

Exercise 5 – Collections

Objective

To understand the use and syntax of containers in Python 3. We'll also compare different ways to access a list.

Questions

1. What's wrong with this?

```
cheese = ['Cheddar', 'Stilton', 'Cornish Yarg']  
cheese += 'Oke'
```

How should 'Oke' be added to the end of the cheese list?

2. What's going on here? Can you explain the output?

```
tup = 'Hello'  
print(len(tup))
```

Prints 5

```
tup = 'Hello',  
print(len(tup))
```

Prints 1

3. Write a Python script called Ex5_3.py that will generate and display six unique random lottery numbers between 1 and 50. Think about which Python data structure is best suited to store the numbers! Use the Python `help()` function to find out which function to use from the python standard library called **random**.

4. We need to do some maintenance on a dictionary of machines:

```
machines = {'user100': 'yogi',
            'user1': 'booboo',
            'user2': 'rupert',
            'user3': 'teddy',
            'user4': 'care',
            'user5': 'winnie',
            'user6': 'sooty',
            'user7': 'padders',
            'user8': 'polar',
            'user9': 'grizzly',
            'user10': 'baloo',
            'user11': 'bungle',
            'user12': 'fozzie',
            'user13': 'huggy',
            'user14': 'barnaby',
            'user15': 'hair',
            'user16': 'greppy'
            }
```

Don't type this in! It should be available for you to edit in Ex5_4.py in the **labs** directory (or your home directory if on Linux).

Without altering the initial definition of the dictionary, write code that will implement the following changes:

- a) user14 no longer has a machine assigned.
- b) The name of user15's machine is changed to 'cinnamon'.
- c) user16 is leaving the company and a new user, user17 will be assigned his machine.

- d) user4, user5, and user6 are all leaving at the same time, but their machine names are to be stored in a list called **unallocated**.

Hint: Pop in a loop.

- e) user8 gets another machine called 'kodiak' in addition to the one they already have.
- f) Print a list of users with their machine in any order. Print each user/machine pair on a separate line.
- g) Print a list of unallocated machines, sorted alphabetically.

Solutions

Question 1

What's wrong with this?

```
cheese = ['Cheddar', 'Stilton', 'Cornish Yarg']  
cheese += 'Oke'
```

If we print the variable cheese we see:

```
['Cheddar', 'Stilton', 'Cornish Yarg', 'O', 'k', 'e']
```

The string 'Oke' is a sequence and using += on a list has broken it down into its constituent parts. It should have been:

```
cheese.append('Oke')
```

Question 2

What's going on here? Can you explain the output?

```
tup = 'Hello'  
print(len(tup))
```

Prints 5

That should be no surprise, 5 is the number of characters in 'Hello'.

```
tup = 'Hello',  
print(len(tup))
```

Prints 1

This is rather more surprising, but did you notice the trailing comma? That extra comma meant that we created a tuple. The **len()** built-in function then reported the number of items in that tuple, which is one.

Question 3

The name of the function to use is `random.randint()` and the Pythonic solution is store the numbers in a set. Objects stored in sets are unique.

```
import random

lotto = set()      # Create an empty set.

while len(lotto) < 6:
    num = random.randint(1, 50)
    lotto.add(num)  # Add new number to set.

print("Lottery numbers = ", lotto)
```

Question 4

There are several solutions, here's one:

```
machines = {'user100': 'yogi',
            'user1': 'booboo',
            'user2': 'rupert',
            'user3': 'teddy',
            'user4': 'care',
            'user5': 'winnie',
            'user6': 'sooty',
            'user7': 'padders',
            'user8': 'polar',
            'user9': 'grizzly',
            'user10': 'baloo',
            'user11': 'bungle',
            'user12': 'fozzie',
            'user13': 'huggy',
            'user14': 'barnaby',
            'user15': 'hair',
```

```
'user16': 'greppy'}
```

a) user14 no longer has a machine assigned.

```
machines['user14'] = None
```

b) The name of user15's machine is changed to 'cinnamon'.

```
machines['user15'] = 'cinnamon'
```

c) user16 is leaving the company, and a new user, user17, will be assigned his machine

```
machines['user17'] = machines['user16']  
del machines['user16']
```

d) user4, user5, and user6 are all leaving at exactly the same time, but their machine names are to be stored in a list called unallocated.

```
unallocated = []  
for user in ('user4', 'user5', 'user6'):  
    unallocated += [machines.pop(user)]
```

e) user8 gets another machine called 'kodiak' in addition to the one they already have.

```
machines['user8'] = [machines['user8'], 'kodiak']
```

f) Print a list of all the users, with their machines, in any order.

```
for kv in machines.items():  
    print(kv)
```

g) Print a list of unallocated machines, sorted alphabetically.

```
print ("Unallocated machines: ", sorted(unallocated))
```