



دليل المعلم

10

الصف العاشر - الفصل الدراسي الثاني



دليل المعلم تقنية المعلومات

الصف العاشر

إشراف

أ/ منى سالم عوض سالم

أ/ فاطمه خليف غنيم العنزوي

د/ ماجدة إبراهيم مزعل الدوسري

أ/ ساره ياسين عبدالله الامير

أ/ إيمان عبد العزيز أحمد الفارسي

أ/ منار مصطفى عبدالحميد جمال

أ/ إبراهيم عبدالله إبراهيم المياس

إخراج

أ/ سنيه محمد علي المؤمن

المحتوى

| | | |
|----|--|---|
| 6 | الوحدة الأولى المعالجة الرقمية Digital Processing Python Programming | 1 |
| 7 | | القوائم Lists |
| 14 | | الصفوف Tuples |
| 17 | | القواميس Dictionaries |
| 22 | | الدوال Functions |
| 26 | الوحدة الثانية المنتجات الرقمية Digital Products Artificial Intelligence | 2 |
| 27 | | الوحدات والحزم Modules and Packages |
| 29 | | مدخل إلى الذكاء الاصطناعي Introduction to Artificial Intelligence |
| 33 | | مشروعات الذكاء الاصطناعي Artificial Intelligence Projects |

المقدمة

عزيزي المعلم

يسعدنا أن نقدم لك هذا الدليل الشامل الذي يهدف إلى دعمك في تقديم مادة تقنية المعلومات لمتعلمي الصف العاشر (الجزء الثاني) بما في ذلك مفاهيم الذكاء الاصطناعي وبرمجة بايثون وبصورة فعالة ومبتكرة، نحن ندرك أن تقديم هذه المادة الحيوية يحتاج إلى أدوات ومعارف حديثة تتماشى مع تطورات العصر الرقمي واحتياجات المتعلمين.

يتضمن هذا الدليل استراتيجيات تعليمية متنوعة تناسب مع الفئة العمرية للمتعلمين، مثل التعلم المباشر (المحاضرة)، التعلم التعاوني، المناقشة والحوار، التعلم القائم على حل المشكلات، والتعلم القائم على المشاريع، كما يوفر الدليل أنواعاً مختلفة من التقويمات، سواء التكوينية منها أو النهائية، لضمان تقييم شامل لتقدم المتعلمين ومساعدتهم على تحسين أدائهم بشكل مستمر.

لضمان تجربة تعليمية شاملة، يحتوي هذا الدليل على خطط مفصلة لكل وحدة دراسية تشمل أهداف الوحدة، الموضوعات الرئيسية، الأنشطة التعليمية، المواد والأدوات اللازمة، وأساليب التقويم. تم تصميم نتائج التعلم بوضوح لكل وحدة لتمكين المتعلمين من معرفة ما يتوقع منهم تحقيقه بنهاية كل درس.

فيما يخص برمجة بايثون، فإن الدليل يحتوي على موارد مفيدة لشرح المفاهيم البرمجية الأساسية، بما في ذلك أمثلة عملية وأنشطة تطبيقية لمساعدة المتعلمين على فهم البرمجة بطريقة سلسة وممتعة. كما يتضمن الدليل قائمة شاملة بالمصطلحات الحاسوبية الهامة، بالإضافة إلى الحلول النموذجية لأوراق العمل لتيسير عملية التصحيح. يهدف هذا الدليل إلى تزويدك بالأدوات والمعارف اللازمة لتحسين عملية التعلم والتعليم، ودعمك في التحضير الفعال للدرس وتنفيذه بنجاح.

أما بالنسبة للذكاء الاصطناعي، إن الهدف من استخدامها هو تمكين المعلمين و المتعلمين من تطوير إنتاجيتهم و كذلك تقديم الحلول المستمرة لتطوير العملية التعليمية وزيادة فاعليتها وتطوير أنظمة وبرمجيات قادرة على محاكاة العقل البشري و تنفيذ مهام معقدة تكون قادرة على التعلم من التجارب و الاستنتاج والتنبؤ وحل المشكلات وغيرها من مهارات يتميز بها الانسان.

نحن على يقين بأن هذا الدليل سيكون مرشدك الأمثل في رحلتك التعليمية، ويساعدك على تقديم دروس ممتعة، تساهم في تطوير مهارات المتعلمين وتعزيز فهمهم للتكنولوجيا، الذكاء الاصطناعي، وبرمجة بايثون. نأمل أن تجد في هذا الدليل كل ما تحتاجه لتحقيق أفضل النتائج في تدريس منهج الصف العاشر

مع أطيب التمنيات

الوحدة الأولى : المعالجة الرقمية

Digital Processing

Python Programming

01



القوائم Lists

نتائج التعلم:

| | |
|---|---|
| التعرف على هياكل البيانات غير الأولية مثل القوائم Lists، واستخداماتها في تنظيم البيانات. | 1 |
| تمييز خصائص القوائم (مثل: القابلية للتغيير، الترتيب، الفهرسة) مقارنة ببقية الهياكل مثل tuples و dictionaries. | 2 |
| إنشاء Lists باستخدام الصيغة الصحيحة في لغة Python، وإدخال البيانات إليها بطرق مختلفة (يدوية أو من المستخدم). | 3 |
| الوصول إلى عناصر Lists باستخدام الفهرسة الإيجابية والسلبية وتفسير النتائج. | 4 |
| تعديل عناصر Lists من خلال إسناد قيم جديدة إلى فهراس محددة. | 5 |
| إجراء عمليات الإضافة والحذف على Lists باستخدام دوال مثل append و insert و remove و pop. | 6 |
| معالجة الأخطاء البرمجية الناتجة عن التعامل مع Lists فارغة أو أخطاء في الفهرسة لضمان سلامة البرنامج. | 7 |

المصطلحات الحاسوبية:



فهرسة القوائم Indexing



القوائم Lists



الدالة Remove



الدالة pop



الدالة Len



الدالة Clear



الدالة Sort



الدالة Copy



تابع/ القوائم Lists

تابع/ المصطلحات الحاسوبية:



الدالة Split



الدالة Join

يمكن إنشاء قائمة من العديد من أنواع البيانات مثل الأعداد الصحيحة والعشرية والسلاسل النصية وأيضا ممكن أن تحتوي القائمة على عدة قوائم مختلفة.



