تم تحميل هذا الملف من موقع المناهج البحرينية

مذكرة ريض 152

موقع المناهج ⇦ المناهج البحرينية ⇦ الصف الأول الثانوي ⇦ رياضيات ⇦ الفصل الثاني ⇦ مذكرات وبنوك ⇦ الملف

تاريخ إضافة الملف على موقع المناهج: 2025-03-11 13:18:34

المزيد من مادة رياضيات:	ملفات ا كتب للمعلم ا كتب للطالب ا اختبارات الكترونية ا اختبارات ا حلول ا عروض بوربوينت ا أوراق عمل منهج انجليزي ا ملخصات وتقارير ا مذكرات وبنوك ا الامتحان النهائي ا للمدرس

التواصل الاجتماعي بحسب الصف الأول الثانوي

صفحة المناهج البحرينية على فيسبوك					
المواد على تلغرام	التربية الاسلامية	اللغة العربية	اللغة الانجليزية	الرياضيات	

المزيد من الملفات بحسب الصف الأول الثانوي والمادة رياضيات في الفصل الثاني

1	مراجعة الاختبار الأول مقرر ريض 152
2	حل بطاقة مراجعة الاختبار الأول
3	ملف أعمال الطالب في مادة الرياضيات
4	مذكرة ريض 152
5	مراجعة الاختبار الثاني مقرر ريض 152

المقرر: تقن 106

البرمجة بلغة البايثون

ملخص شامل لدروس المقرر

من إعداد: أ.لطيفة راشد

2025-2024

الدرس الأول : الأنظمة العددية

أنواع الأنظمة العددية المطلوبة في مقرر تقن 106 :

1. نظام العد الثنائي : لغة الكمبيوتر تمثل الرموز 0 و 1 مثال: 2(100110)
2. نظام العد العشري :اللغة المستخدمة في حياتنا اليومية و تمثل الرموز من 0 الى 9 مثال: 10(50)

الطريقة الصحيحة للتحويل بين الأنظمة العددية هى :

يجب حفظ جدول العد العشري للتمكن من التحويل بين الأنظمة العددية (الجدول يبدأ من 1)

1	2	4	8	16	32	64	128	256	512

- للتحويل بين نظام العد العشري الى نظام العد الثنائي : نستخدم عملية الطرح و نبدأ من أقرب رقم يمكن أن نطبق علية عملية الطرح وهو (الرقم الأصغر) ، نبدأ من جهة اليسار الى اليمين للوصل الى العدد 1 ويكون باقي عملية الطرح 0 ، الرقم الذي يتم استخدامه يوضع رمز 1 في اسفل الخانة و الرقم الذي لم يتم استخدامه يوضع رمز 0 في اسفل الخانة، تنقل الأرقام بشكل صحيح و يتم وضع رقم 2 بجانب الرقم ليمثل نظام العد الثنائي .

مثال :

حول نظام العد العشري الى النظام الثنائي : 10(12)

1	2	4	8
0	0	1	1

12-8=4-4=0

نظام العد الثنائي هو :2(1100)

- للتحويل بين نظام العد الثنائي الى نظام العد العشري : نستخدم عملية الجمع ، نقوم بوضع الأرقام الثنائية بالترتيب الصحيح من جهة اليمين الى اليسار في جدول العد العشري. نقوم بجمع الأعداد العشرية التي تتضمن رمز 1 . يتم وضع رقم 10 بجانب ناتج الجمع ليمثل نظام العد العشري .

مثال :

حول نظام العد الثنائي الى النظام العشري: 2(1011)

1	2	4	8
1	1	0	1

8+2+1=13

نظام العد العشري هو: 10(13)

ملاحظة مهمة جداً :لترتيب الأعداد من الأكبر الى الأصغر أو العكس يجب تحويل الأعداد الثنائية الى أعداد عشرية باستخدام جدول العد العشري (لا تنظر الى الرقم نفسة أنظر الى الرقم الصغير الموجود بجناب الرقم أولاً).

إعداد:أ.لطيفة راشد **مديرة المدرسة: منال سنان**

الدرس الثاني : الخوارزميات و الخريطة التدفقية

الخوارزمية: هي تسلسل منطقي للأحداث قصد الوصول إلى الهدف أو النتيجة المطلوبة.

لكتابة اي خوارزمية نحتاج الى كتابتها على 3 مراحل:

المدخلات: نوضح خلالها جميع القيم المدخلة من قبل المستخدم والتي سيتم تخزينها في متغيرات مختلفة.

المعالجة: يتم تحديد القيم الثابتة والعمليات الحسابية إن وجدت أو المقارنة .

المخرجات : تحديد ما سيتم عرضه للمستخدم .

مثال:

أكتب الخطوات اللازمة لتنفيذ برنامج يستقبل عددين ثم يقوم بجمعهما وطباعة الناتج.

المدخلات : إدخال عددين

المعالجة : جمع العددين

المخرجات: عرض نتيجة الجمع

الخريطة التدفقية: هو حل رسومي للمشكلة البرمجية ، حيث ترتبط مجموعة من الأشكال الهندسية بعضها ببعض في ترتيب منطقي لتسلسل الأحداث و الإجراءات البرمجية للحل الخوارزمي.

الأشكال المكوّنة للخريطة التدفّقيّة:

الشكل	الاسم عربي	الأسم انجليزي	الوصف
	البداية/النهاية	Start/End	يُستخدم في بداية ونهاية الخريطة التدفّقيّة
	مدخلات/مخرجات	Input/output	يُستخدم عند إدخال المعطيات و/أو عرض المخرجات
	معالجة	process	عمليات حسابية | منطقيّة، تعليمة برمجيّة، اسناد قيم، استخدام دوال
	اتخاذ القرار	Decision	عندما يكون هناك اجراء سيُتّخذ بناءً على شرط نتيجته (نعم/لا)
	الاتجاه	Flow arrow	يبيّن اتجاه الارتباط بين مختلف أشكال الخريطة التدفّقيّة

مثال :

ارسم خريطة تدفقية لبرنامج يقوم بحساب نتيجة المعادلة (x*y-5) ثم عرض النتيجة. قبل البدء برسم الخريطة التدفقية، يجب أن تحدد المدخلات، مراحل المعالجة والمخرجات.

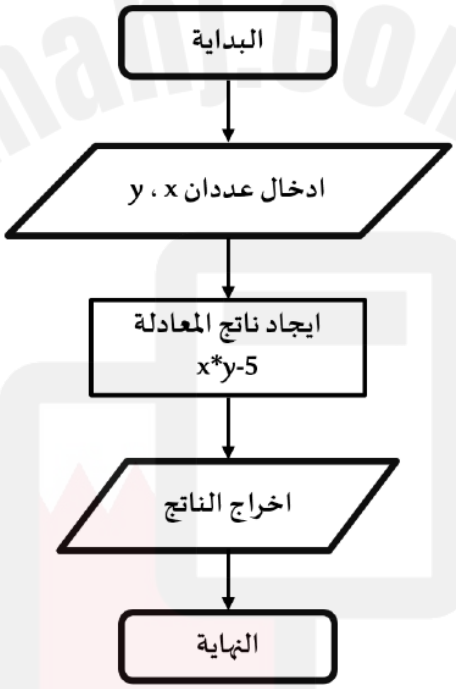
في هذا المثال تلاحظ الآتي:

المدخلات : لم يتم ذكر المدخلات بالتحديد، ولكن بإمكاننا استنتاج ذلك من المعادلة ذاتها التي تشير إلى المتغيران x و y

المعالجة: حساب المعادلة x*y-5

المخرجات: ناتج المعادلة.

