

Finally, it prints out the new tuple `tp11`, which should contain the numbers 2, 4, 6, 8, and 10.

ينشئ هذا الكود صف يسمى `tp1` يحتوي على الأعداد الصحيحة من 1 إلى 10. ثم يقوم بإنشاء صف جديدة تسمى `tp11` تحتوي فقط على الأرقام الزوجية من `tp1` باستخدام `list comprehension`.

يتكرر فوق كل عنصر في `tp1` ويتحقق مما إذا كان قابلاً للقسمة على 2 باستخدام عامل التشغيل `modulo`. إذا كان الأمر كذلك، فسيتم إضافته إلى مجموعة `tp11` الجديد.

أخيرًا، يقوم بطباعة الصف `tp11` الجديد، والذي يجب أن يحتوي على الأرقام 2 و 4 و 6 و 8 و 10.

التمرين 40

الامر:

Write a program which accepts a string as input to print "Yes" if the string is "yes" or "YES" or "Yes", otherwise print "No":

اكتب برنامجًا يقبل سلسلة كمدخلات لطباعة "Yes" إذا كانت السلسلة "yes" أو "YES" أو "Yes"، وإلا اطبع "No":

الكود:

```
input = input("Enter string:")
output = "".join(
    ["Yes" if input == "yes" or input == "YES" or input == "Yes"
    else "No"]
)
print(str(output))
```

شرح ChatGPT:

This code snippet prompts the user to enter a string and then assigns the input to a variable called `input`.

It then uses a `join` method to join a list of strings together. The list is created using a list comprehension that checks if the input is equal to "yes", "YES", or "Yes", if it is it returns "Yes" otherwise it returns "No".

The final result is then stored in the `output` variable and printed as a string. The output will be "Yes" if the user entered "yes", "YES", or "Yes" and "No" for any other input.

يطالب مقتطف الكود هذا المستخدم بإدخال سلسلة ثم يقوم بتعيين الإدخال إلى متغير يسمى `input`.

ثم يستخدم طريقة `join` للانضمام إلى قائمة السلاسل معًا. يتم إنشاء القائمة باستخدام `list comprehension` التي تتحقق مما إذا كان الإدخال يساوي "yes" أو "YES" أو "Yes"، إذا كانت تُرجع "Yes" وإلا فإنها تُرجع "No".

ثم يتم تخزين النتيجة النهائية في متغير الإخراج وطباعتها كسلسلة. سيكون الإخراج "Yes" إذا أدخل المستخدم "yes" أو "YES" أو "Yes" و "No" لأي إدخال آخر.

التمرين 41

الامر:

Write a program which can map() to make a list whose elements are square of elements in [1,2,3,4,5,6,7,8,9,10]:

اكتب برنامجًا يمكنه map() لعمل قائمة تكون عناصرها مربعة من العناصر في [1,2,3,4,5,6,7,8,9,10]:

الكود:

```
li = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
squaredNumbers = map(lambda x: x ** 2, li) # returns map type
object data
print(list(squaredNumbers)) # converting the object into list
```

شرح ChatGPT:

The code above uses the map() function to apply the lambda function lambda x: x ** 2 to each element in the list li.

The lambda function takes one argument x and returns the square of x. The map() function returns a map object, which is an iterator over the results of applying the lambda function to each element in the list. The list() function is then used to convert the map object into a list, so that the squared numbers can be printed using the print() function.

The output will be the list of squared numbers of the elements in the original list.

يستخدم الكود أعلاه دالة map() لتطبيق دالة lambda x: x ** 2 على كل عنصر في القائمة li.

تأخذ دالة lambda وسيطة واحدة x وتعيد مربع x. تُعيد الدالة map() كائن map، وهو مكرر لنتائج تطبيق دالة lambda على كل عنصر في القائمة. يتم بعد ذلك استخدام دالة list() لتحويل كائن map إلى قائمة، بحيث يمكن طباعة الأرقام المربعة باستخدام دالة print().

سيكون الناتج هو قائمة الأرقام التربيعية للعناصر في القائمة الأصلية.

التمرين 42

الامر:

Write a program which can map() and filter() to make a list whose elements are square of even number in [1,2,3,4,5,6,7,8,9,10].

اكتب برنامجاً يمكنه map() و filter() لعمل قائمة تكون عناصرها مربعة لعدد زوجي في [1,2,3,4,5,6,7,8,9,10]:

الكود:

```
def even(x):
    return x % 2 == 0

def squer(x):
    return x * x

li = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
li = map(
    squer, filter(even, li)
) # first filters number by even number and the apply map() on the
resultant elements
print(list(li))
```

شرح ChatGPT:

The code defines two functions: even(x) and square(x). The even function takes a single argument x and returns True if x is even and False otherwise. The square function takes a single argument x and returns x squared.

Then it creates a list of numbers called li. It then uses the map() function to apply the square function to each element in the list. But before applying the square function, it applies the filter function to the list, which filters out all the elements that are not even.

So the final output will be the square of even numbers of the list.

يعرف الكود دالتين: even(x) و square(x). تأخذ الدالة even وسيطة واحدة x وتعيد True إذا كانت x زوجية وتعيد False خلاف ذلك. تأخذ دالة square وسيطة واحدة x وتعيد تربيع x.

ثم يقوم بإنشاء قائمة أرقام تسمى `li`. ثم تستخدم الدالة `map()` لتطبيق دالة التربيع على كل عنصر في القائمة. ولكن قبل تطبيق دالة التربيع، يتم تطبيق دالة `filter` على القائمة، والتي تقوم بتصفية جميع العناصر غير المتساوية.

إذن، سيكون الناتج النهائي هو مربع الأرقام الزوجية في القائمة.

التمرين 43

الامر:

Write a program which can filter() to make a list whose elements are even number between 1 and 20 (both included).

اكتب برنامجاً يمكنه filter() لإنشاء قائمة تتكون من عناصر زوجية بين 1 و 20 (كلاهما مضمن):

الكود:

```
def even(x):
    return x % 2 == 0

evenNumbers = filter(even, range(1, 21))
print(list(evenNumbers))
```

شرح ChatGPT:

The code defines a function called even(x) that takes in a parameter x and returns True if x is even, and False if x is odd.

Then, it uses the filter() function to filter out all the even numbers in the range of 1 to 21 and assigns the resulting filter object to the variable evenNumbers.

Finally, it converts the filter object into a list and prints it. So the output will be a list of all even numbers between 1 and 21.

يعرّف الكود دالة تسمى even(x) تأخذ المعلمة x وترجع True إذا كانت x زوجية، وتعيد False إذا كانت x فردية.

بعد ذلك، تستخدم دالة filter() لتصفية جميع الأرقام الزوجية في النطاق من 1 إلى 21 وتعيين كائن filter الناتج إلى المتغير evenNumbers.

أخيراً، يقوم بتحويل كائن filter إلى قائمة ويطبعه. إذن سيكون الناتج عبارة عن قائمة بجميع الأعداد الزوجية بين 1 و 21.

التمرين 44

الامر:

Write a program which can map() to make a list whose elements are square of numbers between 1 and 20 (both included).

اكتب برنامجاً يمكنه map() لإنشاء قائمة تكون عناصرها مربعة من الأرقام بين 1 و 20 (كلاهما مضمّن):

الكود:

```
def sqr(x):
    return x * x

squaredNumbers = list(map(sqr, range(1, 21)))
print(squaredNumbers)
```

شرح ChatGPT:

In this code snippet, we are defining a function `sqr` that takes in a single argument `x` and returns the square of `x` by multiplying `x` with itself.

Next, we use the `map` function to apply the `sqr` function to every element in the range 1 to 21 (exclusive). This creates a map object, which is a lazy evaluation of the function applied to each element in the range.

To obtain the list of the squared numbers, we use the `list` function to convert the map object into a list and store it in the variable `squaredNumbers`. Finally, we print the `squaredNumbers` list to see the output, which is the list of squares of numbers from 1 to 20.

في مقتطف الكود هذا، نحدد دالة `sqr` تأخذ وسيطة واحدة `x` وتعيد مربع `x` بضرب `x` في نفسها.

بعد ذلك، نستخدم دالة `map` لتطبيق دالة `sqr` على كل عنصر في النطاق من 1 إلى 21 (حصرياً). يؤدي هذا إلى إنشاء كائن `map`، وهو تقييم كسول للدالة المطبقة على كل عنصر في النطاق.

للحصول على قائمة الأرقام المربعة، نستخدم دالة `list` لتحويل كائن الخريطة إلى قائمة وتخزينها في المتغير `squaredNumbers`.

أخيراً، نقوم بطباعة قائمة `squaredNumbers` لرؤية الإخراج، وهي قائمة مربعات الأرقام من 1 إلى 20.

التمرين 45

الامر:

Define a class named American which has a static method called printNationality.

عرف فئة تسمى American لها طريقة ثابتة تسمى printNationality.

الكود:

```
class American:
    @staticmethod
    def printNationality():
        print("I am American")

american = American()
american.printNationality()
American.printNationality()
```

شرح ChatGPT:

The code defines a class American in Python. The class has a static method called printNationality(). This method simply prints the string "I am American".

A object american is created from the American class. The object american invokes the printNationality() method using the object instance american.printNationality().

The class American directly invokes the static method printNationality() by calling American.printNationality(). Both of these method invocations produce the same output "I am American".

يعرف الكود الفئة American في بايثون. يحتوي الفئة على طريقة ثابتة تسمى printNationality(). هذه الطريقة تقوم ببساطة بطباعة السلسلة "I am American".

يتم إنشاء كائن American من الفئة American. يستدعي الكائن american طريقة printNationality() باستخدام مثل الكائن american.printNationality().

تستدعي الفئة American مباشرة الأسلوب الثابت printNationality() عن طريق استدعاء American.printNationality().

كل من هاتين الطريقتين تنتج نفس الناتج "I am American".

التمرين 46

الامر:

Define a class named American and its subclass NewYorker.

عرف فئة تسمى American وفئتها الفرعية NewYorker:

الكود:

```
class American:
    pass

class NewYorker(American):
    pass

american = American()
newyorker = NewYorker()

print(american)
print(newyorker)
```

شرح ChatGPT:

The code defines two classes American and NewYorker in Python. The American class is an empty class which doesn't have any attributes or methods. The NewYorker class is a subclass of American class.

In the code, two objects american and newyorker are created. The first object american is an instance of the American class, while the second object newyorker is an instance of the NewYorker class.

When the print statement is executed, the object american returns the memory location of the american object. The object newyorker returns the memory location of the newyorker object.

The output will be:

```
<__main__.American object at 0x...>
<__main__.NewYorker object at 0x...>
```

يعرف الكود فئتين American و NewYorker في بايثون. الفئة American هي فئة فارغة ليس لها أي سمات أو طرق. فئة NewYorker هي فئة فرعية من فئة American.

في الكود، تم إنشاء كائنين من american و newyorker. الكائن الأول american هو مثيل للفئة American، بينما الكائن الثاني newyorker هو مثيل لفئة NewYorker.

عند تنفيذ تعليمة `print`، يُرجع الكائن `american` موقع ذاكرة الكائن `american`. يقوم `newyorker` في الكائن بإرجاع موقع الذاكرة الخاص بكائن `newyorker`.

سيكون الإخراج:

```
<__main__.American object at 0x...>  
<__main__.NewYorker object at 0x...>
```

التمرين 47

الامر:

Define a class named Circle which can be constructed by a radius. The Circle class has a method which can compute the area.

عرف فئة تسمى Circle والتي يمكن إنشاؤها بواسطة نصف قطر radius. الفئة Circle لديه طريقة يمكنها حساب المساحة:

الكود:

```
class Circle:
    def __init__(self, r):
        self.radius = r

    def area(self):
        return 3.1416 * (self.radius ** 2)

circle = Circle(5)
print(circle.area())
```

شرح ChatGPT:

The code defines a class Circle with a constructor `__init__` method that takes a radius `r` as input and sets it as an instance variable `radius` of the object. The class also has a method `area` which calculates and returns the area of the circle using the formula $3.1416 * (self.radius ** 2)$. Finally, an object `circle` of class `Circle` is created with radius 5 and its area is calculated by calling `circle.area()`. The result of the area calculation is then printed.

يُعرف الكود فئة Circle باستخدام طريقة منشئ `__init__` يأخذ نصف قطر `r` كمدخل ويضعه كمثيل متغير `radius` للكائن. يحتوي الفئة أيضاً على طريقة `area` تقوم بحساب مساحة الدائرة وإرجاعها باستخدام الصيغة $3.1416 * (self.radius ** 2)$.

أخيراً، يتم إنشاء كائن `circle` من فئة Circle بنصف قطر 5 ويتم حساب مساحتها عن طريق استدعاء `circle.area()`. ثم يتم طباعة نتيجة حساب المنطقة.

التمرين 48

الامر:

Define a class named Rectangle which can be constructed by a length and width. The Rectangle class has a method which can compute the area.

