



**G11**

المسار التكنولوجي

# علوم الحاسوب

الفصل الدراسي الأول 2022-2023

**الوحدة الأولى : الحواسيب المصغرة والdarات الإلكترونية**

**1- أوامر التكرار البرمجية**

## الدرس الأول: أوامر التكرار البرمجية

### جملة If وجملة IF else الشرطيتان

- إذا كان الشرط صحيحاً ← فإن الجملة الشرطية التي تتبع الشرط ← سيتم تنفيذها
- إذا كان الشرط غير صحيح ← فإن الجملة الشرطية الثانية ← سيتم تنفيذها

```
x=25
if x<20:
    print(x, " is lower than 20")
else:
    print(x, " is greater than 20")
```

```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093 [MSC v.1914 32 bit (Intel)]) on win32
Type "copyright", "credits" or "license()" for more information.
>>>
=====
RESTART: C:/python/examples.py =====
25 is greater than 20
```

### جمل التكرار For

تستخدم جمل التكرار For عند الحاجة إلى تكرار تنفيذ مقطع برمجي لعدد محدد من المرات معروفة مسبقاً يتم تحديد عدد مرات التكرار في range parameters (نطاق العد)

```
# Prints out the numbers 2,3,4,5
for i in range(2,6):
    print(i)
```

```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093 [MSC v.1914 32 bit (Intel)]) on win32
Type "copyright", "credits" or "license()" for more information.
>>>
=====
RESTART: C:/python/examples.py =====
2
3
4
5
```

## أوامر التكرار Loops

- في بعض الأحيان نحتاج إلى تنفيذ مجموعة أوامر برمجية عدة مرات داخل البرنامج مما يستغرق وقتاً وجهداً كبيرين
- تقدم لغة بايثون مجموعة جمل للتحكم البرمجي تسمى بأوامر التكرار Loop لتجنب إعادة كتابة الأوامر البرمجية مرة أخرى
- تسمح جمل التكرار بتنفيذ جملة برمجية واحدة أو مجموعة جمل برمجية عدة مرات

تعد لغة برمجة بايثون هذه الأنواع من الجمل التكرارية.

<code>for loop_variable in range: statements</code>	تكرار for
<code>while condition: statements</code>	تكرار while

## تكرار While

- يستخدم التكرار While عندما لا يكون عدد مرات التكرار معروفة سلفاً.
- يستمر التكرار في العمل طالما كان شرط التكرار صحيحاً.
- تستخدم المسافة البادئة في بداية الجمل البرمجية التي يحتويها مقطع التكرار تحت الأمر While وذلك لتمييزها أثناء قراءة المقطع.

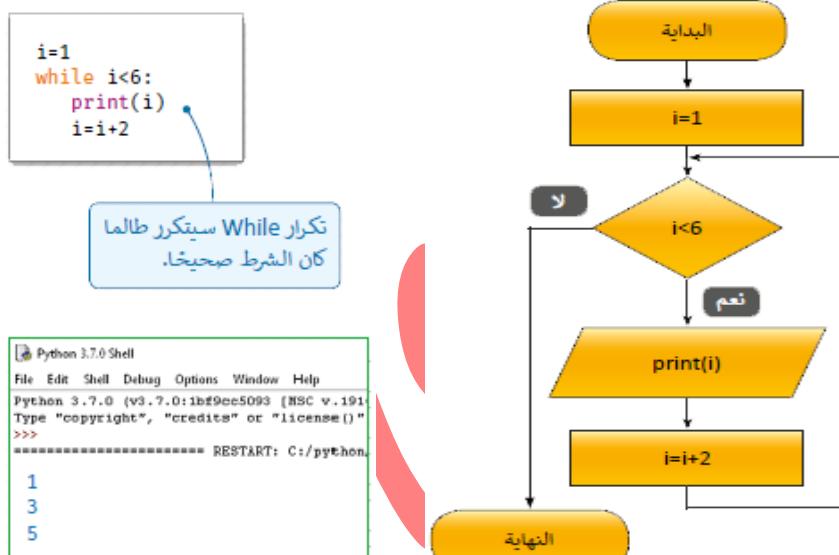
## تابع تنفيذ أوامر تكرار While

- 1- التحقق من الشرط والذي قد يكون صحيحاً أو خطأ
- 2- إذا كان الشرط خطأ يتم الخروج من تكرار While مع متابعة تنفيذ الجملة البرمجية التالية
- 3- إذا كان الشرط صحيحاً يتم تنفيذ جميع الجمل البرمجية الموجودة داخل مقطع التكرار ثم الرجوع إلى خط

## مثال 1

في المثال التالي الأرقام الفردية من صفر إلى ستة باستخدام جملة التكرار While والتي تتطلب الآتي:

- أن يتم تعين متغير كعداد (ولتكن i)
- أن يتم ضبط قيمة المتغير الأولية (ولتكن 1 في مثاناً الحالي)
- أن يتم تنفيذ الجمل البرمجية داخل جملة التكرار طالما كانت نتائجة الشرط صحيحة



اكتب الكود البرمجي السابق مع حذف السطر الرابع ثمنفذ البرنامج.

```
i=1
while i<6:
    print(i)
```

اكتُب الكود البرمجي التالي: ثم لاحظ ماذا سيحدث

- إذا لم تتم عملية الزيادة فإن التكرار سيعمل بدون توقف

حيث أن شرط التكرار While سيكون صحيحاً دائماً

- التكرار الذي يتم تنفيذه بدون توقف سيكون مزعجاً للغاية ولإيقافه يجب ضغط مفاتحي Python shell من نافذة Ctrl+C

## مثال 2

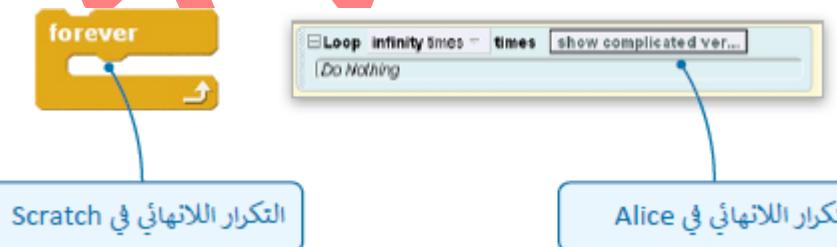
نفذ المقطع البرمجي الآتي:

```
while True:
    word=input("Type a word")
    print("Don't type never again this word: ", word)
```

```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093 [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> -----
RESTART: C:/python/examples.py -----
Type a word:this
Don't type never again this word: this
Type a word:that
Don't type never again this word: that
Type a word:stop
```