ملاحظة للمعلم

الوصول إلى العناصر باستخدام الفهرس (الفهرس يبدأ من 0).



إمكانية التعديل للقيمة المخزنة باستخدام الفهرس.



إضافة عناصر جديدة إلى نهاية القائمة باستخدام الدالة `append()`.



حذف عناصر من القائمة باستخدام `remove()` - `pop()`.



حذف كافة العناصر استخدام الدالة `clear()`.



الحصول على طول القائمة باستخدام الدالة `len()`.



إمكانية فرز القوائم `sort()`.





تابع/ القوائم Lists

استراتيجيات وأمثلة مقترحة :

| الهدف | التعرف على مفهوم القوائم والعمليات عليها. |
|------------------------|--|
| الاستراتيجية التعليمية | العصف الذهني، التعلم التعاوني، الصف المقلوب. |

- بعد بدء الحصة وأخذ الحضور والغياب، يبدأ المعلم بإنشاء حلقة نقاشية حول: ماهي الخطوات التي تم اتباعها لأخذ الحضور والغياب لتحقيق الأهداف التالية:
 - من خلال أسماء الطلاب، يتم شرح القائمة.
 - من خلال الرقم المسلسل، يتم شرح الفهرسة (مع ملاحظة أن ينوه لاحقاً أن البايثون يبدأ من الصفر).
 - من خلال عملية النداء بالأسماء، يتم شرح الحلقة التكرارية لعرض عناصر القائمة.
 - من خلال حذف وإضافة طالب، يتم شرح حذف وإضافة العناصر.
- وغيرها من العمليات.



تابع/ القوائم Lists

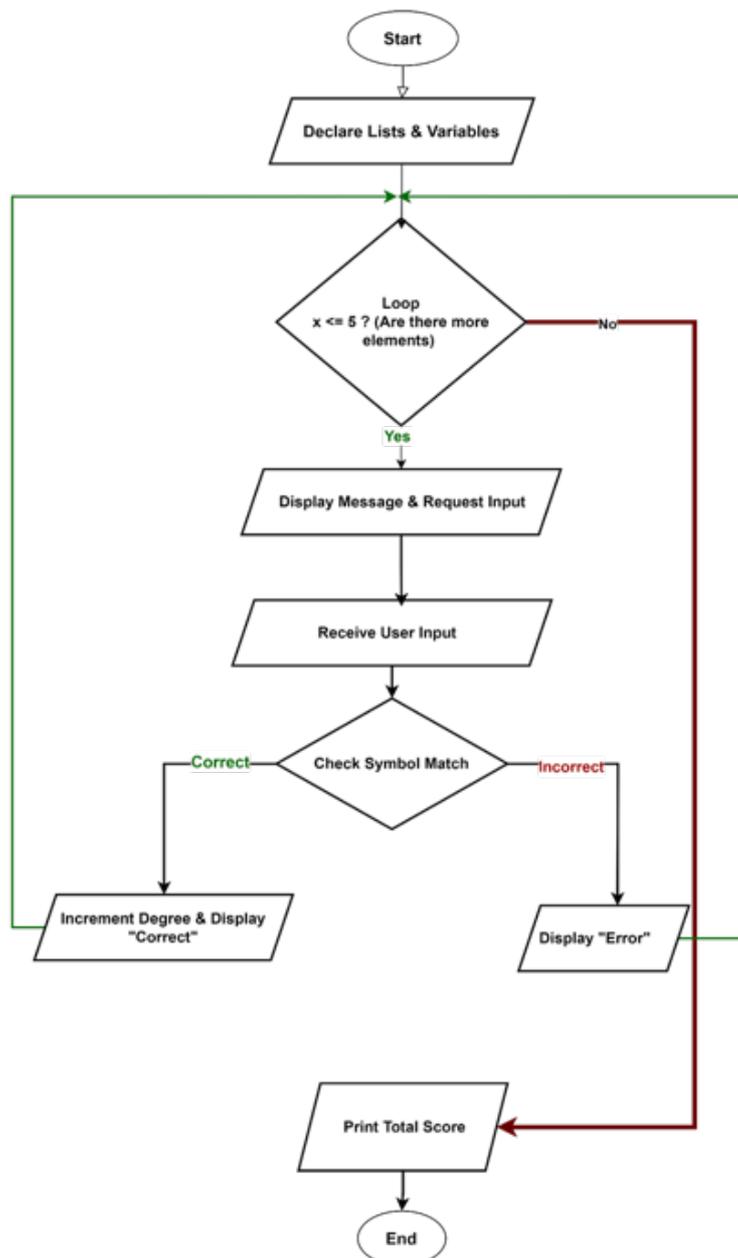
حل أوراق العمل - ورقة عمل 1 ص 27:

برنامج اختبار معلوماتك

مشكلة البرنامج:

استرجاع رموز العناصر الكيميائية، وربطها بأسمائها.

خريطة التدفق:



Input

```

1 # القوائم
2 element_name = ['Sodium', 'Potassium', 'Copper', 'Iron', 'Mercury', 'Lead']
3 element_symbol = ['Na', 'K', 'Cu', 'Fe', 'Hg', 'Pb']
4 # المتغيرات
5 degree = 0
6 x = 0
7 i = 6
8 # حلقة التكرار
9 while x < i:
10     es = input("Enter element symbol " + element_name[x] + ": ")
11     if es == element_symbol[x]:
12         print("Correct")
13         degree += 1
14     else:
15         print("Incorrect", element_symbol[x])
16     x += 1 # زيادة العداد
17 # طباعة الدرجة النهائية
18 print("degree ", degree, "/", i)

```

Output

```

Enter element symbol Sodium: Na
Correct
Enter element symbol Potassium: K
Correct
Enter element symbol Copper: Cu
Correct
Enter element symbol Iron: Fe
Correct
Enter element symbol Mercury: Hg
Correct
Enter element symbol Lead: Pb
Correct
degree 6 / 6