بعد أن ننتهي من تحديد المطلوب من البرنامج، نرسم الخريطة التدفقية للحل البرمجي:



الدرس الثالث : المتغيرات

قواعد تعريف المتغيرات :

- لا يمكن أن تبدأ برقم
- لا يمكن استخدام الكلمات المفتاحية مثل :input,print,float,int,str
- لا يمكن استخدام الرموز الأخرى للفصل بين الحروف و الأرقام مثل (@,$,%,&,*,!,#....)
- يمكن الفصل بين الحروف أو الكلمات و الأرقام باستخدام الرمز (_) فقط
- يمكن الدمج بين الحروف الكبيرة و الصغيرة
- يمكن الدمج بين الحروف و الأرقام

مثال :

حدد إذا ما كانت تسمية المتغيرات التالية صحيحة أم خاطئة مع ذكر السبب :

- Total	إجابة صحيحة
- 5abx	إجابة خاطئة لا يمكن أن تبدأ برقم
- Hell#o	إجابة خاطئة لا يمكن استخدام الرمز #
- f_name	إجابة صحيحة
- Input	إجابة خاطئة لا يمكن استخدام الكلمات المفتاحية
- x1	إجابة صحيحة

ملاحظة مهمة جداً: في البرمجة اتبع قواعد تعريف المتغيرات عند إنشاء أي متغير واحفظ فيه قيمة باستخدام الرمز (=)
مثال : x=5 .

الدرس الرابع : أنواع البيانات

أنواع البيانات :

- عدد صحيح int مثل : 10
- عدد عشري float مثل : 2.2
- قائمة list مثل : [1,3,6]
- قيمة منطقية bool مثل : True
- تاريخ date مثل : datetime.datetime.now()
- بيانات نصية str مثل : "hello"

لتحديد نوع البيانات للمتغيرات بشكل صحيح لابد من ملاحظة بعض الرموز :

مثال :

العدد الصحيح int **يمكن تمييز النوع من خلال النظر لوجود رقم صحيح (بدون بوينتات) أو وجود الدالة int() ، //، %**	**y=30** **x=int(input("Enter a number"))** **t=5.5 العدد هنا عدد عشري** **r=int(t)**
العدد العشري float **يمكن تمييز النوع من خلال النظر لوجود رقم عشري (مع بوينتات) أو وجود الدالة float()، /**	**z=99.9** **u=float(input("Enter a number"))** **c="14" العدد هنا قيمة نصية** **e=float(c)**
قائمة أو مصفوفة أو متسلسلة list **يمكن تمييز النوع من خلال وجود الأقواس []**	**List1=[1,4,7]** **List2=["B"،"S"]** **List3=[4.5,5.7,3.6]**
قيمة منطقية bool **يمكن تمييز النوع من خلال وجود الرموز التالية : >** **,<,>=,<=,!=,==,and,or,not** **أو عندما تكون الإجابة True or False**	**Q=5>=7** **V=False** **B=not(7==4)** **C=True**
تاريخ date **يمكن تمييز النوع من خلال وجود الدوال المضمنة المخصصة لعرض التاريخ والوقت الحالي**	**from datetime import datetime** **d=datetime.now()** **d1=datetime(2024,12,22)**
بيانات نصية str **يمكن تمييز النوع من خلال وجود الرمز "" أو وجود الدالة input()، str()**	**s="Latifa"** **B="True"** **Name=input("Enter your name")**

ملاحظة مهمة جداً: بالنسبة للقيمة المسندة الى المتغير لتحديد نوعه بشكل صحيح انظر الى أول رمز بعد إشارة = ، ركز في قيمة المتغير سواء كانت قيمة نصية ، رقمية ، منطقية ...الخ

الدرس الخامس : العوامل الحسابية

أنواع العوامل للتعامل مع المتغيرات :

1- عوامل الحسابية
2- عوامل المنطقية
3- عوامل المقارنة

العوامل الحسابية :

مثال :

رمز العامل الحسابي : + من خلال هذا الرمز يمكن تطبيق عملية جمع الاعداد بعضهم البعض أو جمع النصوص بعضهم البعض .	x=2 y=6 print(x+y) output: 8 x="hello" y="Latifa " print(x+y) output: helloLatifa x=7 y="L" print(x+y) output: Error
رمز العامل الحسابي : - من خلال هذا الرمز يمكن تطبيق عملية الطرح للأعداد فقط .	x=10 y=20 print(x-y) output: -2 x=5 y=1 print(x-y) output: 4
رمز العامل الحسابي : / من خلال هذا الرمز يمكن تطبيق عملية القسمة للأعداد بالطريقة المعتادة (مع بوينتات)	x=5 y=3 print(x/y) output: 1.6 x=7 y=10 print(x/y) output: 0.7
رمز العامل الحسابي : // من خلال هذا الرمز يمكن الحصول على الناتج الصحيح من عملية القسمة (بدون بوينتات)	x=5 y=3 print(x//y) output: 1 x=7 y=10 print(x//y) output: 0
رمز العامل الحسابي : % من خلال هذا الرمز يمكن الحصول على الباقي من عملية القسمة	x=5 y=3 print(x%y) 5-3=2 output: 2 x=7 y=10 print(x%y) output: 7 x=20 y=4 print(x%y) 20-4=16-4=12-4=8-4=4-4=0 output: 0
رمز العامل الحسابي : * من خلال هذا الرمز يمكن تطبيق عملية الضرب للأعداد أو تكرار النصوص حسب عملية الضرب	x=2 y=2 print(x*y) output: 4 x=3 y="Latifa " print(x*y) output: LatifaLatifaLatifa
رمز العامل الحسابي : ** من خلال هذا الرمز يمكن الحصول على قوة العدد	x=2 y=4 print(x**y) 2^4 output: 8

عوامل المقارنة و العوامل المنطقية :

مثال :

رمز العامل	مثال
== يساوي	print(7==7) output: True print(2==3) output: False
> أكبر من => أكبر من أو يساوي	print(5>6) output: False print(5>=5) output: True
< أصغر من =< أصغر من أو يساوي	print(1<0) output: False print(8<=10) output: True
=! لا يساوي	print(4!=9) output: True print(6!=6) output: False
and و القاعدة : T and T =T T and F=F F and F=F	print(2>=2 and 2<4) output: True print(5>7 and 2>8) output: False print(6+2>15 and 15!=15) output: False
or أو القاعدة : T or T =T T or F=T F or F=F	print(3<7 or 10<40) output: True print(9>2 or 10<1) output: True print(12<4 or 12>18) output: False
not لا القاعدة : not (T)=F not (F)=T	print(not(5!=5)) output:False print(not(6>2)) output: True

ملاحظة مهمة جداً: بالنسبة للعوامل الحسابية عملية الجمع تطبق عندما تكون أنواع البيانات متشابهة سواء كانت نصوص أو ارقام ، ضرب الرقم بالكلمة أو النص سيكرر الكلمة دون ترك مسافة بين الكلمة والأخرى ، القسمة إذا كان الرمز خط / سيكون الناتج عشري وإذا خطان // سيكون الناتج عدد صحيح. بالنسبة لعوامل المقارنة والعوامل المنطيقة لا تنسى التعويض للتمكن من الإجابة بشكل صحيح وستكون النتيجة دائماً True or False.

