

字符串

1.1 实验介绍

字符串是 Python 中最常用的数据类型。

1.2 实验知识点

- 字符串的3种表示
- 字符串的分割、连接、大小写转换、搜索等常用操作

1.3 实验环境

- python3.5
- Xfce终端
- Vim

1.4 适合人群

本课程属于初级级别课程，不仅适用于那些有其它语言基础的同学，对没有编程经验的同学也非常友好。

二、实验步骤

可以通过几种不同的方式表示字符串。如单引号（`'...'`）或双引号（`"..."`）。下面的例子能帮助你更好的理解字符串。

```
>>> s = "I am Chinese"
>>> s
'I am Chinese'
>>> s = 'I am Chinese'
>>> s = "Here is a line \
... split in two lines"
>>> s
'Here is a line split in two lines'
>>> s = "Here is a line \n split in two lines"
>>> s
'Here is a line \n split in two lines'
>>> print(s)
Here is a line
split in two lines
```

如果你想要分几行输入字符串，并且希望行尾的换行符自动包含到字符串当中，可以使用三对引号：`"""..."""` 或 `'''...'''`。

```
>>> print("""\
... Usage: thingy [OPTIONS]
...     -h                        Display this usage message
...     -H hostname              Hostname to connect to
... """)
Usage: thingy [OPTIONS]
    -h                        Display this usage message
    -H hostname              Hostname to connect to
```

2.1 字符串的方法

每一个字符串对象都有几个可用的内建方法，我们已经使用过一些了，比如 `s.split()`。

```
>>> s = "shi yan lou"
>>> s.title()
'Shi Yan Lou'
```

方法 `title()` 返回字符串的标题版本，即单词首字母大写其余字母小写。

```
>>> z = s.upper()
>>> z
'SHI YAN LOU'
>>> z.lower()
'shi yan lou'
```

方法 `upper()` 返回字符串全部大写的版本，反之 `lower()` 返回字符串的全部小写版本。

```
>>> s = "I am A pRoGraMMer"
>> s.swapcase()
'i AM a PrOgRAmmER'
```

方法 `swapcase()` 返回字符串大小写交换后的版本：)

```
>>> s = "jdwb 2323bjb"
>>> s.isalnum()
False
>>> s = "jdwb2323bjb"
>>> s.isalnum()
True
```

方法 `isalnum()` 检查所有字符是否为字母数字，上面的代码中第一行的字符串 `s` 中包含空格字符，所以返回 `False`。

```
>>> s = "SankarshanSir"
>>> s.isalpha()
True
>>> s = "Sankarshan Sir"
>>> s.isalpha()
False
```

方法 `isalpha()` 检查字符串之中是否只有字母。

```
>>> s = "1234"
>>> s.isdigit() # 检查字符串是否所有字符为数字
True
>>> s = "ShiYanLou is coming"
>>> s.islower() # 检查字符串是否所有字符为小写
False
>>> s = "Shiyanlou Is Coming"
>>> s.istitle() # To 检查字符串是否为标题样式
True
>>> s = "CHINA"
>>> s.isupper() # 检查字符串是否所有字符为大写
True
```

我们可以使用 `split()` 分割任意字符串，`split()` 允许有一个参数，用来指定字符串以什么字符分隔（默认为 " "），它返回一个包含所有分割后的字符串的列表。

```
>>> s = "We all love Python"
>>> s.split()
['We', 'all', 'love', 'Python']
>>> x = "shiyanlou:is:waiting"
>>> x.split(':')
['shiyanlou', 'is', 'waiting']
```

相反的，方法 `join()` 使用指定字符连接多个字符串，它需要一个包含字符串元素的列表作为输入然后连接列表内的字符串元素。

```
>>> "-".join("GNU/Linux is great".split())
'GNU/Linux-is-great'
```

在上面的例子中，我们基于空格 " " 分割字符串 "GNU/Linux is great"，然后用 "-" 连接它们。

2.2 字符串剥离

字符串有几个进行剥离操作的方法。最简单的一个是 `strip(chars)`，用来剥离字符串首尾中指定的字符，它允许有一个字符串参数，这个参数为剥离哪些字符提供依据。不指定参数则默认剥离掉首尾的空格和换行符，代码如下：

```
>>> s = "  a bc\n "  
>>> s.strip()  
'a bc'
```

你可以使用 `rstrip(chars)` 或 `rstrip(chars)` 只对字符串左或右剥离。

```
>>> s = "www.foss.in"  
>>> s.lstrip("cwsd.") #删除在字符串左边出现的'c','w','s','d','.'字符  
'foss.in'  
>>> s.rstrip("cnwdi.") #删除在字符串右边出现的'c','n','w','d','i','.'字符  
'www.foss'
```

2.3 文本搜索

字符串有一些方法能够帮助你搜索字符串里的文本或子字符串。下面给出示例：

```
>>> s = "faulty for a reason"  
>>> s.find("for")  
7  
>>> s.find("fora")  
-1  
>>> s.startswith("fa") # 检查字符串是否以 fa 开头  
True  
>>> s.endswith("reason") # 检查字符串是否以 reason 结尾  
True
```

