

PYTHON COURSE

Python course , from scratch
to professionalism ...

Made by : Taha Khaldoun Amro

dictionaries

في هذه الدروس , سنتحدث عن احدى أهم طرق تخزين البيانات , ألا و هي القواميس (dictionaries) .
يعبر مفهوم القواميس عن شكل من أشكال تخزين البيانات على شكل مفتاح و قيمة (key : value) , حيث يمكن أن يكون الـ key نوعاً من أنواع البيانات التي لا يمكن تغييرها (strings , integers , tuples) و يمكن ان تكون الـ value اي نوع من أنواع البيانات .

يعتبر أقرب مثال لهذه القواميس قواعد البيانات , حيث يمكن أن نخلق جدولاً لكل مستخدم , ولكل مستخدم هناك :

1. اسم

2. رقم هوية

3. عمر

4. مواهب

و الى اخره , حيث تكون هذه الخصائص هي المفاتيح و يمكن أن تحتل أي نوع من البيانات كقيمة , ولكن يجب الحذر من أننا لا يمكننا أن نكرر المفتاح مرتين , والا فان النظام سيعتبر قيمة المفتاح اخر ما تم تعيينه في ترتيب الكود .

في النهاية , تعتبر عناصر القواميس غير مرتبة , أي أنه لا يمكن الوصول اليها من خلال رقم موقعها :

(no indexing or slicing)

دورة تعليم لغة بايثون

dictionaries

انشاء ال dictionary و طباعتها :

```
# dicts items are enclosed by curly braces {}
user = {
    'name' : 'Taha',
    'age' : 17 ,
    'country': 'palestine' ,
    'talents': ['drawing' , 'programming' , 'eating'] ,
    'rating': 7.5
}

print(user)
print(type(user)) #dict => dictionary
```

الان و بكل بساطة , نعرف أن البيانات توضع بين {} في القواميس , ولكن , ماذا لو اردنا الحصول على معلومة معينة , مثل عمر المستخدم أو تقييمه أو أيّاً كان ؟

dictionaries

كنا قد ذكرنا سابقا أننا لا نستطيع استخدام المواقع للوصول للبيانات داخل القاموس, حيث أن القواميس غير مرتبة , و لهذا يوجد مفتاح لكل قيمة , حيث يمكننا المفتاح من الوصول الى أي قيمة نريدها .

```
user = {  
    'name': 'Taha',  
    'age': 17,  
    'country': 'palestine',  
    'talents': ['drawing', 'programming', 'eating'],  
    'rating': 7.5  
}  
  
print(user['talents'])  
print(user.get('name'))
```

من أجل الوصول الى أي قيمة في القاموس , نستخدم مفتاحها اما من خلال وضع اسم المفتاح بين الاقواس المربعة [], أو من خلال استخدام تقنية get التي تطلب منا اسم المفتاح .

دورة تعليم لغة بايثون

dictionaries

الوصول الى القيم أو المفاتيح فقط :

```
user = {  
    'name' : 'Taha',  
    'age' : 17,  
    'country': 'palestine',  
    'talents': ['drawing', 'programming', 'eating'],  
    'rating': 7.5  
}  
  
print(user.keys())  
print(user.values())
```

تمكننا تقنيات الـ keys و الـ values من الوصول الى كل المفاتيح , أو كل القيم الموجودة داخل القاموس .

2D dictionaries

سنتكلم الان عن القواميس ثنائية الابعاد , حيث أن مصطلح “كائن ثنائي الابعاد” في البرمجة , يعني أن نوع البيانات يحمل نفسه في داخله كعنصر , مثل أن تكون احدى القيم قاموساً اخر , يوجد بداخله مفاتيح و عناصر أخرى .

```
languages = {  
    'one': {  
        'name': 'Html',  
        'progress': 70,  
    },  
    'two': {  
        'name': 'css',  
        'progress': 50,  
    },  
}  
print(languages)
```

دورة تعليم لغة بايثون

2D dictionaries

عندما قمنا بطباعة القاموس السابق , تبين لنا أن كل العناصر داخله عبارة عن قواميس , بناء على الكود السابق , اصف الاكواد الاتية , و اكتشف ما فائدة كل منها .

```
print(languages['two'])
```

```
print(languages['one']['name'])
```

```
print(len(languages))
```

و الان كشكل من اشكال التدريب , جرب ان تعرف كل مفتاح على انه متغير (قاموس خارجي) و ضع القيمة على انها هذا المتغير .

الان سننتقل الى تقنيات القواميس .

dictionaries

1. clear()

تعمل هذه التقنية على حذف كل العناصر داخل القاموس .

```
user = {  
    'name' : 'Ahmed'  
}  
print(user.clear())
```

2. copy()

تعمل هذه التقنية مثل عملها في الدروس السابقة حيث تأخذ نسخة من القاموس لا تتغير مهما تغيرت الأساسية .

```
user = {  
    'name' : 'Ahmed'  
}  
print(user.copy())  
print(user)
```


dictionaries

3. update()

عندما تريد أن تقوم بإضافة مفتاح جديد مع قيمته الى قاموس معين , فانك بكل بساطة تستخدم احدى الطريقتين الاتيتين.

```
user = {  
    'name' : 'Ahmed'  
}
```

```
print(user)
```

```
user['age'] = 20 # age is the key and it equals to the value 20
```

```
user.update({'rating' : 10}) # the update method takes a dictionary value
```

```
print(user)
```

dictionaries

يمكنك التعرف على المزيد عن الdictionaries من خلال الروابط الآتية :

https://www.youtube.com/watch?v=BQ7jFrysbQU&list=PLDoPjvoNmBAyE_gei5d18qkfle-Z8mocs&index=30&pp=iAQB

https://www.youtube.com/watch?v=oNLaNJrU8r8&list=PLDoPjvoNmBAyE_gei5d18qkfle-Z8mocs&index=31&pp=iAQB

https://www.youtube.com/watch?v=rgOdxQa830Q&list=PLDoPjvoNmBAyE_gei5d18qkfle-Z8mocs&index=32&pp=iAQB