

ملخص شامل لمقرر البرمجة بلغة Python (تقن 108)



تم تحميل هذا الملف من موقع مناهج مملكة البحرين

موقع المناهج ← مناهج مملكة البحرين ← الصف الأول الثانوي ← علوم وتقانة ← الفصل الثاني ← ملفات متنوعة ← الملف

تاريخ إضافة الملف على موقع المناهج: 11:27:09 2026-05-06

ملفات اكتب للمعلم اكتب للطالب | اختبارات الكترونية | اختبارات | حلول | عروض بوربوينت | أوراق عمل
منهج انجليزي | ملخصات وتقارير | مذكرات وبنوك | الامتحان النهائي | للمدرس

المزيد من مادة
علوم وتقانة:

إعداد: شادية محمود الأحمد وفاطمة عبد الله خليفة

التواصل الاجتماعي بحسب الصف الأول الثانوي



صفحة مناهج مملكة
البحرين على
فيسبوك

الرياضيات

اللغة الانجليزية

اللغة العربية

التربية الاسلامية

المواد على تلغرام

المزيد من الملفات بحسب الصف الأول الثانوي والمادة علوم وتقانة في الفصل الثاني

مراجعة و حل مذكرة تقن 107 للفصل الأول

1

نموذج إجابات امتحانات نهاية الفصل الثاني

2

أسئلة امتحانات سابقة

3

كتاب تقن 102

4

إجابة امتحانات نهاية الفصل الثاني

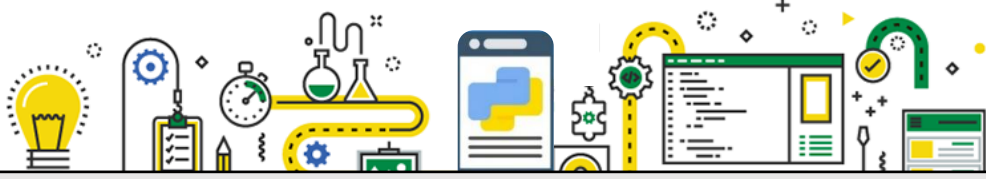
5



ماذا تعلمت؟!

مقرر البرمجة بلغة Python (تقن 108)

الفصل الثاني من العام الدراسي 2026/2025م



بيانات الطالبة

اسم الطالبة:	الشعبة:	
الرقم الأكاديمي:	اسم المعلمة:	
البريد الإلكتروني الرسمي (Moe):		
عزيزتي الطالبة: الرجاء الاحتفاظ بالبريد الإلكتروني وكلمة المرور لاستخدامه في مختبر الحاسوب أثناء الحصة.		
الرؤية والرسالة والاتفاقية	أخلاقيات التعامل مع أدوات الذكاء الاصطناعي للطلبة	نبذة عن المقرر



فضاء التميز



قسم الحاسب الآلي

ماذا أنجزت؟!

مقرر البرمجة بلغة Python (تقن 108)

الفصل الثاني من العام الدراسي 2025/2026م

التقييم الأول: مكتبات بايثون			
مريض ☐	جيد ☐	جيد جداً ☐	ممتاز ☐
التطوير		التعزيز	
الوصف		الوصف	
☐ أرجعي للكتاب أو البوابة التعليمية لتصحيح الأخطاء ☐ ماذا لو زاد انتباهك للدرس بالتدريب ننجز أفضل ☐ ☐ أطلبي المساعدة		☐ متشوقة لرؤية المزيد من إبداعك ☐ سعدت بمحاولتك ☐ لن تتعلمي إلا بالمحاولة ☐ كالعادة متميزة ☐ شكرًا لك ☐ فخورة بك	
☐ إجابتك نموذجية ☐ حاولي لتنجزي ☐ راجعي إجابتك		☐ عملك متقن ☐ محاولتك مقدره ☐	
التقييم الثاني: الجملة الشرطية المتعددة			
مريض ☐	جيد ☐	جيد جداً ☐	ممتاز ☐
التطوير		التعزيز	
الوصف		الوصف	
☐ أرجعي للكتاب أو البوابة التعليمية لتصحيح الأخطاء ☐ ماذا لو زاد انتباهك للدرس بالتدريب ننجز أفضل ☐ ☐ أطلبي المساعدة		☐ متشوقة لرؤية المزيد من إبداعك ☐ سعدت بمحاولتك ☐ لن تتعلمي إلا بالمحاولة ☐ كالعادة متميزة ☐ شكرًا لك ☐ فخورة بك	
☐ إجابتك نموذجية ☐ حاولي لتنجزي ☐ راجعي إجابتك		☐ عملك متقن ☐ محاولتك مقدره ☐	
التقييم الثالث: الجملة التكرارية for			
مريض ☐	جيد ☐	جيد جداً ☐	ممتاز ☐
التطوير		التعزيز	
الوصف		الوصف	
☐ أرجعي للكتاب أو البوابة التعليمية لتصحيح الأخطاء ☐ ماذا لو زاد انتباهك للدرس بالتدريب ننجز أفضل ☐ ☐ أطلبي المساعدة		☐ متشوقة لرؤية المزيد من إبداعك ☐ سعدت بمحاولتك ☐ لن تتعلمي إلا بالمحاولة ☐ كالعادة متميزة ☐ شكرًا لك ☐ فخورة بك	
☐ إجابتك نموذجية ☐ حاولي لتنجزي ☐ راجعي إجابتك		☐ عملك متقن ☐ محاولتك مقدره ☐	
التقييم الرابع: الجملة التكرارية while			
مريض ☐	جيد ☐	جيد جداً ☐	ممتاز ☐
التطوير		التعزيز	
الوصف		الوصف	
☐ أرجعي للكتاب أو البوابة التعليمية لتصحيح الأخطاء ☐ ماذا لو زاد انتباهك للدرس بالتدريب ننجز أفضل ☐ ☐ أطلبي المساعدة		☐ متشوقة لرؤية المزيد من إبداعك ☐ سعدت بمحاولتك ☐ لن تتعلمي إلا بالمحاولة ☐ كالعادة متميزة ☐ شكرًا لك ☐ فخورة بك	
☐ إجابتك نموذجية ☐ حاولي لتنجزي ☐ راجعي إجابتك		☐ عملك متقن ☐ محاولتك مقدره ☐	
التقييم الخامس: بناء الدوال			
مريض ☐	جيد ☐	جيد جداً ☐	ممتاز ☐
التطوير		التعزيز	
الوصف		الوصف	
☐ أرجعي للكتاب أو البوابة التعليمية لتصحيح الأخطاء ☐ ماذا لو زاد انتباهك للدرس بالتدريب ننجز أفضل ☐ ☐ أطلبي المساعدة		☐ متشوقة لرؤية المزيد من إبداعك ☐ سعدت بمحاولتك ☐ لن تتعلمي إلا بالمحاولة ☐ كالعادة متميزة ☐ شكرًا لك ☐ فخورة بك	
☐ إجابتك نموذجية ☐ حاولي لتنجزي ☐ راجعي إجابتك		☐ عملك متقن ☐ محاولتك مقدره ☐	