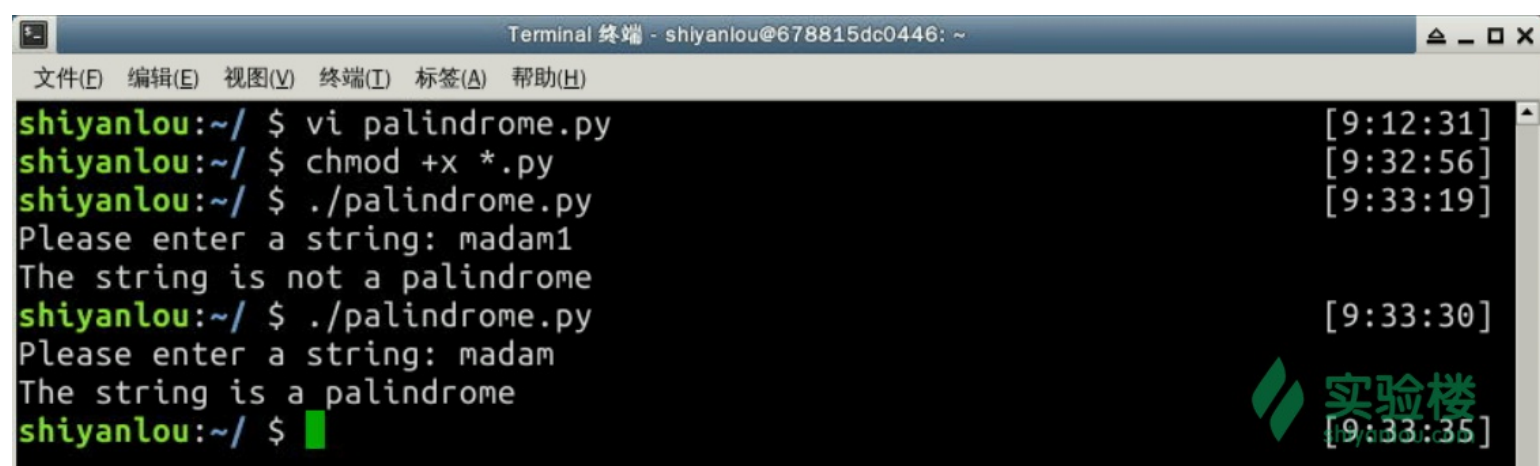
`find()` 能帮助你找到第一个匹配的子字符串，没有找到则返回 -1。

2.4 回文检查

回文是一种无论从左还是从右读都一样的字符序列。比如 “madam”。在这个例子中，我们检查用户输入的字符串是否是回文，并输出结果。

```
#!/usr/bin/env python3
s = input("Please enter a string: ")
z = s[::-1]
if s == z:
    print("The string is a palindrome")
else:
    print("The string is not a palindrome")
```

运行程序：



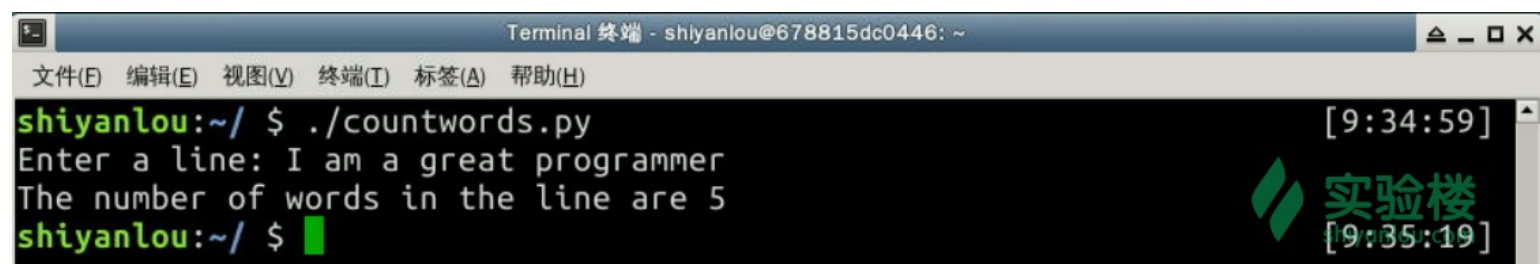
```
Terminal 终端 - shiyanlou@678815dc0446: ~
文件(E) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)
shiyanlou:~/ $ vi palindrome.py [9:12:31]
shiyanlou:~/ $ chmod +x *.py [9:32:56]
shiyanlou:~/ $ ./palindrome.py [9:33:19]
Please enter a string: madam1
The string is not a palindrome
shiyanlou:~/ $ ./palindrome.py [9:33:30]
Please enter a string: madam
The string is a palindrome
shiyanlou:~/ $ [9:33:35]
```

2.5 单词计数

在这个例子中我们对用户输入的一行文本进行单词计数。

```
#!/usr/bin/env python3
s = input("Enter a line: ")
print("The number of words in the line are %d" % (len(s.split(" "))))
```

运行程序：



```
Terminal 终端 - shiyanlou@678815dc0446: ~
文件(E) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)
shiyanlou:~/ $ ./countwords.py [9:34:59]
Enter a line: I am a great programmer
The number of words in the line are 5
shiyanlou:~/ $ [9:35:19]
```

三、总结

本实验学习了字符串的几种表示方法（3种引号）以及对字符串的各种常用操作（分割、连接、大小写转换、搜索...），应当熟练掌握它们。

**本课程内容，由作者授权实验楼发布，未经允许，禁止转载、下载及非法传播。*

上一节：[数据结构 \(/courses/596/labs/2041/document\)](/courses/596/labs/2041/document)

下一节：[挑战：字符串操作 \(/courses/596/labs/2773/document\)](/courses/596/labs/2773/document)