عرف فئة تسمى Rectangle والتي يمكن إنشاؤها بواسطة الطول والعرض. فئة المستطيل لديها طريقة يمكنها حساب المساحة:

الكود:

```
class Rectangle:
    def __init__(self, l, w):
        self.length = l
        self.width = w

    def area(self):
        return self.length * self.width

rect = Rectangle(2, 4)
print(rect.area())
```

شرح ChatGPT:

This code defines a class Rectangle in Python. The class has a constructor method `__init__` which initializes two instance variables `length` and `width` with the values passed as parameters when a Rectangle object is created. The class also has a method `area` which calculates and returns the area of the rectangle by multiplying the `length` and `width` of the object.

Finally, the code creates an instance of the Rectangle class and assigns it to the variable `rect`. The area of the rectangle is then printed by calling the `area` method on the `rect` object. The output of the code will be 8.

يعرف هذا الكود فئة Rectangle في بايثون. يحتوي الفئة على طريقة منشيء `__init__` والتي تهيي متغيري مثل `length` و `width` مع القيم التي تم تمريرها كمعلومات عند إنشاء كائن Rectangle. يحتوي الفئة أيضاً على منطقة طريقة تقوم بحساب مساحة المستطيل وإرجاعها بضرب طول الكائن وعرضه.

أخيراً، ينشئ الكود مثيلاً لفئة Rectangle ويخصّصه للمتغير rect. يتم بعد ذلك طباعة مساحة المستطيل عن طريق استدعاء طريقة area على الكائن rect. سيكون إخراج الكود 8.

التمرين 49

الامر:

Define a class named Shape and its subclass Square. The Square class has an init function which takes a length as argument. Both classes have a area function which can print the area of the shape where Shape's area is 0 by default.

عرف فئة تسمى Shape والفئة الفرعية لها Square. فئة Square لها دالة init التي تأخذ طولاً كوسيلة. كلا الفئتين لهما دالة area يمكنها طباعة مساحة الشكل حيث تكون مساحة الشكل 0 بشكل افتراضي.

الكود:

```
class Shape:
    def __init__(self):
        pass

    def area(self):
        return 0

class Square(Shape):
    def __init__(self, length=0):
        Shape.__init__(self)
        self.length = length

    def area(self):
        return self.length * self.length

Asqr = Square(5)
print(Asqr.area())

print(Square().area())
```

شرح ChatGPT:

The code defines a class Shape that has an `__init__` method which takes no arguments, and an `area` method that returns 0.

It also defines a class Square that inherits from the Shape class. The Square class has an `__init__` method that takes an optional argument `length` and initializes the `length` attribute of the object. The Square class also overrides the `area` method of the Shape class to return the area of a square, which is calculated as the square of its length.

The code then creates an object `Asqr` of the `Square` class with the length 5 and prints its area by calling the `area` method on the object. The code also creates another object of the `Square` class with no arguments and prints its `area`, which is 0 because the length was not specified.

يعرّف الكود فئة `Square` التي لها طريقة `__init__` لا تأخذ أي وسيطات، وطريقة `area` التي ترجع 0.

كما أنها تحدد فئة `Square` التي ترث من فئة `Shape`. تحتوي فئة `Square` على طريقة `__init__` تأخذ وسيطة `length` اختياريًا وتهيئ سمة `length` للكائن. تتجاوز فئة `المربع` أيضًا طريقة المنطقة لفئة الشكل لإرجاع مساحة المربع، والتي يتم حسابها كمربع طوله.

يقوم الكود بعد ذلك بإنشاء كائن `Asqr` من فئة `Square` بطول 5 ويطبع مساحته عن طريق استدعاء طريقة `area` على الكائن. ينشئ الكود أيضًا كائنًا آخر من فئة `Square` بدون وسيطات ويطبع مساحته، وهي 0 لأن الطول لم يتم تحديده.

التمرين 50

الامر:

Please raise a RuntimeError exception.

يُرجى تطبيق استثناء RuntimeError.

الكود:

```
raise RuntimeError("something wrong")
```

شرح ChatGPT:

The code raises a RuntimeError exception with the message "something wrong". When this exception is raised, the program execution stops and an error message is displayed.

يطبق الكود استثناء RuntimeError مع الرسالة "something wrong". عند ظهور هذا الاستثناء، يتوقف تنفيذ البرنامج ويتم عرض رسالة خطأ.

التمرين 51

الامر:

Write a function to compute 5/0 and use try/except to catch the exceptions.

اكتب دالة لحساب 0/5 واستخدم try/except لالتقاط الاستثناءات.

الكود:

```
def divide():
    return 5 / 0

try:
    divide()
except ZeroDivisionError as ze:
    print("Why on earth you are dividing a number by ZERO!!")
except:
    print("Any other exception")
```

شرح ChatGPT:

The code block implements a simple try-except block to handle exceptions raised during the execution of the divide function.

The divide function returns the result of dividing 5 by 0, which will raise a ZeroDivisionError exception.

In the try block, we call the divide function, and in the following except block, we handle the ZeroDivisionError exception by printing a custom error message: "Why on earth you are dividing a number by ZERO!!".

In case any other exception is raised, we catch it in another except block and print a different message: "Any other exception".

This way, if an exception is raised during the execution of the divide function, it will be caught and handled, and the program will not crash.

تنفذ كتلة التعليمات البرمجية كتلة try-except بسيطة للتعامل مع الاستثناءات التي أثيرت أثناء تنفيذ دالة divide.

تعرض دالة divide نتيجة قسمة 5 على 0، والتي سترفع استثناء ZeroDivisionError.

في كتلة `try`، نسمي دالة `divide`، وفي الكتلة `except`، نتعامل مع استثناء `ZeroDivisionError` من خلال طباعة رسالة خطأ مخصصة: "Why on earth you are dividing a number by ZERO!!"

في حالة ظهور أي استثناء آخر، نلحقه في كتلة `except` أخرى باستثناء الكتلة ونطبع رسالة مختلفة: "Any other exception"

بهذه الطريقة، إذا ظهر استثناء أثناء تنفيذ دالة `divide`، فسيتم اكتشافه ومعالجته، ولن يتعطل البرنامج.

التمرين 52

الامر:

Define a custom exception class which takes a string message as attribute.

عرف فئة استثناء exception مخصصة تأخذ رسالة سلسلة كسمة.

الكود:

```
class CustomException(Exception):
    """Exception raised for custom purpose

    Attributes:
        message -- explanation of the error
    """

    def __init__(self, message):
        self.message = message

num = int(input())
try:
    if num < 10:
        raise CustomException("Input is less than 10")
    elif num > 10:
        raise CustomException("Input is grater than 10")
except CustomException as ce:
    print("The error raised: " + ce.message)
```

شرح ChatGPT:

The code defines a custom exception class called CustomException that inherits from the built-in Exception class. The custom exception class takes in a message as an argument and sets it as an attribute.

The code then takes in a number from the user, converts it to an integer, and stores it in the num variable. The code then uses a try-except block to handle the custom exception.

If num is less than 10, the code raises the custom exception with the message "Input is less than 10". If num is greater than 10, the code raises the custom exception with the message "Input is greater than 10".

In the except block, the code prints the message attribute of the custom exception, with the message "The error raised: " concatenated to the front.

يعرف الكود فئة استثناء مخصصة تسمى CustomException والتي ترث من فئة Exception المضمنة. تأخذ فئة الاستثناء المخصص الرسالة كوسيطه وتقوم بتعيينها كسمة.

ثم يأخذ الكود رقمًا من المستخدم، ويحوّله إلى عدد صحيح، ويخزنه في المتغير num. يستخدم الكود بعد ذلك كتلة try-except للتعامل مع الاستثناء المخصص.

```
"إذا كان num أقل من 10، يقوم الكود بتطبيق الاستثناء المخصص برسالة "Input is less than 10"
إذا كانت num أكبر من 10، فإن الكود يرفع الاستثناء المخصص برسالة "Input is greater than 10"
```

```
في كتلة except، يطبع الكود سمة الرسالة الخاصة بالاستثناء المخصص، مع ظهور الرسالة "The error raised: " متسلسلة في المقدمة.
```

التمرين 53

الامر:

Assuming that we have some email addresses in the "[username@companyname.com](#)" format, please write program to print the user name of a given email address. Both user names and company names are composed of letters only.

بافتراض أن لدينا بعض عناوين البريد الإلكتروني بتنسيق "[username@companyname.com](#)"، يُرجى كتابة برنامج لطباعة اسم المستخدم لعنوان بريد إلكتروني معين. تتكون كل من أسماء المستخدمين وأسماء الشركات من أحرف فقط.

الكود:

```
import re

email = "john@google.com elise@python.com"
pattern = "(\\w+)@\\w+.com"
ans = re.findall(pattern, email)
print(ans)
```

شرح ChatGPT:

The given code uses the re module in Python to find all occurrences of a specific pattern in a string. The pattern being searched for is defined as "[\(\\w+\)@\\w+.com](#)" and it represents an email address format.

The findall method of the re module is used to search the string email and return all occurrences of the pattern defined in pattern. The result is stored in the ans variable and finally, the result is printed using the print statement.

The output of the code will be ['john', 'elise'], which are the substrings that match the pattern.

يستخدم الكود المعطى وحدة re في بايثون للعثور على جميع تكرارات نمط معين في سلسلة. يتم تعريف النمط الذي يتم البحث عنه على أنه "[\(\\w+\)@\\w+.com](#)" ويمثل تنسيق عنوان البريد الإلكتروني.

يتم استخدام طريقة findall للوحدة النمطية re للبحث في سلسلة البريد الإلكتروني وإرجاع جميع تكرارات النمط المحدد في النمط. يتم تخزين النتيجة في متغير ans، وفي النهاية تُطبع النتيجة باستخدام تعليمة print.

سيكون ناتج الكود هو ['john', 'elise']، وهي السلاسل الفرعية التي تطابق النمط.

التمرين 54

الامر:

Assuming that we have some email addresses in the "[username@companyname.com](#)" format, please write program to print the company name of a given email address. Both user names and company names are composed of letters only.

بافتراض أن لدينا بعض عناوين البريد الإلكتروني بتنسيق "[username@companyname.com](#)"، يُرجى كتابة برنامج لطباعة اسم الشركة لعنوان بريد إلكتروني معين. تتكون كل من أسماء المستخدمين وأسماء الشركات من أحرف فقط.

الكود:

```
import re

email = "john@google.com elise@python.com"
pattern = "\w+(\w+).com"
ans = re.findall(pattern, email)
print(ans)
```

شرح ChatGPT:

The code imports the re (regular expression) module. Then it defines a string `email` that contains two email addresses.

A regular expression `pattern` is then defined to match words (`\w+`) before the `@` symbol, followed by another word (`\w+`) and `.com`.

The `re.findall` function is called, passing in the `pattern` and `email` strings. The function returns a list of all the matches found in the `email` string and it is assigned to the variable `ans`.

Finally, `ans` is printed, which would output: `['google', 'python']`.

يستورد الكود الوحدة النمطية re (التعبير المنتظمة). ثم يعرف سلسلة بريد إلكتروني تحتوي على عنوان بريد إلكتروني.

ثم يتم تحديد نمط التعبير العادي لمطابقة الكلمات (`\w+`) قبل الرمز @، متبوعة بكلمة أخرى (`\w+`) و `.com`

يتم استدعاء دالة `re.findall`، وتمرير النمط وسلاسل البريد الإلكتروني. تقوم الدالة بإرجاع قائمة بجميع التطابقات الموجودة في سلسلة البريد الإلكتروني ويتم تخصيصها للمتغير `ans`.
أخيراً، تتم طباعة الإجابة، والتي ستخرج: `['google', 'python']`.

التمرين 55

الامر:

Write a program which accepts a sequence of words separated by whitespace as input to print the words composed of digits only.

اكتب برنامجاً يقبل سلسلة من الكلمات مفصولة بمسافة بيضاء كمدخلات لطباعة الكلمات المكونة من أرقام فقط.

الكود:

```
email = input().split()
ans = [word for word in email if word.isdigit()] # using list
comprehension method
print(ans)
```

شرح ChatGPT:

This code takes a list of words as input and splits it into separate words. Then it uses a list comprehension to extract all the words that are numbers. Finally, it prints the resulting list of numbers.

يأخذ هذا الكود قائمة بالكلمات كمدخلات ويقسمها إلى كلمات منفصلة. ثم يستخدم list comprehension لاستخراج جميع الكلمات التي هي أرقام. أخيراً، يقوم بطباعة قائمة الأرقام الناتجة.

التمرين 56

الامر:

Print a unicode string "hello world".

اطبع سلسلة يونيكود "hello world".

الكود:

```
s = u"hello world"
print(s)
```

شرح ChatGPT:

The following code is written in Python, it defines a Unicode string `s` with the value "hello world" and then prints it to the console.

Unicode strings in Python are denoted by prefixing the string with the letter 'u' as in `u"hello world"`. This means that the string is encoded in the Unicode character set, which allows for the representation of a wide range of characters from many different languages.