```



تابع/ القوائم Lists

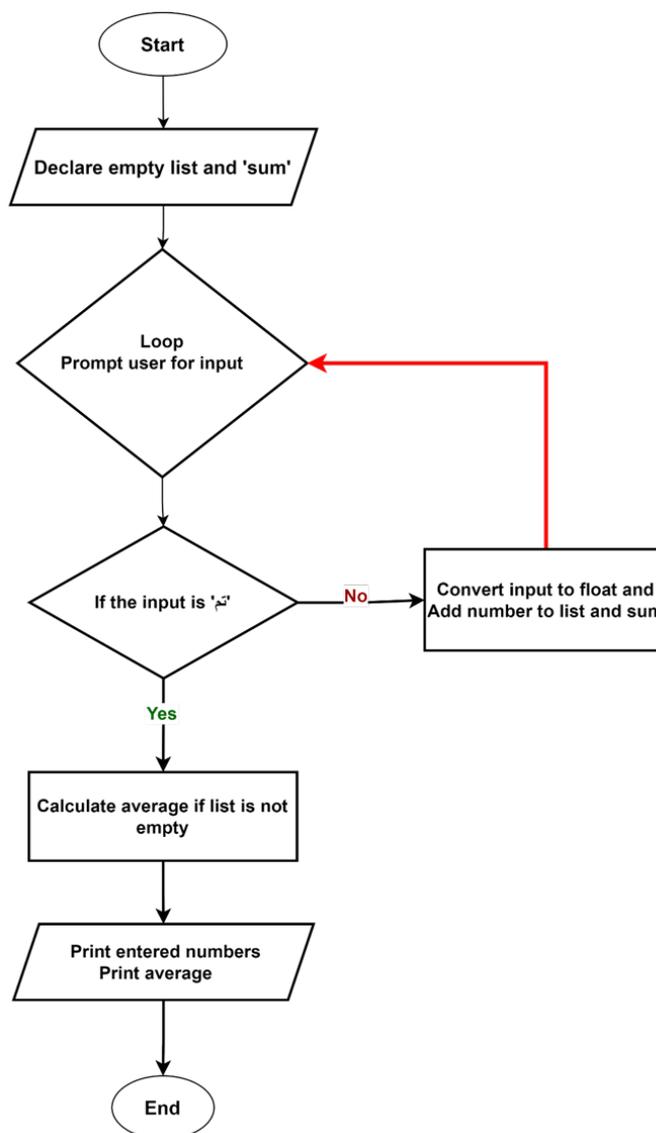
تابع/ حل أوراق العمل - ورقة عمل 2 ص 30:

برنامج حساب المتوسط الحسابي

مشكلة البرنامج:

حاجة المستخدم إلى برنامج يُدخل من خلاله أعداداً موجبة أو صفرية، ثم يحسب المتوسط الحسابي تلقائياً بعد الانتهاء من الإدخال.

خريطة التدفق:





تابع/ القوائم Lists

التعليمات البرمجية:

Input

```
1 number_List = []
2 sum_1 = 0
3 input_number = 0
4 while True:
5     input_number = int (input("Enter a number (negative to stop): "))
6     if input_number >= 0:
7         number_List.append(input_number)
8     else:
9         break
10
11     sum_1 = sum_1 + input_number
12 average = sum_1/ len(number_List)
13 print("Numbers you entered: ")
14 print (number_List)
15 print (f"Average value {average}")
```

Output

```
Enter a number (negative to stop): 5
Enter a number (negative to stop): 10
Enter a number (negative to stop): 20
Enter a number (negative to stop): -1
Numbers you entered:
[5, 10, 20]
Average value 11.666666666666666
```



الصفوف Tuples

نتائج التعلم: الصفوف Tuples

| | |
|---|---|
| 1 | التعرف على مفهوم Tuples كنوع من أنواع البيانات غير القابلة للتغيير Immutable في Python. |
| 2 | التمييز بين خصائص Tuples والأنواع الأخرى من البيانات، مثل Lists. |
| 3 | إنشاء Tuple يحتوي على عناصر مرتبة باستخدام الأقواس الدائرية. |
| 4 | الوصول إلى عناصر Tuple باستخدام الفهرسة الموجبة والسالبة. |
| 5 | تحويل Tuple إلى List بهدف تعديل العناصر، ثم إعادته إلى Tuple. |
| 6 | تخصيص عناصر Tuple إلى متغيرات متعددة باستخدام تقنية Sequence Unpacking. |
| 7 | كتابة برامج بسيطة تتعامل مع Tuples لتمثيل بيانات متعددة الأنواع. |
| 8 | استخدام التكرار لطباعة محتويات Tuples وتنظيم إخراج البيانات. |
| 9 | اختبار البرامج والتأكد من خلوها من الأخطاء البرمجية أثناء التنفيذ. |

المصطلحات الحاسوبية:



سلسلة Sequence



قابل للتغيير Immutable



فك ترتيب التسلسل Sequence Unpacking



أقواس دائرية Parentheses



تابع/ الصفوف Tuples

استراتيجيات وأمثلة مقترحة :

| الهدف | التعرف على مفهوم الصفوف |
|------------------------|--|
| الاستراتيجية التعليمية | العصف الذهني، التعلم التعاوني، الصف المقلوب. |

- استخدام أيام الأسبوع لتمثيل الصفوف Tuples، وشرح طريقة الإعلان عنها وإنشائها، ثم تكليف كل مجموعة لإنشاء صف Tuple خاص بها.

