Practical Portfolio Task 02

Python / Raspberry Pi Intermediate

Instructions to the Assessor

Type of Assessment: Practical Portfolio Task – Graded by: observational checklist

Duration of Assessment: 1 Session (3 hours)

Location of Assessment:

- Thornlie Campus: Block 8, Room 8G27

Conditions:

- This assessment is to be completed by individual students.
- Student must be deemed satisfactory for all listed skills in the *observational checklist* document.
- The assessment is “open book” to simulate a practical work environment; “open book” refers, but is not limited to:
    - Language and API reference documentation
    - Manual pages
    - Web searches
    - Student typed or hand-written notes.

Elements and Performance Criteria: Refer to DAP and Unit Mapping documents.

Required Skills and Knowledge: Refer to DAP and Unit Mapping documents.

Instructions to Students

Student Outcomes: Successful completion of this assessment should demonstrate the following outcomes:

- Have knowledge of and observe OHS policies and procedures when carrying out work
- Object-oriented programming language features
- Object-oriented programming language operators
- Object-oriented language control structures
- File access
- Classes and Objects

What is Assessed: Refer to DAP

Duration of Assessment: 1 Session (3 hours)

Materials Allowed:

- Object-oriented language reference documentation
- API documentation
- Hand-written or typed notes

Resources Required:

- Provided by Assessor:
    - Raspberry Pi
    - Raspberry Pi Sense HAT
    - Micro SD Card
    - USB Power Brick
    - HDMI to DVI-D cable
    - Internet-connected Lab PC

Assessment Instructions

For this assessment you will be required to create a basic Python program that: gets sensor data from the Raspberry Pi Sense Hat, stores data in a file, runs the program for a set amount of time. You will be required to demonstrate knowledge of established OHS procedures and best practices (e.g. safe handling practices), Python language features, operators, control structures, syntax, Classes, Objects and file access.

Be sure to read all task requirements carefully before you begin this assessment.

Task requirements:

You are required to write a Python program that fulfils the following requirements:

- Must take measurements for:
  - Temperature
  - Humidity
  - Pressure
- The above data must be gathered **once every hour** for a time period of **one week**.
- Every measurement must have a timestamp of when the measurement was taken in the following format:
  - YYYY-MM-DDThh:mm (e.g. 2019-05-27T09:05)
- Data from measurements must be stored in a file called “env_data.csv” on a new line for each measurement in the following format (an example file is included in the assessment folder on Blackboard):
  - temperature,humidity,pressure,timestamp (e.g. 23.58,65.81,1013.42,2019-05-27T09:05)
- Program is to exit once it has run for one week (as per previous requirement) and when it exits it should print the content of the “env_data.csv” file to the Terminal
- Program must contain reasonable documentation. Programs with no documentation will require resubmission.

In addition to the above requirements you must also write a brief paragraph detailing how you tested your program (i.e. debugging). You may use dot points for formatting purposes; your brief must be at least four sentences explaining your testing methodology (e.g. running the program without timer delays to check core functionality).

The above above are the **minimum** program requirements. Minimum requirements must be met for you to be deemed competent for this assessment. If **any** requirements are not fulfilled you will be required to resubmit. Extra functionality is permitted but will not be assessed.