In this code snippet, `s` is defined as a Unicode string and then printed to the console using the `print` function. The output of the code will be `hello world`.

الكود التالي مكتوب بلغة بايثون، وهو يعرف سلسلة Unicode بالقيمة "hello world" ثم يطبعها على وحدة التحكم.

يتم الإشارة إلى سلاسل Unicode في بايثون عن طريق إضافة بادئة للسلسلة بالحرف 'u' كما في "u"hello world". هذا يعني أن السلسلة مشفرة في مجموعة أحرف Unicode، مما يسمح بتمثيل مجموعة واسعة من الأحرف من العديد من اللغات المختلفة.

في مقتطف الكود هذا، يتم تعريف `s` كسلسلة Unicode ثم يتم طباعتها على وحدة التحكم باستخدام دالة `print`. سيكون إخراج الكود هو `hello world`.

التمرين 57

الامر:

Write a program to read an ASCII string and to convert it to a unicode string encoded by utf-8.

اكتب برنامجًا لقراءة سلسلة ASCII وتحويلها إلى سلسلة Unicode مشفرة بواسطة utf-8.

الكود:

```
s = input()
u = s.encode("utf-8")
print(u)
```

شرح ChatGPT:

The code first takes input from the user and stores it in the variable s. Then it uses the encode method of the str class to encode the string s into utf-8 format and stores the result in a variable u. Finally, it prints the encoded string u.

The encode method is used to convert the string into a specific encoding format, in this case, utf-8. The utf-8 format is a widely used character encoding that can handle a wide range of characters from different languages and scripts.

By converting the string into utf-8, it can be transmitted over the internet and stored in various data storage systems in a consistent manner, ensuring that the characters are properly displayed on different platforms.

يأخذ الكود أولاً الإدخال من المستخدم ويخزنه في المتغير s. ثم تستخدم طريقة الترميز لفئة str لترميز السلسلة s إلى تنسيق utf-8 وتخزين النتيجة في متغير u. أخيراً، يقوم بطباعة السلسلة المشفرة u.

تُستخدم طريقة الترميز لتحويل السلسلة إلى تنسيق ترميز محدد، في هذه الحالة، utf-8. تنسيق utf-8 هو ترميز أحرف مستخدم على نطاق واسع يمكنه التعامل مع مجموعة كبيرة من الأحرف من لغات ونصوص مختلفة.

من خلال تحويل السلسلة إلى utf-8، يمكن نقلها عبر الإنترنت وتخزينها في أنظمة تخزين بيانات مختلفة بطريقة متسقة، مما يضمن عرض الأحرف بشكل صحيح على منصات مختلفة.

التمرين 58

الامر:

Write a special comment to indicate a Python source code file is in unicode.

اكتب تعليقاً خاصاً للإشارة إلى أن ملف شفرة مصدر بايثون موجود في Unicode.

الكود:

```
# -*- coding: utf-8 -*-
```

شرح ChatGPT:

The line `# -*- coding: utf-8 -*-` is a special comment line in the Python code that indicates the encoding of the source code file. In this case, the source code is encoded using UTF-8, which is a widely used character encoding that can handle a large number of characters from various scripts and languages.

This line is usually placed at the beginning of the file to specify the encoding of the file, which helps the Python interpreter understand and process the file correctly.

السطر `# -*- coding: utf-8 -*-` هو سطر تعليق خاص في كود بايثون يشير إلى ترميز ملف الكود المصدري. في هذه الحالة، يتم ترميز الكود المصدري باستخدام UTF-8، وهو ترميز أحرف مستخدم على نطاق واسع يمكنه التعامل مع عدد كبير من الأحرف من نصوص ولغات مختلفة.

يوضع هذا السطر عادة في بداية الملف لتحديد ترميز الملف، مما يساعد مترجم بايثون على فهم الملف ومعالجته بشكل صحيح.

التمرين 59

الامر:

Write a program to compute $1/2+2/3+3/4+\dots+n/n+1$ with a given n input by console ($n>0$).

اكتب برنامجًا لحساب $2/1 + 3/2 + 4/3 + \dots + n/n + 1$ مع إدخال n محدد بواسطة وحدة التحكم ($n > 0$).

الكود:

```
n = int(input())
sum = 0
for i in range(1, n + 1):
    sum += i / (i + 1)
print(round(sum, 2)) # rounded to 2 decimal point
```

شرح ChatGPT:

The code snippet computes and prints the sum of a series. The series formula is defined as follows:

$$\text{sum} = (1/2) + (2/3) + (3/4) + \dots + (n/(n + 1))$$

The input n is expected to be a positive integer and determines the number of terms in the series. The code uses a for loop to iterate through the range 1 to $n + 1$ and calculates each term of the series by dividing i by $(i + 1)$ where i is the current iteration value. Finally, the sum of all terms is calculated and rounded to 2 decimal places using the round function. The result is then printed.

يُحسب مقتطف الكود مجموع سلسلة ويطبّعها. يتم تعريف معادلة السلسلة على النحو التالي:

$$\text{sum} = (1/2) + (2/3) + (3/4) + \dots + (n/(n + 1))$$

من المتوقع أن يكون الإدخال n عددًا صحيحًا موجبًا ويعرف عدد المصطلحات في السلسلة. يستخدم الكود حلقة for للتكرار خلال النطاق 1 إلى $n + 1$ ويحسب كل مصطلح في السلسلة بقسمة i على $(i + 1)$ حيث i هي قيمة التكرار الحالية. أخيرًا، يتم حساب مجموع كل الحدود وتقريبه إلى منزلتين عشريتين باستخدام دالة round. ثم يتم طباعة النتيجة.

التمرين 60

الامر:

Write a program to compute: $f(n) = f(n - 1) + 100$, $n > 0$ with a given n input by console ($n > 0$).

اكتب برنامجًا لحساب: $f(n) = f(n - 1) + 100$, مع إدخال n محدد بواسطة وحدة التحكم ($n > 0$).

الكود:

```
def f(n):
    if n == 0:
        return 0
    return f(n - 1) + 100

n = int(input())
print(f(n))
```

شرح ChatGPT:

The following code is a recursive function in Python that calculates the sum of the serie $100 + 200 + 300 + \dots + 100 * n$ where n is the input argument.

The function $f(n)$ takes an integer n as input. If n is 0, it returns 0. Otherwise, it calls itself with the argument $n-1$ and adds 100 to the result. This process repeats until the base case of $n=0$ is reached, and then the final result is returned up the call stack.

The code also takes a user input n , converts it to an integer, and passes it as an argument to the f function. The result of the f function is then printed.

الكود التالي هو دالة استدعاء ذاتي في بايثون تحسب مجموع السلسلة $100 + 200 + 300 + \dots + 100 * n$ حيث n هي وسيطة الإدخال.

تأخذ الدالة $f(n)$ عددًا صحيحًا n كمدخل. إذا كانت n تساوي 0، فإنها ترجع 0. وإلا فإنها تستدعي نفسها بالمتغير $n-1$ وتضيف 100 إلى النتيجة. تتكرر هذه العملية حتى يتم الوصول إلى الحالة الأساسية لـ $n = 0$ ، ثم يتم إرجاع النتيجة النهائية إلى مكس الاستدعاءات.

يأخذ الكود أيضًا إدخال المستخدم n ، ويحوّله إلى عدد صحيح، ويمرره كوسيلة إلى الدالة f . ثم تتم طباعة نتيجة الدالة f .

التمرين 61

الامر:

The Fibonacci Sequence is computed based on the following formula:

يتم حساب متتالية فيبوناتشي على أساس الصيغة التالية:

$$f(0)=0,$$

$$f(1)=1,$$

$$f(n) = f(n - 1) + f(n - 2), n > 1$$

Please write a program to compute the value of $f(n)$ with a given n input by console.

يُرجى كتابة برنامج لحساب قيمة $f(n)$ بإدخال n محدد بواسطة وحدة التحكم.

الكود:

```
def f(n):
    if n < 2:
        return n
    return f(n - 1) + f(n - 2)

n = int(input())
print(f(n))
```

شرح ChatGPT:

This code implements the recursive function $f(n)$, which returns the n -th number in the Fibonacci sequence. The Fibonacci sequence is a sequence of numbers where each number is the sum of the two preceding numbers. In this case, the sequence starts with 0 and 1, so the first two numbers in the sequence are 0 and 1.

The function first checks if n is less than 2, and if so, it returns n . This serves as the base case for the recursion. If n is greater than or equal to 2, the function calls itself twice, each time with an argument that is one less than the current value of n . The two calls to $f(n - 1)$ and $f(n - 2)$ represent the two preceding numbers in the sequence, and the sum of these two calls gives the next number in the sequence. This process continues until n becomes less than 2, at which point the base case is reached and the recursion stops.

Finally, the code takes an input n from the user, and passes it as an argument to the function $f(n)$. The function returns the n -th number in the Fibonacci sequence, which is printed to the screen.

يقوم هذا الكود بتنفيذ دالة الاستدعاء ذاتي $f(n)$ ، والتي تُرجع الرقم n في متتالية فيبوناتشي. متتالية فيبوناتشي هو سلسلة من الأرقام حيث يكون كل رقم هو مجموع الرقمين السابقين. في هذه الحالة، يبدأ التسلسل بالرقم 0 و 1، لذا فإن أول رقمين في التسلسل هما 0 و 1.

تتحقق الدالة أولاً مما إذا كان n أقل من 2، وإذا كان الأمر كذلك، فإنها ترجع n . هذا بمثابة الحالة الأساسية للتكرار. إذا كانت n أكبر من أو تساوي 2، فإن الدالة تستدعي نفسها مرتين، في كل مرة مع وسيطة أقل بواحد من القيمة الحالية لـ n . الاستدعاءات إلى $f(n-1)$ و $f(n-2)$ يمثلان الرقمين السابقين في التسلسل، ومجموع هذين الاستدعاءين يعطي الرقم التالي في التسلسل. تستمر هذه العملية حتى يصبح n أقل من 2، وعند هذه النقطة يتم الوصول إلى الحالة الأساسية ويتوقف التكرار.

أخيراً، يأخذ الكود إدخال n من المستخدم ويمرره كوسيطة للدالة $f(n)$. تقوم الدالة بإرجاع الرقم n في تسلسل فيبوناتشي، والذي تتم طباعته على الشاشة.

التمرين 62

الامر:

The Fibonacci Sequence is computed based on the following formula:

يتم حساب متتالية فيبوناتشي على أساس الصيغة التالية:

```
f(0)=0,
f(1)=1,
f(n) = f(n - 1) + f(n - 2), n > 1
```

Please write a program to compute the value of $f(n)$ with a given n input by console and prints all the values all along.

يُرجى كتابة برنامج لحساب قيمة $f(n)$ مع إدخال n محدد بواسطة وحدة التحكم وطباعة جميع القيم طوال الوقت.

الكود:

```
def f(n):
    if n < 2:
        fibo[n] = n
        return fibo[n]
    fibo[n] = f(n - 1) + f(n - 2)
    return fibo[n]

n = int(input())
fibonacci = [0] * (n + 1) # initialize a list of size (n+1)
f(n) # call once and it will set value to fibonacci[n]
fibonacci = [str(i) for i in fibonacci] # converting integer data to string type
ans = ",".join(fibonacci) # joining all string element of fibonacci with ',' character
print(ans)
```

شرح ChatGPT:

The code above calculates the first n terms of the Fibonacci sequence using a recursive function f .

The function takes an argument n and checks if it is less than 2. If it is, it stores n in the `fibonacci` list and returns n .

Otherwise, it calculates the n -th term of the sequence by calling the function with $n-1$ and $n-2$ as arguments, adds the results and stores it in the `fibonacci` list.

The function returns the n -th term.

The function is called with the input value of n and the fibo list is initialized with 0's. The function is called to calculate the first n terms of the sequence and the results are stored in the fibo list.

Finally, the list is converted to a string with each element separated by a comma and the result is printed.

يحسب الكود أعلاه أول n من المصطلحات من متتالية فيبوناتشي باستخدام دالة استدعاء ذاتي f . تأخذ الدالة وسيطة n وتتحقق مما إذا كانت أقل من 2. إذا كانت كذلك، فإنها تخزن n في قائمة $fibo$ وتعيد n .

خلاف ذلك، فإنه يحسب الحد n من المتتالية عن طريق استدعاء الدالة مع $n-1$ و $n-2$ كوسيطتين، ويضيف النتائج ويخزنها في قائمة $fibo$. ترجع الدالة الحد رقم n .

يتم استدعاء الدالة بقيمة الإدخال n ويتم تهيئة قائمة $fibo$ بـ 0. يتم استدعاء الدالة لحساب المصطلحات n الأولى من التسلسل ويتم تخزين النتائج في قائمة الألياف البصرية.

أخيراً، يتم تحويل القائمة إلى سلسلة مع فصل كل عنصر بفاصلة ويتم طباعة النتيجة.

