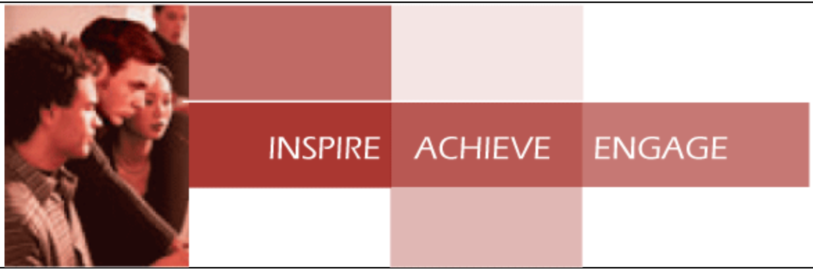
HOLMES INSTITUTE

FACULTY OF HIGHER EDUCATION



HOLMES INSTITUTE
Cum Propris Suis Alis Volat

Assessment Details and Submission Guidelines	
Trimester	T2 2019
Unit Code	HS1031
Unit Title	Introduction to Programming
Assessment Type	Individual
Assessment Title	Individual Assignment
Purpose of the assessment (with ULO Mapping)	Assess student's ability to develop algorithmic solutions to programming problems using Python language.
Weight	20 % of total assessments
Total Marks	20
Word limit	N/A
Due Date	Week 8
Submission Guidelines	• There are three questions in this assignment which require you to write and submit three Python scripts. Please save each script in two different formats: .py and .txt prior to submission. For example, for the first question you need to submit **checkWord.py** and **checkWord.txt**. • Combine all .py and .txt files into one zip or rar file (6 files in total). Finally, upload your compressed file (zip or rar) to Black Board and proceed with submission. • Code must be appropriately commented. Make sure to add comments at each segment of your code to explain what it does. You may lose grades if you do not add comments. You may lose grades if you do not add comments. • Make sure that your code runs successfully for all possible entries. **Hint:** test your code against the examples given in the question (if any). • Try to approach the solution with the least number of steps. Your code must be clear, logical, and easy to understand. • Your code must be written in **Python 3**. You get no marks if the code is written in Python 2 or any other language. • All work must be submitted to Blackboard by the due date (**Monday, 09 September 2019 11:55PM**). • You are encouraged to avoid last minute submissions so that you do not run into technical difficulties. • You are allowed up to three attempts. All attempts must take place prior to assignment deadline. • Please note that the Self-check Safeassign link has been removed from Blackboard. You can still check your work for plagiarism by directly submitting your assignment. If the score for plagiarism is high, you are welcome to resubmit which will count as a second/third attempt. • Please note that plagiarism is treated seriously. All those caught plagiarising will get zero and get their names added to an institutional register. Remember,

	submitting genuine work and scoring lower is better than submitting non-genuine work and risking being added to the plagiarism watch-list. • than falling under the plagiarism watch-list. • Please download the attached submission template and see the step-by-step instructions for submission.

Individual Assignment Specifications

Purpose:

This assignment evaluates your understanding of basic programming principles using Python language. In particular, it assesses your ability to develop algorithms to solve simple problems, successfully run python programs, and your ability to write meaningful comments when required.

Marking criteria

Question	**Marking criteria**	**Marks**
Question 1	Appropriate commenting	2
	Sound logic	2
	Code running successfully	2
Total		**6**
Question 2	Appropriate commenting	2
	Sound logic	2
	Code running successfully	2
Total		**6**
Question 3	Appropriate commenting	2
	Sound logic	3
	Code running successfully	3
Total		**8**
Total Marks		**20**

1. You may have noticed that some English words can be read both ways (from left to right or right to left). For example, **Kayak**, **Racecar** and **Madam** can all be read both ways. Most words however can only be read from left to right, for example, **Melbourne**, **Python** and **Holmes**. Write a program (name it, **checkWord.py**) that finds out if a given word can be read both ways. The program prompts the user to enter a word, and then either prints "Both ways" or "One way" as an output.

   For example:
   If the input word is: "Kayak"
   The output would be: Both ways

   If the input word is: "Holmes"
   The output would be: One way

   Hint:
   - Start by prompting the user for a word
   - Convert the word into small letters
   - Write the rest of the program

   Marks Distribution

Criteria	**Comments**	**Logic**	**Execution**	**Total**
Mark	2	2	2	**6**

2. Write a program (name it, **unique.py**) that takes a text message as an input and prints out the same message, however with unique words only. In other words, all words repeated within the text must only appear once. You need to remove all punctuation marks from the returned message. Punctuation marks include "**. , ; : ? !**"

   For example:
   If the input message is: "You're an insomniac, you tell yourself: there are profound truths revealed only to the insomniac by night like those phosphorescent minerals veined and glimmering in the dark but coarse and ordinary otherwise; you have to examine such minerals in the absence of light to discover their beauty, you tell yourself."

   The output message would be: "you're an insomniac you tell yourself there are profound truths revealed only to the by night like those phosphorescent minerals veined and glimmering in dark but coarse ordinary otherwise have examine such absence of light discover their beauty"

   Hint:
   - Use the message in the example as a test case. Store it in a variable called **text**.
   - You will need to convert the message into small case letters, and remove punctuation marks.
   - Your final output must be similar to the output in the example above.

Marks Distribution

Criteria	Comments	Logic	Execution	Total
Mark	2	2	2	**6**

3. Three pets live in a happy house: dog, cat and a mouse. The dog often chases the cat, the cat likes to chase the mouse, however the mighty dog runs away when it sees the mouse. Develop a game (let's call it **happyHouse.py**) which can be played by two players. The program asks the two players (**player1** and **player2**) to choose either 1 for dog, 2 for cat, or 3 for mouse. The rules for this game are simple:

The dog prevails over the cat
The cat prevails over the mouse
The mouse prevails over the dog

So if **player1** chooses a dog and **player2** chooses a cat, **player1** *wins*. If **player1** chooses a dog and **player2** chooses a mouse, **player2** *wins*, and so forth. If both players choose the same pet, then the result will be *draw*. The players continue playing until either player hits enter without choosing a number (1, 2 or 3). Each time the game is run, the result is printed on the screen (for example, Player 1 wins or Player 2 wins).

Hint:

- Define an indefinite loop which only breaks when either player enters an empty string (hits enter with no entry).
- Within the loop, prompt Player 1 and Player 2 to enter 1 for dog, 2 for cat or 3 for mouse
- Based on the choices made by Player 1 and Player 2, your program prints out the appropriate message, which can either be "draw", "Player 1 wins" or "Player 2 wins".

Marks Distribution

Criteria	Comments	Logic	Execution	Total
Mark	2	3	3	**8**