**Mindworx Academy**

**Python Programming Continuous Assessment**

**Total time: 2hrs**

**Guessing Number Game Project Specs:**

--------------------------------------------------------------------------------------------------------------------------------

**The Following are the specs or instructions to follow to develop the game:**

-------------------------------------------------------------------------------------------------------------------------------

1. This is a game that allows a user to play guessing one single number to try to win at a maximum of 3 attempts. The possible numbers to win must be stored in a list or tuple or dictionary.
2. It must be able to store **6 randomly** generated possible numbers (between 0-60, thus 1-59) to win from in a list or tuple or dictionary.
3. A person playing the game must have only three chances of playing to win or loose
4. One can win in the first or second or third attempt. The moment one had a correct guess, the game must then end telling someone that he or she has won and end the game.
5. If you did not win in all three attempts, then the program must inform the user that he has lost the game.
6. The game only allow some to play the game using numbers between 0-60, thus 1-59.
7. To win is to play only anyone correct number from our six numbers stored in list or tuple or dictionary. **Please note** that, playing a number below 1 or above 59 intentionally or by mistake must be counted as an attempt and chance of playing the game. **A TOUCH IS A MOVE!!**
8. **Please Note:** winning is playing one correct number of the 6 stored numbers.
9. The application can allow the user to enter his or her name to play as the game is ready to start, and welcomes him or her. NB***. The name of the user can be used during playing.***
10. As the game is getting ready to start, the program can give a user options 'OK' to continue playing the game or 'CANCEL' to exit the game playing the game.

--------- Additional info----------------------------------------------------------------------------------------------

1. Use a combination of any **looping structures, decision structures and lists or tuples or dictionary** to achieve your results.
2. Make sure your program is logically organised and
3. Each instructions carries **4marks**