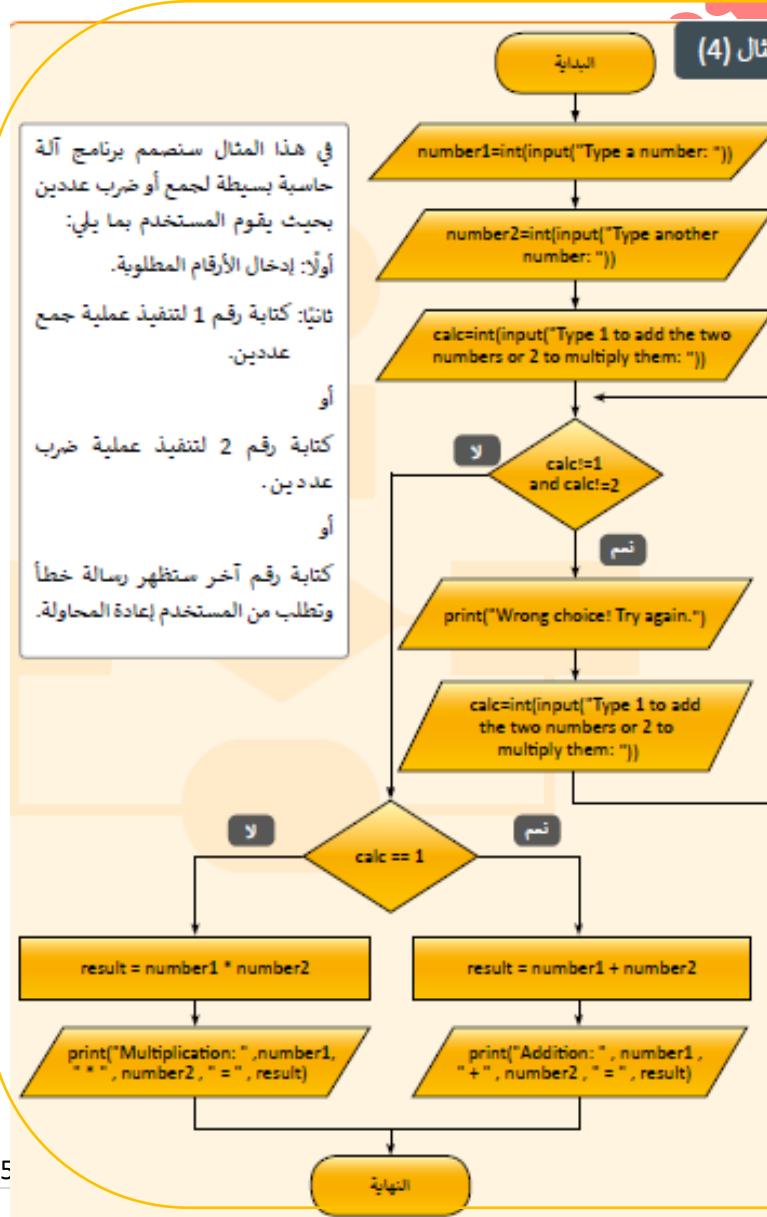
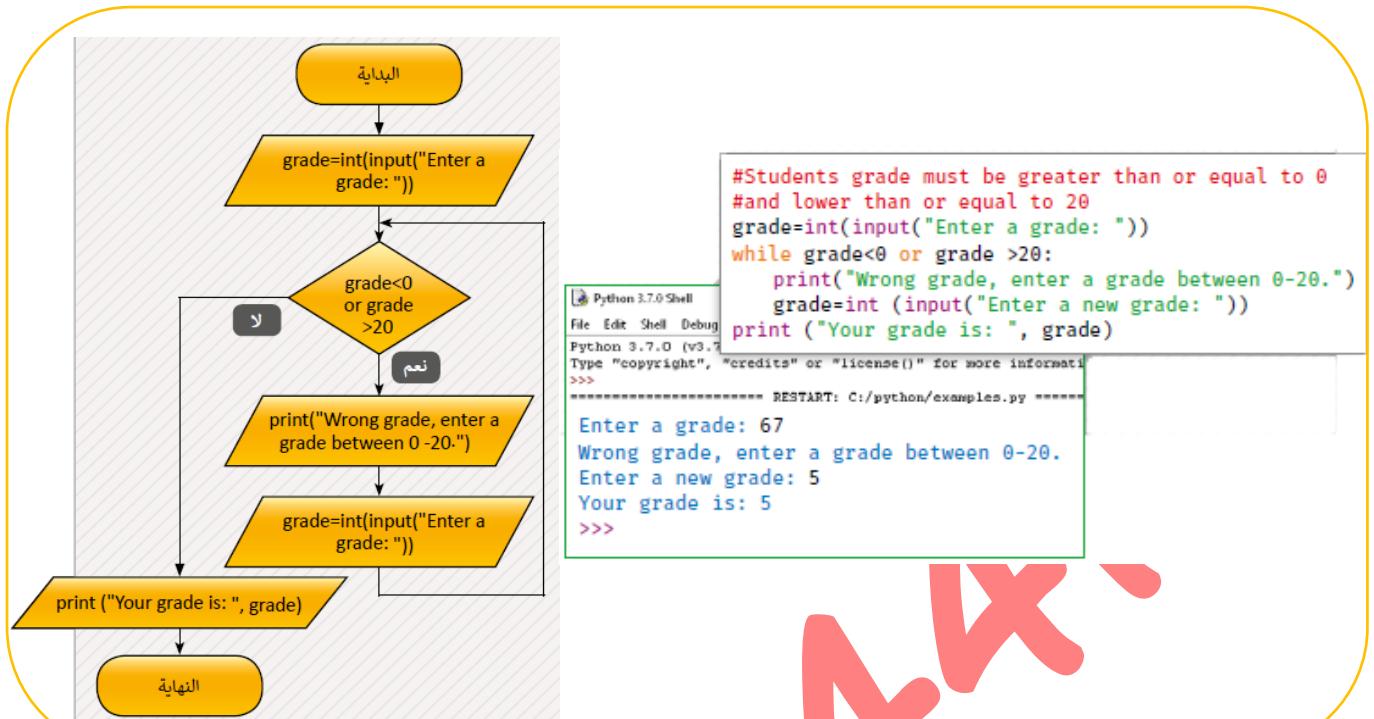
## Infinite loop

يحدث التكرار اللانهائي عندما تكون حالة الشرط صحيحة في البرمجة القائمة على البناء مثل **forever** يتم تمثيل التكرار اللانهائي ببلنة **Alice** و **Scratch**



## مثال 2

يطلب البرنامج من المستخدم إدخال درجة الطالب **grade** ويجب أن تكون هذه الدرجة أكبر من أو تساوي 0 وأقل من أو تساوي 20 إذا أدخل المستخدم قيمة خارج هذا النطاق فإن البرنامج سيعرض رسالة خطأ ويطلب من المستخدم إدخال درجة جديدة



```

number1=int(input("Type a number: "))
number2=int(input("Type another number: "))
calc=int(input("Type 1 to add the two numbers or 2 to multiply them: "))
while calc!=1 and calc!=2:
    print("Wrong choice! Try again.")
    calc=int(input("Type 1 to add the two numbers or 2 to multiply them: "))
if calc == 1:
    result = number1 + number2
    print("Addition: " , number1 , " + " , number2 , " = " , result)
else:
    result = number1 * number2
    print("Multiplication: " , number1 , " * " , number2 , " = " , result)

```

التحقق من الأخطاء

الخطأ في برنامج  
الحاسوب

### 1- الأخطاء اللغوية Syntax Errors

يتضمن الأخطاء في كتابة الأوامر والمتغيرات وهذا النوع من السهل اكتشافه وتصحيحه

- استخدام أمر التكرار While لإدخال كلمة مرور صحيحة

```

passw=input ("Enter the password")
while passw != "G10" •
    passw=input ("Enter the password: ")

```

خطأ لغوي  
عدم كتابة علامة : بعد الشرط.