التمرين 63

الامر:

Please write a program using generator to print the even numbers between 0 and n in comma separated form while n is input by console:

يرجى كتابة برنامج باستخدام المولد لطباعة الأرقام الزوجية بين 0 و n في شكل مفصول بفاصلة بينما يتم إدخال n بواسطة وحدة التحكم:

الكود:

```
def EvenGenerator(n):
    i = 0
    while i <= n:
        if i % 2 == 0:
            yield i
            i += 1

n = int(raw_input())
values = []
for i in EvenGenerator(n):
    values.append(str(i))

print ",".join(values)
```

[شرح ChatGPT](#):

The following code generates a list of even numbers up to a given number n. This is achieved using a generator function `EvenGenerator(n)`. The generator function `EvenGenerator(n)` generates even numbers up to n using a while loop.

The `yield` statement is used in the generator function to return each even number. This allows the generator to pause its execution after each iteration and return the current value of `i`. This way, we can generate a series of even numbers without having to store them all in memory.

The generator function is called with the input n and the generated even numbers are stored in the list `values`. The `values` list is then converted to a string separated by commas using the `join` method and printed as the final result.

يُنشئ الكود التالي قائمة بالأرقام الزوجية حتى رقم معين n . يتم تحقيق ذلك باستخدام دالة المولد `EvenGenerator(n)`. تولد دالة المولد `EvenGenerator(n)` أعدادًا زوجية تصل إلى n باستخدام حلقة `while`.

يتم استخدام جملة `yield` في دالة المولد لإرجاع كل رقم زوجي. يسمح هذا للمولد بإيقاف تنفيذه مؤقتًا بعد كل تكرار وإرجاع القيمة الحالية لـ `i`. بهذه الطريقة، يمكننا إنشاء سلسلة من الأرقام الزوجية دون الحاجة إلى تخزينها جميعًا في الذاكرة.

يتم استدعاء دالة المولد مع الإدخال n ويتم تخزين الأرقام الزوجية المتولدة في القائمة `values`. يتم بعد ذلك تحويل قائمة `values` إلى سلسلة مفصولة بفواصل باستخدام طريقة `join` وطباعتها كنتيجة نهائية.

التمرين 64

الامر:

Please write a program using generator to print the numbers which can be divisible by 5 and 7 between 0 and n in comma separated form while n is input by console:

يُرجى كتابة برنامج باستخدام المولد لطباعة الأرقام التي يمكن أن تكون قابلة للقسمة على 5 و 7 بين 0 و n في شكل مفصول بفاصلة بينما يتم إدخال n بواسطة وحدة التحكم:

الكود:

```
def generate(n):
    for i in range(n + 1):
        if (
            i % 35 == 0
        ): # 5*7 = 35, if a number is divisible by a & b then it
            is also divisible by a*b
            yield i

n = int(input())
resp = [str(i) for i in generate(n)]
print(",".join(resp))
```

شرح ChatGPT:

The following code snippet defines a function called generate that takes an integer n as input. The function uses the range function to generate a range of numbers from 0 to n and uses a for loop to iterate through this range. The for loop checks if the number i is divisible by 35, if it is divisible, it is added to the generator using the yield keyword. This allows us to generate a sequence of numbers that are divisible by 35.

Finally, the list comprehension resp takes the numbers generated from the generator and converts them into a list of strings. The list is then joined using the join method and separated by a comma. The final result is then printed to the console.

يعرف مقتطف الكود التالي دالة تسمى generate تأخذ عددًا صحيحًا n كمدخل. تستخدم الدالة دالة range لإنشاء نطاق من الأرقام من 0 إلى n وتستخدم حلقة for للتكرار خلال هذا النطاق.

تتحقق حلقة for مما إذا كان الرقم i قابلاً للقسمة على 35، وإذا كان قابلاً للقسمة، فسيتم إضافته إلى المولد باستخدام الكلمة المفتاحية yield. هذا يسمح لنا بتوليد سلسلة من الأرقام التي تقبل القسمة على 35.

أخيراً، تأخذ resp (list comprehension) الأرقام الناتجة من المولد وتحولها إلى قائمة من السلاسل. ثم يتم ضم القائمة باستخدام طريقة join ويفصل بينها بفاصلة. ثم يتم طباعة النتيجة النهائية على وحدة التحكم.

التمرين 65

الامر:

Please write assert statements to verify that every number in the list [2,4,6,8] is even.

يُرجى كتابة بيانات التأكيد للتحقق من أن كل رقم في القائمة [2,4,6,8] هو زوجي.

الكود:

```
data = [2, 4, 5, 6]
for i in data:
    assert i % 2 == 0, "{} is not an even number".format(i)
```

شرح ChatGPT:

This code is checking if each element in the list data is even or not. If the element is not even, it will raise an AssertionError with the message "x is not an even number", where x is the current element in the iteration.

The assert statement is used to check if a given condition is met, and if not, raise an exception. In this case, the condition being checked is if the element divided by 2 has a remainder of 0 (i.e. if it is even). If this condition is met for all elements in the list, the code will execute without raising any exceptions.

يتحقق هذا الكود مما إذا كان كل عنصر في بيانات القائمة متساوياً أم لا. إذا لم يكن العنصر زوجياً، فإنه سيرفع خطأ AssertionError مع الرسالة "x is not an even number"، حيث x هو العنصر الحالي في التكرار.

يتم استخدام عبارة assert للتحقق مما إذا تم استيفاء شرط معين، وإذا لم يكن الأمر كذلك، فقم بتطبيق استثناء. في هذه الحالة، الشرط الذي يتم التحقق منه هو إذا كان العنصر مقسوماً على 2 يحتوي على باقي 0 (أي إذا كان زوجياً). إذا تم استيفاء هذا الشرط لجميع العناصر في القائمة، فسيتم تنفيذ الكود دون تطبيق أي استثناءات.

التمرين 66

الامر:

Please write a program which accepts basic mathematic expression from console and print the evaluation result.

يُرجى كتابة برنامج يقبل التعبير الرياضي الأساسي من وحدة التحكم وطباعة نتيجة التقييم.

الكود:

```
expression = input()
ans = eval(expression)
print(ans)
```

شرح ChatGPT:

The code reads an expression as input from the user and then evaluates it using the eval function in Python. The result of the evaluation is then printed to the console.

For example, if the user inputs $2 + 3$, the code will evaluate $2 + 3$ to 5 and then print 5 to the console.

يقرأ الكود تعبيراً كمدخلات من المستخدم ثم يقيمه باستخدام دالة eval في بايثون. ثم تتم طباعة نتيجة التقييم على وحدة التحكم.

على سبيل المثال، إذا قام المستخدم بإدخال $2 + 3$ ، فسيقوم الكود بتقييم $2 + 3$ إلى 5 ثم طباعة 5 إلى وحدة التحكم.

التمرين 67

الامر:

Please write a binary search function which searches an item in a sorted list. The function should return the index of element to be searched in the list.

يُرجى كتابة دالة بحث ثنائية تبحث عن عنصر في قائمة مرتبة. يجب أن تقوم الدالة بإرجاع فهرس العنصر المراد البحث عنه في القائمة.

الكود:

```
def binary_search_Ascending(array, target):
    lower = 0
    upper = len(array)
    print("Array Length:", upper)
    while lower < upper:
        x = (lower + upper) // 2
        print("Middle Value:", x)
        value = array[x]
        if target == value:
            return x
        elif target > value:
            lower = x
        elif target < value:
            upper = x
```

```
Array = [1, 5, 8, 10, 12, 13, 55, 66, 73, 78, 82, 85, 88, 99]
print("The Value Found at Index:", binary_search_Ascending(Array,
82))
```

[شرح ChatGPT](#):

The code performs a binary search algorithm in an ascending sorted array to find the index of a target value.

The function `binary_search_Ascending` takes two arguments, `array` and `target`. It first initializes two variables, `lower` and `upper`, to be the start and end of the array, respectively. The `while` loop runs until `lower` becomes greater than or equal to `upper`. In each iteration, `x` is calculated as the middle value of the current subarray defined by `lower` and `upper`, and the value of the array at that index is stored in `value`.

Then, the target value is compared with the middle value, if they are equal, the index x is returned as the answer. If the target value is greater than the middle value, lower is updated to be equal to x so that the next search will be in the upper half of the array. If the target value is less than the middle value, upper is updated to be equal to x , so that the next search will be in the lower half of the array.

The print statements are used for debugging purposes to see the intermediate values of x and the length of the array.

The code then calls the function with an example array and target value and prints the index at which the target value is found.

تقوم الكود بتنفيذ خوارزمية بحث ثنائية في مصفوفة مرتبة تصاعدياً للعثور على فهرس القيمة المستهدفة.

تأخذ الدالة `binary_search_Ascending` وسيطتين، المصفوفة `array` والهدف `target`. يقوم أولاً بتهيئة متغيرين، `lower` و `upper`، ليكونا بداية المصفوفة ونهايتها، على التوالي. تعمل الحلقة `while` حتى يصبح الجزء السفلي أكبر من أو يساوي `upper`. في كل تكرار، يتم حساب x على أنها القيمة الوسطى للمصفوفة الفرعية الحالية المحددة بواسطة `lower` و `upper`، ويتم تخزين قيمة الصفيف في هذا الفهرس بالقيمة.

بعد ذلك، تتم مقارنة قيمة `target` بقيمة `middle`، إذا كانت متساوية، يتم إرجاع الفهرس x كإجابة. إذا كانت قيمة `target` أكبر من قيمة `middle`، فسيتم تحديث القيمة الأقل لتكون مساوية لـ x بحيث يكون البحث التالي في النصف العلوي من المصفوفة. إذا كانت قيمة `target` أقل من قيمة `middle`، يتم تحديث `upper` ليكون مساوياً لـ x ، بحيث يكون البحث التالي في النصف السفلي من المصفوفة.

تُستخدم عبارات `print` لأغراض التصحيح لمعرفة القيم الوسيطة لـ x وطول المصفوفة.

ثم يستدعي الكود الدالة بمصفوفة مثال وقيمة مستهدفة ويطبع الفهرس الذي توجد فيه القيمة الهدف.

التمرين 68

الامر:

Please generate a random float where the value is between 10 and 100 using Python module.

يُرجى إنشاء عدد عشوائي حيث تتراوح القيمة بين 10 و 100 باستخدام وحدة بايثون.

الكود:

```
import random

rand_num = random.uniform(10, 100)
print(rand_num)
```

شرح ChatGPT:

The code is generating a random floating-point number between 10 and 100 using the `uniform` function from the `random` module in Python. The `uniform` function generates a random float value between two given parameters (inclusive). In this case, it's generating a random float value between 10 and 100.

يُنشئ الكود رقمًا عشوائيًا للفاصلة العائمة بين 10 و 100 باستخدام دالة `uniform` من الوحدة النمطية `random` في بايثون. تولد الدالة `uniform` قيمة قائمة عشوائية بين معلمتين معينتين (شاملة). في هذه الحالة، يتم إنشاء قيمة قائمة عشوائية بين 10 و 100.

التمرين 69

الامر:

Please generate a random float where the value is between 5 and 95 using Python module.

يُرجى إنشاء عدد عشوائي حيث تتراوح القيمة بين 5 و 95 باستخدام الوحدة النمطية من بايثون.

الكود:

```
import random

rand_num = random.uniform(5, 95)
print(rand_num)
```

شرح ChatGPT:

The code generates a random floating-point number between 5 and 95.

The first line `import random` is used to import the random module in Python, which provides functions for generating random numbers.

The second line `rand_num = random.uniform(5, 95)` generates a random floating-point number between 5 and 95 (including 5 but not including 95) and assigns it to the variable `rand_num`.

Finally, the line `print(rand_num)` prints the generated random number.

يولد الكود عددًا عشوائيًا من النقطة العائمة بين 5 و 95.

يتم استخدام أول سطر `import random` لاستيراد وحدة عشوائية في بايثون، والتي توفر دوال لتوليد أرقام عشوائية.

يولد السطر الثاني `rand_num = random.uniform(5, 95)` رقمًا عشوائيًا للفاصلة العائمة بين 5 و 95 (بما في ذلك 5 ولكن لا يشمل 95) ويخصصه للمتغير `rand_num`.

أخيرًا، طباعة `print(rand_num)` تطبع الرقم العشوائي الذي تم إنشاؤه.

التمرين 70

الامر:

Please write a program to output a random even number between 0 and 10 inclusive using random module and list comprehension.

يُرجى كتابة برنامج لإخراج عدد زوجي عشوائي بين 0 و 10 باستخدام وحدة random و list comprehension.

الكود:

```
import random

resp = [i for i in range(0, 11, 2)]
print(random.choice(resp))
```

شرح ChatGPT:

The code imports the random module in Python, which provides a suite of functions to generate pseudo-random numbers.

Then, a list resp is created using list comprehension, with the numbers from 0 to 10 with a step of 2.

Finally, the random.choice method is used to select a random element from the list resp. The selected random element is then printed to the console. This code will output a random even number between 0 and 10.