الدرس السادس : جمل الادخال و الإخراج

جمل الادخال أو المدخلات: هي قيمة مجهولة مدخلة من قبل المستخدم ويتم تخزينها في متغير.

القاعدة العامة للمدخلات : المتغير =input("الرسالة التي ستظهر للمستخدم")
مثال عند إدخال نص : اكتب بلغة البايثون برنامج يطلب من المستخدم إدخال اسمة . x=input("Enter your name ")
مثال عند إدخال عدد صحيح : اكتب بلغة البايثون برنامج يطلب من المستخدم إدخال عمره. x=int(input("Enter your age "))
مثال عند إدخال عدد عشري : اكتب بلغة البايثون برنامج يطلب من المستخدم إدخال وزنة. x=float(input("Enter your weight "))

جمل الإخراج أو المخرجات: هي النتيجة التي ستظهر للمستخدم .

القاعدة العامة للمخرجات : print()
مثال لطباعة نص : اكتب بلغة البايثون برنامج يقوم بطباعة العبارة "welcome" print("welcome")
مثال لطباعة متغير : x=10 اكتب بلغة البايثون برنامج يقوم بطباعة قيمة x . print(x)
مثال لطباعة نص مع متغير : x=10 اكتب بلغة البايثون برنامج يقوم بطباعة العبارة "num=" متبوع بالمتغير x . print("num=",x)
مثال لطباعة معادلة: x=3 y=10 اكتب بلغة البايثون برنامج يقوم بطباعة مجموع العددين. print(x+y)

العلامات الخاصة تستخدم مع النصوص :

"\n": تستخدم لإدراج سطر جديد.

"\t":تستخدم لإدراج مسافة.

"\'":تستخدم لإدراج علامة تنصيص مفردة.

"\"":تستخدم لإدراج علامة تنصيص مزدوجة.

ملاحظة مهمة جداً:

المدخلات : يجب تعريف متغير لتخزين كل قيمة مدخلة على حده، هناك بعض الكلمات توضح استخدام الدلة مثل (إدخال ، يستقبل ، القيمة المدخلة، تعبئة..الخ)

المخرجات: عند طباعة نص يجب استخدام الرمز " " ، للفصل بين المتغيرات و النصوص يتم استخدام الفاصلة (,) ، تكتب المتغيرات كما هي من غير أي إضافة.

إعداد:أ.لطيفة راشد
مديرة المدرسة: منال سنان

الدرس السابع : الجملة الشرطية

أنواع الجمل الشرطية :

1- البسيطة
2- الكاملة
3- المركبة

الجملة الشرطية البسيطة :

القاعدة العامة للجملة الشرطية البسيطة :

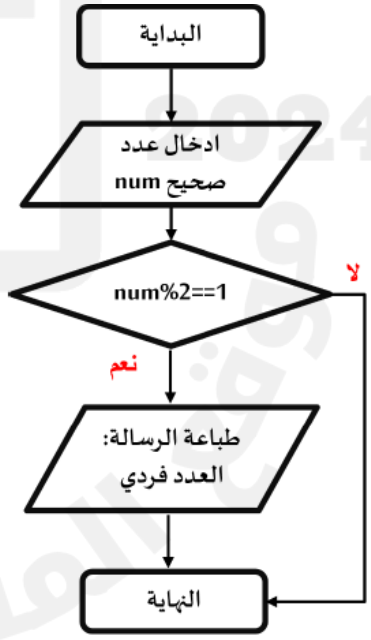
if condition(*الشرط) :
التعليمات ⟵ يجب ترك مسافه بادئة بعد الشرط

مثال : أرسم الخريطة التدفقية، ثم أكتب برنامجا بلغة بايثون يمكنك من:

1. إدخال عدد صحيح وحفظه في متغير مناسب.
2. إذا كان العدد المدخل:
   a. فردياً يقوم بطباعة الرسالة التالية "العدد فردي."

علماً بأن العدد الفردي هو الذي يكون باقي قسمته على الرقم اثنان يساوي واحد.
لحساب باقي القسمة على 2 تنفذ المعادلة الآتية: result=num%2
ملاحظة: ستكون النتيجة إما 0 ويعني أن العدد زوجي أو 1ويعني أن العدد فردي.

الخريطة التدفقية :



الحل البرمجي :

```
odd.py
1 num =int(input("Enter The Number: "))
2 if num%2==1:
3    print("فردي العدد")
4
```

```
Shell
>>> %Run odd.py
Enter The Number: 5
العدد فردي
```

الجملة الشرطية الكاملة :

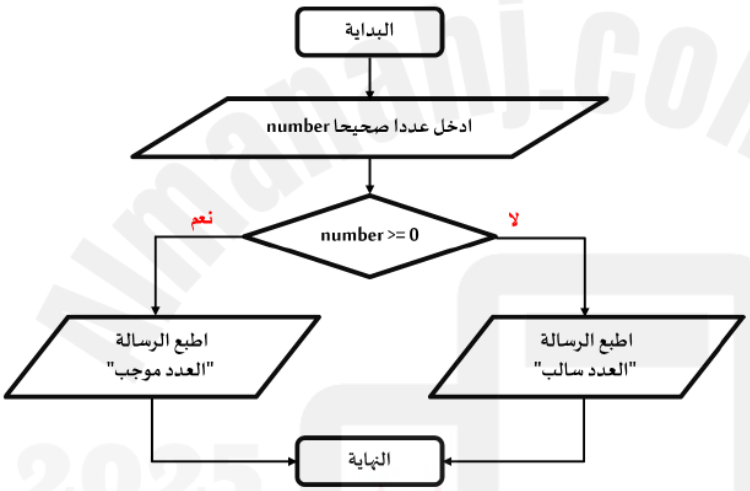
القاعدة العامة للجملة الشرطية الكاملة :

```
if condition(*الشرط) :
    التعليمات        ← يجب ترك مسافه بادئة بعد الشرط
else:
    التعليمات        ← يجب ترك مسافه بادئة بعد الشرط
```

مثال:أرسم الخريطة التدفقية ثم أكتب برنامجاً بلغة بايثون يمكنك من:

1. إدخال عدد صحيح وحفظه في متغير باسم number .
2. إذا كان العدد المدخل أكبر من أو يساوي صفر:
   أ. يقوم بطباعة الرسالة "العدد موجب."
   ب.ما عدا ذلك فيطبع الرسالة "العدد سالب".