## - أخطاء أثناء التنفيذ أو - Semantic Errors

يحدث عادة أثناء تصميم البرنامج مثل قيام المبرمج بعمل ترتيب غير صحيح للأوامر البرمجية أو عمليات حسابية أو منطقية خاطئة عند تنفيذ هذا البرنامج فإنه سيعمل ولكن سيعطي نتائج غير صحيحة

- يحسب عدد المحاولات لإدخال كلمة مرور صحيحة

خطأ منطقي  
يجب رفع قيمة i بمقدار 1

```
i=1
passw=input ("Enter the password")
while passw != "G10":
    passw=input ("Enter the password : ")
    i=i+3
print (" عدد المحاولات " , i)
```

## - الأخطاء المنطقية Logical Errors

يشير خطأ وقت التشغيل Runtime error إلى خطأ يحدث أثناء تنفيذ البرنامج على سبيل المثال إذا حاولنا القسمة على صفر

- يحسب متوسط درجة الطالب

في المرة الأولى التي ندخل فيها -1 كدرجة نحصل على رسالة الخطأ التالية.

```
Python 3.7.0 Shell
File Edit Shell Debug Options Window
Python 3.7.0 (v3.7.0:1bf9cc509f, Oct 12 2018, 14:35:35)
Type "copyright", "credits" or "license()" for more information.
>>>
```

```
sumGrade=0
counter=0
grade=int(input("Enter a grade: "))
while grade>0:
    sumGrade=sumGrade+grade
    counter=counter+1
    grade=int(input("Enter a grade"))
avgGrade=sumGrade/counter
print("Your average grade is: ",avgGrade)
```

## التدريبات

### - التحقق من الأخطاء :

1- في أي الحالات يفضل استخدام كل من :

- جملة التكرار While :

- جملة التكرار For :

2- اختر الإجابة الصحيحة :

<pre>i=1 while i&lt;6:     i=i+1     if i == 3:         print("Hello")     print(i)</pre>	<p>6 مرات</p> <p>حتى تصبح قيمة <math>i=3</math></p> <p>5 مرات</p>
<pre>i=5 while i&gt;1:     print(i)     i=i-1</pre>	<p>سيتم تنفيذه 5 مرات</p> <p>عدد مرات التكرار غير معروف</p> <p>سيتم تنفيذه 4 مرات</p>
<p>يتم التتحقق من الشرط ..... على الأقل ..... قبل تنفيذ جملة التكرار While</p>	<p>مرة واحدة</p> <p>مرتان</p> <p>ثلاث مرات</p>
<p>تنفذ الأوامر في تكرار While ..... على ..... الأقل</p>	<p>مرة واحدة</p> <p>مرتان</p> <p>أكثر من مرة</p>
<p>يتم تنفيذ تكرار While طالما كان الشرط ..... الموجود</p>	<p>صحيح</p> <p>خطأ</p>

## - ما ناتج تنفيذ كل من الجمل البرمجية التالية:

.1  
`for i in range(5,1,-1):  
 print (i*i)`

.2  
`for k in range(5,5):  
 print (k)`

.3  
`i=6  
while i!=2:  
 print (i+2)  
 print ("====")  
 i=i-1`

.4  
`sum=0  
i=1  
while i<=6:  
 sum=sum+i  
 i=i+2  
print(sum)`

## - حول المقاطع البرمجية التالية باستخدام جملة التكرار While

.1  
`for i in range(5):  
 print (i)`

.2  
`for i in range(10,2,2):  
 print (i)`

.3  
`for i in range(3,9):  
 print (i+2)  
print (100)`

- اكتب برنامجاً باستخدام جملة التكرار While يطلب من المستخدم إدخال عدد محصور بين 1 و 18 لطباعة جملة I love Qatar | بحيث يعود البرنامج ويطلب إدخال العدد مرة أخرى إذا تم إدخال عدد من خارج النطاق (1-18)

- اكتب برنامجاً يطلب إدخال كلمة المرور الصحيحة وبحد أقصى 5 محاولات

\* يطلب البرنامج إدخال كلمة مرور رقمية

\* إذا تمت كتابة الرقم الصحيح فإن البرنامج يطبع "كلمة مرور صحيحة" وإنما يطبع "حاول مرة أخرى!"

\* بعد المحاولة الخامسة يتوقف ويطبع "تجاوزت عدد المحاولات"

- اكتب برنامجاً يطبع الأعداد المحصورة بين 1 إلى 10 باستثناء العدد 5 باستخدام جملة التكرار

مع تمنياتي بدوام النجاح والتفوق