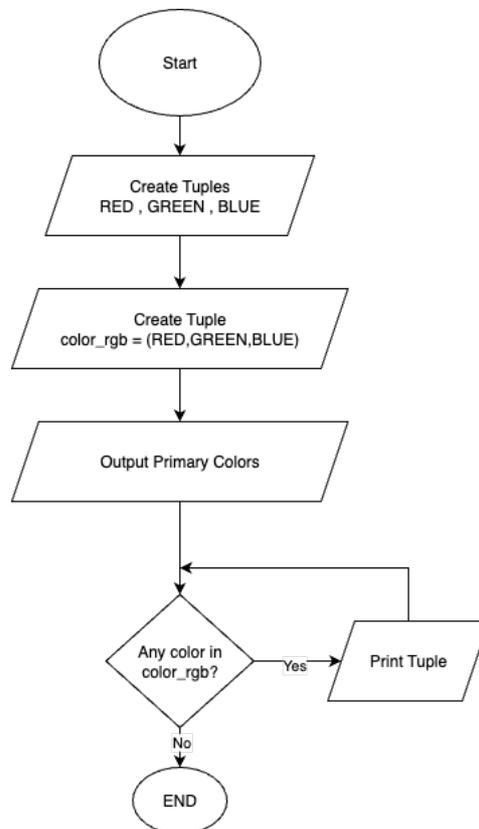
حل أوراق العمل - ورقة عمل 3 ص 42

نظام الألوان الضوئية الأساسية

مشكلة البرنامج:

إنشاء مصفوفة تمثل الألوان الضوئية الأساسية وطباعتها.

خريطة التدفق:





تابع / الصفوف Tuples

التعليمات البرمجية:

Input

```
1 # إنشاء مصفوفة تمثل الألوان الضوئية الأساسية وطباعة عناصرها
2 # RGB الألوان الأساسية في نظام
3 RED = (255, 0, 0)
4 GREEN = (0, 255, 0)
5 BLUE = (0, 0, 255)
6
7 color_rgb = (RED, GREEN, BLUE)
8 print ("Primary Colors: ")
9 for color in color_rgb:
10     print(color)
```

Output

```
Primary Colors:
(255, 0, 0)
(0, 255, 0)
(0, 0, 255)
```



القواميس Dictionaries

نتائج التعلم :

| | |
|----|--|
| 1 | التعرف على Dictionaries كنوع من أنواع البيانات غير الأولية القابلة للتغيير Mutable. |
| 2 | التمييز بين خصائص Dictionaries مقارنة بالقوائم Lists والصفوف Tuples. |
| 3 | إنشاء Dictionary باستخدام الأقواس المعقوفة {} يحتوي على أزواج من المفاتيح والقيم. |
| 4 | استخدام الفهرسة بواسطة المفاتيح للوصول إلى القيم المرتبطة بها. |
| 5 | إضافة وتعديل عناصر داخل Dictionary باستخدام التعيين المباشر. |
| 6 | حذف عناصر من Dictionary باستخدام الدوال مثل clear() - popitem() - pop(). |
| 7 | طباعة مفاتيح وقيم Dictionary باستخدام دالة items() والتكرار for. |
| 8 | تحليل البيانات داخل Dictionaries واستخلاص نتائج مثل مجموع الدرجات أو أسماء الطلاب الناجحين/الراسبين. |
| 9 | توظيف Dictionary في سيناريوهات عملية، مثل تخزين بيانات الطالب أو درجاته في المواد المختلفة. |
| 10 | اختبار البرامج والتأكد من خلوها من الأخطاء البرمجية أثناء التنفيذ. |

المصطلحات الحاسوبية :

| | |
|------------------------------------|-----------------------------|
| | |
| الأقواس المعقوفة Curly Braces {} . | القواميس Dictionaries |
| | |
| المفاتيح Key | القيم Value |
| | |
| فريدة Unique | عناصر غير مرتبة Not Ordered |
| | |
| الدالة dict | |



تابع/ القواميس Dictionaries

استراتيجيات وأمثلة مقترحة :

| الهدف | التعرف على مفهوم القواميس. |
|------------------------|--|
| الاستراتيجية التعليمية | العصف الذهني، التعلم التعاوني، الصف المقلوب. |

- ينشئ المعلم شهادة طالب وهمية ثم يطلب من كل مجموعة استخراج بيانات بدلالة اسم المادة، ثم يوضح المعلم أن القواميس تعمل على نفس الآلية، يتم استخراج القيمة بالدلالة الفتحاح.

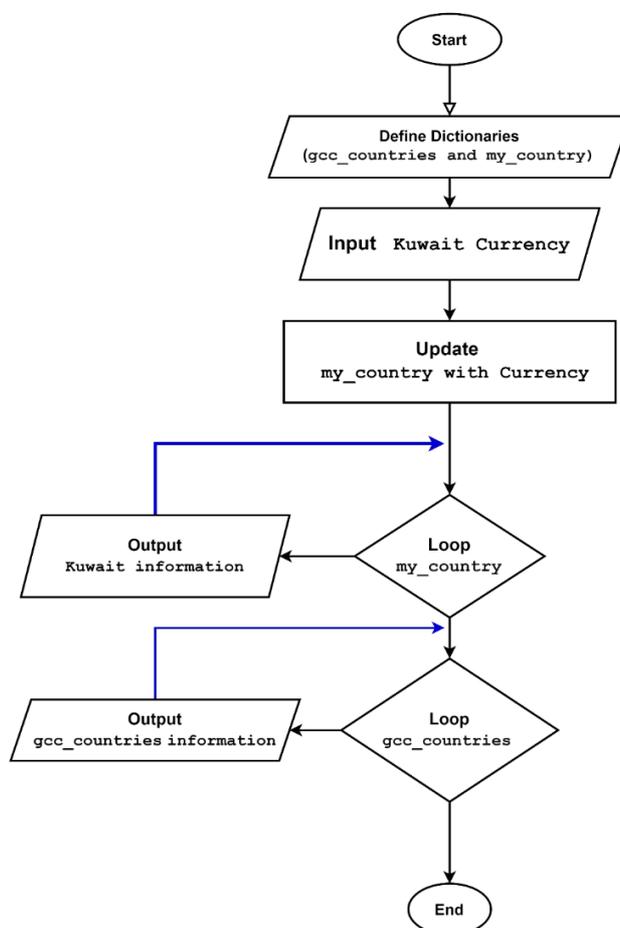
حل أوراق العمل - ورقة عمل 4 ص 54:

دول مجلس التعاون الخليجي GCC

مشكلة البرنامج:

تنفيذ عمليات الإنشاء وإضافة العناصر وتعديلها وحذفها على Dictionaries.