الخريطة التدفقية :



الحل البرمجي :

If positive Negtive.py

```
1 number =int(input("Enter The Number:  "))
2 if number>=0:
3     print("موجب العدد")
4 else:
5     print(" سالب العدد")
```

Shell

```
%Run 'If positive Negtive.py'
Enter The Number:  -3
 العدد سالب
```

الجملة الشرطية المركبة :

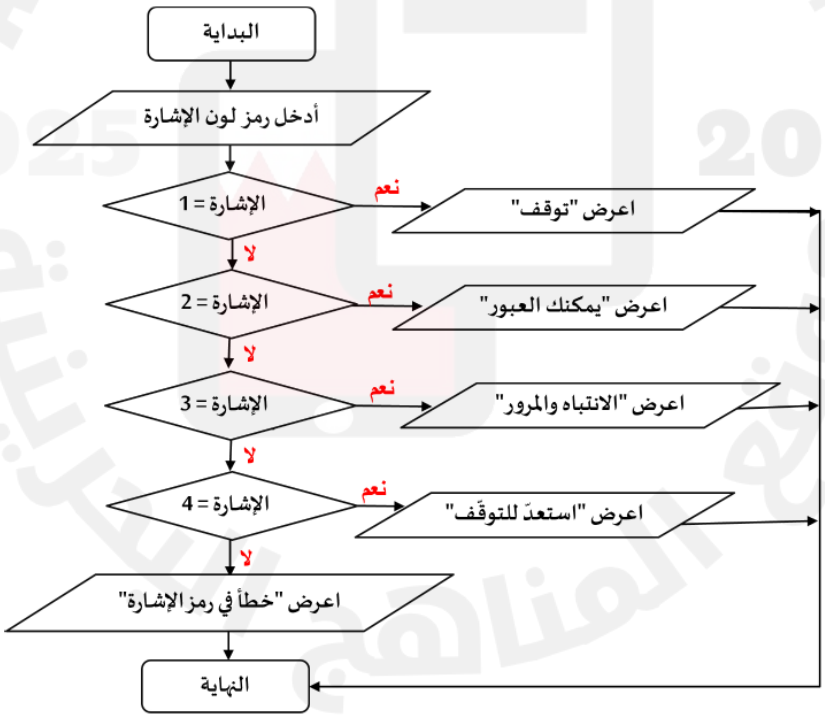
القاعدة العامة للجملة الشرطية المركبة :

if condition(الشرط*) :
    التعليمات ⟷ يجب ترك مسافه بادئة بعد الشرط
elif condition(الشرط*):
    التعليمات ⟷ يجب ترك مسافه بادئة بعد الشرط
    .
    .
else:
    التعليمات ⟷ يجب ترك مسافه بادئة بعد الشرط

مثال:أرسم خريطة تدفقية لبرنامج يقوم بتحديد حالات اشارة المرور، حيث سترمز إليها بالأرقام (1) 1-أحمر|| 2-الأخضر || 3- الأخضر المتقطع || 4- البرتقالي

- الأحمر (1) اعرض الرسالة توقف.
- الأخضر (2) اعرض الرسالة: يمكنك العبور.
- الأخضر المتقطع (3) اعرض الرسالة الانتباه والمرور.
- البرتقالي (4) اعرض الرسالة استعد للتوقف.
- فيما عدا ذلك: اعرض الرسالة خطأ في رمز الإشارة

الخريطة التدفقية :



الحل البرمجي :

```
File  Edit  Format  Run  Options  Window  Help
print("اختر الرقم حسب حالة الإشارة الضوئيّة")
print("1-red 2-green 3-blinking green 4-Yellow")
sign=input("أدخل رقما من 1 إلى 4:")

if int(sign)==1:
    print("لا يسمح لك بالعبور")
elif int(sign)==2:
    print("يسمح لك بالعبور")
elif int(sign) == 3:
    print("انتبه واعبر ")
elif int(sign)==4:
    print("خفف السرعة واستعدّ للوقوف")
else:
    print("الرقم الذي أدخلته خاطئ")

```

ملاحظة مهمة جداً:

مكونات الجمل الشرطية : البسيطة if ، الكاملة if,else ، المركبة if,elif,else

يجب اتباع قواعد الجملة الشرطية مراعياَ المسافة و :

النتيجة ستكون واحدة عند تحقق الشرط فقط .

رسم الخريطة التدفقية للجمل الشرطية مهم جداً جداً جداً

الدرس الثامن: القائمة list

القائمة أو المصفوفة أو المتسلسلة:

تستخدم لجمع العناصر بعضها البعض باستخدام رمز [] وتستخدم الفاصلة "," للفصل بين العناصر .. يتم حفظ العناصر في متغير واحد .

من خصائص القوائم:

1. يمكن أن تضم أي عدد من العناصر.
2. بمجرد استخدام القوس المربع [] اثناء تعريف المتغير، يستطيع البرنامج تمييز أن هذا المتغير من نوع قائمة.
3. لكل عنصر من عناصر القائمة موقعا خاصا ويمكن توظيفه لتحديد العنصر واستخدامه في البرمجة.

أنواع القوائم :

```
list1 = [] # قائمة فارغة
list2 = [1,2,3,4,5] #قائمة تحتوي بيانات رقمية
list3 = ["Red","Yellow","Green"] #قائمة تحتوي بيانات نصية
```

موقع العنصر index :

لتحديد موقع العناصر في القوائم يتم استخدام طريقتين وهي كالتالي:

1. من اليسار لليمين : يكون موقع العنصر الأول هو هو 0 و من ثم يتم إضافة 1 على العناصر التالية

مثال: اذا كان list1=["a","b","c","d","e"] فإن مواقع العناصر ستكون كالتالي

العنصر	a	b	c	d	e
الموقع	0	1	2	3	4

print(list1[2])	c

2. من اليمين لليسار: يكون موقع العنصر الأول هو هو -1 و من ثم يتم إضافة 1- على العناصر التالية

- مثال: اذا كان list1=["a","b","c","d","e"] فإن مواقع العناصر ستكون كالتالي

العنصر	a	b	c	d	e
الموقع	-5	-4	-3	-2	-1

print(list1[-4])	b

مثال : اكتب بلغة البايثون برنامج ينفذ التعليمات التالية:

1. **يعرف قائمة باسم names ويحفظ فيها القيم التالية :**

Fatima	sara	amal	noora

2. **يقوم بطباعة العبارة "الاسم الثاني هو" متبوعة بالعنصر الثاني.**
3. **يطبع العنصر الأول الموجود في القائمة .**
4. **يطبع العنصر الأخير الموجود في القائمة .**
5. **يطبع القائمة.**

```
names=["noora","amal","sara","fatima"]
print("الاسم الثاني هو",names[1])
print(names[0])
print(names[-1])
print(names)
```

ملاحظة مهمة جداً:

موقع العناصر يبدأ من 0 مثلاً العنصر الأول موقعة صفر ، العنصر الثاني موقعة 1 وهكذا .. لتحديد عنصر محدد في القائمة يكتب كالتالي: [الموقع]اسم القائمة .

عند طباعة القائمة يجب أن تتضمن الطباعة رمز [] مع القيم الموجودة.

الدرس التاسع : الدوال المضمنة

أنواع الدوال المضمنة :

1. دوال مع نقطة
2. دوال بدون نقطة
3. دوال الاختيار العشوائي
4. دوال عرض التاريخ و الوقت الحالي

دوال مع نقطة:

طريقة استخدام الدوال المضمنة مع نقطة :

المتغير.الدالة()

الدوال المضمة مع نقطة :

الدالة	الوصف	الدالة	الوصف
append(العنصر المراد إضافته)	تستخدم الدالة لإضافة عنصر في القائمة	index(العنصر المراد تحديد موقعة)	تستخدم لتحديد موقع العنصر
remove(العنصر المراد حذفة)	تستخدم لحذف عنصر من القائمة بناءً على قيمته	pop(موقع العنصر المراد حذفة)	تستخدم لحذف عنصر من القائمة بناءً على موقعة
spilt()	تستخدم لتقسيم النص الى كلمات منفردة وتوضعة في قائمة		

مثال:اكتب بلغة البايثون برنامج ينفذ التعليمات التالية:

1. يعرف قائمة باسم names ويحفظ فيها القيم التالية :

sameera	noora ali	zain	latifa

2. يطبع العنصر الثاني من القائمة.
3. يدرج العنصر"mariam"آخر القائمة .
4. يحذف العنصر الأخير بناء على موقعة .
5. يحذف العنصر الأول بناء على قيمته.
6. يطبع موقع العنصر "zain" الموجود في القائمة .
7. يقسم النص الموجود في العنصر الثالث الى كلمات منفردة .

```
names=["latifa","zain","noora ali","sameera"]
print(names[1])
names.append("mariam")
names.pop() or names.pop(-1) or names.pop(3)
names.remove("latifa")
names.index("zain")
x=names[2].spilte()
print(x)    or print(names[2].spilte())
```

إعداد:أ.لطيفة راشد
مديرة المدرسة: منال سنان

دوال بدون نقطة:

طريقة استخدام الدوال المضمنة بدون نقطة:

(المتغير)الدالة

الدوال المضمة مع نقطة :

الدالة	الوصف	الدالة	الوصف
max(المتغير)	تستخدم لإيجاد العدد الأكبر الموجود ضمن القائمة أو لإيجاد أكبر حرف z	sorted(اسم القائمة)	تستخدم لترتيب العناصر من الأصغر الى الأكبر، تضع العناصر في قائمة
min(المتغير)	تستخدم لإيجاد العدد الأصغر الموجود ضمن القائمة أو لإيجاد أصغر حرف A	sorted(اسم القائمة, reverse=True)	تستخدم لترتيب العناصر من الأكبر الى الأصغر، تضع العناصر في قائمة
len(المتغير)	تستخدم لإيجاد عدد العناصر الموجودة ضمن القائمة ، أو لإيجاد حجم النص (عدد الحروف)	sum(المتغير)	تستخدم لإيجاد مجموع القيم أو العناصر الموجودة ضمن القائمة .
round(المتغير)	تستخدم لجبر القيمة الى اقرب رقم (من 0.1 الى 0.5 تأخذ القيمة الأصغر ، من 0.6 الى 0.9 تأخذ القيمة الأكبر)	abs(المتغير)	تستخدم لتحويل القيمة السالبة الى عدد موجب فقط .
str(المتغير)	تستخدم لتحويل القيمة الرقمية الى قيمة نصية	float(المتغير)	تستخدم لتحويل القيم الى عدد عشري
int(المتغير)	تستخدم لتحويل القيم الى عدد صحيح	del(المتغير)	تستخدم لحذف المتغير

مثال:اكتب بلغة البايثون برنامج ينفذ التعليمات التالية:

1. يعرف قائمة باسم num ويحفظ فيها القيم التالية :

2.2	-12	10	7

2. يطبع العدد الأكبر من القائمة
3. يطبع العدد الأصغر من القائمة
4. يطبع مجموع العناصر الموجودة في القائمة
5. يطبع ترتيب عناصر القائمة ترتيباً تنازلياً (من الأكبر الى الأصغر)
6. يطبع ترتيب عناصر القائمة ترتيباً تصاعدياً (من الأصغر الى الأكبر)
7. يطبع عدد العناصر الموجودة في القائمة
8. يطبع اقرب قيمة للعنصر الرابع .
9. يحول العنصر الثالث الى قيمة مطلقة ويطبع قيمته.

```
num=[7,10,-12,2.2]
print(max(num))
print(min(num))
print(sum(num)
print(sorted(num,reverse=True))
print(sorted(num))
print(len(num))
print(round(num[3]))
print(abs(num[2]))
```

دوال الاختيار العشوائي :

دالة الاختيار العشوائي :

- **تستخدم الدالة لإيجاد عدد عشوائي.**
- **يتم تعريف متغير لتخزين القيمة المختارة.**
- **يقوم البرنامج باختيار عدد عشوائي ضمن مجال محدد .**

دوال الاختيار العشوائي:

الدالة	الوصف
import random random.random()	ارجاع قيمة عشرية بين 0 و 1 مثال : 0.5
import random random.randint(start,end)	ارجاع عدد صحيح من القيمة الأولى حتى النهاية
import random random.randrange(end) random.randrange(start,end) random.randrange(start,end, step)	ارجاع عدد صحيح من 0 حتى (نهاية العدد -1) ارجاع عدد صحيح من البداية حتى (نهاية العدد -1) ارجاع عدد صحيح من البداية حتى (نهاية العدد -1) مع ترك خطوة حسب قيمة (step)

مثال:

1. **استخدم الدالة المناسبة لاختيار عدد عشري عشوائي**
2. **استخدم الدالة المناسبة لاختيار عدد صحيح عشوائي بين 14 و 24**
3. **استخدم الدالة المناسبة لاختيار عدد صحيح عشوائي بين 100 و 200 وتكون خطواته 50**

```
Import random
x=random.random()
#تختار عدد عشري عشوائي مثل: x=0.6
y=random.randint(14,24) or y=random.randrange(14,25)
# تختار عدد صحيح عشوائي مثل: y=14
z=random.randrange(100,201,50)
# تختار عدد صحيح عشوائي مثل: z=100 or z=150
```

دوال عرض التاريخ و الوقت الحالي:

استخدامات دوال عرض التاريخ و الوقت الحالي :

- لعرض التاريخ والوقت الحالي المحلي
- يمكن أسناد تاريخ محدد وعرضه
- يمكن عرض اليوم ، الشهر، السنة كلا على حده

دوال عرض التاريخ و الوقت الحالي :

- date
- datetime

يجب استدعاء الدالة
from datetime import .....
تعريف متغير لتخزين قيمة .....

الدالة	الوظيفة
date.today()	عرض تاريخ اليوم فقط
datetime.now()	عرض الوقت والتاريخ الحالي
..... (y,m,d)	اسناد تاريخ محدد
.year	عرض السنة
.month	عرض الشهر
.day	عرض اليوم
.strftime("%A")	عرض اسم اليوم
.strftime("%B")	عرض اسم الشهر

مثال: نشاط :اكتب بلغة البايثون برنامجاً يكمنني من الآتي:

- تعين متغير من نوع تاريخ وقيمتة 2022,12,31 .
- استخدم الدوال المناسبة لعرض الآتي:
  1. التاريخ
  2. السنة
  3. الشهر
  4. اليوم
  5. اسم الشهر
  6. اسم الشهر

```
from datetime import date
d=date(2022,12,31)
print(d)
print(d.year)
print(d.month)
print(d.day)
print(d.strftime("%A"))
print(d.strftime("%B"))
```

الدرس العاشر : الجملة التكرارية

أنوع الجمل التكرارية المطلوبة في مقرر تقن106 :

1. for loop : تستخدم عند وجود عدد محدود من التكرار فقط.
2. while loop : تستخدم عند وجود عدد محدود و غير محدود من التكرار .

القاعدة العامة للجملة التكرارية for loop:

متغير
for counter in range() or list or string :
التعليمات

يتم تحديد مجال واحد

يجب ترك مسافة بادئة

القاعدة العامة للبنية التكرارية in range():

for counter in range() :
التعليمات

دالة range() تتعامل مع الأرقام دون احتساب الرقم الأخير، تبدأ من 0 إذ لم يتم تحديد البداية.
حالات الدالة :

1. range(end)
2. range(start,end)
3. range(start,end,steps)

counter :عبارة عن أي متغير وعند استخدامه in range تعتبر قيمته عدد.

