Lists and Tuples

Exercises

۱۸	10	0	k	6
v v			r v -	

Prior to attempting these exercises ensure you have read the lecture notes and/or viewed the video, and followed the practical. You may wish to use the Python interpreter in interactive mode to help work out the solutions to some of the questions.

Download and store this document within your own filespace, so the contents can be edited. You will be able to refer to it during the test in Week 7.

Enter your answers directly into the highlighted boxes.

For more information about the module delivery, assessment and feedback please refer to the module within the MyBeckett portal.

©2021 Mark Dixon / Tony Jenkins

Would you describe the following Python statement as a function call? Or a method call?

names.reverse()

Answer:

Method call

Write a Python statement that appends a single element to the end of the specified *List* using a **method** call.

```
prices = [2.65, 7.65, 8.25, 9.56]
```

Answer:

```
append() method
prices = [2.65, 7.65, 8.25, 9.56]
prices.append(10.20)
```

Write another statement that appends three elements to the end of the specified *List* using a single **method** call.

Answer:

```
extend() method

prices.extend([10.20,11.21,12.22])
```

Now write a for loop that iterates over each value in the list and prints it to the screen.

Answer:

```
prices = [2.65, 7.65, 8.25, 9.56]
for I in prices:
    print(I)
```

Is a method that changes the contents of the associated value referred to as a **mutator**? Or an **accessor**?

Answer:

Mutator

What would the contents of the primes list look like after execution of the following statements?

```
primes = [ 2, 3, 5, 7, 11, 13, 17, 19 ]
primes.pop()
```

Answer:

```
primes = [ 2, 3, 5, 7, 11, 13, 17, 19 ]
primes.pop()
19
```

primes.reverse()

Answer:

```
[17,13,11,7,5,3,2]
```

primes.remove(7)

Answer:

```
[2,3,5,11,13,17]
```

Provide an example of how the insert() method could be used to add a value of 10 to the beginning of the list shown below.

```
temps = [32, 46, 95, 10, 50]
```

Answer:

```
temps = [32, 46, 95, 10, 50]
temps.insert(0, 10)
print(temps)
```

Now write a statement that uses an *accessor* method to find the index of the value 95 within the list.

Answer:

```
index() method

temps = [32, 46, 95, 10, 50]
index_of_95 = temps.index(95)

index_of_95
2
```

Finally write a statement that uses another *accessor* method to count how many times the number 10 appears within the list.

Answer:

```
temps = [32, 46, 95, 10, 50]
count_of_10 = temps.count(10)
count_of_10
1
```

What would be stored in the list samples after the following statements were executed?

```
samples = [ 100.2, 100.6, 99.2, 765.2, 900.2, 400 ]
samples = samples.reverse()
```

Answer:

error

Explain why this is the case.

Answer:

Because the reverse() method does not return a value that can be assigned to the samples variable.

Write a Python program that uses a **List-Comprehension** to produce the same list as the following code -

```
values = []
for n in range(100,200):
     values.append(x*x)
```

Answer:

```
values = [n*n for n in range(100, 200)]
values
```

Now, amend your code so that it only includes even numbers.

Answer:

```
even_values = [n for n in range(100, 200) if n % 2 == 0]
even_values
```

What is the data-type of the following value?

```
info = ("Ken", "bae-192", 62)
```

Answer:

Tuple

Is a Tuple **mutable** or **immutable**?

Answer:

Immutable

Write a statement that creates a Tuple that contains a single element.

Answer:

my_tuple=(13,)

Write a single Python statement that **unpacks** the following Tuple into three variables, called x, y and z.

```
coord = (100, 200, 150)
```

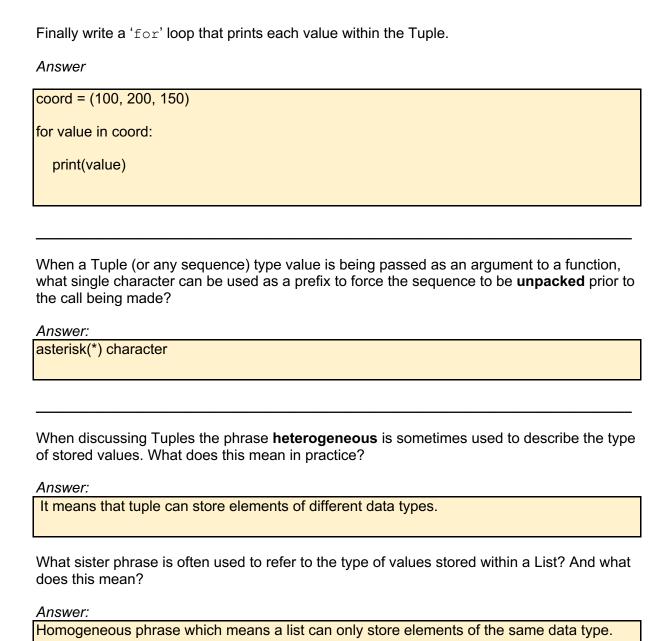
Answer:

x, y, z = coord

Write another statement that uses indexing to access the second element of the Tuple and store it in a variable called 'height'

Answer:

```
coord = (100, 200, 150)
height = coord[1]
height
```



Exercises are completed