قسم الحاسب الآلي

مقرر البرمجة بلغة Python (تقن 108)
الفصل الثاني من العام الدراسي 2025/2026م

الوحدة الأولى: مكتبات البايثون

مكتبات البايثون			
■ مفهوم المكتبات: هي عبارة عن مجلدات وملفات تحتوي قطعاً برمجية جاهزة مكتوبة ومختبرة من قبل مبرمجين خبراء. ■ مثال: البحث عن أكبر عدد ضمن مصفوفة أو أصغر عدد أو معدل الأعداد، تقوم بإستدعاء المكتبة المناسبة عند الحاجة ضمن برنامجك الرئيسي، مما يغنيك عن كتابة أكواد جديدة من الصفر.			
أمثلة على بعض أهم مكتبات بايثون:			
1. Datetime : للتعامل مع بيانات من نوع تاريخ.			
2. random : للتعامل مع الأعداد العشوائية.			
مكتبة datetime			
■ مكتبة datetime: للتعامل مع المتغيرات التي تحفظ تاريخاً من خلال استدعاء مكتبة مضمنة تسمى datetime. 1. للتعامل مع التاريخ فقط يتم استدعاء الدالة date من مكتبة datetime : from datetime import date 2. للتعامل مع التاريخ والوقت يتم استدعاء الدالة datetime من مكتبة datetime : from datetime import datetime			
التعامل مع دوال التاريخ			
نوع البيانات	الدالة	الغرض	الطريقة
تاريخ date	today()	عرض تاريخ اليوم فقط	from datetime import date print(date.today())
تاريخ date	now()	عرض الوقت والتاريخ الحالي	from datetime import datetime print(datetime.now())
في الأمثلة السابقة، تم طباعة التاريخ مباشرة، لكن يمكننا أن نُسند التاريخ إلى متغير، كالتالي :		اسناد تاريخ محدد لمتغير:	
from datetime import date d = date(2024 , 5 , 23) print(d)		from datetime import date d = date.today() print(d)	
ملاحظة: نوع المتغير d في المثال السابق هو تاريخ date			



قسم الحاسب الآلي

التعامل مع دوال التاريخ

أمثلة	الغرض	الدالة المستخدمة
<pre>from datetime import date print(date.today().year)</pre> مثال التاريخ الحالي هو: 2022 - 11 - 20 النتيجة: 2022	لعرض السنة فقط من التاريخ الحالي	year
<pre>from datetime import date print(date.today().month)</pre> مثال التاريخ الحالي هو: 2022 - 11 - 20 النتيجة: 11	لعرض الشهر فقط من التاريخ الحالي	month
<pre>from datetime import date print(date.today().day)</pre> مثال التاريخ الحالي هو: 2022 - 11 - 20 النتيجة: 20	لعرض اليوم فقط من التاريخ الحالي	day

أمثلة	الغرض	الدالة المستخدمة
<pre>from datetime import date x=date(2022,11,20) print(x.strftime("%A"))</pre> النتيجة: Sunday	لعرض اسم اليوم فقط من التاريخ	strftime("%A")
<pre>from datetime import date x=date(2022,11,20) print(x.strftime("%B"))</pre> النتيجة: November	لعرض اسم الشهر فقط من التاريخ	strftime("%B")

مكتبة random

■ **مكتبة random:** هي مجموعة من الدوال يتم توظيفها لاستخراج قيمة بشكل عشوائي، من خلال استدعاء المكتبة الخاصة بها وتسمى random، لذا يتم كتابة السطر التالي في ملف البايثون:

```
import random
```

ولاحقاً عند استخدام اي دالة تابعة للمكتبة random، يتم ذكر اسم المكتبة يليها اسم الدالة المطلوب استخدامها.



