الوحدة 2 الدرس 4

المكتبات البرمجية

وصف الدرس

الغرض العام من هذا الدرس أن يتعلم الطلبة حول مكتبات **Python**، وكيفية استيراد الوحدات القياسية والدوال واستخدامها لتنفيذ مهام معينة.

ما سيتعلمه الطالب

- ما المكتبات البرمجية وما خصائصها.
- مكتبة **Python** القياسية وبعض وحداتها البرمجية.
- كيفية استخدام المكتبة واستيراد وحداتها البرمجية والدوال الموجودة بها.
- مدير الحزم **Python PIP** ودوره في استخدام المكتبات الخارجية.

نتاجات التعلم

- الاستفادة من المكتبة القياسية والمكتبات الخارجية في **Python** لاختصار الوقت المستغرق في عملية البرمجة.

المصطلحات

اللغة العربية	اللغة الإنجليزية
إعادة استخدام الكود	Code Reuse
التوثيق	Documentation
دالة	Function

اللغة العربية	اللغة الإنجليزية
استيراد	Import
وحدة برمجية	Module
إجراء	Procedure
مكتبة بايثون	Python Library
المكتبة القياسية	Standard Library

التحديات المتوقعة

> قد يواجه الطلبة صعوبة في التعرف على هيكلية مكتبة **Python** القياسية. شجع الطلبة على مراجعة المخطط المدرج في كتاب الطالب لفهم كيفية تضمين الوحدات والدوال داخل المكتبة القياسية.

> قد يواجه الطلبة صعوبة في تذكر المعاملات والدوال في الوحدة الجديدة.

> قد يواجه الطلبة صعوبة في استخدام الألوان من خلال التحكم بمعاملات **RGB**. يجب تعريف الألوان كمتغيرات بقيم نظام **RGB** في بداية البرامج.

> قد يواجه الطلبة صعوبة في تحديد الوحدة القياسية المناسبة للاستخدام في كل مهمة. فعلى سبيل المثال، عندما يرغب الطلبة في تشغيل ملف صوتي يمكنهم استخدام الوحدات القياسية **winsound** أو **pygame** أو **tkinter**، ولكن إذا أرادوا إنشاء كائن نصي فيمكنهم استخدام الوحدات القياسية **pygame** و **tkinter** ، وليس وحدة **winsound** القياسية.

> قد يواجه الطلبة صعوبة في تذكر الدوال الخاصة بكل من الوحدات القياسية. حث الطلبة على الاستعانة بجداول كتاب الطالب لمساعدتهم على تمييز الأوامر المختلفة.

التمهيد

> استخدم طريقة التدريس بالنقاش لتقديم غرض الدرس في أن يتعامل الطلبة مع مكتبات **Python**.

> يمكنك البدء بطرح بعض الأسئلة على الطلبة على سبيل المثال:

- ما هو استخدام المكتبات في حياتنا اليومية؟
- كيف نصنف الكتب من أجل عملية تنظيمها؟
- كيف يمكن البحث عن كتاب معين في مكتبة؟

> قم بإجراء نقاش مع الطلبة حول مصطلح "المكتبات" في لغة برمجة **Python**.

> يمكنك البدء بطرح بعض الأسئلة على الطلبة مثل:

- ما الذي تعرفونه حول الدوال والوحدات القياسية؟
- هل ترغبون بمعرفة المزيد من الدوال التي تساعدنا في حل المشكلات بشكل أسرع كيف يمكن إيجادها؟

التلميحات الخاصة بالتنفيذ

> باستخدام استراتيجية الحوار والمناقشة، عرِّف الطلبة على مفهوم مكتبة البرامج. اطلب من الطلبة تأمل الرسم التخطيطي في كتاب الطالب بعناية من أجل مساعدة الطلبة على فهم هيكلية مكتبة البرامج بالتوازي مع المكتبة الفعلية بالواقع، ثم اشرح لهم أن الوحدة القياسية لمكتبة البرامج يمكن تشبيهها بأحد أقسام المكتبة العامة مثلًا، وأن الدالة تشبه أحد الكتب في ذلك القسم. بالطبع فإن علينا الذهاب إلى القسم المحدد في المكتبة للعثور على الكتاب الذي نريده، وكذلك الحال بالنسبة للدوال في الوحدات القياسية.

> أثناء النقاش حول مكتبة **Python** القياسية، اشرح للطلبة كيفية استيراد وحدات ودوال محددة من أجل استخدامها في برامجهم.

