Government of Western Australia
North Metropolitan TAFE

Assessment Task: Lab 5

Qualification national code and title	22603VIC Certificate IV in Cybersecurity
Unit/s national code/s and title/s	**ICTPRG434 - Automate processes** **ICTPRG435 - Write script for software applications**

Assessment type (☑):

☐ Questioning (Oral/Written)
☐ Practical Demonstration
☐ 3rd Party Report
☒ Other – Lab

Assessment Resources:

The base requirements this assessment task include:

- IDE or editor for developing Python programs *(only IDLE and PyCharm supported by the college)*
- Access to Office 365 & Microsoft Word
- Virtual machine

You may not need all these for every part in this assessment

Assessment Due:

This assessment is due after the weekly session, **Week 5, Friday 17:00**.

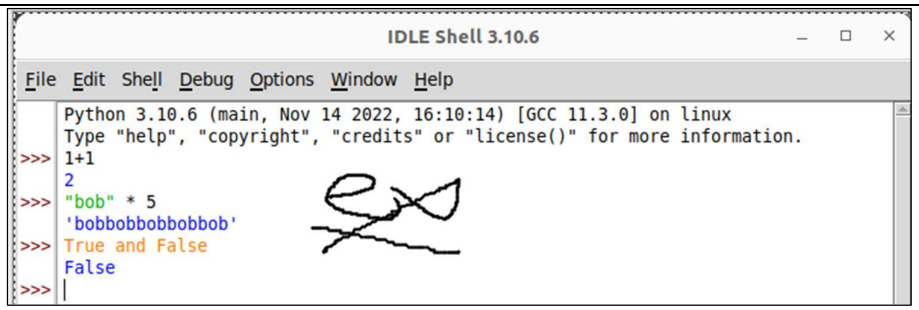
Assessment Instructions:

1. Your code must be written in IDLE or PyCharm IDEs. If you are using a different IDEs or a different structure for your application, then assistance from your lecturers may be limited (at best). Discuss with your lecturer before straying too far off the path!
2. All resources used should be referenced with the question. Answers may not be copied and pasted from any resource. All answers must be reworded to display your understanding.
3. You may only use Python functionality, methods and libraries which were taught in this unit.
4. First line of code in a program should have the student's name and number, as proof of authenticity.
5. Screenshots of all programs must be included **in** this document, with the appropriate question.
6. Screenshots of testing, showing your code works as intended, should be included with the relevant question.
7. Python programs should be named: XXX_Lab##_SYY_QZZ
   Replace XXX with your initials
   Replace ## with Lab number
   Replace YY with Section number,
   Replace ZZ with Question number
8. It is a submission requirement that all screen shots be signed in some way. Some acceptable examples of signed screen shots are shown below.

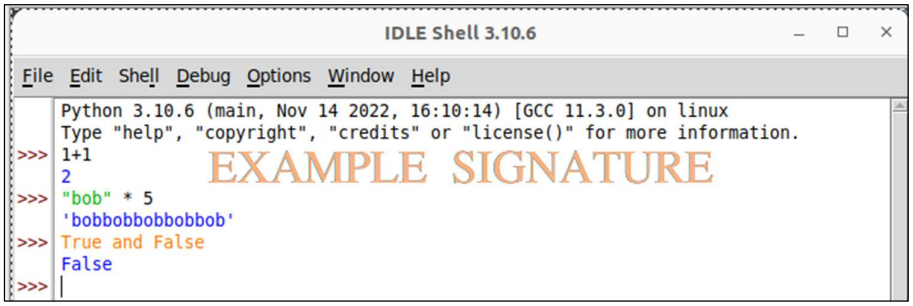
Government of Western Australia
North Metropolitan TAFE