يستورد الكود الوحدة random في بايثون، والتي توفر مجموعة من الدوال لإنشاء أرقام شبه عشوائية.

بعد ذلك، يتم إنشاء قائمة متتالية باستخدام list comprehension، مع الأرقام من 0 إلى 10 بخطوة 2.

أخيراً، يتم استخدام طريقة random.choice لتحديد عنصر عشوائي من قائمة resp. ثم يتم طباعة العنصر العشوائي المحدد على وحدة التحكم. سيُنتج هذا الكود عدداً زوجياً عشوائياً بين 0 و 10.

التمرين 71

الامر:

Please write a program to output a random number, which is divisible by 5 and 7, between 10 and 150 inclusive using random module and list comprehension.

يُرجى كتابة برنامج لإخراج رقم عشوائي، والذي يقبل القسمة على 5 و 7، بين 10 و 150 شاملاً باستخدام وحدة random و list comprehension .

الكود:

```
import random

resp = [i for i in range(10, 151) if i % 35 == 0]
print(random.choice(resp))
```

شرح ChatGPT:

The code imports the random module in Python, which provides functions for generating pseudo-random numbers.

Then, it defines a list called resp using a list comprehension that generates a list of numbers from 10 to 150 (inclusive) that are divisible by 35 (i.e., have a remainder of 0 when divided by 35).

Finally, the code prints a random element from this list using the random.choice function, which returns a randomly selected element from a given sequence (in this case, the resp list).

يستورد الكود الوحدة random في بايثون، والتي توفر دوال لتوليد أرقام شبه عشوائية.

بعد ذلك، تحدد قائمة تسمى resp باستخدام list comprehension تقوم بإنشاء قائمة من الأرقام من 10 إلى 150 (ضمنًا) قابلة للقسمة على 35 (أي أن يكون لديك الباقي 0 عند القسمة على 35).

أخيرًا، تطبع الكود عنصرًا عشوائيًا من هذه القائمة باستخدام دالة random.choice، التي تُرجع عنصرًا تم اختياره عشوائيًا من تسلسل معين (في هذه الحالة، قائمة resp).

التمرين 72

الامر:

Please write a program to generate a list with 5 random numbers between 100 and 200 inclusive.

يُرجى كتابة برنامج لإنشاء قائمة بها 5 أرقام عشوائية بين 100 و 200 ضمناً.

الكود:

```
import random

resp = random.sample(range(100, 201), 5)
print(resp)
```

شرح ChatGPT:

This code snippet is written in Python and uses the random module to generate a list of 5 random numbers within the range of 100 to 200 (inclusive). First, we import the random module, which provides a suite of functions for generating pseudo-random numbers.

Next, we use the random.sample method to generate a list of 5 random numbers within the range of 100 to 200. The random.sample method takes two arguments: the first argument is the range of numbers to choose from (in this case, it's range(100, 201)), and the second argument is the number of items to choose from that range (in this case, it's 5).

Finally, we use the print function to output the list of 5 randomly generated numbers.

تمت كتابة مقتطف الكود هذا بلغة بايثون ويستخدم الوحدة النمطية random لإنشاء قائمة من 5 أرقام عشوائية ضمن النطاق من 100 إلى 200 (ضمناً).

أولاً، نقوم باستيراد الوحدة random، والتي توفر مجموعة من الدوال لتوليد أرقام شبه عشوائية.

بعد ذلك، نستخدم طريقة random.sample لإنشاء قائمة من 5 أرقام عشوائية ضمن النطاق من 100 إلى 200. تأخذ طريقة random.sample وسيطتين: الوسيطة الأولى هي نطاق الأرقام للاختيار من بينها (في هذه الحالة، النطاق (100، 201))، والوسيطة الثانية هي عدد العناصر للاختيار من هذا النطاق (في هذه الحالة، 5).

أخيراً، نستخدم دالة print لإخراج قائمة من 5 أرقام تم إنشاؤها عشوائياً.

التمرين 73

الامر:

Please write a program to randomly generate a list with 5 even numbers between 100 and 200 inclusive.

يرجى كتابة برنامج لإنشاء قائمة عشوائية مكونة من 5 أعداد زوجية بين 100 و 200.

الكود:

```
import random

resp = random.sample(range(100, 201, 2), 5)
print(resp)
```

شرح ChatGPT:

This code snippet generates 5 random numbers from a range of even numbers between 100 and 201 (inclusive).

It starts by importing the random module, which contains a number of useful functions for generating random numbers, shuffling lists, and choosing random items from a list.

Next, the line `resp = random.sample(range(100, 201, 2), 5)` generates a list of 5 random numbers from a range of even numbers between 100 and 201 (inclusive).

The `range(100, 201, 2)` function generates a list of all even numbers between 100 and 201 (inclusive), and `random.sample(range(100, 201, 2), 5)` randomly chooses 5 numbers from this list.

Finally, the line `print(resp)` prints the list of 5 random numbers.

يُنشئ مقتطف الكود هذا 5 أرقام عشوائية من نطاق من الأرقام الزوجية بين 100 و 201 (ضمنياً).

يبدأ باستيراد الوحدة random، والتي تحتوي على عدد من الدوال المفيدة لتوليد الأرقام العشوائية، وخط القوائم، واختيار العناصر العشوائية من القائمة.

بعد ذلك، يُنشئ السطر `resp = random.sample(range(100, 201, 2), 5)` من 5 أرقام عشوائية من نطاق من الأرقام الزوجية بين 100 و 201 (ضمنياً).

تقوم دالة `range(100, 201, 2)` بإنشاء قائمة بجميع الأرقام الزوجية بين 100 و 201 (ضمنًا) ، و `random.sample(range(100, 201, 2), 5)` تختار عشوائيًا 5 أرقام من هذه القائمة.

أخيرًا، يطبع السطر `print(resp)` قائمة من 5 أرقام عشوائية.

التمرين 74

الامر:

Please write a program to randomly generate a list with 5 numbers, which are divisible by 5 and 7 , between 1 and 1000 inclusive.

يُرجى كتابة برنامج لإنشاء قائمة عشوائية مكونة من 5 أرقام قابلة للقسمة على 5 و 7 ، بين 1 و 1000 ضمناً.

الكود:

```
import random

lst = [i for i in range(1, 1001) if i % 35 == 0]
resp = random.sample(lst, 5)
print(resp)
```

شرح ChatGPT:

This code snippet imports the random module in Python, which provides functions to generate random numbers.

The first line defines a list called `lst` which contains all the numbers from 1 to 1000 that are divisible by 35. This is done using a list comprehension.

The second line uses the `random.sample` function from the random module to generate 5 random numbers from the list `lst`. This function takes two arguments, the first is the list from which the random samples should be selected, and the second is the number of samples to select.

Finally, the last line prints the result of the `resp` variable, which contains the 5 random numbers selected from the `lst` list.

يستورد مقتطف الكود هذا الوحدة النمطية `random` في بايثون، والتي توفر دوال لإنشاء أرقام عشوائية.

يعرف السطر الأول قائمة تسمى `lst` تحتوي على جميع الأرقام من 1 إلى 1000 القابلة للقسمة على 35. ويتم ذلك باستخدام `list comprehension`.

يستخدم السطر الثاني دالة `random.sample` من الوحدة `random` لتوليد 5 أرقام عشوائية من القائمة الأولى. تأخذ هذه الدالة وسيطين، الأولى هي القائمة التي يجب اختيار العينات العشوائية منها، والثانية هي عدد العينات المراد تحديدها.

أخيراً، يطبع السطر الأخير نتيجة متغير Resp، الذي يحتوي على 5 أرقام عشوائية مختارة من القائمة الأولى.

التمرين 75

الامر:

Please write a program to randomly print a integer number between 7 and 15 inclusive.

يُرجى كتابة برنامج لطباعة عدد صحيح بشكل عشوائي بين 7 و 15 ضمناً.

الكود:

```
import random
print(random.randrange(7, 16))
```

شرح ChatGPT:

The following code snippet is written in Python and it uses the random module.

The `random.randrange(start, stop)` method is used to generate a random number within a range, in this case between 7 and 16 (excluding 16). The result of the `random.randrange(7, 16)` method is then printed to the console. This means that each time the code is run, a different random number between 7 and 16 will be generated and printed to the console.

تمت كتابة مقتطف الكود التالي بلغة بايثون ويستخدم الوحدة النمطية `random`.

يتم استخدام طريقة `random.randrange(start, stop)` لتوليد رقم عشوائي ضمن نطاق ، في هذه الحالة بين 7 و 16 (باستثناء 16).

ثم تتم طباعة نتيجة طريقة `random.randrange(7, 16)` على وحدة التحكم. هذا يعني أنه في كل مرة يتم فيها تشغيل الكود، سيتم إنشاء رقم عشوائي مختلف بين 7 و 16 وطباعته على وحدة التحكم.

التمرين 76

الامر:

Please write a program to compress and decompress the string "hello world!hello world!hello world!hello world!".

يُرجى كتابة برنامج لضغط وفك ضغط السلسلة "hello world! hello world! hello world! hello world!".

الكود:

```
import zlib

t = zlib.compress(b'hello world!hello world!hello world!hello world!')
print(t)
print(zlib.decompress(t))
```

شرح ChatGPT:

The code is using the `zlib` library to perform data compression and decompression. The `zlib` library provides in-memory compression and decompression functions, including the `compress` and `decompress` functions. In the code, the string "hello world!hello world!hello world!hello world!" is first converted to a byte object using `b'hello world!hello world!hello world!hello world!'`. This byte object is then passed to the `compress` function of the `zlib` library which returns a compressed version of the byte object.

The compressed version of the byte object is then printed, and finally, the compressed version of the byte object is passed to the `decompress` function of the `zlib` library which returns the original byte object. The decompressed byte object is then printed.

In this way, the code demonstrates the use of `zlib` library for data compression and decompression in Python.

يستخدم الكود مكتبة `zlib` لأداء ضغط البيانات وإلغاء ضغطها. توفر مكتبة `zlib` دوال ضغط وفك ضغط في الذاكرة، بما في ذلك دوال الضغط وفك الضغط.

في الكود، السلسلة "hello world!hello world!hello world!hello world!" تم تحويله أولاً إلى كائن بايت باستخدام `b'hello world!hello world!hello world!hello world!'`.

'world!hello world!'. ثم يتم تمرير كائن البايت هذا إلى دالة `compress` لمكتبة `zlib` التي تقوم بإرجاع نسخة مضغوطة من كائن البايت.

يتم بعد ذلك طباعة النسخة المضغوطة من كائن البايت، وفي النهاية، يتم تمرير الإصدار المضغوط من كائن البايت إلى دالة `decompress` في مكتبة `zlib` التي تُرجع كائن البايت الأصلي. ثم تتم طباعة كائن البايت الذي تم فك ضغطه.

بهذه الطريقة، يوضح الكود استخدام مكتبة `zlib` لضغط البيانات وإلغاء الضغط في بايثون.

التمرين 77

الامر:

Please write a program to print the running time of execution of "1+1" for 100 times.

يرجى كتابة برنامج لطباعة وقت تشغيل تنفيذ "1 + 1" 100 مرة.

الكود:

```
import time

before = time.time()
for i in range(100):
    x = 1 + 1
after = time.time()
execution_time = after - before
print(execution_time)
```

شرح ChatGPT:

The code is measuring the execution time of a simple operation. It's using the `time` module in Python which provides various time-related functions.

At the start of the code, the current time is recorded by `before = time.time()`.

Then, the code runs a for loop 100 times where a simple operation `x = 1 + 1` is performed.

At the end of the loop, the current time is recorded again by `after = time.time()`.

Finally, the execution time is calculated by subtracting `before` from `after` and stored in the variable `execution_time`. The execution time is then printed using `print(execution_time)`. This shows how much time it took to run the simple operation 100 times.

الكود يقيس وقت تنفيذ عملية بسيطة. إنها تستخدم وحدة `time` في بايثون والتي توفر دوال مختلفة متعلقة بالوقت.

في بداية الكود، يتم تسجيل الوقت الحالي `.before = time.time()`.

بعد ذلك، يقوم الكود بتشغيل حلقة `for 100` مرة حيث يتم تنفيذ عملية بسيطة `x = 1 + 1`.

في نهاية الحلقة، يتم تسجيل الوقت الحالي مرة أخرى `.after = time.time()`.
أخيراً، يتم حساب وقت التنفيذ عن طريق طرح `before` من `after` وتخزينه في متغير `execution_time`. ثم تتم طباعة وقت التنفيذ باستخدام `.print(execution_time)`.
يوضح هذا مقدار الوقت المستغرق لتشغيل العملية البسيطة 100 مرة.

التمرين 78

الامر:

Please write a program to shuffle and print the list [3,6,7,8].

الرجاء كتابة برنامج لتبديل القائمة وطباعتها [3,6,7,8].

