الوحدة 1 | الدرس 2

الجمل الشرطية

وصف الدرس

الغرض العام من هذا الدرس هو أن يتعلم الطلبة كيفية استخدام الجمل الشرطية والمعاملات المنطقية والشرطية في **Python**.

ما سيتعلمه الطالب

> استخدام الجمل الشرطية لإنشاء حل لبعض المشكلات البرمجية.

> المقارنة بين استخدام عدة طرق مختلفة لتنفيذ نفس الخوارزمية مع تقييم كفاءة وسرعة كل طريقة.

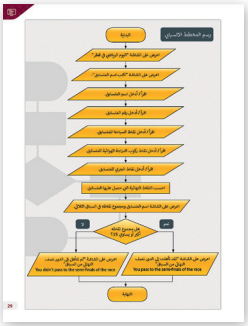
المصطلحات

اللغة العربية	اللغة الإنجليزية
الشرط	Condition
شروط متعددة	Multiple conditions
المعاملات الشرطية	Conditional operators
المعاملات المنطقية	Logical operators

التحديات المتوقعة

> قد يجد الطلبة صعوبة في فهم استخدام بعض المعاملات الشرطية أو المنطقية مثل "**not**" و "**!=**" (لا يساوي). تأكد من أنك تعطي الصف أمثلة كافية لاستخدامها في الجمل الشرطية.

> قد يواجه الطلبة مشاكل في استخدام البنية الصحيحة في **Python** لجمل **If** و **If...else**. أكد على أنه يجب وضع النقطتين دائمًا في نهاية السطر "**if**" أو "**elif**". قد يواجه الطلبة صعوبة أخرى، وهي محاذاة الأوامر تحت وظيفة الشرط. اشرح لهم أنه من أجل توضيح أنه سيتم تنفيذ الأمر عندما يكون الشرط صحيحًا، يجب أن يكون هناك مسافة بادئة قبل الأوامر داخل "**if**" أو "**elif**".



> عند إنشاء المخططات الانسيابية، قد يواجه الطلبة صعوبات في استخدام الأشكال. قد لا يكون من السهل عليهم ترتيب الأشكال لأن المخططات الانسيابية أكثر تقدمًا مع المزيد من الأشكال في هذا الصف. الصعوبة الأخرى التي قد يواجهها الطلبة هي الروابط بين الأشكال. في هذه الحالة، وضح لهم كيفية إنشاء المخطط الانسيابي خطوة بخطوة وكيفية توخي الحذر عند إضافة الأشكال بعد حالات "نعم" أو "لا".

التمهيد

> قدِّم الغرض من الدرس عن طريق تحفيز اهتمام الطلبة باتخاذ القرارات الشرطية بلغة برمجة **Python**.

> يمكنك البدء بطرح أسئلة على الطلبة، مثل:

- في أي حالات يتعين علينا اتخاذ قرارات؟
- هل سبق لك أن اتخذت قرارات في البرمجة؟ إذا كان الجواب نعم، فما هي الأداة التي استخدمتها؟
- هل تتذكر كيف جمعت بين قيمتين من أجل القيام بالإجراءات المناسبة؟

> اسألهم إذا كان بإمكانهم تحديد مشكلة ما، بحيث يكون حلها معتمداً على القرارات المختلفة التي يتعين علينا اتخاذها. أعط مثالاً عن السباحة حسب حالة الطقس ودرجة الحرارة. ما هو القرار الذي يتعين عليهم اتخاذه، وما هي النتائج المختلفة؟

> اعرض اشارة مرور بلونين ثم بثلاثة ألوان (أخضر أحمر أصفر)، واطلب من الطلبة صياغة جمل لوصف كيفية التعامل مع الإضارات الضوئية الثلاث وكيفية اتخاذ القرار المناسب للعبور أو الانتباه أو التوقف. ثم وجه الطلاب لتحديد مكان الشرط وأي قرار سيقع تنفيذه حسب الحالة.

التلميحات الخاصة بالتنفيذ

> ابدأ بتعريف الطلبة على الجمل الشرطية **If** و **If...else**، وأظهر لهم المخططات الانسيابية في كتاب الطالب موضحًا بنية كل عبارة. اشرح اختلافاتها وتأكد من فهم الطلبة لها. اذكر أن الجملة **if ... else** تتيح لنا استخدام أكثر من شرط واحد، مثل استخدام أكثر من حالة **if**. لذلك فنحن نستخدمها عندما نريد اتخاذ قرارات أكثر تعقيدًا.

> أثناء مناقشة الجمل الشرطية المختلفة، أكد على الفرق بين المعاملات الشرطية الأكبر من أو الأصغر من (>، <) مع أكبر من أو يساوي، أصغر من أو يساوي (> = ، <=). قد يكون من المفيد استخدام **Python** لكتابة بعض الأمثلة وتشغيلها.