مثال1:

باستخدام الجملة التكرارية for اكتبي برنامج يقوم بطباعة عبارة "welcome" 7 مرات.

```
for i in range(7):
    print("welcome")
```

مثال2:

صمم برنامجا يطلب من المستخدم إدخال رقما صحيحا ويقوم

البرنامج بطباعة أول 5 مضاعف من مضاعفات هذا الرقم.

```
x=int(input("Enter a number"))
for i in range(1,6):
    print(x*i)
```

مثال3:

لدينا المصفوفة التالية لتخزين درجات الطلاب
student_grade=[]

باستخدام الجملة التكرارية for ادخلي درجات اربع طالبات الى القائمة student_grade ثم اطبعي المصفوفة .

```
student_grade=[]
for i in range(8):
    x=float(input("Enter grade:"))
    student_grade.append(i)
print(student_grade)
```

مثال 4: اكتب برنامجاً يطبع الأرقام الزوجية من 0 الى 20.

```
for i in range(0,21,2):
    print(i)
```

القاعدة العامة للبنية التكرارية in list:

for counter in list :

التعليمات

in list : يتم تحديد اسم القائمة بدلاً من كلمة **list**

counter : عبارة عن أي متغير وعند استخدامه **in list** تعتبر قيمته عنصر من عناصر القائمة، تبدأ حلقة التكرار من العنصر الأول حتى الوصول الى العنصر الأخير.

مثال1:

لدينا المصفوفة التالية باسم **prices** والتي تحتوي على أسعار أجهزة الهاتف

prices=[89.5,360,120,430,220]

باستخدام الجملة التكرارية **for** قومي بكتابة برنامج يغيير كل سعر موجود بالقائمة بزيادة قدرها **20** ثم يعرض المحتوى النهائي للمصفوفة.

```
prices=[89.5,360,120,430,220]
list1=[]
for i in prices:
    i=i+20
    list1.append(i)
print(list1)
```

مثال 2 :

لدينا مصفوفة تحتوي على أسماء :

names=["amal","Amna","noora","areej","manal"]

باستخدام الجملة التكرارية **for** :

1. اطبعي أسماء الطالبات التي تبدأ بحرف "a" أو "A".
2. اطبعي الحرف الأخير من كل اسم في المصفوفة .

```
names=["amal","Amna","noora","areej","manal"]
for i in names:                    #1
    if i[0]=="A" or i[0]=="a":
        print(i)
for i in names:                    #2
    print(i[-1])
```

القاعدة العامة للبنية التكرارية in string:

for counter in string :

التعليمات

in string : يتم تحديد اسم المتغير وهو دائماً من نوع **string**

counter : عبارة عن أي متغير وعند استخدامه **in string** تعتبر قيمته حروف، تبدأ حلقة التكرار من الحرف الأول حتى الوصول الى الحرف الأخير *ملاحظة يتم احتساب المسافة.

مثال1: باستخدام الجملة التكرارية **for**، اطبعي الحروف الموجودة في النص التالي كلا على حدة:

x="nice to see you"

```
x="nice to see you"
for i in x:
    print(i)
```

مثال2:باستخدام الجملة التكرارية **for**، اطبعي عدد تكرار الحرف **o** الموجودة في النص التالي: **x="nice to see you"**

```
x="nice to see you"
c=0
for i in x:
    if i=="o":
        c=c+1
print(c)
```

جميع الأسئلة يمكن حلها باستخدام دالة range:

حل اخر :

مثال:

لدينا مصفوفة تحتوي على أسماء :

```
names=["amal","Amna","noora","areej","manal"]
```

باستخدام الجملة التكرارية for :

1. اطبعي أسماء الطالبات التي تبدأ بحرف "a" أو "A".
2. اطبعي الحرف الأخير من كل اسم في المصفوفة .

```
names=["amal","Amna","noora","areej","manal"]
for i in range(len(names)):                    #1
    if names[i][0]=="A" or names[i][0]=="a":
        print(names[i])
for i in range(len(names)):                    #2
    print(names[i][-1])
```

مثال: باستخدام الجملة التكرارية for، اطبعي عدد تكرار الحرف o الموجودة في النص التالي :

x="nice to see you"

```
x="nice to see you"
c=0
for i in range(len(x)):
    if x[i]=="o":
        c=c+1
print(c)
```

- الأمر break

في الجملة التكرارية يمكن إيقاف حلقة التكرار باستخدام الأمر break بناء على شرط محدد if (يخرج من loop وينفذ باقي التعليمات إن وجدت)

مثال :

lst=["A","B","C","D"]

أرغب في عرض محتوى المتسلسلة شرط التوقف عند الحرف C .

```
lst=["A","B","C","D"]
for i in lst:
    if i=="C":
        break
    print(i)
```

- الأمر continue

في الجملة التكرارية يمكن مواصلة حلقة التكرار(تجاوز عنصر محدد) باستخدام الأمر continue بناء على شرط محدد if (يتخطى التعليمات ويواصل حلقة التكرار)

مثال :

lst=["A","B","C","D"]

أرغب في عرض محتوى المتسلسلة بحيث يتم تجاوز الحرف C .

```
lst=["A","B","C","D"]
for i in lst:
    if i=="C":
        continue
    print(i)
```

القاعدة العامة للجملة التكرارية while loop:

القاعدة العامة للبنية التكرارية while :

يتوقف البرنامج في حال عند تحقق

```
while condition :
    التعليمات
```

يجب ترك مسافة بادئة

مثال1:صمم برنامج يطلب من المستخدم إدخال رقماً صحيحاً ويتوقف عند إدخال الرقم 10 .

```
x=0
while x!=10:
    x=int(input("Enter number"))
```

مثال 2: اكتب برنامجاً بلغة البايثون يستقبل مجموعة من الكلمات ويدرجها ضمن قائمة ، يتوقف البرنامج عند إدخال كلمة "finish"، ثم قم بطباعة القائمة .

```
x=""
list1=[]
while x!="finish":
    x=input("Enter word")
    if x=="finish":
        break
    list1.append(x)
print(list1)
```

القاعدة العامة للبنية التكرارية while مع else: يتوقف البرنامج في حال عند تحقق الشرط

```
while condition :
    التعليمات
else:
    التعليمات
```

يجب ترك مسافة بادئة

يجب ترك مسافة بادئة

يقرأ التعليمات في حال عندم تحقق الشرط

مثال:صمم برنامج يطلب من المستخدم إدخال مجموعة من أسماء العملاء وعند إدخال النص stopسيتوقف البرنامج ويعرض رسالة "لقد أكتفينا بهذا القدر".

```
x=""
while x!="stop":
    x=input("Enter customer name:")
else:
    print("لقد أكتفينا بهذا القدر")
```