الكود:

```
import random

lst = [3, 6, 7, 8]
random.shuffle(lst)
print(lst)
```

شرح ChatGPT:

This code is in the Python programming language. It makes use of the random module which provides a set of functions for generating pseudo-random numbers.

The code defines a list called `lst` that contains the values [3, 6, 7, 8].

Then, the function `shuffle` from the `random` module is used to shuffle the items in the list `lst`. The `shuffle` function randomly reorders the items in the list.

Finally, the shuffled list is printed using the `print` statement.

هذا الكود موجود في لغة برمجة بايثون. إنها تستخدم الوحدة `random` التي توفر مجموعة من الدوال لتوليد أرقام شبه عشوائية.

يعرف الكود قائمة تسمى `lst` تحتوي على القيم [3 ، 6 ، 7 ، 8].

بعد ذلك، يتم استخدام الدالة `shuffle` من الوحدة النمطية `random` لتبديل العناصر الموجودة في القائمة `lst`. تقوم دالة `shuffle` بإعادة ترتيب العناصر الموجودة في القائمة بشكل عشوائي.

أخيراً، تتم طباعة القائمة التي تم خلطها عشوائياً باستخدام جملة `print`.

التمرين 79

الامر:

Please write a program to generate all sentences where subject is in ["I", "You"] and verb is in ["Play", "Love"] and the object is in ["Hockey", "Football"].

يُرجى كتابة برنامج لتوليد جميع الجمل حيث يكون الفاعل في ["I", "You"] والفاعل في ["Play", "Love"] ويكون الكائن في ["Hockey", "Football"].

الكود:

```
subjects = ["I", "You"]
verbs = ["Play", "Love"]
objects = ["Hockey", "Football"]

for sub in subjects:
    for verb in verbs:
        for obj in objects:
            print("{} {} {}".format(sub, verb, obj))
```

شرح ChatGPT:

The following code snippet in Python generates all possible combinations of a sentence using elements from three different lists: subjects, verbs, and objects.

The subjects list contains the words "I" and "You". The verbs list contains the words "Play" and "Love". The objects list contains the words "Hockey" and "Football".

The code uses a nested loop, with each outer loop iterating through each element in one of the lists (subjects, verbs, objects). The inner loop then iterates through the elements of the other two lists, combining each subject-verb-object combination and printing the resulting sentence.

For example, the first combination would be "I Play Hockey". The code will generate a total of $(2 * 2 * 2 =) 8$ combinations of sentences in total.

يولد مقتطف الكود التالي في بايثون جميع التركيبات الممكنة للجمل باستخدام عناصر من ثلاث قوائم مختلفة: الفاعل subjects والفعل verbs والمفعول به objects.

تحتوي قائمة subjects على الكلمتين "I" و "You". تحتوي قائمة verbs على الكلمتين "Play" و "Love". تحتوي قائمة objects على الكلمتين "Hockey" و "Football".

يستخدم الكود حلقة متداخلة، مع كل حلقة خارجية تتكرر خلال كل عنصر في إحدى القوائم (objects, verbs, subjects). ثم تتكرر الحلقة الداخلية من خلال عناصر القائمتين الأخيرتين، وتجمع بين كل تركيبة subject-verb-object وطباعة الجملة الناتجة.

على سبيل المثال، المجموعة الأولى ستكون "I Play Hockey". سيقوم الكود بإنشاء ما مجموعه 2 * 2 * 2 = 8 مجموعات من الجمل في المجموع.

التمرين 80

الامر:

Please write a program to print the list after removing even numbers in [5,6,77,45,22,12,24].

يُرجى كتابة برنامج لطباعة القائمة بعد إزالة الأرقام الزوجية في [5,6,77,45,22,12,24].

الكود:

```
li = [5, 6, 77, 45, 22, 12, 24]
lst = list(filter(lambda n: n % 2 != 0, li))
print(lst)
```

شرح ChatGPT:

This is a Python code snippet that uses the `filter` method and `lambda` function to extract the odd numbers from a list.

The list `li` is initialized with a set of integers. The `filter` method and `lambda` function are used to extract only the odd numbers from the list. The `filter` method takes a function and a list as arguments. The `lambda` function is an anonymous function that takes an argument `n` and returns `True` if `n` is not divisible by 2, and `False` otherwise. The `filter` method then applies the `lambda` function to each element of the `li` list and returns a new list containing only the elements for which the function returns `True`. This new list is stored in the `lst` variable. Finally, the resulting list is printed using the `print` function.

هذا مقتطف كود بايثون يستخدم طريقة `filter` ودالة `lambda` لاستخراج الأرقام الفردية من القائمة.

تتم تهيئة القائمة `li` بمجموعة من الأعداد الصحيحة. يتم استخدام طريقة `filter` ودالة `lambda` لاستخراج الأرقام الفردية فقط من القائمة.

تأخذ طريقة `filter` دالة وقائمة كوسيطات. دالة `lambda` هي دالة مجهولة تأخذ وسيطة `n` وتعيد `True` إذا كانت `n` غير قابلة للقسمة على 2، وتعيد `False` خلاف ذلك. تقوم طريقة `filter` بعد ذلك بتطبيق دالة `lambda` على كل عنصر من عناصر قائمة `li` وتعيد قائمة جديدة تحتوي فقط على العناصر التي ترجع الدالة `True` لها. يتم تخزين هذه القائمة الجديدة في المتغير الأول.

أخيراً، تتم طباعة القائمة الناتجة باستخدام دالة `print`.

التمرين 81

الامر:

By using list comprehension, please write a program to print the list after removing numbers which are divisible by 5 and 7 in [12,24,35,70,88,120,155].

باستخدام list comprehension، يُرجى كتابة برنامج لطباعة القائمة بعد إزالة الأرقام القابلة للقسمة على 5 و 7 في [12،24،35،70،88،120،155].

الكود:

```
li = [12, 24, 35, 70, 88, 120, 155]
li = [x for x in li if x % 35 != 0]
print(li)
```

شرح ChatGPT:

The code defines a list of integers called `li`. Then, it uses a list comprehension to create a new list of integers, filtering out the elements of `li` that are divisible by 35.

The `if x % 35 != 0` part in the list comprehension checks whether `x` is not divisible by 35. If this condition is true, then `x` is added to the new list.

Finally, the code prints the new filtered list.

يعرف الكود قائمة من الأعداد الصحيحة تسمى `li`. بعد ذلك، يستخدم list comprehension لإنشاء قائمة جديدة من الأعداد الصحيحة، مع تصفية عناصر `li` القابلة للقسمة على 35.

يتحقق الجزء `if x % 35 != 0` في استيعاب القائمة مما إذا كان `x` غير قابل للقسمة على 35. إذا كان هذا الشرط صحيحاً، فسيتم إضافة `x` إلى القائمة الجديدة.

أخيراً، يطبع الكود القائمة المصفاة الجديدة.

التمرين 82

الامر:

By using list comprehension, please write a program to print the list after removing the 0th, 2nd, 4th, 6th numbers in [12,24,35,70,88,120,155].

باستخدام list comprehension، يُرجى كتابة برنامج لطباعة القائمة بعد إزالة الأرقام 0، 2، 4، 6 في [12،24،35،70،88،120،155].

الكود:

```
li = [12, 24, 35, 70, 88, 120, 155]
li = [li[i] for i in range(len(li)) if i % 2 != 0]
print(li)
```

شرح ChatGPT:

This code snippet uses Python to create a list `li` of numbers, and then creates a new list `li` with only the elements located at odd indices of the original list `li`.

It starts by defining a list `li` of 7 integers.

Then, it creates a new list `li` using a list comprehension. The list comprehension includes all elements located at odd indices (indexes 1, 3, 5, etc.) of the original list `li`.

Finally, the new list `li` is printed to the console.

يستخدم مقتطف الكود هذا بايثون لإنشاء قائمة `li` بالأرقام، ثم يقوم بإنشاء قائمة جديدة `li` تحتوي فقط على العناصر الموجودة في مؤشرات فردية من القائمة الأصلية `li`.

يبدأ بتعريف قائمة `li` تتكون من 7 أعداد صحيحة.

ثم يقوم بإنشاء قائمة جديدة `li` باستخدام list comprehension. يشمل استيعاب القائمة جميع العناصر الموجودة في مؤشرات فردية (الفهارس 1، 3، 5، إلخ) من القائمة الأصلية `li`.

أخيراً، تتم طباعة القائمة الجديدة `li` على وحدة التحكم.

التمرين 83

الامر:

By using list comprehension, please write a program to print the list after removing the 2nd - 4th numbers in [12,24,35,70,88,120,155].

باستخدام list comprehension، يُرجى كتابة برنامج لطباعة القائمة بعد إزالة الأرقام من الثاني إلى الرابع في [12،24،35،70،88،120،155].

الكود:

```
li = [12, 24, 35, 70, 88, 120, 155]
li = [li[i] for i in range(len(li)) if i < 3 or 4 < i]
print(li)
```

شرح ChatGPT:

This code snippet is written in Python and it defines a list called `li` which contains 7 numbers.

The next line of code creates a new list called `li` which contains elements of the original list `li` based on the conditions specified in a list comprehension. A list comprehension is a concise way of creating a new list from an existing one.

The condition specified in this list comprehension is `if i < 3 or 4 < i`, which means the elements will be selected if the index of the element (`i`) is less than 3 or greater than 4.

Finally, the new list is printed using the `print()` function.

تمت كتابة مقتطف الكود هذا بلغة بايثون وهو يعرف قائمة تسمى `li` تحتوي على 7 أرقام.

ينشئ السطر التالي من التعليمات البرمجية قائمة جديدة تسمى `li` تحتوي على عناصر من القائمة الأصلية `li` استناداً إلى الشروط المحددة في list comprehension.

يعد list comprehension طريقة موجزة لإنشاء قائمة جديدة من قائمة موجودة.

الشرط المحدد في list comprehension هذه هو إذا كان `if i < 3 or 4 < i`، مما يعني أنه سيتم تحديد العناصر إذا كان فهرس العنصر (`i`) أقل من 3 أو أكبر من 4.

أخيراً، تتم طباعة القائمة الجديدة باستخدام دالة `print()`.

التمرين 84

الامر:

By using list comprehension, please write a program generate a 358 3D array whose each element is 0.

باستخدام list comprehension، يُرجى كتابة برنامج لإنشاء مصفوفة 358 ثلاثية الأبعاد يكون كل عنصر فيها صفرًا.

الكود:

```
array = [[[0 for col in range(8)] for col in range(5)] for row in range(3)]
```

شرح ChatGPT:

The code creates a 3-dimensional list in Python, also called an array. This array has 3 layers. The first layer has 3 rows. The second layer has 5 columns. The third layer has 8 columns. Each value in the 3rd layer is set to zero.

In other words, the code creates a 3x5x8 matrix filled with zeros. The matrix has 3 pages, with each page having 5 rows and 8 columns, making a total of $3 \times 5 \times 8 = 120$ zeros in the array.

ينشئ الكود قائمة ثلاثية الأبعاد في بايثون، تسمى أيضاً مصفوفة. هذه المجموعة لديها 3 طبقات. الطبقة الأولى تتكون من 3 صفوف. الطبقة الثانية بها 5 أعمدة. الطبقة الثالثة بها 8 أعمدة. يتم تعيين كل قيمة في الطبقة الثالثة على الصفر.

بمعنى آخر، يُنشئ الكود مصفوفة 3 x 5 x 8 مليئة بالأصفار. تحتوي المصفوفة على 3 صفحات، كل صفحة بها 5 صفوف و 8 أعمدة، مما يجعل إجمالي $3 \times 5 \times 8 = 120$ صفراً في المصفوفة.

التمرين 85

الامر:

By using list comprehension, please write a program to print the list after removing the 0th, 4th, 5th numbers in [12,24,35,70,88,120,155].

باستخدام list comprehension، يُرجى كتابة برنامج لطباعة القائمة بعد إزالة الأرقام 0، 4، 5 في [12،24،35،70،88،120،155].

الكود:

```
li = [12, 24, 35, 70, 88, 120, 155]
li = [li[i] for i in range(len(li)) if i not in (0, 4, 5)]
print(li)
```

شرح ChatGPT:

This code defines a list of numbers named `li`. It then creates a new list named `li` which consists of elements from the original `li` list where the index is not equal to 0, 4, and 5.

The new `li` is constructed using a list comprehension. A list comprehension is a concise way to create a new list in Python. The syntax is:

```
new_list = [expression for item in iterable if condition]
```

In this code, `expression` is `li[i]`, `iterable` is `range(len(li))`, and `condition` is `i not in (0, 4, 5)`. This expression generates a new list with elements from the `li` list where the index is not 0, 4, or 5. Finally, the code prints the new `li`.

يعرف هذا الكود قائمة الأرقام المسماة `li`. ثم يقوم بإنشاء قائمة جديدة تسمى `li` والتي تتكون من عناصر من قائمة `li` الأصلية حيث لا يساوي الفهرس 0 و 4 و 5.