قسم الحاسب الآلي

التعامل مع دوال العد العشوائي

اسم الدالة	الوصف	مثال
random()	اختيار عدد عشوائي بين 0.0 إلى 1.0	ضرورة استدعاء المكتبة random <pre>import random x= random.random() print(x)</pre> النتيجة الممكنة: 0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1
randint()	لاختيار عدد عشوائي صحيح، مع تحديد المجال (البداية والنهاية) مثال: (بين 10 و 15)	<pre>import random x= random.randint(10, 15) print(x)</pre>
	تحديد المجال عن طريق استخدام المتغيرات	<pre>import random s=5 ; e=11 x= random.randint(s, e) print(x)</pre>
	تحديد المجال عن طريق المستخدم وذلك بإدخال قيمتين صحيحتين	<pre>import random s= int(input("start= ")) e= int(input("end= ")) x= random.randint(s, e) print(x)</pre>

اسم الدالة	الوصف	مثال
randrange()	لاختيار عدد عشوائي صحيح ضمن مجال تُحدّد بدايته ونهايته وخطوته. مثال: (بين 5 و 15 خطوة 2)	<pre>import random x= random.randrange(5, 15, 2) print(x)</pre>
	لاختيار عدد عشوائي صحيح ضمن مجال تُحدّد بدايته ونهايته. مثال: (بين 5 و 11)	<pre>import random x= random.randrange(5,11) print(x)</pre>
	لاختيار عدد عشوائي صحيح ضمن مجال محدد بدءاً من الصفر مثال: (بين 0 و 8)	<pre>import random x= random.randrange(8) print(x)</pre>

10 /



قسم الحاسب الآلي

أنشطة صفية 1 – مكثبات بايثون

النشاط الأول: أقرئ البرنامج التالي ونفذه ذهنياً، ثم اكتب الناتج في خانة output:

البرنامج	output
from datetime import date	-
theDate=date(2026, 5, 5)	-
print(theDate)	
print(theDate.year)	
print(theDate.month)	
print(theDate.day)	

النشاط الثاني: أقرئ البرنامج التالي ونفذه ذهنياً، ثم اكتب الناتج في خانة output ، المطلوب كتابة كل القيم المحتملة:

#	Python code	output
1	import random n=8 x= random.randrange(n) print(x)	
2	import random x= random.randrange(5,10,3) print(x)	
3	import random x= random.randint(8,10) print(x)	

النشاط الثالث: أكمل البرنامج التالي الذي يقوم باختيار عدد عشوائي زوجي ضمن المجال (25 و5):

import

R=random._____ (____ ,____ ,____) # >=5 and < 25 اختیار رقم زوجی عشوائی

```
print(          )
```

النشاط الرابع: أكملى البرنامج التالى الذى يقوم بطباعة السنة من تاريخ اليوم الحالى today:

from import

```
print(_____._____()._____)
```



قسم الحاسب الآلي

مقرر البرمجة بلغة Python (تقن 108)

الفصل الثاني من العام الدراسي 2026/2025 م

الوحدة الثانية: الجملة الشرطية متعددة الشروط

الجملة الشرطية متعددة الشروط

■ مفهوم الجملة الشرطية متعددة الشروط:

هي جملة شرطية تتضمن جمل شرطية أخرى لفحص شروط متعددة لاتخاذ القرار المناسب بناءً على تحقق شروط متعددة من عدمه.

■ أنواع الجملة الشرطية: (البسيطة - الكاملة - متعددة الشروط)

بنية الجملة الشرطية المتعددة:

الشرط 1 :
if
إجراء تحقق الشرط 1
الشرط 2 :
elif
إجراء تحقق الشرط 2
else :
إجراء عدم تحقق الشرط 1 و 2

بنية الجملة الشرطية الكاملة:

الشرط :
if
إجراء تحقق الشرط
else :
إجراء عدم تحقق الشرط

بنية الجملة الشرطية البسيطة:

الشرط :
if
إجراء تحقق الشرط

■ تطبيق عملي على الجملة الشرطية الكاملة:

صممي برنامج لإدخال رقماً صحيحاً إلى الحاسوب ثم يعرض الرسالة "هذا الرقم زوجياً" إذا كان زوجياً أو الرسالة "هذا الرقم فردياً" إذا كان فردياً.

```
number=int(input( "Plz Enter an integer : " ))
if number % 2 == 0 :
    print("هذا الرقم زوجي")
else:
    print("هذا الرقم فردي")
```

■ تطبيق عملي على الجملة الشرطية البسيطة:

صممي برنامج يطلب من المستخدم إدخال رقم صحيح وإذا كان الرقم فردي يطبعه على الشاشة.

```
num=int(input( "Plz Enter an integer : " ))
if num% 2 == 1 :
    print(num)
```

■ تطبيق عملي 1 على الجملة الشرطية المتداخلة:

صممي برنامج لحساب سعر تذكرة الدخول إلى حلبة سباق البحرين للفورمولا 1 بناءً على المنطقة بحيث يتم إدخال المنطقة ويعرض سعر التذكرة (على أن يتم إدخال المنطقة في المتغير ar مع عبارة "Plz Enter Area A or B" بحسب الشروط التالية):

1. إذا كانت المنطقة: A - عرض "سعر التذكرة 35 دب"	2. إذا كانت المنطقة: B - عرض "سعر التذكرة 50 دب"	3. وإلا عرض "الرمز الذي أدخلته غير صحيح"
--	--	--

رسالة إدخال المنطقة: # ar=input("Plz Enter Area A or B : ")

```
الشرط 1: إذا كانت المنطقة تساوي #A :
print("سعر التذكرة 35 دب")
الشرط 2: إذا كانت المنطقة تساوي #B :
print("سعر التذكرة 50 دب")
إجراء تحقق الشرط 1: طباعة ("سعر التذكرة 35 دينار) #
إجراء تحقق الشرط 2: طباعة ("سعر التذكرة 50 دينار) #
إجراء عدم تحقق الشرط 1 و 2: طباعة ("الرمز الذي أدخلته غير صحيح) #
else :
print("الرمز الذي أدخلته غير صحيح")
```