> عند تحليل عبارات **if** و **If...else** في لغة برمجة **Python**. اطلب من الطلبة فتح بيئة **Python** وقم بمساعدتهم على تطبيق أمثلة كتاب الطالب. هل يمكنهم فهم الفرق بين الجملتين في هذه الأمثلة؟ باستخدام استراتيجية المناقشة، اسأل عن أي صعوبات قد يواجهها الطلبة.

> أثناء تعلم كيفية استخدام شروط متعددة، استخدم المثال الثاني في كتاب الطالب، لتوضيح أنه في بعض الحالات لا نحتاج إلى إجراء تحكم ثالث في العنصر الثالث للحالة الثالثة. على سبيل المثال، عندما نريد التحقق مما إذا كان الرقم إيجابياً أم سلبياً أم صفرًا، فإننا نتحقق من الحالتين وستكون الحالة الباقية هي الحالة الثالثة. باستخدام استراتيجية التجارب الاستقصائية، اطلب منهم العثور على مثال مشابه للكتاب لتقديمه إلى الصف.

> باستخدام المثال التالي في كتاب الطالب (المثال 3)، حث الطلبة على التفكير أكثر مثل المبرمج وإنشاء برامج أقصر وأكثر تقدمًا. على سبيل المثال، اشرح لهم كيف يمكن أن يكون لديهم فحص مزدوج في جمل If و **If...else** آخر باستخدام العوامل المنطقية.

استراتيجيات غلق الدرس

في نهاية الدرس تأكد من تحقيق الطلبة لجميع أهداف الدرس وقيّم معرفتهم من خلال الأسئلة على سبيل المثال لا الحصر:

> هل تستطيع أن تتذكر:

- ما هي العبارات الشرطية التي تعرفها؟ صف وظائفها.
- صف وظيفة معاملات التشغيل المنطقية.

> ذكّر الطلبة بالمصطلحات الهامة وكرّرها معهم.

> قم بتقسيم الطلبة إلى مجموعات، ولتقم كل مجموعة بتقديم حل برمجي بلغة بايثون لمشكلة معينة، ثم اطلب من كل فريق ترشيح أحد الطلاب لشرح البرنامج الذي قاموا بكتابته.

> يمكنك الاستعانة بتدريبات الكتاب ضمن الاستراتيجيات التي ستستخدمها لغلق الدرس.

> جهز مجموعة من المسائل البرمجية وقم بطباعة المخططات الانسيابية الكلية أو الجزئية لكل مسألة من المسائل المقترحة، وقم بقصها. قسم الطلبة إلى مجموعات ثنائية، واطلب منهم ترتيب المخططات الانسيابية بالشكل الصحيح لتصلح كحل للمسائل المقترحة، ثم اطلب منهم ترجمتها لبرامج بايثون.

التدريبات المقترحة لغلق الدرس

يمكنك استخدام هذا التدريب ضمن استراتيجيات غلق الدرس، مع التأكد من فهم الطلبة للمفاهيم الرئيسة التي تعلموها في هذا الدرس.

الصف العاشر | الفصل الدراسي الأول | كتاب الطالب | صفحة 37

الفروق الفردية

تمارين إضافية للطلبة ذوي التحصيل المرتفع

> بعد الانتهاء من التمرين السادس، اطلب من الطلبة إنشاء مخطط انسيابي للبرنامج.

- اطلب من المستخدم إدخال رقمين.
- اطلب من المستخدم إدخال إحدى المعاملات الحسابية(*، /، -، +).
- قم بعمل الحسابات اللازمة بناءً على المعامل الحسابي الذي تم إدخاله.
- كن حذراً عند التعامل مع عمليات القسمة.

الإجابات النموذجية للتدريبات:

1

اختر الإجابة الصحيحة (استعن بحاسوبك للتحقق من الإجابة):

عند تشغيل البرنامج التالي فسوف:

```
Number= 12
if Number > 0 :
   print(“positive number”)
```

لن يعمل البرنامج لأن صيغة الأوامر غير صحيحة.	○
لن يتم عرض أي رسالة على الشاشة لأن الجمل الشرطية غير صحيحة.	○
يتم عرض الرسالة "positive number" على الشاشة.	✓

عند تشغيل البرنامج التالي فسوف:

```
Number = 12
If Number > 0 :
   print(“positive number”)
```

لن يعمل البرنامج لأن صيغة الأوامر غير صحيحة.	✓
لن يتم عرض أي رسالة على الشاشة لأن الجمل الشرطية غير صحيحة.	○
يتم عرض الرسالة "positive number" على الشاشة.	○

2

يقوم البرنامج التالي بعرض القيمة المطلقة للمتغير العددي "Num1".

