

Contents

Python for Data Analysis 101	1
Homework	1
Instructor: Evelyn J. Boettcher, DiDacTex, LLC	1
Week 1: Lecture 2	1
1 Lists	1
2 Dictionaries	1
3 Strings	1
4 Strings Multiline	2
5 join	2
6 String Format.	2
Solutions	2

Python for Data Analysis 101

Homework

Instructor: Evelyn J. Boettcher, DiDacTex, LLC

Week 1: Lecture 2

1 Lists

- Create an empty list. Append 4 strings to the list. Then pop one item off the end of the list.

2 Dictionaries

- Create a dictionary using with a zip and two lists
- Add to this dictionary using the key “HW2” with value “Done”
- Define a dictionary using both string literals and variables containing strings.

3 Strings

- Use a literal to create a string containing:
 - a single quote,
 - a double quote,

- both a single and double quote.

4 Strings Multiline

- Write a string literal that spans multiple lines.

5 join

- Use the string `join` operation to create a string that contains a colon as a separator.

6 String Format.

- Use string formatting to produce a string containing your last and first names, separated by a comma.

Solutions

Solution ##1 :

```
In [25]: a = []
In [26]: a.append('aaa')
In [27]: a.append('bbb')
In [28]: a.append('ccc')
In [29]: a.append('ddd')
In [30]: print a
['aaa', 'bbb', 'ccc', 'ddd']
In [31]: a.pop()
Out[31]: 'ddd'
```

Solution ##2 :

```
first = 'Dave'
last = 'Kuhlman'
name_dict = {first: last, 'Elvis': 'Presley'}
print( name_dict)
#      {'Dave': 'Kuhlman', 'Elvis': 'Presley'}
```

Solution ##3 :

```
"Some 'quoted' text."  
'Some "quoted" text.'  
'Some "quoted" \'extra\' text.'
```

Solution ## 4: - Write a string literal that spans multiple lines. Solution:

```
"This string\  
spans several lines\  
because it is a little long.\  
"
```

Solution ##5 - Use the string `join` operation to create a string that contains a colon as a separator.

Solution:

```
content = []  
content.append('finch')  
content.append('sparrow')  
content.append('thrush')  
content.append('jay')  
new_str = ':'.join(content)  
print( new_str)  
#      finch:sparrow:thrush:jay
```

Solution 6

- Use string formatting to produce a string containing your last and first names, separated by a comma.

Solution:

```
first = 'Dave'  
last = 'Kuhlman'  
full = '%s, %s' % (last, first, )  
print( full)  
#      Kuhlman, Dave
```