Assessment Task: Lab 5

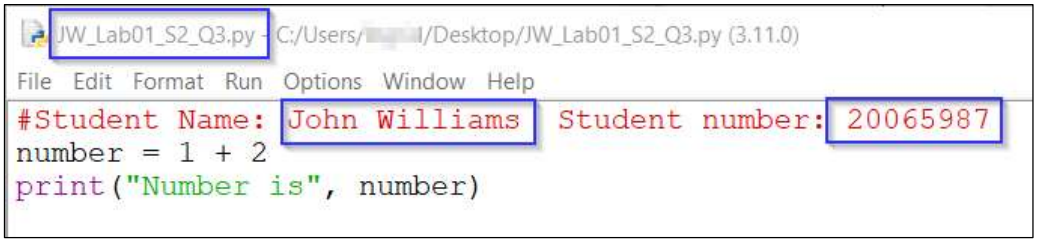
Qualification national code and title	22603VIC Certificate IV in Cybersecurity
Unit/s national code/s and title/s	**ICTPRG434 - Automate processes** **ICTPRG435 - Write script for software applications**



Example 1: Signed using a simple drawing tool.



Example 2: Water marked signature.



Example 3: Program named as prescribed, as well as first line comment with student name and number. Program saved as pre-described.

9. All python programs must be included in the submission, as well as this document.

Assessment Instrument:

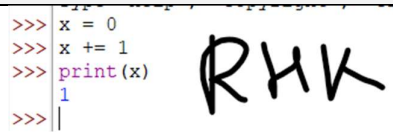
Government of Western Australia
North Metropolitan TAFE

Assessment Task: Lab 5

Qualification national code and title	22603VIC Certificate IV in Cybersecurity
Unit/s national code/s and title/s	**ICTPRG434 - Automate processes** **ICTPRG435 - Write script for software applications**

Section 1: Incremental short cuts

Begin this lab by exploring some of the short cuts commonly used in a loop. Run the following exercises in the Python shell/interpreter.
Run the following operations and evaluate the output:

Code	Screenshot of output
x = 0 x+=1 print(x)	
x-=1 print(x)	
x+=5 x*=10 print(x)	

Government of Western Australia
North Metropolitan TAFE

Assessment Task: Lab 5

Qualification national code and title	22603VIC Certificate IV in Cybersecurity
Unit/s national code/s and title/s	**ICTPRG434 - Automate processes** **ICTPRG435 - Write script for software applications**

Section 2: While loops

Occasionally, you may need to repeat a set of instructions while any given condition evaluates to true. Complete the following tasks in the Python shell/interpreter.

1. Run the following code and evaluate the output:

```
>>> import random
>>> x=0
>>> while x!=5:
        print("x is",x)
        x = random.randint(0,10)
```

Screenshot of output

```
>>> import random
>>> x = 0
>>> while x != 5:
...     print("x is", x)
...     x = random.randint(0,10)
...
x is 0
x is 6
x is 7
x is 3
x is 0
x is 7
x is 8
>>>
```

RHK.

2. How many times did the above loop run?

7 times

3. What condition must exist for the loop to stop running?

x != 5

4. Create an infinite loop by running the following code (stop execution with control + C):

```
>>> x = 0
>>> while True:
        print(x)
        x+=1
```

Screenshot of output

```
lab-5.py > ...
1  x = 0
2  while True:
3      print(x)
4      x += 1
5

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  GITLENS  POLYGLOT NOTEBO

34850
34851
34852
34853
34854
```

RHK

Government of Western Australia
North Metropolitan TAFE

Assessment Task: Lab 5

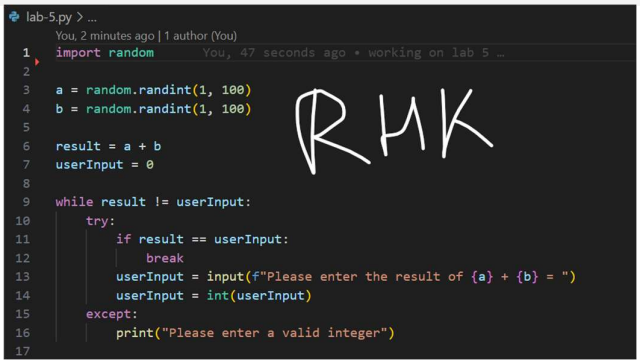
Qualification national code and title	22603VIC Certificate IV in Cybersecurity
Unit/s national code/s and title/s	**ICTPRG434 - Automate processes** **ICTPRG435 - Write script for software applications**