> بعد تقديم جميع الوحدات الجديدة للطلبة، وباستخدام استراتيجية التعلم من خلال الاستكشاف، شجع الطلبة على تجربة الوحدات والدوال الجديدة. حثهم على إيجاد المعاملات المناسبة لكل دالة مع الاستعانة بكتاب الطالب إذا لزم الأمر.

> أثناء النقاش حول مدير حزم **Python pip**، اشرح للطلبة أن المكونات التركيبية للتعليمات البرمجية التي يمكن استيرادها إلى برنامج **Python** الخاص بك يشار إليها على

أنها وحدات. إن مكتبة **Python** القياسية هي مجموعة منسقة وواسعة من الوحدات التي تم توثيقها بشكل جيد. تتكون الوحدات القياسية من تعليمات برمجية بلغة **Python** ويتم تجميعها في أرشيفات مضغوطة تُعرف باسم الحزم (**packages**)، وذلك إلى جانب معاملات التثبيت وكود الاختبار وغيرها. يمكن القول ببساطة بأن الحزم هي الملفات التي يتم تنزيلها وتثبيتها بواسطة المستخدمين، ويعتبر pip أكثر مثبتات الحزم شيوعًا.

> تحقق أثناء الدرس من فهم كل طالب للخطوات المطلوبة واتباعها بشكل صحيح، وقدِّم إرشاداتك الفردية لمن يحتاج إلى مساعدة. تحقق من أي جزء من الدرس لم يفهمه الطلبة تمامًا وقم بإجراء أي تغييرات مطلوبة في عملية التدريس لكي يفهموه.

استراتيجيات غلق الدرس

> في نهاية الدرس تأكد من فهم الطلبة لجميع أهداف الدرس وتقييم معرفتهم من خلال أسئلة على سبيل المثال لا الحصر:

> هل تستطيع أن تتذكر:

- هل يمكنكم ذكر خصائص مكتبة البرامج؟
- مم تتكون هيكلية مكتبة **Python** القياسية؟
- ما هي بعض المكتبات التي لم يتم تضمينها في مكتبة **Python** القياسية؟

> ذكِّر الطلبة بالمصطلحات الهامة وكررها معهم.

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

التدريبات المقترحة لغلق الدرس

يمكنك استخدام التمرين الخامس في هذا الدرس كاستراتيجية ختامية لتقييم وتعزيز قدرة الطلبة على تطبيق المهارات التي تم تقديمها في هذا الدرس.

الصف الثاني عشر | الفصل الثاني | كتاب الطالب | صفحة 190

الفروق الفردية

تمارين إضافية للطلبة ذوي التحصيل المرتفع

> بعد الانتهاء من تنفيذ التمرين الثالث في هذا الدرس، اطلب من الطلبة إنشاء عرض تقديمي حول مكتبات **Python** القياسية. اطلب منهم البحث عبر الإنترنت والعثور على معلومات حول وحدات ودوال **Python** التي تنتمي إلى مكتبة **Python** القياسية، وتلك التي لا تنتمي إليها. يجب على الطلبة إنشاء مخطط لتنظيم معلوماتهم حول الوحدات ودوالها.

الإجابات النموذجية للتدريبات

1 أنشئ المقطع البرمجي لكي تجيب عن الأسئلة التالية.

```
from datetime import datetime

odds=[1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33,35,37,39,
41,43,45,47,49,51,53,55,57,59 ]

right_this_minute = datetime.today().minute

if right_this_minute in odds:
    print("This minute is odd.")
else:
    print("Not odd.")
```

1. المكتبة القياسية Library التي تم استيرادها في البرنامج هي **datetime**.
2. اسم الوحدة البرمجية Module التي تم استيرادها في البرنامج من المكتبة القياسية هو **datetime**.
3. اسم الدالة Function التي استدعيناها من الوحدة البرمجية هو **datetime.today( ).minute**.
4. وضح وظيفة البرنامج السابق.

يستورد البرنامج دالة datetime من الوحدة القياسية datetime. تقوم هذه الدالة بتمييز الأرقام الفردية، واستخراج الدقيقة الحالية من وقت النظام، ثم تخزن الإجابة في المتغير right_this_minute . وأخيرًا تتحقق هذه الدالة فيما إذا كانت الدقيقة الحالية فردية أم لا وتُبلِّغ المستخدم برسالة مطبوعة.

5. قم بتشغيل البرنامج وفسر النتائج التي ستحصل عليها.

تعتمد نتيجة هذا التمرين على قيمة الدقيقة الحالية، أهي فردية أم لا.