تم إنشاء لغة `li` الجديدة باستخدام `list comprehension`. `list comprehension` هي طريقة موجزة لإنشاء قائمة جديدة في بايثون. الصيغة هي:

```
new_list = [expression for item in iterable if condition]
```

في هذا الكود، التعبير `expression` هو `li[i]`، والقابل للتكرار `iterable` هو `range(len(li))`، والشرط `condition` هو `i not in (0, 4, 5)`. ينشئ هذا التعبير قائمة جديدة تحتوي على عناصر من قائمة `li` حيث لا يكون الفهرس 0 أو 4 أو 5. وأخيراً، يطبع الكود `li` الجديد.

التمرين 86

الامر:

By using list comprehension, please write a program to print the list after removing the value 24 in [12,24,35,24,88,120,155].

باستخدام list comprehension، يُرجى كتابة برنامج لطباعة القائمة بعد إزالة القيمة 24 في [12,24,35,24,88,120,155].

الكود:

```
li = [12, 24, 35, 24, 88, 120, 155]
li.remove(24) # this will remove only the first occurrence of 24
print(li)
```

شرح ChatGPT:

This code defines a list named `li` with some values. Then, it removes the first occurrence of the value 24 from the list using the `remove()` method. Finally, it prints the resulting list.

يعرّف هذا الكود قائمة باسم `li` مع بعض القيم. بعد ذلك، يزيل التواجد الأول للقيمة 24 من القائمة باستخدام طريقة `remove()`. أخيرًا، يقوم بطباعة القائمة الناتجة.

التمرين 87

الامر:

With two given lists [1,3,6,78,35,55] and [12,24,35,24,88,120,155], write a program to make a list whose elements are intersection of the above given lists.

باستخدام قائمتين محددتين [1,3,6,78,35,55] و [12,24,35,24,88,120,155]، اكتب برنامجاً لعمل قائمة تتقاطع عناصرها مع القوائم المذكورة أعلاه.

الكود:

```
list1 = [1, 3, 6, 78, 35, 55]
list2 = [12, 24, 35, 24, 88, 120, 155]
set1 = set(list1)
set2 = set(list2)
intersection = set.intersection(set1, set2)
print(intersection)
```

شرح ChatGPT:

The code defines two lists, `list1` and `list2` with some values. Then, it creates two sets `set1` and `set2` from these two lists respectively.

The `intersection` variable is assigned the result of the intersection of `set1` and `set2` using the `intersection` method of the `set` class. This method returns a set that contains all the elements that are present in both `set1` and `set2`.

Finally, the code prints the `intersection` variable which contains the result of the intersection of `set1` and `set2`.

يعرف الكود قائمتين، `list1` و `list2` مع بعض القيم. بعد ذلك، يقوم بإنشاء مجموعتين `set1` و `set2` من هاتين القائمتين على التوالي.

يتم تعيين متغير `intersection` نتيجة تقاطع `set1` و `set2` باستخدام طريقة `intersection` لفئة `set`. تقوم هذه الطريقة بإرجاع مجموعة تحتوي على جميع العناصر الموجودة في كل من `set1` و `set2`.

أخيراً، يطبع الكود متغير `intersection` الذي يحتوي على نتيجة تقاطع `set1` و `set2`.

التمرين 88

الامر:

With a given list [12,24,35,24,88,120,155,88,120,155], write a program to print this list after removing all duplicate values with original order reserved.

باستخدام قائمة معينة [12,24,35,24,88,120,155,88,120,155]، اكتب برنامجاً لطباعة هذه القائمة بعد إزالة جميع القيم المكررة بالترتيب الأصلي المحجوز.

الكود:

```
li = [12, 24, 35, 24, 88, 120, 155, 88, 120, 155]
for i in li:
    if li.count(i) > 1:
        li.remove(i)
print(li)
```

شرح ChatGPT:

This code snippet creates a list of integers called `li` and then loops through each element `i` in the list.

For each iteration of the loop, it checks if the count of `i` in the list is greater than 1. If the count is greater than 1, it removes the first occurrence of `i` from the list using the `remove` method.

Finally, it prints the updated list after all the duplicates have been removed.

ينشئ مقتطف الكود هذا قائمة من الأعداد الصحيحة تسمى `li` ثم حلقات عبر كل عنصر `i` في القائمة.

لكل تكرار للحلقة، فإنه يتحقق مما إذا كان عدد `i` في القائمة أكبر من 1. إذا كان العدد أكبر من 1، فإنه يزيل التكرار الأول لـ `i` من القائمة باستخدام طريقة `remove`.

أخيراً، يقوم بطباعة القائمة المحدثة بعد إزالة جميع التكرارات.

التمرين 89

الامر:

Define a class Person and its two child classes: Male and Female. All classes have a method "getGender" which can print "Male" for Male class and "Female" for Female class.

عرف فئة Person وفئتيه الفرعيين: Male و Female. كل الفئات لديها طريقة "getGender" والتي يمكنها طباعة "Male" لفئة الذكور و "Female" لفئة الإناث.

الكود:

```
class Person(object):
    def getGender(self):
        return "Unknown"

class Male(Person):
    def getGender(self):
        return "Male"

class Female(Person):
    def getGender(self):
        return "Female"

aMale = Male()
aFemale = Female()
print(aMale.getGender())
print(aFemale.getGender())
```

شرح ChatGPT:

The code creates a Python class called Person with a method getGender that returns the string "Unknown".

The class Male is a subclass of the Person class, and it overrides the getGender method to return the string "Male". Similarly, the class Female is a subclass of the Person class and it overrides the getGender method to return the string "Female".

Next, the code creates two objects, aMale and aFemale, from the classes Male and Female, respectively. Finally, the code prints the result of

calling the `getGender` method on each object. This will print "Male" and "Female".

يقوم الكود بإنشاء فئة بايثون تسمى `Person` مع أسلوب `getGender` الذي يقوم بإرجاع السلسلة "Unknown".

الفئة `Male` هي فئة فرعية من فئة `Person`، وهي تلغي طريقة `getGender` لإرجاع السلسلة النصية "Male". وبالمثل، فإن الفئة `Female` هي فئة فرعية من فئة `Person` وتتوارث طريقة `getGender` لإرجاع السلسلة "Female".

بعد ذلك، يقوم الكود بإنشاء كائنين، `aFemale` و `aMale`، من فئتي `Female` و `Male`، على التوالي. أخيراً، يطبع الكود نتيجة استدعاء طريقة `getGender` على كل كائن. سيؤدي هذا إلى طباعة "Male" و "Female".

التمرين 90

الامر:

Please write a program which count and print the numbers of each character in a string input by console.

يُرجى كتابة برنامج يقوم بحساب وطباعة أرقام كل حرف في سلسلة إدخال بواسطة وحدة التحكم.

الكود:

```
import string

s = input()
for letter in string.ascii_lowercase:
    cnt = s.count(letter)
    if cnt > 0:
        print("{} , {}".format(letter, cnt))
```

شرح ChatGPT:

This code is a program that takes in a string as input from the user and then counts the number of occurrences of each letter in the string (case-insensitive).

It uses the string module's `ascii_lowercase` constant to get a list of all lowercase letters in the alphabet. The program loops over each letter in `ascii_lowercase`, and for each letter, it counts the number of occurrences of that letter in the input string `s` using the `count` method. If the count of a letter is greater than 0, the program prints the letter and its count in the format "`{}, {}`".

هذا الكود عبارة عن برنامج يأخذ سلسلة كمدخلات من المستخدم ثم يحسب عدد تكرارات كل حرف في السلسلة (غير حساس لحالة الأحرف).

ويستخدم ثابت `ascii_lowercase` الخاص بوحدة `string` للحصول على قائمة بجميع الأحرف الصغيرة في الأبجدية. يدور البرنامج فوق كل حرف في `ascii_lowercase`، ولكل حرف، يقوم بحساب عدد تكرارات هذا الحرف في سلسلة الإدخال `s` باستخدام طريقة `count`. إذا كان عدد الحرف أكبر من 0، يقوم البرنامج بطباعة الحرف وعدده بالتنسيق "`{}, {}`".

التمرين 91

الامر:

Please write a program which accepts a string from console and print it in reverse order.

يُرجى كتابة برنامج يقبل سلسلة من وحدة التحكم وطباعتها بترتيب عكسي.

الكود:

```
s = input()
s = "".join(reversed(s))
print(s)
```

شرح ChatGPT:

This code takes an input string from the user and reverses its characters. The `input()` function is used to get the string from the user. Then, the `reversed` function is used to reverse the order of characters in the string. Finally, the `join` function is used to join the reversed characters together into a new string, which is then printed to the console.

يأخذ هذا الكود سلسلة إدخال من المستخدم ويعكس أحرفها. تُستخدم `input()` للحصول على السلسلة من المستخدم. بعد ذلك، يتم استخدام دالة `reversed` لعكس ترتيب الأحرف في السلسلة.

أخيراً، يتم استخدام دالة `join` لربط الأحرف المعكوسة معاً في سلسلة جديدة، والتي يتم طباعتها بعد ذلك على وحدة التحكم.

التمرين 92

الامر:

Please write a program which accepts a string from console and print the characters that have even indexes.

يُرجى كتابة برنامج يقبل سلسلة من وحدة التحكم وطباعة الأحرف التي تحتوي على فهارس زوجية.

الكود:

```
s = "H1e2l3l4o5w6o7r8l9d"
s = [s[i] for i in range(len(s)) if i % 2 == 0]
print("".join(s))
```

شرح ChatGPT:

The given code takes a string `s` and creates a new list of characters by picking characters at even-indexed positions in the input string `s`.

The final output is a concatenation of all the characters in the new list, represented as a string. The result is printed on the console.

يأخذ الكود المعطى سلسلة `s` وينشئ قائمة جديدة من الأحرف عن طريق اختيار الأحرف في المواضع ذات الفهرسة الزوجية في سلسلة الإدخال `s`.

الإخراج النهائي هو عبارة عن سلسلة من جميع الأحرف في القائمة الجديدة، ممثلة كسلسلة. تتم طباعة النتيجة على وحدة التحكم.

التمرين 93

الامر:

Please write a program which prints all permutations of [1,2,3].

يُرجى كتابة برنامج يطبع جميع التباديل [1,2,3].

الكود:

```
import itertools
print(list(itertools.permutations([1, 2, 3])))
```

شرح ChatGPT:

This code uses the `itertools` module in Python to generate all possible permutations of a given list [1, 2, 3].

The `permutations` function from the `itertools` module takes a list as input and returns an iterable object containing all possible arrangements (permutations) of the elements in that list.

In this code, the permutations are computed for the list [1, 2, 3], and the result is then converted to a list using the `list` function.

The final output of the code will be a list of all possible permutations of the list [1, 2, 3].

يستخدم هذا الكود وحدة `itertools` في بايثون لإنشاء جميع التباديل الممكنة لقائمة معينة [1, 2, 3].

تأخذ دالة `permutations` من الوحدة النمطية `itertools` قائمة كمدخلات وتعيد كائناً متكرراً يحتوي على جميع الترتيبات الممكنة (التباديل) للعناصر في تلك القائمة.

في هذا الكود، يتم حساب التباديل للقائمة [1, 2, 3]، ثم يتم تحويل النتيجة إلى قائمة باستخدام دالة `list`.

سيكون الناتج النهائي للكود عبارة عن قائمة بجميع التباديل الممكنة للقائمة [1, 2, 3].

التمرين 94

الامر:

Write a program to solve a classic ancient Chinese puzzle: We count 35 heads and 94 legs among the chickens and rabbits in a farm. How many rabbits and how many chickens do we have?

اكتب برنامجاً لحل لغز صيني قديم كلاسيكي: نحسب 35 رأساً و 94 ساقاً بين الدجاج والأرانب في مزرعة. كم عدد الأرانب وكم عدد الدجاجات لدينا؟

الكود:

```
def solve(numheads, numlegs):
    ns = "No solutions!"
    for i in range(numheads + 1):
        j = numheads - i
        if 2 * i + 4 * j == numlegs:
            return i, j
    return ns, ns

numheads = 35
numlegs = 94
solutions = solve(numheads, numlegs)
print(solutions)
```

شرح ChatGPT:

The code defines a function named `solve` that takes two parameters `numheads` and `numlegs`. It then uses a `for` loop to iterate over a range of values starting from 0 to the value of `numheads + 1`. Within the loop, it calculates the value of `j` by subtracting the current value of `i` from `numheads`. Then, it checks if the result of the expression $2 * i + 4 * j$ is equal to `numlegs`. If it is, it returns a tuple of values `i` and `j`.

If the loop completes and no solutions are found, the function returns a tuple `ns, ns` which contains the string "No solutions!".

Finally, the code sets `numheads` to 35 and `numlegs` to 94, then calls the `solve` function passing these values. The returned solution is then printed.