خريطة التدفق:





تابع/ القواميس Dictionaries

التعليمات البرمجية:

Input

```
1 gcc_countries = {
2     "Kuwait": "Kuwait City",
3     "Saudi Arabia": "Riyadh",
4     "United Arab Emirates": "Abu Dhabi",
5     "Qatar": "Doha",
6     "Oman": "Muscat",
7     "Bahrain": "Manama",
8     "Yemen": "Sana'a"
9 }
10 my_country = {
11     "country": "Kuwait",
12     "capital": "Kuwait City",
13     "Population": 8000000,
14 }
15 #Update Key population value to 5000000
16 my_country["Population"] = 5000000
17 #Enter new value to Currency key
18 my_country["Currency"] = input("Enter Kuwait Currency: ")
19 print("Info About Kuwait")
20 print ('-' * 40)
21 for key, value in my_country.items():
22     #print element of my_country dictionary
23     print(key, ":", value)
24 # Delete an element from GCC_countries
25 gcc_countries.pop("Yemen")
26 print("The Gulf Cooperation Council Countries")
27 print ('-' * 40)
28 for country, capital in gcc_countries.items():
29     #Print element of gcc_countries dictionary
30     print(country, ":", capital)
```

Output

Enter Kuwait Currency: *Kuwaiti Dinar*

Info About Kuwait

country : Kuwait
capital : Kuwait City
Population : 5000000
Currency : Kuwaiti Dinar

The Gulf Cooperation Council Countries

Kuwait : Kuwait City
Saudi Arabia : Riyadh
United Arab Emirates : Abu Dhabi
Qatar : Doha
Oman : Muscat
Bahrain : Manama



تابع/ القواميس Dictionaries

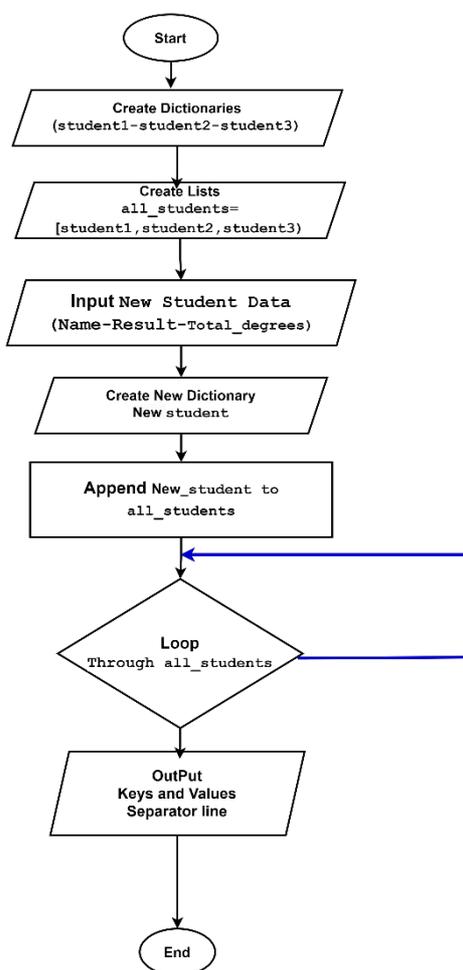
تابع/ حل أوراق العمل - ورقة عمل 5 ص 58:

برنامج سجل الطالب

مشكلة البرنامج:

التعامل مع عناصر القاموس Dictionary, والقوائم Lists.

خريطة التدفق:





تابع/ القواميس Dictionaries

التعليمات البرمجية:

Input

```
1 # create dictionaries
2 student1 = {"name": "ali", "result": "success", "total_degrees": 650.0} # بيانات المتعلم الأول
3 student2 = {"name": "khalid", "result": "success", "total_degrees": 700.0} # بيانات المتعلم الثاني
4 student3 = {"name": "ahmed", "result": "failed", "total_degrees": 250.0} # بيانات المتعلم الثالث
5 #Create a list includes the three dictionaries
6 all_students = [student1, student2, student3]
7 #Enter new student data through user input
8 name = input("Enter new name: ") # إدخال اسم المتعلم الجديد
9 result = input("Enter student result(success/failed): ") # إدخال نتيجة المتعلم (نجاح أو رسوب)
10 total_degrees = float(input("Enter total degrees: ")) # إدخال مجموع الدرجات (يجب أن يكون رقمًا)
11 #Create new Student's Dictionary
12 new_student = {"name": name, "result": result, "total_degrees": total_degrees}
13 #Add Student's Dictionary to list (all_students)
14 all_students.append(new_student) # إضافة القاموس للمتعلم الجديد إلى القائمة
15 print()
16 #Print the data of all students
17 for i in all_students: # المرور عبر قائمة (قواميس) المتعلمين
18     for x, y in i.items(): # المرور عبر مفاتيح القاموس وقيمه
19         print(f"{x:<20}{y}") # طباعة المفتاح والقيمة بتنسيق عرض 20 خانة
20     print("-" * 40) # طباعة خط فاصل بين بيانات كل طالب
```

Output

```
Enter new name: Fahed
Enter student result(success/failed): success
Enter total degrees: 500
```

```
name          ali
result        success
total_degrees 650.0
```

```
-----
name          khalid
result        success
total_degrees 700.0
```

```
-----
name          ahmed
result        failed
total_degrees 250.0
```

```
-----
name          Fahed
result        success
total_degrees 500.0
-----
```