تلميح:

اشرح للطلبة أن Python تسمح بأن تتشابه أسماء الوحدات والدوال، فقد يكون للدالة نفس اسم الوحدة القياسية الخاصة بها. على سبيل المثال، يستورد السطر "from datatime import datatime" الدالة datatime من الوحدة القياسية datetime.

2

نفذ الكود البرمجي التالي.

```
import sys
print(dir(sys))
```

اشرح الوظيفة التي يقوم بها هذا الكود البرمجي.

تلميح:

حث الطلبة على الاطلاع على الجزء النظري الخاص بدالة ()dir، والذي تم شرحه في الدرس للإجابة على السؤال.

3

وضِّح الفرق بين مكتبات Python القياسية ومكتبات Python الأخرى، واذكر بعض الأمثلة على كل منهما.

تلميح:

يمكنك الرجوع للصفحتين 167 و 168 من كتاب الطالب.

4 وضح اثنتين من مميزات إنشاء الكود البرمجي الخاص بك بدلًا من استخدام وحدات برمجية خارجية.

تلميح:

شجع الطلبة على الاطلاع على الجانب النظري الذي تم تغطيته في الدرس للجمع بين المعرفة المكتسبة وإكمال التمرين.

5 أنشئ برنامج يستخدم الدوال التالية:

1. دالة تحسب الجذر التربيعي لرقم يقوم المستخدم بإدخاله.
2. دالة تحسب القيمة التقريبية للجذر التربيعي لذلك الرقم.
3. دالة تطبع رسالة تعرض للمستخدم الرقم الذي قام بإدخاله وجذره التربيعي، والقيمة التقريبية للجذر التربيعي.

على سبيل المثال، إذا كان هناك رقم "30"، فالرسالة التي ستطبع هي "the given number is 30"، والجذر التربيعي "its root is 5.478" والقيمة التقريبية للجذر التربيعي "round value of the root is 5".

```
#calculate the square root of a number
def squareRoot(a):
    return math.sqrt(number)

#calculate the approximate value of a number
def roundValue(a):
    return round(sq_root_number)

#print the numbers
def print_numbers():
    print("The given number is :" , number)
    print("The square root of the number is :" ,sq_root_number)
    print("The round value of the square root is :" ,round_v_number)

#main program
#import the math module
import math
number = int(input("Enter a number:"))
#calculate the root of the input number
sq_root_number = squareRoot(number)
#calculate the round value of the square root
round_v_number = roundValue(squareRoot)
#call the print_number function
print_numbers()
```

تلميح:

وضِّح للطلبة أنه ليس بالضرورة أن يكون هناك حل واحد صحيح فقط لمشكلة برمجية، بل يمكن قبول أي برنامج يؤدي إلى النتيجة الصحيحة.

> حث الطلبة على الاستعانة بإرشادات كتاب الطالب عند الضرورة.

> حث الطلبة على إضافة التعليقات المناسبة في برامجهم.

6

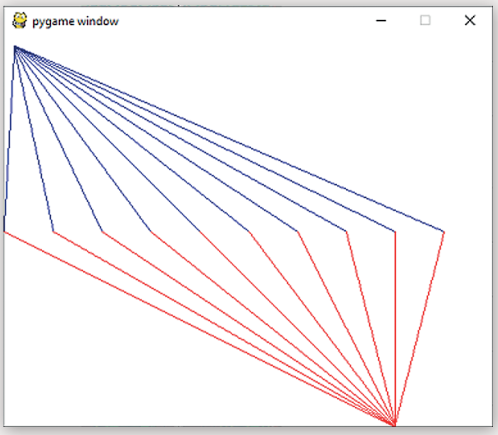
اذكر المقصود بالمكتبة البرمجية وعدد خصائصها.

تلميح:

شجع الطلبة على دراسة الجانب النظري حول مكتبات Python التي تمت تغطيته في الدرس للجمع بين المعرفة المكتسبة وإكمال التمرين.

7 املأ الفراغ في البرنامج من أجل إنشاء الأشكال التالية.

```
import pygame
pygame.init()

darkBlue = (0,0,128)
pink = (255,200,200)
green = (0, 255, 0)
red = (255, 0, 0)
white = (255, 255, 255)


screen = pygame.display.set_mode((500,400))
screen.fill(white)

for i in range(0,500,50):
    pygame.draw.lines(screen, darkBlue , False,
     [(10,10), (i,200)], 1)

for i in range(0,500,50):
    pygame.draw.lines(screen, red , False,
     [(400,400), (i,200)], 1)

pygame.display.update()
```



تلميح:

شجع الطلبة على تجربة استخدام المعاملات في البرمجة لتنفيذ التمرين.