يعرف الكود دالة تسمى `solve` تأخذ معلمتين `numheads` و `numlegs`. ثم تستخدم حلقة `for` للتكرار على مدى مجموعة من القيم تبدأ من 0 إلى قيمة `numheads + 1`. داخل الحلقة، تحسب

قيمة j عن طريق طرح القيمة الحالية لـ i من numheads . ثم يتحقق مما إذا كانت نتيجة التعبير $2j + 4i + \text{numlegs}$ تساوي numheads . إذا كانت كذلك، فإنها تُرجع مجموعة من القيم i و j . إذا اكتملت الحلقة ولم يتم العثور على حلول، فستُرجع الدالة الصفوف ns , ns التي تحتوي على السلسلة "No solutions!".

أخيراً، يعرف الكود numheads إلى 35 و numlegs إلى 94، ثم يستدعي دالة solve لتمرير هذه القيم. ثم يتم طباعة الحل الذي تم إرجاعه.

التمرين 95

الامر:

Given the participants' score sheet for your University Sports Day, you are required to find the runner-up score. You are given scores. Store them in a list and find the score of the runner-up.

If the following string is given as input to the program:

- 5 2 3 6 6 5

Then, the output of the program should be:

- 5

بالنظر إلى ورقة نتائج المشاركين ليوم الرياضة الجامعي، يتعين عليك العثور على درجة الوصيف. لقد حصلت على درجات. قم بتخزينها في قائمة وابحث عن نتيجة الوصيف.

إذا تم إعطاء السلسلة التالية كمدخلات للبرنامج:

- 5 6 6 3 2 5

بعد ذلك، يجب أن يكون إخراج البرنامج:

- 5

الكود:

```
arr = map(int, input().split())
arr = list(set(arr))
arr.sort()
print(arr[-2])
```

شرح ChatGPT:

The code first takes an input from the user of a list of integers separated by spaces. The input is then passed to the `map` function which converts the strings into integers and returns a map object.

Next, the map object is converted into a list using the `list` function, and duplicates are removed by converting the list into a set and then back into a list.

Finally, the list is sorted in ascending order and the second last element is printed which is the second largest element in the list.

يأخذ الكود أولاً مُدخلًا من المستخدم لقائمة أعداد صحيحة مفصولة بمسافات. ثم يتم تمرير الإدخال إلى دالة map التي تحول السلاسل إلى أعداد صحيحة وتعيد كائن map.

بعد ذلك، يتم تحويل كائن map إلى قائمة باستخدام دالة list، وتتم إزالة التكرارات عن طريق تحويل القائمة إلى مجموعة ثم العودة إلى قائمة.

أخيرًا، يتم فرز القائمة بترتيب تصاعدي ويتم طباعة العنصر الثاني الأخير وهو ثاني أكبر عنصر في القائمة.

التمرين 96

الامر:

You are given a string S and width W . Your task is to wrap the string into a paragraph of width.

If the following string is given as input to the program:

- ABCDEFGHIJKLMNOPQRSTUVWXYZ 4

Then, the output of the program should be:

- ABCD EFGH IJKL IMNO QRST UVWX YZ

يتم منحك سلسلة S وعرض W . مهمتك هي لف السلسلة في فقرة من العرض.

إذا تم إعطاء السلسلة التالية كمدخلات للبرنامج:

- ABCDEFGHIJKLMNOPQRSTUVWXYZ 4

بعد ذلك، يجب أن يكون إخراج البرنامج:

- ABCD EFGH IJKL IMNO QRST UVWX YZ

الكود:

```
import textwrap

string = input()
width = int(input())

print(textwrap.fill(string, width))
```

[شرح ChatGPT](#):

The code snippet is using the `textwrap` module to wrap a given text string into multiple lines.

The user inputs the text string and the width (number of characters per line) they want to wrap the text into.

The code then calls the `fill` method from the `textwrap` module and passes in the text string and the desired width as arguments. This method returns

the text string, wrapped into multiple lines so that no line exceeds the width specified.

Finally, the wrapped text is printed to the console.

يستخدم مقتطف الكود وحدة `textwrap` لالتفاف سلسلة نصية معينة إلى عدة أسطر.

يقوم المستخدم بإدخال سلسلة النص والعرض (عدد الأحرف في كل سطر) الذي يريد التفاف النص فيه.

ثم يستدعي الكود طريقة `fill` من وحدة `textwrap` ويمرر في السلسلة النصية والعرض المطلوب كوسائط. تقوم هذه الطريقة بإرجاع السلسلة النصية، ملفوفة في عدة أسطر بحيث لا يتجاوز أي سطر العرض المحدد.

أخيراً، تتم طباعة النص الملتف على وحدة التحكم.

التمرين 97

الامر:

You are given a date. Your task is to find what the day is on that date.

لقد أعطيت موعداً. مهمتك هي العثور على اليوم في ذلك التاريخ.

الكود:

```
import calendar

month, day, year = map(int, input().split())
dayId = calendar.weekday(year, month, day)
print(calendar.day_name[dayId].upper())
```

شرح ChatGPT:

This code is using the `calendar` module in Python to find the day of the week for a given date.

First, it takes three inputs from the user: `month`, `day`, and `year`. These three inputs are combined into a single string, then separated and converted into integers using the `map` function and the `int` function.

Next, the `calendar.weekday` function is used to determine the weekday for the given date. This function takes three arguments: the year, the month, and the day. The result is an integer representing the day of the week, where Monday is 0 and Sunday is 6.

Finally, the `calendar.day_name` array is used to convert the integer representation of the day of the week into a string, and the `upper` function is used to convert the string to uppercase. The resulting string is then printed.

يستخدم هذا الكود وحدة `calendar` في بايثون للعثور على يوم الأسبوع لتاريخ معين.

أولاً، يأخذ ثلاث مدخلات من المستخدم: الشهر `month` واليوم `day` والسنة `year`. يتم دمج هذه المدخلات الثلاثة في سلسلة واحدة، ثم يتم فصلها وتحويلها إلى أعداد صحيحة باستخدام دالة `map` والدالة `.int`.

بعد ذلك، يتم استخدام دالة `calendar.weekday` لتحديد يوم الأسبوع للتاريخ المحدد. تأخذ هذه الدالة ثلاث وسيطات: السنة والشهر واليوم. والنتيجة هي عدد صحيح يمثل يوم الأسبوع، حيث يكون الاثنين 0 والأحد هو 6.

أخيراً، يتم استخدام مصفوفة `calendar.day_name` لتحويل تمثيل العدد الصحيح ليوم الأسبوع إلى سلسلة، ويتم استخدام دالة `upper` لتحويل السلسلة إلى أحرف كبيرة. ثم يتم طباعة السلسلة الناتجة.

التمرين 98

الامر:

Given 2 sets of integers, M and N, print their symmetric difference in ascending order. The term symmetric difference indicates those values that exist in either M or N but do not exist in both.

بالنظر إلى مجموعتين من الأعداد الصحيحة، M و N، اطبع الفرق المتماثل بينهما بترتيب تصاعدي. يشير مصطلح الاختلاف المتماثل إلى تلك القيم الموجودة في M أو N ولكنها غير موجودة في كليهما.

الكود:

```
n = int(input())
set1 = set(map(int, input().split()))

m = int(input())
set2 = set(map(int, input().split()))

ans = list(set1 ^ set2)
ans.sort()
for i in ans:
    print(i)
```

شرح ChatGPT:

This code takes two sets of integers as input and finds the symmetric difference between them. The symmetric difference of two sets is defined as the set of elements that are in either of the sets, but not in their intersection.

1. The first input line n is the number of elements in set 1, and the next line is the elements of set1, separated by spaces. These elements are converted to integers using the map function and stored in set1.
 2. The next two lines are similar, with m being the number of elements in set2, and the next line being the elements of set 2. These elements are stored in set2.
 3. The symmetric difference between the two sets is found using the ^ operator, and the result is stored in the list ans.
 4. The list ans is sorted and the elements are printed one by one.
- يأخذ هذا الكود مجموعتين من الأعداد الصحيحة كمدخلات ويجد الفرق المتماثل بينهما. يتم تعريف الاختلاف المتماثل بين مجموعتين على أنه مجموعة العناصر الموجودة في أي من المجموعتين، ولكن ليس في تقاطعهما.

1. سطر الإدخال الأول n هو عدد العناصر في set1، والسطر التالي هو عناصر set1، مفصولة بمسافات. يتم تحويل هذه العناصر إلى أعداد صحيحة باستخدام دالة map وتخزينها في set1.
2. يتشابه السطران التاليان، حيث يمثل m عدد العناصر في set2، والسطر التالي يمثل عناصر set2. ويتم تخزين هذه العناصر في set2.
3. تم العثور على الاختلاف المتماثل بين المجموعتين باستخدام عامل التشغيل ^، ويتم تخزين النتيجة في قائمة ans.
4. يتم فرز القائمة ans ويتم طباعة العناصر واحدة تلو الأخرى.

التمرين 99

الامر:

You are given words. Some words may repeat. For each word, output its number of occurrences. The output order should correspond with the input order of appearance of the word. See the sample input/output for clarification.

If the following string is given as input to the program:

- 4 bcdef abcdefg bcde bcdef

Then, the output of the program should be:

- 3 2 1 1.

لقد أُعطيت كلمات. قد تتكرر بعض الكلمات. لكل كلمة، قم بإخراج عدد تكراراتها. يجب أن يتوافق ترتيب الإخراج مع ترتيب الإدخال الخاص بظهور الكلمة. انظر عينة المدخلات / المخرجات للتوضيح.

إذا تم إعطاء السلسلة التالية كمدخلات للبرنامج:

- 4bcdef abcdefg bcde bcdef

بعد ذلك، يجب أن يكون إخراج البرنامج:

- 1 1 2 3

الكود:

```
n = int(input())

word_list = []
word_dict = {}

for i in range(n):
    word = input()
    if word not in word_dict:
        word_list.append(word)
    word_dict[word] = word_dict.get(word, 0) + 1

print(len(word_list))
for word in word_list:
```



```
print(word_dict[word], end=" ")
```

شرح ChatGPT:

This code takes in an integer n as input which represents the number of words to be entered. Then it takes n words as input and stores them in a list word_list. It also creates a dictionary word_dict where the key is a word and the value is the frequency of that word in the input words.

It then checks if a word is not already present in the dictionary, if yes, it appends the word to the list word_list. It then increments the value of the word in the dictionary by 1.

In the end, it prints the length of the list word_list and the frequency of each word in word_dict separated by a space.

يأخذ هذا الكود عددًا صحيحًا n كمدخل يمثل عدد الكلمات المراد إدخالها. ثم يأخذ n من الكلمات كمدخلات ويخزنها في قائمة word_list. يقوم أيضًا بإنشاء قاموس word_dict حيث يكون المفتاح كلمة والقيمة هي تكرار تلك الكلمة في كلمات الإدخال.

ثم يتحقق مما إذا كانت الكلمة غير موجودة بالفعل في القاموس، وإذا كانت الإجابة بنعم، فإنها تقوم بإلحاق الكلمة بقائمة word_list. ثم يقوم بزيادة قيمة الكلمة في القاموس بمقدار 1.

في النهاية، يقوم بطباعة طول قائمة word_list وتردد كل كلمة في word_dict مفصولة بمسافة.

التمرين 100

الامر:

You are given a string. Your task is to count the frequency of letters of the string and print the letters in descending order of frequency.

- If the following string is given as input to the program: - aabbbccde
- Then, the output of the program should be: - b 3 a 2 c 2 d 1 e 1

يتم إعطاء سلسلة ومهمتك هي حساب تكرار أحرف السلسلة وطباعة الحروف بترتيب تنازلي للتكرار.

- إذا تم إعطاء السلسلة التالية كمدخلات للبرنامج: - aabbbccde
- بعد ذلك، يجب أن يكون إخراج البرنامج: - b 3 a 2 c 2 d 1 e 1.

الكود:

```
s = list(input())

dict_count_ = {k: s.count(k) for k in s}
list_of_tuples = [(k, v) for k, v in dict_count_.items()]
list_of_tuples.sort(key=lambda x: x[1], reverse=True)

for item in list_of_tuples:
    print(item[0], item[1])
```

شرح ChatGPT:

The following code in Python takes an input string, *s*, and performs the following actions:

1. Creates a dictionary, dict count, where each character in the string is a key and its value is the count of that character in the string.
2. Creates a list of tuples, list of tuples, where each tuple consists of a character and its count.
3. Sorts the list of tuples in descending order of the count of characters.
4. Prints the character and its count for each tuple in the sorted list.

تأخذ الكود التالي في بايثون سلسلة إدخال، *s*، وتنفيذ الإجراءات التالية:

1. يُنشئ قاموساً، dict count، حيث يكون كل حرف في السلسلة مفتاحاً وتكون قيمته هي عدد هذا الحرف في السلسلة.

2. يقوم بإنشاء قائمة المجموعات، `list_of_tuples` ، حيث تتكون كل مجموعة من حرف وعددها.
3. يفرز `list_of_tuples` بترتيب تنازلي لعدد الأحرف.
4. يطبع الحرف وعدده لكل مجموعة في القائمة التي تم فرزها.