نتائج التعلم :

| | |
|---|--|
| 1 | التعرف على مفهوم الدالة في Python بوصفها وحدة برمجية مستقلة تنفذ مهام محددة عند استدعائها. |
| 2 | التمييز بين أنواع الدوال المدمجة Built-in والدوال المعرفة من المستخدم User-defined. |
| 3 | توضيح مميزات استخدام الدوال في تنظيم البرامج وتسهيل الصيانة وإعادة الاستخدام. |
| 4 | كتابة دوال باستخدام الكلمة المفتاحية def مع تحديد الاسم والمعاملات Parameters. |
| 5 | تحديد الفرق بين المعاملات Parameters والقيم المرسلة Arguments. |
| 6 | استدعاء الدالة داخل البرنامج بالطريقة الصحيحة وتمرير القيم إليها. |
| 7 | استخدام تعليمة return لإرجاع نتائج من داخل الدالة عند الحاجة. |
| 8 | التمييز بين المتغيرات المحلية والعالمية داخل البرنامج. |
| 9 | اختبار البرنامج والتأكد من خلوه من الأخطاء البرمجية أثناء التنفيذ. |

المصطلحات الحاسوبية:



الدوال المدمجة Built-in functions



الدوال Functions



تعريف الدالة Define a function



الدوال المعرفة من المستخدم User-defined functions



Function Head



Function body



معاملات الدالة / Function Parameter



صيغة الدالة Function Syntax



تابع/ الدوال Functions

تابع/ المصطلحات الحاسوبية:



الإرجاع Return Expression



التعليمات Statements



القيم المرسلة Arguments



المعاملات parameters



مجهولة الاسم Anonymo



القيم الافتراضية لمعاملات الدوال default parameters



المتغير المحلي Local Variable



نطاق المتغيرات Scope



متوالية فيبوناتشي Fibonacci Generator



المتغير العام global variable



النسبة الذهبية Golden Ratio



تابع/ الدوال Functions

استراتيجيات وأمثلة مقترحة :

الهدف

التعرف على مفهوم الدوال وأهميتها.

الاستراتيجية التعليمية

العصف الذهني، التعلم التعاوني، لعب الأدوار.

- يفترض المعلم أن الصف يمثل شركة لتصميم الألعاب، ثم يوضح المعلم أن لو اجتمعوا جميعاً كل مرة لتصميم اللعبة ستأخذ وقتاً وطويلاً، فما الحل؟ تبدأ الآن مرحلة العصف الذهني.
- يوضح المعلم أن أفضل طريقة لإنشاء اللعبة وإدارة العمل هي تسييم المهام بين أعضاء الفريق، حيث كل عضو في الفريق يكلف بعمل جزء من البرنامج، ثم يتم تجميع الأجزاء من قبل فريق العمل وتوليف التعليمات البرمجية.

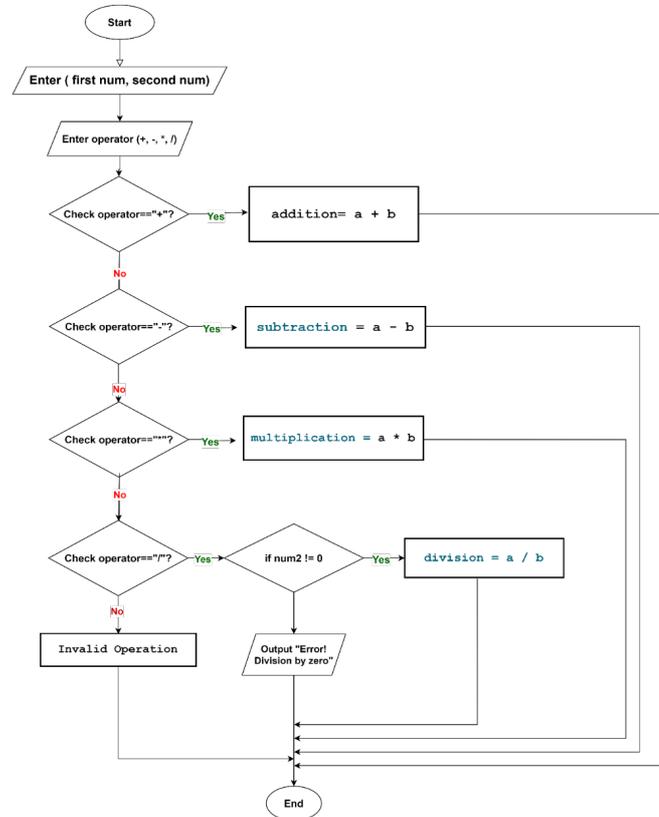
حل أوراق العمل - ورقة عمل 6 ص 74

برنامج الآلة الحاسبة

مشكلة البرنامج:

توظيف الدوال لإجراء العمليات الحسابية بين عددين مدخلين.

خريطة التدفق:





تابع/ الدوال Functions

التعليمات البرمجية:

Input

```
1 # تعريف دالة الجمع
2 def addition(a, b):
3     return a + b # ترجع ناتج جمع العددين
4 # تعريف دالة الطرح
5 def subtraction(a, b):
6     return a - b # ترجع ناتج طرح العدد الثاني من الأول
7 # تعريف دالة الضرب
8 def multiplication(a, b):
9     return a * b # ترجع ناتج ضرب العددين
10 # تعريف دالة القسمة
11 def division(a, b):
12     if b == 0: # يتحقق من أن القاسم ليس صفراً
13         return "Sorry, Can not divide by zero!!" # رسالة خطأ إذا كان القاسم صفراً
14     return a / b # ترجع ناتج القسمة
15 # تعريف دالة الحاسبة البسيطة
16 def simple_calculator():
17     # طلب الأعداد من المستخدم
18     number1 = float(input("Enter first number: ")) # إدخال العدد الأول
19     number2 = float(input("Enter second number: ")) # إدخال العدد الثاني
20     # طلب العملية الحسابية المطلوبة
21     user_input = input("Enter your choice (+, -, *, /): ") # إدخال نوع العملية بناءً على اختيار المستخدم
22     # تنفيذ العملية بناءً على اختيار المستخدم
23     if user_input == "+":
24         result = addition(number1, number2) # استدعاء دالة الجمع
25     elif user_input == "-":
26         result = subtraction(number1, number2) # استدعاء دالة الطرح
27     elif user_input == "*":
28         result = multiplication(number1, number2) # استدعاء دالة الضرب
29     elif user_input == "/":
30         result = division(number1, number2) # استدعاء دالة القسمة
31     else:
32         print("Invalid input.") # إذا كانت العملية المدخلة غير صحيحة
33         exit() # إنهاء البرنامج
34     # طباعة ناتج العملية
35     print(result)
36 # استدعاء دالة الحاسبة
37 simple_calculator()
```