5. Why was the last block of code run infinitely?

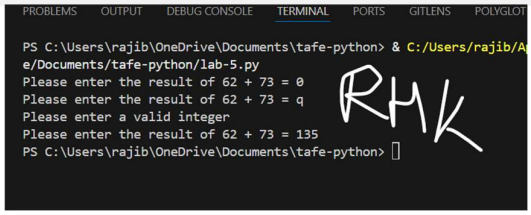
Bcs True will always be evaluated as True & won't allow to break the loop

6. Using a code file, create a working similar block of code that will do the following:
   a. Generate two random numbers.
   b. Display the maths equation the user needs to complete E.g. a + b =
   c. Repeatedly ask the user to input the correct answer to the sum of those two numbers until they input the correct answer.
   d. Ensure error handling and helpful user messages.

Screenshot of Code



Screenshot of Output



Government of Western Australia
North Metropolitan TAFE

Assessment Task: Lab 5

Qualification national code and title	22603VIC Certificate IV in Cybersecurity
Unit/s national code/s and title/s	**ICTPRG434 - Automate processes** **ICTPRG435 - Write script for software applications**

Section 3: For loops

Sometimes you will need to iterate through a set number of things. Complete the following tasks in the Python shell/interpreter.

1. Run the following code and evaluate the output:

```
>>> letterCount = 0
>>> theWord = "Kraftfahrzeughaftpflichtversicherung"
>>> for i in theWord:
        letterCount+=1
        print("Counted",letterCount,"letters in the word!")
```

Screenshot of output

```
Counted  1  letters in the word!
Counted  2  letters in the word!
Counted  3  letters in the word!
Counted  4  letters in the word!
Counted  5  letters in the word!
Counted  6  letters in the word!
Counted  7  letters in the word!
Counted  8  letters in the word!
Counted  9  letters in the word!
Counted  10  letters in the word!
Counted  11  letters in the word!
Counted  12  letters in the word!
Counted  13  letters in the word!
Counted  14  letters in the word!
Counted  15  letters in the word!
Counted  16  letters in the word!
Counted  17  letters in the word!
Counted  18  letters in the word!
Counted  19  letters in the word!
Counted  20  letters in the word!
Counted  21  letters in the word!
Counted  22  letters in the word!
Counted  23  letters in the word!
Counted  24  letters in the word!
Counted  25  letters in the word!
Counted  26  letters in the word!
Counted  27  letters in the word!
Counted  28  letters in the word!
Counted  29  letters in the word!
Counted  30  letters in the word!
Counted  31  letters in the word!
Counted  32  letters in the word!
Counted  33  letters in the word!
Counted  34  letters in the word!
Counted  35  letters in the word!
Counted  36  letters in the word!
```

RHK

Evaluation of code

For loop is iterating through the string of the theWord variable and on each iteration incrementing the value of the letterCount by 1

2. There is a useful Python string method called count, it will count the number of occurrences of any character you want with in a provided object. You can replicate its functionality using a for loop. Run the following code and evaluate the output:

```
>>> numberOfLetter = 0
>>> for i in "Handschuhfach":
        if i == "a":
                numberOfLetter+=1
                print(numberOfLetter)
```

Government of Western Australia
North Metropolitan TAFE

Assessment Task: Lab 5

Qualification national code and title	22603VIC Certificate IV in Cybersecurity
Unit/s national code/s and title/s	**ICTPRG434 - Automate processes** **ICTPRG435 - Write script for software applications**

Screenshot of output

```
>>> numberOfLetter = 0
>>> for i in "Handschuchfach":
...     if i == "a":
...         numberOfLetter += 1
...         print(numberOfLetter)
...
1
2
>>>
```

3. Describe why the above code output the observed result.

We are running the loop through the string object and if any item matches “a” we are incrementing the numberOfLetter variable by one and as there is only 2 “a” in the string object we are getting 2

4. Using a code file, create a working similar block of code that will do the following:
   a. Ask the user to input a word
   b. Ask the user to input another word
   c. Ask the user to input a letter.
   d. Use loops to count how many occurrences of that letter occur in both words.
   e. Tell the user which word has the most occurrences of the selected letter, if the words have the same number of occurrences of the letter, the program should state that.