قسم الحاسب الآلي

الجملة الشرطية متعددة الشروط

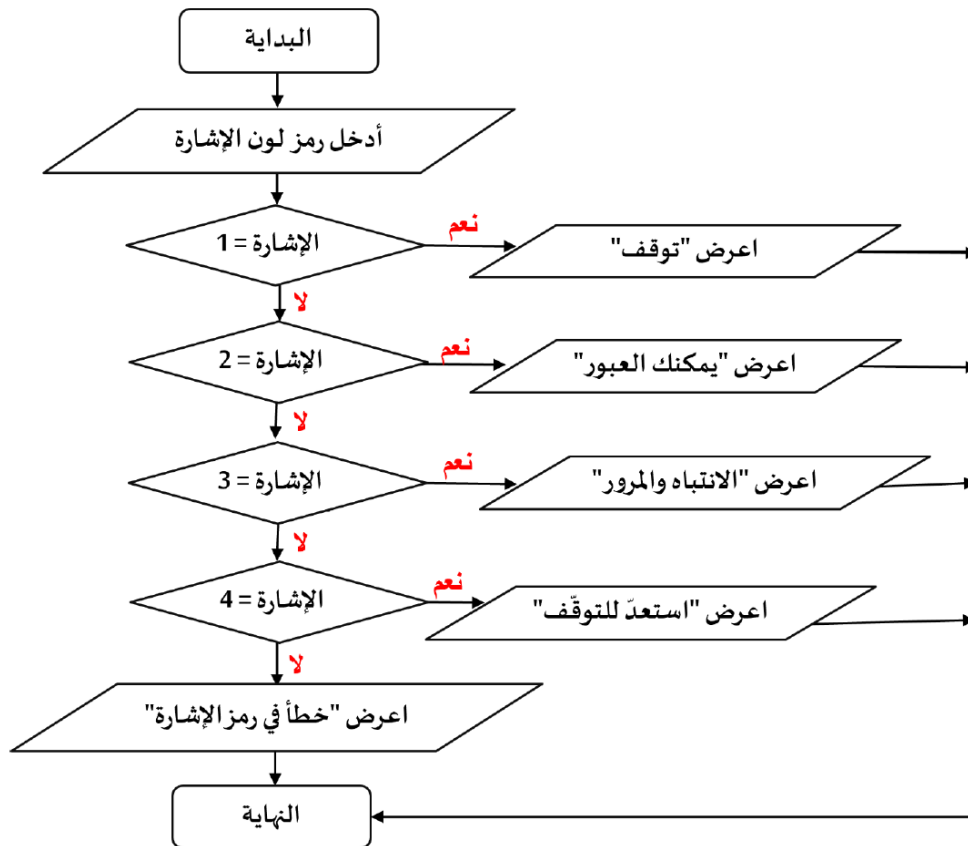
```
print("اختر الرقم حسب حالة الإشارة الضوئية")
print("1-red 2-green 3-blinking green 4-Yellow")
sign=input("أدخل رقماً من 1 إلى 4:")

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 واستعد للوقوف")
else:
    print("الرقم الذي أدخلته خاطئ")
```

■ تطبيق عملي 2 على الجملة الشرطية المتداخلة:

1. اكتب برنامجاً يقوم بتحديد حالات إشارة المرور حيث سنرمز لها بالأرقام (1- أحمر || 2- الأخضر || 3- الأخضر المتقطع || 4- البرتقالي):
 - الأحمر (1): اعرضي الرسالة: "توقف".
 - الأخضر (2): اعرضي الرسالة: "يمكنك العبور".
 - الأخضر المتقطع (3): اعرضي الرسالة: "الانتباه للمرور".
 - البرتقالي (4): اعرضي الرسالة: "استعد للوقوف".
 - فيما عدا ذلك: اعرضي الرسالة: "خطأ في رمز الإشارة".

ارسمي الخريطة التدفقية للبرنامج السابق.





10 /

قسم الحاسب الآلي

أنشطة صفية 2 – الجملة الشرطية المتعددة

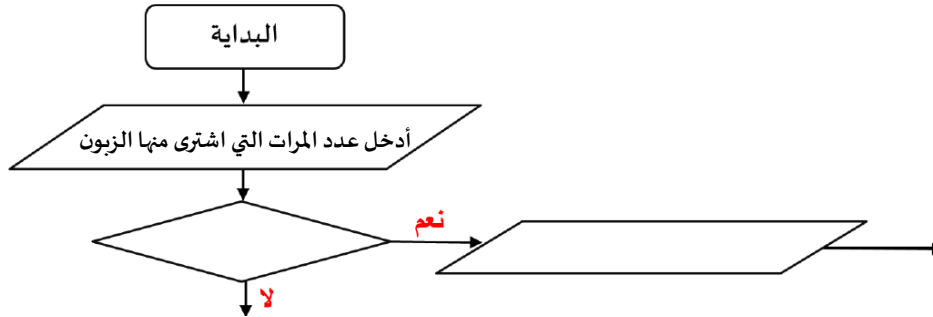
النشاط الأول:

1. اكتب برنامجاً بلغة Python يطلب من المستخدم إدخال عدد المرات `nbr_bay` التي اشترى فيها الزبون من المحل التجاري ثم عرض رسالة

حسب التالي:

- عرض رسالة "زبون وفي للمحل" إذا كان عدد مرات الشراء `nbr_bay` أكبر من أو يساوي 7.
- عرض رسالة "زبون متوسط الوفاء للمحل" إذا كان عدد مرات الشراء `nbr_bay` أكبر من أو يساوي 3.
- عرض رسالة "زبون عادي" إذا كان عدد مرات الشراء `nbr_bay` أقل من 3.