Output

```
Enter first number: 9
Enter second number: 0
Enter your choice (+, -, *, /): /
Sorry, Can not divide by zero!!
```

الوحدة الثانية : المنتجات الرقمية

Digital Products

Artificial Intelligence

02



الوحدات والحزم Modules and Packages

نتائج التعلم

| | |
|---|---|
| 1 | التعرف على مفهوم الوحدات والحزم. |
| 2 | توضيح أهمية الوحدات والحزم. |
| 3 | التمييز بين بعض الوحدات المدمجة مثل datetime-random والحزم الخارجية NumPy – OpenCV. |
| 4 | استيراد الوحدات والحزم. |
| 5 | استيراد دالة أو عنصر محدد من الوحدات والحزم. |
| 6 | استخدام الحزم الجاهزة لأداء مهمة معينة. |
| 7 | تثبيت الحزم الخارجية. |
| 8 | تنفيذ برامج بسيطة توظف حزم خارجية. |
| 9 | ربط استخدام حزم البرمجية بتطبيقات حقيقية في مجالات متنوعة. |

المصطلحات : الوحدات والحزم Modules and Packages



الذكاء الاصطناعي Artificial Intelligence



تحليل البيانات Data Analysis



التعلم العميق Deep Learning



تعلم الآلة Machine Learning



تطوير الألعاب Gaming Development



الأمن السيبراني Cybersecurity



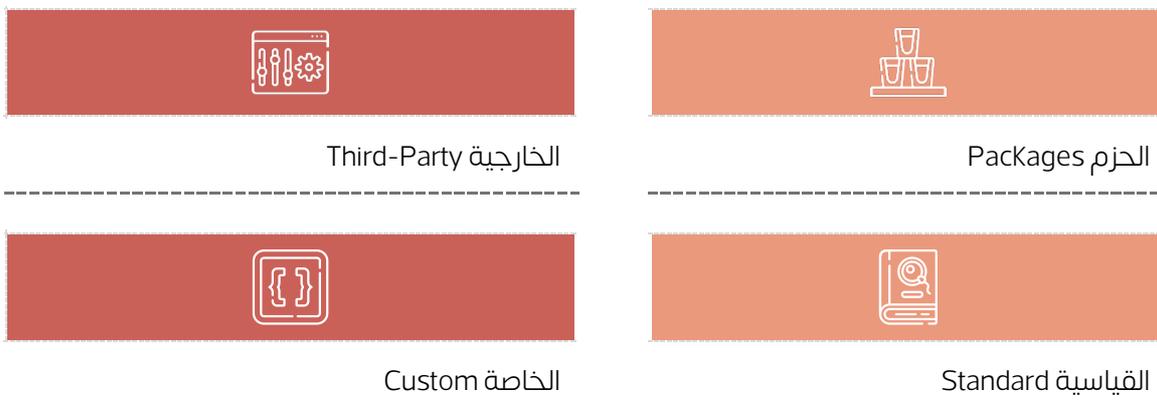
الرؤية الحاسوبية Computer Vision



الوحدات Modules



تابع/ الوحدات والحزم Modules and Packages



استراتيجيات وأمثلة مقترحة :

| الهدف | التعرف على مفهوم الحزم والوحدات وأهميتها. |
|------------------------|---|
| الاستراتيجية التعليمية | العصف الذهني، التعلم التعاوني، لعب الأدوار. |

- يفترض المعلم أن الصف يمثل شركة لتصميم الألعاب، ثم يوضح المعلم لإنشاء لعبة من الصفر ستأخذ سنوات، فما هي السبل للاقتصار الوقت والجهد؟ حلقة عصف ذهني
- يوضح المعلم أن استخدام محرك تصميم الألعاب مثل Unity أو Unreal سيوفر وقتاً ومجهود جبار في تطوير اللعبة حيث تتضمن مجموعة من الأدوات الجاهزة والبيئة المناسبة. وكذلك الوحدات والحزم في بايثون.



مدخل إلى الذكاء الاصطناعي

نتائج التعلم :

| | |
|---|--|
| 1 | التعرف على مفهوم الذكاء الاصطناعي. |
| 2 | التمييز بين وظائف تعلم الآلة. |
| 3 | توضيح مفهوم التعلم العميق والذكاء التوليدي واستخداماته. |
| 4 | تحليل العلاقة بين الذكاء الاصطناعي وانترنت الأشياء. |
| 5 | توظيف مهارات تصميم الأوامر Prompt Design للحصول على مخرجات دقيقة من أدوات الذكاء التوليدي. |
| 6 | تحليل مراحل إعداد البيانات. |
| 7 | مناقشة المبادئ الأخلاقية للذكاء الاصطناعي وأثرها على الاستخدام المسؤول للتقنيات. |
| 8 | استكشاف الأدوات البرمجية الذكية لتطبيقات الذكاء الاصطناعي وإجراء تجارب بسيطة عليها. |

المصطلحات الحاسوبية:



تعلم الآلة **Machine Learning**



الذكاء الاصطناعي **Artificial Intelligence**



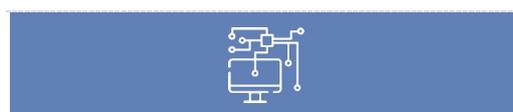
الذكاء الاصطناعي التوليدي
Generative Artificial Intelligence



التعلم العميق **Deep Learning**



نموذج اللغة الكبير
Large Language Model (LLM)



الشبكات العصبية الاصطناعية
Artificial Neural Networks



تابع/ مدخل إلى الذكاء الاصطناعي

تابع/ المصطلحات الحاسوبية:



فجوة القوة الحوسبية

Computing Power Gap



المحول التوليدي سابق التدريب

pretrained Transform A Generative



انتهاك حقوق الملكية

Violation Intellectual Property



التطور السريع Rapid Development



التزييف العميق Deep fake



الحد من تنوع الآراء

Limiting Diversity of Opinions



توليد الصور Image Generation



توليد نص Text Generation



شبكات تنافسية توليدية

Generative Adversarial Networks(GANs)



نماذج الانتشار Diffusion Mode



تابع/ مدخل إلى الذكاء الاصطناعي

تابع/ المصطلحات الحاسوبية:



تقنية الانتشار العكسي Inverse Diffusion



توليد الفيديو Video Generate



أدوات معالجة اللغة الطبيعية: (NLP)



توليد الصوت Voice Generation



أداة لإنشاء الصور الواقعية والفنية. DALL-E 2



أداة إنشاء الصور الفنية Midjourney



منصة بناء وتدريب ونشر نماذج التعلم الآلي

Amazon Sage MaKer .