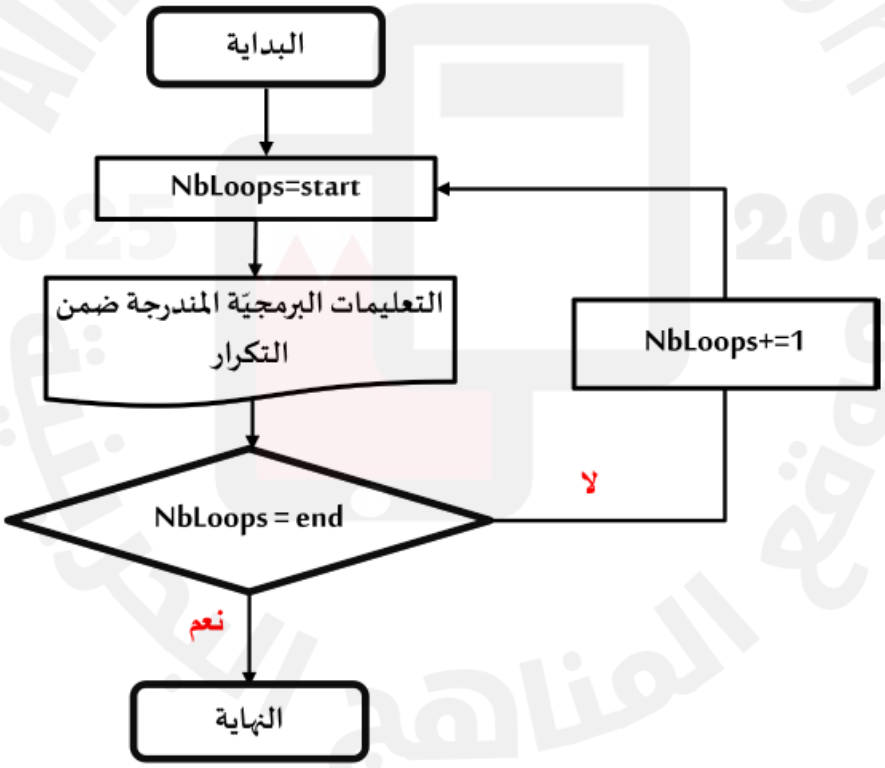
مثال :عند وجود تكرار محدود باستخدام الجملة التكرارية while loop :

صمم برنامج يقوم بطباعة 5 أرقام عشوائية تقع بين 10 و30.

```
import random
x=0
while x!=5:
    y=random.randint(10,30)
    print(y)
    x=x+1
```

ملاحظة: رسم الخريطة التدفقية للجمل التكرارية مهم جداً جداً جداً

رسم الخريطة التدفّقيّة للجملة التكرارية for *var* in range([start],end,[step]) :

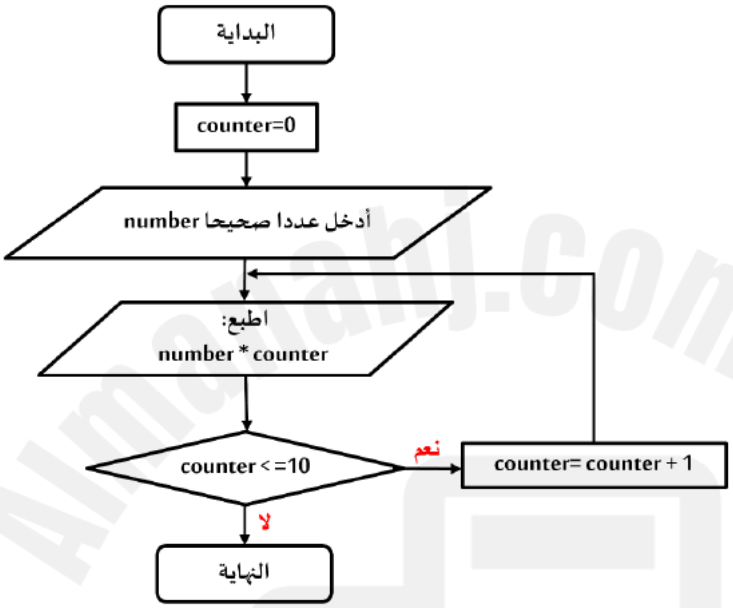


مثال1:

أرغب في كتابة برنامج بلغة بايثون يعرض لي جدول الضرب لأي عدد أدخله للحاسوب.

1. **أعلم أن جدول الضرب يكون من 0 إلى 10 أي 11 تكرارا لعملية الضرب**
2. **اعرض جدول الضرب بحيث يظهر قيمتي الضارب والمضروب. مثال: 35 = 5 * 7**

الخريطة التدفقية :



الحل البرمجي:

```
1 n=int(input("Show the muliple table of the number: "))
2 for i in range(1,11):
3     print(n,"*",i,"=",n * i)
```

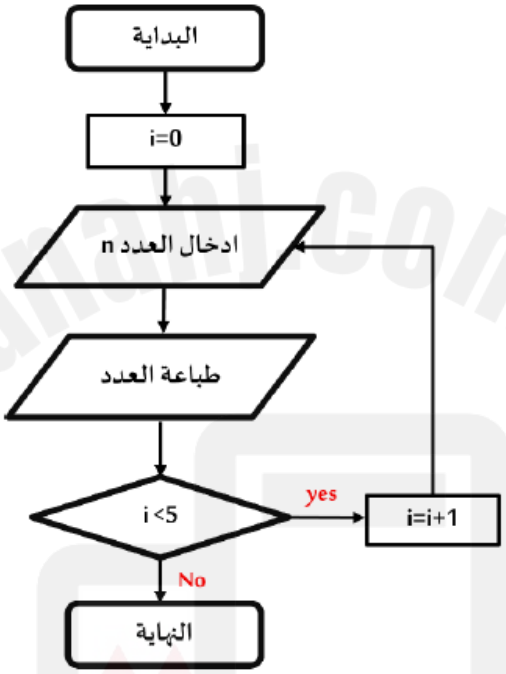
```
>>> %Run -c $EDITOR_CONTENT
Show the muliple table of the number: 6
6 * 1 = 6
6 * 2 = 12
6 * 3 = 18
6 * 4 = 24
6 * 5 = 30
6 * 6 = 36
6 * 7 = 42
6 * 8 = 48
6 * 9 = 54
6 * 10 = 60
```

مثال2:

اكتب برنامجاً بلغة البايثون يقوم بالتالي:

1. **ادخال أرقام صحيحة وطباعتها.**
2. **يتوقف البرنامج عن ادخال الأرقام، عند ادخال الرقم 0**

الخريطة التدفقية :



الحل البرمجي:

```
n=1
while n != 0:
    n=int(input("Number= "))
    print(n)
```

الدرس الحادي عشر : بناء الدوال

بناء الدوال : هي دالة يتم بناؤها لإداء مهمة معينة في البرمجة .

القاعدة العامة لبناء دالة من غير إرجاع:

العوامل : هي المتغيرات المطوبة لإداء البرمجة أن وجدت

```
def اسم الدالة(العوامل) :
    التعليمات
    print(....)
```

يجب ترك مسافة بادئة

القاعدة العامة لبناء دالة مع إرجاع قيمة:

العوامل : هي المتغيرات المطوبة لإداء البرمجة أن وجدت

```
def اسم الدالة(العوامل) :
    التعليمات
    return var
```

يجب ترك مسافة بادئة

var:هي القيمة المرجعة

مثال1:

اكتب بلغة البايثون برنامج ينفذ التعليمات التالية :

1. طباعة عبارة (welcome to my page) عن طريق توظيف دالة show()
2. إرجاع عبارة (welcome to my page) عن طريق توظيف دالة show()

```
def show():                    #1
    print("welcome to my page")
البرنامج الرئيسي#
show() توظيف الدالة في البرنامج الرئيسي#
```

```
def show():                    #2
    return("welcome to my page")
البرنامج الرئيسي#
x=show()#توظيف الدالة في البرنامج الرئيسي
print(x)  #or  print(show())
```

مثال2 :

اكتب بلغة البايثون برنامج ينفذ التعليمات التالية :