2. أكمل رسم الخريطة التدفقية للبرنامج السابق.





10 /

قسم الحاسب الآلي

أنشطة صفية 3 – الجملة الشرطية المتعددة



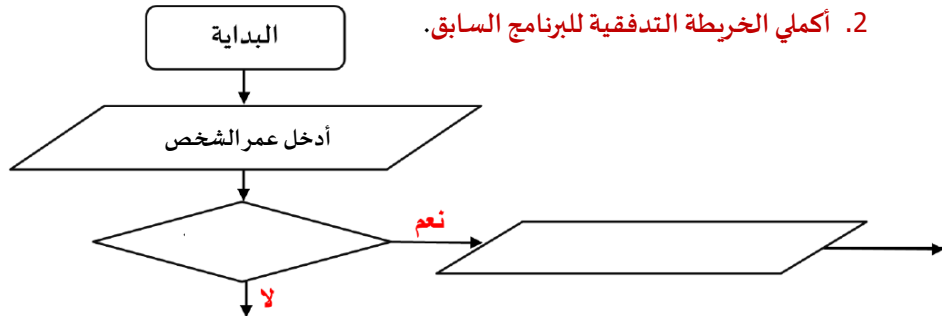
النشاط الثاني:

1. أكمل البرنامج التالي بلغة Python لعرض الوصف المناسب لشخص ما حسب عمره بناء على الشروط الآتية:

```
age= _____ ("ادخل عمر الشخص:")  
_____ < 2:  
    print( " _____ " )  
_____ age < _____:  
    print( " _____ " )  
elif _____ < _____:  
    print( " _____ " )  
_____ age < 30:  
    print( " _____ " )  
_____ age < _____:  
    print("كهل")  
_____  
    print( "شيخ" )
```

الوصف	الشرط
رضيع	أصغر من 2 سنة
طفل	أصغر من 12 سنة
مراهق	أصغر من 18 سنة
شاب	أصغر من 30 سنة
كهل	أصغر من 60 سنة
شيخ	أخرى

2. أكمل الخريطة التدفقية للبرنامج السابق.





قسم الحاسب الآلي

مقرر البرمجة بلغة Python (تقن 108)
الفصل الثاني من العام الدراسي 2025/2026م

الوحدة الثالثة: الجمل التكرارية

الجمل التكرارية for

تطبيق عملي على بنية الجمل التكرارية for:

- اكتب برنامج بلغة Python (استخدم for loop) لعرض الأرقام من 5 إلى 10:

```
for c in range(5,11):
    print(c)
```

- اكتب برنامج بلغة Python (استخدم for loop) لعرض الأرقام الفردية 1 و 3 و 5:

```
for i in range(1,7,2):
    print(i)
```



مفهوم الجمل التكرارية for:

تمكنني الجمل التكرارية for من تكرار التعليمات instructions عدد محدد من المرات حيث يبدأ العداد counter من القيمة start وحتى القيمة end وهي عدد مرات التكرار.

أنواع الجمل التكرارية for (range-list-string):

الجمل التكرارية for باستخدام range:

مثال 1: اكتب برنامجاً يستقبل رقماً صحيحاً ، و اعرضي ناتج جمع الأعداد من 1 الى الرقم الذي تم ادخاله.

```
num=int(input("Enter the number \t")) #المدخل الرقم
total=0
for i in range(1,num+1):
    total=total+i

print("the total SUM = : ",total)
```

الجمل التكرارية for باستخدام القائمة list: تمكنني الجمل التكرارية for من تكرار

التعليمات instructions بحيث يكون مجال التكرار هو القائمة ذاتها.

مثال 2: اكتب برنامجاً يقرأ القيم من القائمة ls ، و اعرضها على الشاشة.

```
1 ls=[2,4,6,8,10]
2 for i in ls:
3     print(i)
```

Shell

```
>>> %Run -c $EDITOR_0
2
4
6
8
10
```

الجمل التكرارية for باستخدام النص string: تمكنني الجمل التكرارية for من تكرار التعليمات instructions بحيث يكون مجال التكرار

يساوي عدد أحرف نص محدد.

مثال 3: اكتب برنامجاً يقوم بإدخال عبارة نصية وحفظها في متغير، ثم إدخال حرف وحفظه في متغير، وتأكد إن كان الحرف الذي أبحث عنه

موجوداً ضمن العبارة التي أدخلتها.

```
text='Welcome to bahrain'
char=input("Enter a letter:")
for i in text:
    if char == i:
        print("The letter ",char, " exists in the text")
```



10 /

قسم الحاسب الآلي

أنشطة صفية 4 – الجملة التكرارية for

النشاط الأول: أقرئي البرنامج ونفذه ذهنياً، ثم اكتب الناتج في خانة output:

#	Python code	output	#	Python code	output
1	for i in range(2,10,2): print(i*2)		4	A=[20,13,8,7,25] for i in A: if i>10: print(i+3)	
2	for i in range(30,0,-5): print(i)		5	for i in "Bahrain": print(i)	
3	Fact=1 for i in range(1,5): Fact= Fact *i print(Fact)		6	Counter=0 Str_var="Bahrain" for c in Str_var: if (c=='a'): Counter=Counter+1 print("The number of char. a is",Counter)	

النشاط الثاني: أكمل البرنامج التالي الذي يقوم بطباعة حروف اسم الطالب المُدخل:

Stn= _____ ("ادخل اسم الطالب")
_____ i in _____:
print(_____)

النشاط الثالث: أكمل البرنامج التالي الذي يقوم بطباعة الأرقام الزوجية من القائمة :

mylist=[3,4,8,7,10]
_____ i in _____:
_____ i _____ 2 _____:
print(_____)



قسم الحاسب الآلي

مقرر البرمجة بلغة Python (تقن 108)
الفصل الثاني من العام الدراسي 2025/2026م

الوحدة الثالثة: الجمل التكرارية

الجمل التكرارية while

■ مفهوم الجمل التكرارية while:

تقوم بتنفيذ الأوامر instructions الموجودة ضمنها طالما الشرط condition متحقق عدّة مرات.