أدوات الذكاء الاصطناعي الخاصة بجوجل

Google Cloud AI



Google Gemini

أحد نماذج الذكاء الاصطناعي تقدما من جوجل.



Microsoft Copilot



تابع/ مدخل إلى الذكاء الاصطناعي

تابع/ المصطلحات الحاسوبية:



الذكاء الاصطناعي من أجل الأرض

Artificial Intelligence for Earth



Internet of Things IoT

انترنت الأشياء



الذكاء الاصطناعي لمساعدة المكفوفين

Seeing Artificial Intelligence AI



مشروعات الذكاء الاصطناعي Artificial Intelligence Projects

نتائج التعلم :

| | |
|---|---|
| 1 | التعرف على مفهوم الرؤية الحاسوبية ودور مكتبة OpenCV في الذكاء الاصطناعي. |
| 2 | تمييز الفرق بين الكشف عن الوجوه والكشف عن المشاة من حيث المفهوم والخوارزمية. |
| 3 | توضيح خطوات استخدام مكتبة OpenCV في تحليل الصور واكتشاف الكائنات. |
| 4 | تنفيذ مشروع برمجي لاكتشاف الوجوه باستخدام خوارزمية Haar Cascade. |
| 5 | تعديل البرنامج البرمجي ليتفاعل من كاميرا الحاسوب في الوقت الحالي. |
| 6 | تصميم برنامج برمجي لاكتشاف المشاة باستخدام خوارزمية HOG والمصنف SVM. |
| 7 | تحليل معايير ضبط الكواشف مثل scaleFactor و minNeighbors وتأثيرها على نواتج الكشف. |
| 8 | استكشاف تطبيقات الذكاء الاصطناعي في الحياة الواقعية باستخدام الرؤية الحاسوبية. |

المصطلحات الحاسوبية:



الشبكات العصبية العميقة Deep Neural Networks



خوارزمية الرؤية الحاسوبية Haar Cascade



تعلم الآلة Machine Learning



تقنيات التعلم العميق Customize Libraries



تابع/ المصطلحات الحاسوبية:



مكتبة مفتوحة المصدر لمعالجة الصور والرؤية الحاسوبية

Open-Source Computer Vision Library



الشبكات التلافيفية

Convolutional Neural Networks



صورة بتدرج الرمادي Grayscale Image

مصادر التعلم :

| | | |
|--|--|---|
|  <p>الوحدة ١ - بايثون</p> | <p>الوحدة الأولى : برمجة بايثون Python</p> |  |
|  <p>١- أوراق العمل</p> | | |
|  <p>الوحدة ٢ - بايثون</p> | <p>الوحدة الثانية: الذكاء الاصطناعي Artificial Intelligence</p> |  |
|  <p>٢- أوراق العمل</p> | | |
|  <p>الوحدة ٣ - بايثون</p> | <p>الوحدة الثالثة : المنتجات الرقمية</p> |  |
|  <p>٣- أوراق العمل</p> | | |
|  <p>10 - part 2</p> | <p>قناة التوجيه العام للحاسوب مشغل فيديوهات الصف العاشر الفصل الدراسي الثاني</p> |  |

الخاتمة

نختتم هذا الدليل بتفاؤل وامل في ان يكون له الأثر الإيجابي على رحلتكم التعليمية في تدريس منهج الصف العاشر وأن يكون معيناً للمعلم أثناء تقديم المحتوى العلمي , فقد تم تصميم هذا الدليل لمساعدة المعلم في تقديم المادة العلمية من خلال فيديوات معدة وجاهزة لاستخدام المعلم يمكن أن يستعين بها لتقديم المحتوى العلمي وعرضه للمتعلمين كوسيلة مساعدة أثناء الشرح , وهي غير ملزمة له , ويترك للمعلم حرية الاختيار في اعداد وتصميم ما يناسبه من فيديوات

وندعو جميع المعلمين للاستفادة القصوى من الموارد والمعلومات المقدمة في هذا الدليل بهدف اثراء مهاراتهم وتعزيز معلوماتهم وخبراتهم , ونتطلع بشغف الى رؤية تفاعلاتهم ومساهماتهم في هذا المجال .

نحن ملتزمون بدعمكم في رحلتكم التعليمية التي ستستمر بفضل جهودكم المخلصة في تقديم الأفضل لمتعلميكم وحرصكم على استكشاف وتطوير مهاراتهم والاطلاع على كل ما هو جديد في مجال التكنولوجيا والبرمجة , حيث تحتم علينا ظروف العالم المتسارعة والتطور السريع في مجال مادة تقنية المعلومات التي ضرورة مواكبة كل ما هو جديد على الساحة .

نتطلع الى رؤية نتائجكم المذهلة في مجال البرمجة متمثلة في نتاجات المتعلمين من مشاريع جيدة تعكس مدى استفادة المتعلمين من معلومات أثناء الفصل الدراسي , وندعوكم الى المشاركة النشطة في المسابقات والتحديات التقنية لتطوير المهارات البرمجية لكل من المعلم والمتعلم .

نتمنى لكم كل التوفيق في رحلتكم التعليمية والمهنية ونأمل أن يكون هذا الدليل بمثابة مورد ومرجع ملهم لكم عند الحاجة.

10.2