

PYTHON COURSE

Python course , from scratch
to Professionalism ...

Made by : Taha Khaldoun Amro 

تنسيق النصوص

لنفترض أنك قد بنيت موقعًا ، يقوم المستخدم بتحديد اسمه ، عمره ، جنسه وأمور أخرى ، حسنا ، أنت تريد أن يقوم النظام بطباعة بيانات هذا الشخص على شكل رسالة : أهلا أنا اسمي "س" و عمري "ص" سنين ... وهكذا .

بطبيعة الحال ، لابد من أن البيانات قد خزنت في متغيرات ، حسناً ، تسمى عملية طباعة متغيرات داخل نص من خلال جمع النصوص (كما تعلمنا سابقاً) concatenation ، هذه الطريقة ليست فعالة ، لأننا عندما نريد طباعة متغير يحمل قيمة رقمية (int) نحصل على error يخبرنا أننا لا يمكننا جمع النصوص والأرقام .
الحل يكمن في استخدام مفهوم string formatting الذي يمكننا من وضع المتغيرات في وسط الكلام من دون الحصول على error .

في البداية ، يوجد عدة طرق للقيام بالعملية سنتعرف عليها جميعها ، ولكننا سنركز على الجديدة منها ، ولكن هذا لا يعني إهمال القديمات من الطرق ، فحتى تقول عن نفسك مبرمجة يجب أن تكون ملماً بكل التفاصيل .

دورة تعليم لغة بايثون

تنسيق النصوص

سنتعرف الان إلى الطريقة القديمة (هذه الطريقة فقط لأغراض المعرفة و العلم بها في حال واجهتها)

```
name = 'Taha'  
age = 16  
lang = 'Python'  
exp = 3.3
```

```
print('Hi , my name is %s , and im %d years old , im a %s programmer , with %f years of experience' % (name , age , lang , exp))
```

حسنا ، يمكننا رؤية علامات النسبة المئوية ، والتي تعمل على إخبار النظام بوجود قيمة معينة ستأتي هنا ، حيث سيتم تحديدها في الأقواس ، خارج النص الذي سيطبع بعد وضع علامة % ، والأحرف التي تأتي بعدها لها دلالات :

%s --> string

%d --> int

%f --> float

و يمكننا التحكم في عدد المنازل التي تكتب بعد الفاصلة في ال f% بوضع نقطة بعد العلامة و من ثم عدد المنازل التي نريد بالصيغة الآتية على سبيل المثال : (2f.%)

دورة تحليم لغة بايثون

تنسيق النصوص

في التحديات الأخيرة للغة بايثون ، سئم الناس من طريقة تنسيق النصوص القديمة ، حيث أنها أصبحت صعبة و عديمة الفائدة مقارنة باللغات الأخرى في العالم ، لهذا السبب ، قام مطورو اللغة بتطوير طريقتين جديدين فعالتين بشكل أكبر في تنسيق النصوص.

قد تتشابه الطريقة الثانية مع الطريقة القديمة إلا أنها قد أتاحت العديد من الأمور التي كانت غير ممكنة في التحديات السابقة .

الآن سوف نتكلم عن هاتين الطريقتين و نغطي كل شيء فيما بإذن الله.

و لأن هذه الدروس تعتبر مهمة جداً ، وقد تفرق ما بين الخبر و اللعوب في هذه اللغة ، سنرفق لكم رابط شرح الطرق القديمة و الجديدة :

القديمة : https://youtu.be/m_OUIywn_XE?si=MEBA0qclJP37HQs4

الجديدة : <https://youtu.be/nm4qN90A7X4?si=hWMpOwaEkQWBQ9nt>

دورة تعليم لغة بايثون

تنسيق النصوص

في هذه الطريقة ، سيكون الأمر مشابهاً إلى حد ما للطريقة القديمة ، إلا أنها ستعود لموضوع التقنيات ، حيث تعتبر طريقة التنسيق هذه إحدى تقنيات النصوص.

تعمل هذه التقنية على أخذ نص ، في داخل النص ، يتم وضع علامة {} في كل مكان سنضع فيه متغيراً ، و بعد النص ، يمكننا وضع التقنية مع إدخال المتغيرات فيها.

```
name = 'Taha'  
lang = 'Python'  
exp = 3.3  
  
print('Hi , my name is {} , im a {} programmer , with {} years of experience'.format(name , lang , exp))
```

بسهولة أليس كذلك ؟
حسنا ، السؤال هنا ، ما الذي أضافته لنا هذه الخاصية ، في البداية ، لنتعلم ماذا يمكن أن نضع داخل الأقواس المعقوفة.

تنسيق النصوص

1. يمكننا تحديد نوع القيمة :

```
print('Hi , my name is {:s} , im a {:s} programmer , with {:f} years of experience'.format(name , lang , exp))
```

نلاحظ أن الموضوع لم يتغير و ذلك لأن التقنية الجديدة قادرة على تحديد نوع العنصر تلقائيا .

2. يمكننا تحديد عدد المنازل المراد عرضها من العنصر :(float and strings)

```
print('Hi , my name is {:s} , im a {:.2s} programmer , with {:.2f} years of experience'.format(name , lang , exp))
```

3. يمكننا فصل منازل الأرقام الكبيرة ، لتسهيل قرائتها :

```
money = 2219329048
```

```
print('my bank cash is : {:.} '.format(money))
```

تنسيق النصوص

4. يمكننا ترتيب عرض المتغيرات :

```
name = 'Taha'  
lang = 'Python'  
exp = 3.3  
# use the index of each element in the brackets  
print('name : {2} , language : {1} , experience : {0}'.format(exp , lang , name))
```

و هكذا نكون قد وصلنا إلى نهاية الطريقة الجديدة ، نلاحظ أنها ليست بعيدة عن القديمة إلا أنها شكلت خطوة هائلة في هذا المجال.

تنسيق النصوص

هنا سنتحدث عن الطريقة الجديدة و هي أسهل طريقة .
في أول النص , نضع حرف f و من ثم في كل مكان نريد أن نقوم فيه بذكر متغير , نضع اسم المتغير في أقواس معقوفة { }

```
name = 'Taha'  
languages = ['python' , 'css' , 'html']  
exp = 3.3  
age = 16  
  
print(f'Hi , my name is {name} , im {age} years old')  
print(f'I can program good websites using {languages[0]} , {languages[1]} , {languages[2]}' )  
print(f'I have {exp} years of experience .')
```

حاليا , الطريقة الجديدة هي الأكثر شيوعاً و استخداماً في أوساط المبرمجين , إلا أننا يجب أن نعطي اهتماماً إلى الطرق الأخرى , فقد يمر علينا كود يستعملها أو قد تسأل عنها في مقابلة عمل , لذلك يجب معرفتها رغم أننا قد لا نستخدمها.

PYTHON COURSE

Python course , from scratch
to professionalism ...

Made by : Taha Khaldoun Amro 