■ تطبيق عملي على بنية الجمل التكرارية while:

اكتب برنامج بلغة Python (استخدم while loop) يقوم بالآتي:

1. يطلب من المستخدم إدخال أعداد.
2. يحسب مجموع الأعداد المدخلة طالما المجموع أقل من أو يساوي 100 ثم يعرض المجموع على الشاشة.

```
total=0
while total<=100 :
    n=int(input("الرجاء إدخال الرقم "))
    total=total+n
print("Total= ",total)
```

■ مثال 1: اكتب برنامجاً بلغة البايثون يقوم بالتالي:

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

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

■ مثال 2: اكتب برنامجاً بلغة البايثون يقوم بالتالي:

1. إدخال عدد صحيح بصورة مكررة.
2. بعد كل إدخال يقوم بحساب المجموع للأعداد المدخلة إلى أن يفوق المجموع القيمة 100 عندها يتوقف البرنامج عن طلب إدخال الأعداد.
3. عرض المجموع النهائي.

```
1 total=0
2 while total < 100:
3     mynum=int(input("Number= "))
4     total=total+mynum
5 print(total)
```



10 /

قسم الحاسب الآلي



أنشطة صفية 5 – الجملة التكرارية while

النشاط الأول: أقرئي البرنامج ونفذه ذهنياً، ثم اکتبي الناتج في خانة outputs:

#	Python code	Outputs				
1	<pre>c=1 while c<=4: print(c/2) c=c+1</pre>	loop	c	print(c/2)		
2	<pre>lis=["Bahrain","Kuwait","Saudia"] t=0 while t < len(lis): #len(lis) = ...? f=lis[t] print(f[0]) t=t+1</pre>	loop	t	f	f[0]	print(f[0])

النشاط الثاني: اكملی البرنامج التالي بلغة البايثون لعرض جدول الضرب للعدد 6 بحسب المخرجات المعطاة.

المخرجات

0*6=0
 1*6=6
 2*6=12
 3*6=18
 4*6=24
 5*6=30
 6*6=36
 7*6=42
 8*6=48
 9*6=54
 10*6=60

i = _____
_____ i _____ :
print("6 x", _____ , "=", _____)
i = _____ + _____



قسم الحاسب الآلي

مقرر البرمجة بلغة Python (تقن 108)
الفصل الثاني من العام الدراسي 2025/2026م

الوحدة الثالثة: الجمل التكرارية

كسر حلقة التكرار - break

■ الأمر break:

يمكنني الأمر break من الخروج من حلقة التكرار بناءً على شرط محدد.

- **مثال:** وظيفي الأمر break لكسر الحلقة التكرارية والخروج من البرنامج عند الوصول للعدد 4 في القائمة.

```
break
nbr=[1,2,3,4,5]
for x in nbr:
    if x == 4:
        break
    else:
        print(x)
```

المخرجات

```
1
2
3
```

تخطي حلقة التكرار - continue

■ الأمر continue:

يمكنني هذا الأمر من تجاوز عنصر محدد ضمن الجملة التكرارية ومواصلة أوامر الحلقة التكرارية.

- **مثال:** وظيفي الأمر continue لتخطي تنفيذ الحلقة التكرارية عند الوصول للعدد 4 في القائمة.

```
continue
nbr=[1,2,3,4,5]
for x in nbr:
    if x == 4:
        continue
    else:
        print(x)
```

المخرجات

```
1
2
3
5
```

■ تطبيق عملي:

اكتبي برنامج بلغة بايثون يقوم بالآتي:

- تعريف قائمة باسم ls تحتوي على الأرقام [4,3,0,2,0,5]
- إدخال رقم صحيح بواسطة المستخدم وحفظه في متغير باسم Num
- تكرار قسمة الرقم Num بواسطة جميع عناصر القائمة مع تخطي القسمة على الرقم صفر لتفادي خطأ division by zero.

```
1 ls=[4,3,0,2,0,5]
2 Num=int(input("Enter the divisor Number"))
3 for i in range(len(ls)):
4     if ls[i]==0:
5         continue
6     else:
7         print(Num/ls[i])
```

Shell

```
>>> %Run continue.py
Enter the divisor Number 12
3.0
4.0|
6.0
2.4
```



10 /

قسم الحاسب الآلي

أنشطة صفية 6 – كسر / تخطي حلقة التكرار

النشاط الأول: أقرئي البرنامج ونفذه ذهنياً، ثم اكتب الناتج في خانة output:

#	البرنامج	output
1	<pre>ls=["Ahmed","Amna","Mohamed","Amel","Ayman"] name="Mohamed" for i in range(0,5): if ls[i]==name: print("found in location",i,"Good bye") break else: print("it is not in location",i)</pre>	
2	<pre>ls=[2,3,0,3,0,5] for i in range(len(ls)): if ls[i]==0: continue else: print(ls[i])</pre>	
3	<pre>x = 20 while x!=0: x = x - 5 if x == 5: break print(x)</pre>	
4	<pre>counter = 1 while counter <= 30: counter = counter * 2 if counter == 8 or counter == 4: continue print(counter)</pre>	



