

3/30/2020

التعبير المنتظمة في البايثون 3

Regular Expression in python 3

الاستاذ: حمزة جميل

العنوان/ العراق - النجف

قناتي على اليوتيوب /

<https://www.youtube.com/channel/UCz6zmCSR3nZEs-0elgW507Q>

التعبير المنتظم (Regular Expression): هو سلسلة من الحروف والرموز تكوّن نمط معين. ويستخدم للتحقق من اذا كان النص يحتوي على نمط خاص يتم البحث عنه.

لاستخدام التعبيرات المنتظمة في البايثون يجب استدعاء re module في بداية البرنامج كما يلي:

Import re

مثال يتم البحث عن نص يبدأ ب (I) وينتهي (language).

```
import re
str="I like python language"
x=re.search("^I.*language$",str)
if (x):print("Match")
else:print("Not match")
```

output\
match

دوال التعبيرات المنتظمة [Regex Functions]:

تحتوي وحدة re على عدة دوال تستخدم للبحث عن النص المطلوب كما في الجدول الاتي:

الوصف (Description)	الدالة (Function)
Returns a list containing all matches تعيد جميع القيم المتشابهة	Findall
Returns a Match object if there is a match anywhere in the string تعيد موقع القيم المتشابهة في النص	Search
Returns a list where the string has been split at each match تقسيم النص بالاعتماد على رمز محدد	Split
Replace one or many matches in the string. استبدال قيم من النص باخرى محددة	Sub

الرموز التعبيرات المنتظمة:

هي رموز تستخدم مع التعبير المنتظم

الرمز Character	الوصف Description	مثال Example
[]	Set of character مجموعة من الرموز	[a-n]
\	It represents the special sequence. يمثل سلسلة خاصة	“\d”
.	Any character اي رمز	“py.o.”
^	It represents begin of the string. يمثل بداية النص	“^python”
\$	Ends string	“hello\$”
*	Zero or more occurrences يسمح برمز واحد او اكثر او لا شيء	“World*”
+	One or more occurrence. يسمح برمز واحد او اكثر	“hero+”
{ }	Specified number of occurrences of pattern the string يحدد عدد مرات تكرار نسق معين في النص	“hi{2}”
	Either or اما او	“c c++”
()	Capture and group يسمح بمجموعة	

سلسلة خاصة [special sequence]:

يستخدم \ متبوع باحد الرموز في الجدول ادناه لتصبح ذات دلالة في التعبيرات المنتظمة.

الرمز Character	الوصف Description	مثال Example
\A	specified characters are at the beginning of the string. تحديد رمز او نص تبدأ به السلسلة النصية	“\Athe”
\b	specified characters are at the beginning or at the end of a word تحديد رمز تبدأ به او تنتهي به الكلمة	r“\bin” Or r”in\b”
\d	It allows numbers (0... 9) يسمح بالارقام فقط	“\d”

\s	the string contains a white space character. يستخدم للمسافة	"\s"
\w	the string contains any word characters (characters from a to Z, digits from 0-9, and the underscore _ character). يسمح بالارقام والاحرف الكبير والصغيرة	"\w"

دالة البحث findall() function

تُعيد قائمة تحتوي على كل الحروف او الكلمات المتطابقة.
مثال/ يعيد قائمة خاصة بظهور الحرفين "ai" في كل مرة.

```
import re
str = "The rain in Spain"
x = re.findall("ai", str)
print(x)
output/ ['ai', 'ai']
```

ملاحظة: لطباعة الكلمات كاملة يكون التعبير المنتظم بالشكل الاتي:

```
x = re.findall(".ai.", str)
output\ ['rain', 'spain']
```

للاطلاع على القوائم في البايثون اليك الرابط الاتي:

<https://www.youtube.com/watch?v=FGbcBsVG7us>

مثال اخر يعيد كلمة من سلسلة نصية اذا تطابق التعبير

```
import re
str="I like python language, life is funny"
x=re.findall("\w+e",str)
print(x)
```

```
output\ ['like', 'language', 'life']
```

اذا لم يحصل التطابق فان الدالة تعيد اقواس فارغة. مثال/

```
import re
str="I like python language, life is funny"
x=re.findall("live",str)
print(x)
```

```
if(x):print("Match")
else:print("No match")
```

output\ [], no match

دالة البحث search() function

تستخدم هذه الدالة في البحث عن نص للتطابق، وتعيد الجزء المتطابق من النص. اذا كان هناك اكثر من تطابق فان الدالة تعيد التطابق الاول فقط.

مثال/ البحث عن كلمة تنتهي بحرف e.

```
import re
txt = "I like python language"
x = re.search("\w+e", txt)
```

```
print(x)
```

output\ <re.Match object; span=(2, 6), match='like'>

اذا لم يحصل تطابق في النص تكون النتيجة none كما في المثال الاتي:

```
import re
txt = "I like python language"
x = re.search("html", txt)
```

```
print(x)
```

دالة الفصل split() function :

تعيد قائمة من للانماط المتطابقة

مثال/ تقسم النص بالاعتماد على الفراغات الموجودة بين الكلمات

```
import re
txt = "How are you"
x = re.split("\s", txt)
```

```
print(x)
```

output\ ['How', 'are', 'you']

ملاحظة: يمكن التحكم في شرط تقسيم النص كما يأتي:

مثال/ فصل الكلمة الاولى من النص

```
import re
txt = "Regular expression in python"
x = re.split("\s", txt, 1)
print(x)
```

Output \ ['Regular', 'expression in python']

دالة الابدال function sub():

تستخدم لابدال رمز باخر حسب التعبير المطلوب
مثال/ ابدال الفراغات بين الكلمات بالرمز _

```
import re
txt = "Regular expression in python"
x = re.sub("\s", "_", txt)
print(x)
```

output \ Regular_expression_in_python

يمكن التحكم في عدد معين من شرط التطابق كما في المثال الاتي:

```
import re
txt = "Regular expression in python"
x = re.sub("\s", "#", txt)
print(x)
```

output \ Regular#expression#in python

هناك المزيد من الدوال التي تستخدم مع search في ارجاع وطباعة القيمة المطلوبة
في التعبير المنطقي راجع الروابط الاتية:

https://www.youtube.com/watch?v=5r3_RI5j4PI

<https://www.youtube.com/watch?v=vmkOeiwGvUk>

مثال على دوال span(), string, group()

<https://www.youtube.com/watch?v=yU7NGCY6sRo>