1. يعرف الدالة show (x,y) حيث x وy هما اعداد مدخلة من قبل المستخدم .
2. تقوم الدالة بطباعة حاصل طرح العدد x من y .
3. يتم إدخال الأرقام الصحيحة في البرنامج الرئيسي.
4. يتم توظيف الدالة في البرنامج الرئيسي لإظهار الناتج.

```
def show(x,y):
    print(x-y)
البرنامج الرئيسي#
a=int(input("Enter first number"))
b=int(input("Enter second number"))
show(a,b) توظيف الدالة في البرنامج الرئيسي#
```

مثال3:
اكتب بلغة البايثون برنامج ينفذ التعليمات التالية :

1. **تعبئة القائمة من خلال دالة يتم بناؤها لهذا الغرض.**
2. **اسم الدالة randlst(c,lst) وتتضمن متغييرين، عدد العناصر في القائمة c، والقائمةlst .**
3. **تكرر الطالبة عملية تعبئة القائمة lst بأرقام عشوائية تقع ما ٥ و ٦٠ بحسب عدد العناصر المحددة.**
4. **تقوم الدالة بإرجاع القائمة lst.**
5. **عدد عناصر القائم يكون ما بين 3 و 5 حسب اختيار المستخدم في البرنامج الرئيسي.**
6. **يتم عرض القائمة lst2 بعد تنفيذ الدالة في البرنامج الرئيسي.**

```
def randlst(c,lst):
    import random
    for i in range(c):
        x=random.randint(5,60)
        lst.append(x)
    return lst
#البرنامج الرئيسي
a=int(input("ادخل عدد بين 3 و 5"))
lst2=[]
print(randlst(a,lst2))#توظيف الدالة في البرنامج الرئيسي
```

أسئلة outputs

اقرأ الكود ونفذة ذهنياً ثم أكتب نواتج المخرجات في العمود Output :

Python code	Output
a="hello" b="noora" print(a," \t",b)	hello noora
x=10 print(x,"BD")	10BD
x=2 z=12 print(z-x) print(z+x) print(z*x) print(z**x) print(z*"L") print(z/x) print(z//x) print(z%x)	6 10 16 64 LLLLLLLL 4.0 4 0
c=3 u=7 print(c>u and u>c) print(c+u>c and c!=u) print(not(u<c) or c-u==-4) print(not(c<u or u>c)) print(c+4==u)	False True True False True

x=9
if x==1:
  print(x+10)
else:
  print(x-10)
	-1

C="pink"
if C=="red":
  print("nice color")
elif C=="yellow":
  print("perfect color")
elif C=="pink":
  print("loved \n color")
	Loved
	color

lst=[10,6,8,4] print(lst) print(lst[3]) print(lst[-3])	[10, 6, 8, 4] 6 4

Python code	Output
nameLs=["ameera","fatima","hajer"] nameLs.append("Latifa") print(nameLs) nameLs.pop() print(nameLs) nameLs.remove("fatima") print(len(nameLs))	['ameera', 'fatima', 'hajer', 'Latifa'] ['ameera', 'fatima', 'hajer'] 2
z="great jop" print(z.index("j")) print(z.split())	6 ['great', 'jop']
Lst=[2,8,10,4,6,0] print("max=",max(Lst)) print("min=",min(Lst)) print("sum=",sum(Lst)) print("len=",len(Lst)) print("list1=",sorted(Lst)) print("list2=",sorted(Lst,reverse=True))	max= 10 min= 0 sum= 30 len= 6 list1= [0, 2, 4, 6, 8, 10] list2= [10, 8, 6, 4, 2, 0]
import random x=random.random() y=random.randint(1,5) z=random.randrange(2,7,3) print(x,"",y,"",z)	0.86 3 2
import datetime d=datetime.datetime(2024,12,22) print(d.year) print(d.day) print(d.month) print(d)	2024 22 12 2024-12-22 00:00:00
for i in range(2,10,5): print(i)	2 7
Lst=[1,2,3,4] for i in Lst: print(i+5)	6 7 8 9
b="evening" for i in b: print(i+"1")	e1 v1 e1 n1 i1 n1 g1

Python code	Output

st=[11,12,13,14,15,16]
for i in range(len(st)):
  if st[i]%3==0:
  print(st[i])
	12
	15

st=[11,12,13,14,15,16]
for i in range(len(st)):
  if st[i]%3==0:
  st.remove(st[i])
  print(st)
	[11, 13, 14, 15, 16]
	[11, 13, 14, 16]

Ls=[20,21,22,23,24,25]
for i in range(len(Ls)):
  if Ls[i]%2==0:
  Ls.remove(Ls[i])
  print(Ls)
	[21, 22, 23, 24, 25]
	[21, 23, 24, 25]
	[21, 23, 25]

num=[9,7,10,2,1]
for i in num:
  if i==10:
  break
  print(i)
	9
	7

n="dala"
for i in n:
  if i=="a":
  continue
  print(i)
	d
	l

t=[30,15,5,20,25,10]
for i in t:
  if i==5:
  break
  print(i)
print(sum(t))
	30
	15
	105

j=["salwa","hanof","smar","mariam"]
for i in j:
  if i[0]=="s":
  continue
  print(i)
	hanof
	mariam

**Lst=[2,5,4]
i=0
while i<len(Lst):
  print(Lst[i])
  i=i+1
else:
  print("Thankyou")**
	2
	5
	4
	Thankyou

Python code	Output

Lst=[4,2,1,4,6]
i=0
while i!=len(Lst):
  if Lst[i]==4:
  Lst.pop(i)
  i=i+1
print(Lst)
	[2, 1, 6]

import random
i=2
while i!=5:
  x=random.random()
  print(x)
  i=i+1
	0.82
	0.20
	0.44

x=5
while x!=7:
  print(x)
  x=x+1
print(x+2)
	5
	6
	9

t=0
i=1
while i!=10:
  t=t+i
  i=i+1
print(t)
	45

a="I Love Math Classes"
i=0
lst=[]
while i!=len(a):
  if a[i]=="s":
  lst.append(a[i])
  i=i+1
print(len(lst))
	3

n=[70,20,55,40,10]
e=-1
while e!=len(n):
  e=e+1
  if n[e]%4==0:
  continue
  print(n[e])
	70
	55
	10

Python code	Output

num=[5,10,15]
i=-1
while i!=3:
  i=i+1
  if num[i]==10:
  break
  print(num[i])
	5

def s(x,y): return x/y z,r=2,7 print(s(r,z))	3.5

def nList(Lst):
  for i in Lst:
  if i=="green":
  continue
  print(i)
colors=["black","yellow","green","red","pink"]
nList(colors)
	black
	yellow
	red
	pink

def Ls(Lst):
  for i in Lst:
  if i%2==0:
  print(i)
mlst=[2,3,4,5,6]
Ls(mlst)
	2
	4
	6

z="zain" v="Ahmed" print(z+"\n",v)	zain Ahmed
b="keep going" print(len(b))	10
h=["Amal","Noora"] print(h[1][-3])	o

"التعب يزول و الإنجاز يبقى"

جميل أن نتحدى الوقت ونعلم يقيناً أن الدقيقة تسجل لنا إنجازاً عظيماً

مع خالص تحياتي تمنياتي لكم بالتوفيق والنجاح..

معلمتكم المحبة: أ.لطيفة راشد

إعداد:أ.لطيفة راشد **مديرة المدرسة: منال سنان**