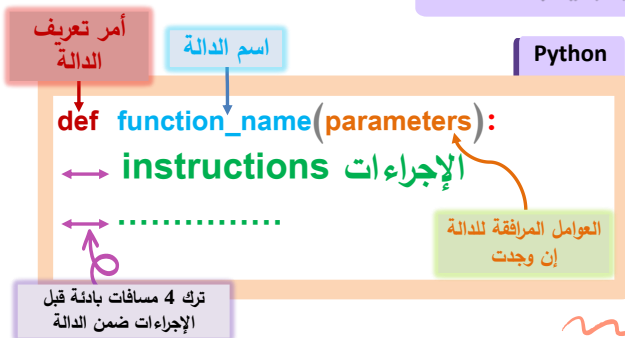
قسم الحاسب الآلي

مقرر البرمجة بلغة Python (تقن 108) الفصل الثاني من العام الدراسي 2025/2026م

الوحدة الرابعة: بناء الدوال

بناء الدوال

تحديد بنية بناء الدالة



■ **مفهوم الدالة:** هي مجموعة من الجمل البرمجية الجاهزة مسبقاً، يتم استدعاؤها عند اللزوم، عادةً ما ترافقها عوامل خاصة بها تسمى "parameters" لاستخدامها للوصول إلى النتيجة المطلوبة. عادةً ما تقوم الدالة بإرجاع النتيجة إلى البرنامج الرئيسي الذي تم استدعاؤها منه.

■ أنواع الدوال وحل المشاكل البرمجية:

1. دالة مع إرجاع نتيجة:

- إنشاء دالة لحساب مجموع عددين.
- عدد عوامل الدالة parameters هو 2: إذاً نحتاج لمتغيرين (a,b) وهو عدد الأعداد التي سأجمعها.
- Return هو نتيجة الدالة التي ستحول إلى الجزء الرئيسي من البرنامج.

```
File Edit Format Run Options Window Help
def sumTwo(a, b):
    s=a+b
    return s
##### نهاية الدالة #####
##### الجزء الرئيسي من البرنامج #####
x=5
y=4
z=sumTwo(x, y)
print(z)
```

منطقة الدالة
منطقة البرنامج الرئيسي

يجب استدعاء الدالة بنفس عدد العوامل parameters التي أنشأت بها وبنفس الترتيب.

2. دالة بدون إرجاع نتيجة:

- إنشاء دالة تقوم بطباعة رسالة فقط ولا أحتاج منها إرجاع أي نتيجة.

```
File Edit Format Run Options Window Help
def msg():
    print("This is a function without parameters.")
```

لكي تنفذ هذه الدالة سأقوم باستدعائها كالآتي:

```
File Edit Format Run Options Window Help
def msg():
    print("This is a function without parameters.")
#call of the function msg()
msg()
```

استدعاء الدالة يأتي بعد تعيينها.

المخرجات

This is a function without parameters.



10 /

قسم الحاسب الآلي

أنشطة صفية 7 – بناء الدوال



النشاط الأول: اكمل البرنامج التالي بلغة البايثون لتنفيذ الآتي:

1. إنشاء دالة باسم CreateList لإدخال عناصر سلسلة list مكوّنة من أعداد صحيحة تُحدّد عددها.
2. إنشاء دالة باسم SumList لحساب مجموع عناصر السلسلة .
3. إظهار النتيجة لعرض السلسلة ثم عرض مجموع عناصر السلسلة في البرنامج الرئيسي.

دالة لإنشاء قائمة وإدخال عناصرها

```
_____( ):
chars = []
n = _____ ("أدخلي عدد العناصر")

for i in range(_____):
    value = int(input("أدخلي عدد صحيح: "))
    _____._____ (value)

_____ chars
```

دالة لحساب مجموع عناصر القائمة

```
def _____ (_____):
    total = 0

    for i in range(len(_____)):
        total += chars[i]

    return total
```

البرنامج الرئيسي

```
chars = _____ () # استدعاء الدالة لإنشاء القائمة
print("عناصر السلسلة:", _____)

result = _____ (chars) # استدعاء حساب المجموع
print("مجموع عناصر السلسلة:", _____)
```



10 /

قسم الحاسب الآلي

أنشطة صفية 8 – بناء الدوال



النشاط الثاني: أكمل البرنامج التالي بلغة البايثون لتعبئة قائمة باسم chars بمجموعة من الحروف، متبعةً الآتي:

1. تعبئة القائمة من خلال دالة يتم بناؤها لهذا الغرض.
2. اسم الدالة Fill(c,chars) ، وتحتوي على متغيرين: عدد عناصر القائمة c ، والقائمة char.
3. الدالة تقوم بإرجاع (return) القائمة chars.
4. اسم القائمة في البرنامج الرئيس هو FillChars .
5. عدد عناصر القائمة يكون من 5 إلى 10 عناصر ويتم تحديدها عشوائياً في متغير باسم nb ضمن البرنامج الرئيس.
6. عرض القائمة FillChars بعد تعبئتها

import random

تعريف الدالة Fill

```
_____ Fill( _____ , _____ ):  
    for i in range( _____ ):  
        ch = input("أدخلي حرف:")  
        _____ (ch)  
        _____ chars
```

البرنامج الرئيسي

```
nb = random. _____ ( _____ , 10) # تحديد عدد عناصر عشوائي من 5 إلى 10  
FillChars = [] # إنشاء القائمة في البرنامج الرئيسي  
  
FillChars = Fill( _____ , _____ ) # استدعاء الدالة لتعبئة القائمة  
  
print("عدد العناصر العشوائي هو", _____)  
print("FillChars محتويات القائمة:", _____)
```

