# LC CS Python

## Student Exercise Book

LEAVING CERTIFICATE
COMPUTER SCIENCE

# Section 4

## Lists

**Name: _____**

## Task 1

Read the program carefully and see if you can figure out how it works. Type up the code and run it.

```
1.  # Program to simulate a fruit machine!
2.  import random
3.
4.  # initialise the list of fruits - 5 elements
5.  fruits = ['Strawberry', 'Lemon', 'Orange', 'Raspberry', 'Cherry']
6.
7.  # generate three random numbers between 0 and 4 incl.
8.  selection1 = random.randint(0, 4)
9.  selection2 = random.randint(0, 4)
10. selection3 = random.randint(0, 4)
11
12. # show the results - display the fruits
13. print(fruits[selection1])
14. print(fruits[selection2])
15. print(fruits[selection3])
```

What does:

- Line 5 do? _____
- Lines 8, 9 and 10 do? _____

  _____
- Lines 13, 14 and 15 do? _____

  _____

Can you predict the screen output for the above program? Justify your answer.

_____

_____

***Look up the online documentation for the Python `random` library.***
***What does the `choice` command do?***
***How might `choice` be used in the above program?***

_____

_____

_____

_____

_____

_____

## Task 2

Read through the following program. Do you understand what each line of code does?

```
1.  fruits = ['Strawberry', 'Lemon', 'Orange', 'Raspberry', 'Cherry']
2.  print(fruits)
3.
4.  fruits[0] = 'Apple'
5.  fruit = 'Melon'
6.  fruits[1] = fruit
7.  fruits[2] = 'Raspberry'
8.  fruits[3] = fruits[4]
9.  fruits[4] = 'Pineapple'
10.
11. print(fruits)
```

- Line 1 initialises a list of fruits and line 2 displays the contents of the list
- Lines 4-9, each make separate changes to the individual elements in the list that are 'housed' at the given index
- Line 11 displays the list again

- Use program tracing to predict the output of the above program.

- Type up the program and run it. Were your predictions correct?

## Task 3

```
# A program to demonstrate list slicing
fruits = ['apple', 'pear', 'orange', 'banana', 'kiwi']

print(fruits[1:3])  # ['pear', 'orange']
print(fruits[2:4])  # ['orange', 'banana']
print(fruits[2:5])  # ['orange', 'banana', 'kiwi']

print(fruits[1:])   # ['pear', 'orange', 'banana', 'kiwi']
print(fruits[:5])   # ['apple', 'pear', 'orange', 'banana', 'kiwi']
```

1. ***Can you figure out how the program generates the output shown inside the comments?***
2. ***How could we 'slice out' the fruits that grow in Ireland (i.e. apple and pear)?***
3. ***What output would be generated from the line*** print(fruits[ : ])**?**

---

---

---

**Task 4**

***Time to experiment!***
***Let's say we have the following initialisation.***

```
fruits = ['Strawberry', 'Lemon', 'Orange', 'Raspberry', 'Cherry']
```

***Predict the output that would be displayed by each of the `print` statements in the program snippet below. Record your predictions in the left column.***

```
1.  print(fruits[0])
2.  print(fruits[3])
3.  print(fruits[2])
4.  print(fruits[len(fruits)-1])
5.  print(fruits[1])
6.
7.  fruit = fruits[2+2]
8.  print(fruit)
9.  print(fruit[0])
10.
11. orange = fruits[1]
12. print(orange)
13. lemon = fruits[1]
14. print(lemon)
```

| ***Prediction*** | ***Actual*** |
|---|---|
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |

***Now key in the program and run it. Record the actual output in the right column.***

***What value do you think the following expression would generate? Try it!***

```
fruits[0][0]+fruits[1][0]+fruits[2][0]
```

***How does this answer shape your thinking in terms of the relationship between strings and lists?***

**Task 5**

***Can you find and suggest a fix for the syntax error contained in the code below?***

```
# Initialise two lists
1.  suits = ['Hearts','Diamonds','Spades','Clubs']
2.  cardFaces = ['Ace','1','2','3','4','5','6','7','8','9','Jack','Queen','King']
3.
4.  print(cardFaces[1:10])
5.
6.  colourCards = cardFaces[10:23]
7.  print(colourCards)
8.
9.  redSuits = suits[:2]
10. blackSuits = [2:]
11.
12. print(pinStripSuits)
13. print(redSuits)
14. print(blackSuits)
```

Syntax Error:

Fix:

***Does it surprise you that line 6 does <u>not</u> generate a syntax error? How does this extend your thinking in relation to range errors and slicing?***

## Task 6

```
fruits = ['pear', 'apple', 'orange', 'banana', 'kiwi']
fruit = 'apple'
vegs = ['peas', 'carrots']


fruits.append(fruit)
print(fruits) # ['pear', 'apple', 'orange', 'banana', 'kiwi', 'apple']

fruits.extend(vegs)
print(fruits) # ['pear', 'apple', 'orange', 'banana', 'kiwi', 'apple', 'peas', 'carrots']

fruits.insert(2, fruit)
print(fruits) # ['pear', 'apple', 'apple', 'orange', 'banana', 'kiwi', 'apple', 'peas', 'carrots']

fruits.pop()
print(fruits) # ['pear', 'apple', 'apple', 'orange', 'banana', 'kiwi', 'apple', 'peas']

fruits.remove(fruit)
print(fruits) # ['pear', 'apple', 'orange', 'banana', 'kiwi', 'apple', 'peas']

fruits.reverse()
print(fruits) # ['peas', 'apple', 'kiwi', 'banana', 'orange', 'apple', 'pear']

fruits.sort()
print(fruits) # ['apple', 'apple', 'banana', 'kiwi', 'orange', 'pear', 'peas']

print(fruits.index(fruit)) # 0
print(fruits.count(fruit)) # 2
```

1. ***Can you figure out how the program generates the output shown inside the comments?***
2. ***Type up the code and run it.***
   ***Tip! In order to fully understand the code, you may want to make some small changes to it (one at a time) and see how they affect the output.***
3. ***Log any questions you still have in relation to this code.***

## Task 7

```
# Sample program to build up a list of user details
userList = []
userName = input("Enter your name: ")
userList.append(userName)
userAge = int(input("What age are you "+userName+"?"))
userList.append(userAge)
userCountry = input("What is your country of birth?")
userList.append(userCountry)

print("My database for "+userName+"\n"+str(userList))
```

1. ***Type up the code and run it. Can you figure out what each line of code does?***
2. ***Extend the code to get the user to input their year of birth. What datatype will you use for this?***
3. ***At the end of the program, display the list again but in reverse order.***
4. ***Log any questions you still have in relation to this code.***

---

---

---

**Task 8**

- Write a program which splits the pangram "*The five boxing wizards jump quickly".*
- Rearrange the list so that the pangram becomes "*Quickly jump the five boxing wizards*". (You may ignore capitals)

  Hint! You could use the *pop()* method to pop off the last two words and use variables to store them until you have moved along the other four words.

**Task 9**

```
# Program to simulate a fruit machine!
import random

# Open the fruits file (already created)
fruitFile = open("fruits.txt","r")

# Read the entire file
fileContents = fruitFile.read()

# Close the file
fruitFile.close()

# Split the content into a list
fruits = fileContents.split()

# Spin! Display three fruits
print(random.choice(fruits))
print(random.choice(fruits))
print(random.choice(fruits))
```

1. ***Type up the code and run it. Can you figure out what each line of code does?***
2. ***Log any questions you still have in relation to this code.***

---

---

---