- أكمل البرنامج ليعمل بشكل صحيح.
- قم بتشغيل البرنامج والتأكد من عمله بشكل صحيح.

```
print(“Give a number”)
Num1=float(input())
if Num1<0:
       print(“The absolute value of ”,Num1,” is ”,-Num1)
else:
       print(“The absolute value of ”,Num1,” is ”,Num1)
```

3 ما ناتج تنفيذ كل من المقاطع البرمجية التالية:

1. S is Not Equal 5

```
s=10
if s==5:
   print("S is Equal 5")
else:
  print("S is Not Equal 5")
```

2. 20

```
x=3/2
if x!=1.5:
   print(10)
else:
  print(20)
```

4 قم بتحويل عبارات "if" التالية إلى برامج:

> إذا كانت السماء تمطر عندها سآخذ مظلة، وإلا سوف أرتدي قبعة.

> إذا كان اليوم هو يوم السبت فسأذهب إلى السباحة، وإلا فسوف أقرأ كتابًا.

```
print("Is it raining? Please answer yes or no.")
answer=input()
if answer=="yes":
   print("I will take an umbrella.")
else:
   print("I will wear a hat.")
```

```
print("What day is today?")
answer=input()
if answer=="Saturday":
   print("I will go for swimming.")
else:
   print("I will read a book.")
```

5

أنشئ برنامجًا للتحقق مما إذا كنت في عُمر يسمح لك بقيادة السيارة. فكر في طريقتين مختلفتين للتحقق من عمرك.

```
print("How old are you?")
answer= float(input())
if answer<18:
   print("You are young to drive get.")
else:
   print("You can drive.")
```

```
print("Which chronology did you born?")
answer= float(input())
age=2021-answer
if age<18:
   print("You are young to drive get.")
else:
   print("You can drive.")
```

6

أنشئ برنامج يتحقق مما إذا كان الرقم يتكون من رقمين أم لا، ويعرض الرسالة المناسبة.

```
print("Enter a number.")
number= float(input())
if 9<number<100:
   print("The number has two digits.")

else:
   print("The number doesn't has two digits.")
```

7 أكتب برنامج بلغة البايثون لإدخال نتيجة طالب من قبل المستخدم، وطباعة كلمة Pass إذا كانت النتيجة أكبر أو يساوي 50 ، وعكس ذلك يطبع كلمة Fail.

```
print("Enter the result: ")
x=int(input())
if x>=50:
    print("Pass")
else:
   print("Fail")
```

8 أكتب برنامج لإدخال درجة حرارة اليوم، وطباعة إحدى الجمل التالية:

> "Fair Weather", اذا كانت درجة الحرارة محصورة بين 15-30.
> "Cold Weather", اذا كانت درجة الحرارة أقل من 15.
> "Hot Weather", إذا كانت درجة الحرارة أكبر من 30.

```
print("Enter the day's temperature: ")
temp=int(input())
if temp>15 and temp<30:
    print("Fair Weather")
elif temp<=15:
    print("Cold Weather")
else:
   print("Hot Weather")
```

9 إنشاء آلة حاسبة

> اطلب من المستخدم إدخال رقمين.

> اطلب من المستخدم إدخال إحدى المعاملات الحسابية(*، /، -، +).

> قم بعمل الحسابات اللازمة بناءً على المعامل الحسابي الذي تم إدخاله.

> كن حذراً عند التعامل مع عمليات القسمة.

```
print("Enter the first number.")
x= float(input())

print("Please type in the math operation you would like
to complete: + for addition, - for subtraction, * for
multiplication, / for division")
operation=input()

print("Enter the second number.")
y= float(input())

if operation=="+":
   print(x,operation, y, "=", x+y)

elif operation=="-":
   print(x,operation, y, "=", x-y)

elif operation=="*":

   print(x,operation, y, "=", x*y)

elif operation=="/":

   print(x,operation, y, "=", x/y)
else:

   print("The operation that entered is not matching")
```

10

قم بكتابة برنامج بلغة بايثون لحساب المبلغ النهائي للمشتريات علما سيتم تخفيض بنسبة 20% إذا كانت قيمة المشتريات أكثر من 1000 ر.ق و 10% إذا كانت قيمة المشتريات بين 500 ر.ق و1000 ر.ق.

لنبدأ بكتابة الكود البرمجي:

> اطلب من المستخدم إدخال اسمه.

> اطلب من المستخدم إدخال قيمة المشتريات.

> احسب المبلغ النهائي للمشتريات بعد التخفيض.

> اعرض على الشاشة رسالة بقيمة المبلغ النهائي للمشتريات بعد التخفيض.

```
print("Enter your name.")
name=input()

print("Enter the value of your purchase.")
purchase=float(input())

if purchase>1000:
   purchase=purchase-purchase*0.2

elif 500<purchase<1000:
   purchase=purchase-purchase*0.1

print("Your final purchase is ",purchase)
```