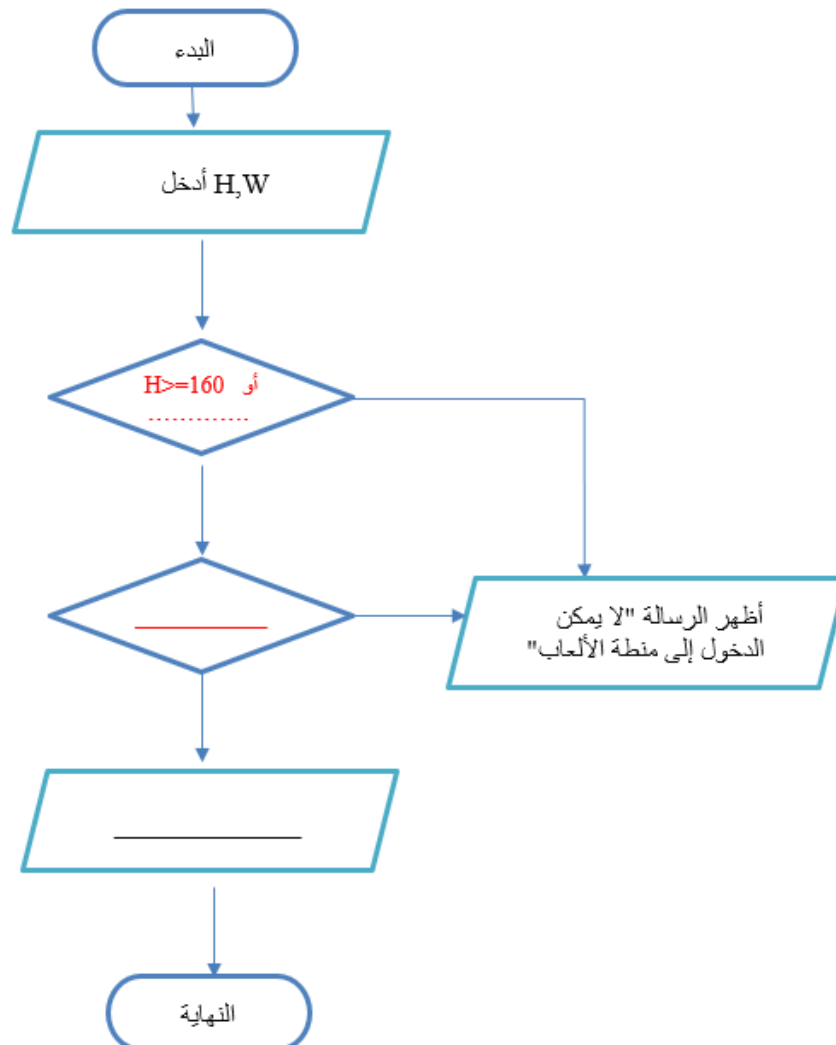


المهمة

السؤال الأول:

6

1. أكمل الإجراءات الناقصة في الخريطة التدفقية للتخطيط لحل مسألة برمجية تمكّنك من الدخول لإحدى مناطق الألعاب، على أن تتوفّر فيك المواصفات الآتية:
 - الطول (H) أكبر من أو يساوي 160 سم أو وزنك (W) أكبر من 65 كغ.
 - أخذت جميع التطعيمات (C) المضادة لفيروس كورونا.
2. على الخريطة التدفقية، اكتب الشرط الصحيح في الشكل المناسب ثم بيّن المسار "نعم" والمسار "لا" بناءً على الشرط وأكمل بالشكل المناسب لإظهار الرسالة.





قسم الحاسب الآلي

السؤال الثاني:

6

أقري الكود ونفذه ذهنيًا ثم أكتبي نواتج المخرجات في العمود Output:

#	Python Code	Output
1	<pre>from datetime import date Today = date.today() print("Today's date:", Today)</pre>	
2	<pre>from datetime import date d = date(2025,11,5) dtxt=d.strftime("%B") print(dtxt)</pre>	
3	<pre>import random x=random.randint(6,8)</pre>	
4	<pre>import random ran=random.randrange(22,30,3) print(ran)</pre>	
5	<pre>b=40 while b>=20: b=b-6 if b==22: continue print(b)</pre>	
6	<pre>def FloatPart(a,b): c=(a/b) - (a//b) return c x,y = 8 , 12 #(x=8 & y=12) F=FloatPart(y,x) Print(F)</pre>	



قسم الحاسب الآلي

السؤال الثالث:

4

أكمل الفراغ لتنفيذ الهدف المطلوب من البرنامج.

```
import _____  
  
rnd=random._____(_____,_____)  
  
if rnd == 15:  
  
    print("هذا أكبر عدد عشوائي يمكنك الحصول عليه")  
  
elif rnd == 10:  
  
    print("هذا أصغر عدد عشوائي يمكنك الحصول عليه")  
  
    _____:  
  
    print("هذا العدد ضمن المجال العشوائي")
```

السؤال الرابع:

4

أكمل البرنامج التالي بلغة البايثون لتنفيذ الآتي:

دالة لحساب كتلة الجسم

```
_____ CBMI(_____, _____):  
    _____ = _____ / (_____ * _____)  
    _____ BodyMass
```

البرنامج الرئيسي

```
height = _____ (input("أدخل لي الطول بالمتر " ))  
weight = _____ (input("أدخل لي الوزن " ))  
bmi = _____ (_____, _____)  
_____ ("كتلة الجسم", _____)
```

1. إنشاء دالة لحساب كتلة الجسم حسب الآتي:

- سأحتاج إلى متغير الطول H بالمتر.
- سأحتاج إلى الوزن W بالكيلو.
- معادلة حساب كتلة الجسم BodyMass.

$$\text{BodyMass} = W / (H * H)$$

2. طباعة الكتلة بعد حسابها.