Pierian-Data PYTHON 3 UPDATES

1 contributor

528 lines (528 sloc) | 10.3 KB ...

Objects and Data Structures Assessment Test

Test your knowledge.

** Answer the following questions **

Write a brief description of all the following Object Types and Data Structures we've learned about:

Numbers:

Strings:

Lists:

Tuples:

Dictionaries:

Numbers

Write an equation that uses multiplication, division, an exponent, addition, and subtraction that is equal to 100.25.

Hint: This is just to test your memory of the basic arithmetic commands, work backwards from 100.25

In [ ]:

Answer these 3 questions without typing code. Then type code to check your answer.

```
What is the value of the expression 4 * (6 + 5)

What is the value of the expression 4 * 6 + 5

What is the value of the expression 4 + 6 * 5
```

In [ ]:

What is the *type* of the result of the expression 3 + 1.5 + 4?

What would you use to find a number's square root, as well as its square?

In [ ]:

```
# Square root:
```

In [ ]:
```
# Square:
```

Strings

Given the string 'hello' give an index command that returns 'e'. Enter your code in the cell below:

In [ ]:
```
s = 'hello'
# Print out 'e' using indexing
```

Reverse the string 'hello' using slicing:

In [ ]:
```
s ='hello'
# Reverse the string using slicing
```

Given the string hello, give two methods of producing the letter 'o' using indexing.

In [ ]:
```
s ='hello'
# Print out the 'o'

# Method 1:
```

In [ ]:
```
# Method 2:
```

Lists

Build this list [0,0,0] two separate ways.

In [ ]:
```
# Method 1:
```

In [ ]:
```
# Method 2:
```

Reassign 'hello' in this nested list to say 'goodbye' instead:

In [ ]:
```
list3 = [1,2,[3,4,'hello']]
```

Sort the list below:

In [ ]:
```
list4 = [5,3,4,6,1]
```

Dictionaries

Using keys and indexing, grab the 'hello' from the following dictionaries:

In [ ]:
```
d = {'simple_key':'hello'}
# Grab 'hello'
```

In [ ]:
```
d = {'k1':{'k2':'hello'}}
# Grab 'hello'
```

In [ ]:
```
# Getting a little tricker
d = {'k1':[{'nest_key':['this is deep',['hello']]}]}

#Grab hello
```

In [ ]:
```
# This will be hard and annoying!
d = {'k1':[1,2,{'k2':['this is tricky',{'tough':[1,2,['hello']]}]}]}
```

Can you sort a dictionary? Why or why not?

Tuples

What is the major difference between tuples and lists?

How do you create a tuple?

Sets

What is unique about a set?

Use a set to find the unique values of the list below:

In [ ]:
```
list5 = [1,2,2,33,4,4,11,22,3,3,2]
```

Booleans

For the following quiz questions, we will get a preview of comparison operators.
In the table below, a=3 and b=4.

Operator	Description	Example
==	If the values of two operands are equal, then the condition becomes true.	(a == b) is not true.
!=	If values of two operands are not equal, then condition becomes true.	(a != b) is true.

	then condition becomes true.	
>	If the value of left operand is greater than the value of right operand, then condition becomes true.	(a > b) is not true.
<	If the value of left operand is less than the value of right operand, then condition becomes true.	(a < b) is true.
>=	If the value of left operand is greater than or equal to the value of right operand, then condition becomes true.	(a >= b) is not true.
<=	If the value of left operand is less than or equal to the value of right operand, then condition becomes true.	(a <= b) is true.

What will be the resulting Boolean of the following pieces of code (answer fist then check by typing it in!)

In [ ]:
```
# Answer before running cell
2 > 3
```

In [ ]:
```
# Answer before running cell
3 <= 2
```

In [ ]:
```
# Answer before running cell
3 == 2.0
```

In [ ]:
```
# Answer before running cell
3.0 == 3
```