Screenshot of Code

```
oneWord = input("Please enter one word: ")
anotherWord = input("Please enter another word: ")
matchingLetter = input("Please enter a letter you want to match: ")
countOfLetterInFirstWord = 0
countOfLetterInSecondWord = 0

for i in oneWord:
    if matchingLetter == i:
        countOfLetterInFirstWord += 1

for c in anotherWord:
    if matchingLetter == c:
        countOfLetterInSecondWord += 1

if countOfLetterInFirstWord > countOfLetterInSecondWord:
    print(f"{matchingLetter} has more mtaches in {oneWord}")
elif countOfLetterInSecondWord > countOfLetterInFirstWord:
    print(f"{matchingLetter} has more mtaches in {anotherWord}")
else:
    print(f"{matchingLetter} has same number of matches in {oneWord} & {anotherWord}")
```

Screenshot of Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  GITLENS  POLYGL

PS C:\Users\rajib\OneDrive\Documents\tafe-python> & C:/Users/rajib/
e/Documents/tafe-python/lab-5.py
Please enter one word: January
Please enter another word: February
Please enter a letter you want to match: a
a has more mtaches in January
PS C:\Users\rajib\OneDrive\Documents\tafe-python>
```

Section 4: A challenge

Government of Western Australia
North Metropolitan TAFE

Assessment Task: Lab 5

Qualification national code and title	22603VIC Certificate IV in Cybersecurity
Unit/s national code/s and title/s	**ICTPRG434 - Automate processes** **ICTPRG435 - Write script for software applications**

Complete the following challenge as described and demonstrated using a code file. Using the sample screen shots and pseudocode, create a script that will do the following:

1. Ask the user to input numbers until the user inputs the word “done”.
2. Use a try and except to validate and ask the user to re-input any numbers if the input is not valid as per requirement number 1.
3. The try and except block should be as *narrow* as possible; only code that may crash the code should be in the try block
4. Tell the user how many numbers they input and calculate the running total of those numbers.

```
Enter 'done' or number 0: 22
Enter 'done' or number 1: 3
Enter 'done' or number 2: a
Your input was invalid, try again
Enter 'done' or number 2: 24
Enter 'done' or number 3: done
You input a total of 3 numbers.
The sum of those numbers is 49
```

Note: To be marked as satisfactory for this task, your solution must:

- Have output formatted the same way as the demonstration screen shots.
- Contain comprehensive commenting.
- Demonstrate through screen shots, that your solution addresses all the above points.

You must submit the code file with this document

Screenshot of output:

```
PS C:\Users\rajib\OneDrive\Documents\tafe-python> & C:/Users/rajib/
e/Documents/tafe-python/lab-5.py
Enter 'done' or number 0: a
Your input was invalid, try again
Enter 'done' or number 0: b
Your input was invalid, try again
Enter 'done' or number 0: 6
Enter 'done' or number 1: 7
Enter 'done' or number 2: y
Your input was invalid, try again
Enter 'done' or number 2: done
You input a total of 2 numbers.
The sum of those numbers is 13
PS C:\Users\rajib\OneDrive\Documents\tafe-python>
```

RHK

Screenshot of code:

Government of Western Australia
North Metropolitan TAFE

Assessment Task: Lab 5

Qualification national code and title	22603VIC Certificate IV in Cybersecurity
Unit/s national code/s and title/s	**ICTPRG434 - Automate processes** **ICTPRG435 - Write script for software applications**

```
numCount = 0
total = 0
userNum = input(f"Enter 'done' or number {numCount}: ")

# running a loop to check if user wants to finish the task
while userNum != "done":
    # checking if the user input is a number
    if userNum.isnumeric():
        try:
            userNum = int(userNum)
            total += userNum
            numCount += 1
        except ValueError:
            print(ValueError)
        userNum = input(f"Enter 'done' or number {numCount}: ")
    else:
        print("Your input was invalid, try again")
        userNum = input(f"Enter 'done' or number {numCount}: ")

# printing the output of the task      You, 1 second ago • Uncommitted changes
print(f"You input a total of {numCount} numbers.")
print(f"The sum of those numbers is {total}")

```

RHK