**Assessment type (🗹):**

Questioning (Written)

Practical Demonstration

3rd Party Report

Portfolio

**Assessment Resources:**

|  |
| --- |
| Python3 interpreter.  Python IDE, like PyCharm, or VSCode (the latter is not supported by the college), with the ability to use Python Virtual Environments.  Access to Office 365 and Microsoft Word.  Git and access to GitHub.  Use of some of these items may not occur in this part of the assessment task. |

**Assessment Due**

|  |
| --- |
| This item is due:   * **Week 12, 17:00 (5pm) on the day of the scheduled lecture**   Refer to Blackboard for most accurate dates, which may alter due to unforeseen circumstances. We also endeavour to update these documents at the same time.  It is advantageous for you to attempt to meet this deadline. |

**Assessment Instructions:**

|  |
| --- |
| Scenario (updated!) You are employed as a junior software developer at a boutique software house called Softwares-R-Us in Perth. Instructions Your code must adhere to certain style guides, the most important being PEP-8 – Style Guide for Python Code. You should familiarise yourself with [PEP-8](https://pep8.org/) before continuing.  Use of a Git repository is standard practice at Softwares-R-Us and most of the steps require the use of Git and GitHub.  There are multiple steps in this task, and you must perform each step to a satisfactory level.  You may answer any questions in the provided template (if available) and the use of screenshots is encouraged. However, you must also provide the actual source code as a ZIP-file of the project for any programming tasks. Make sure to remove the Virtual Environment folder (**venv** or **.venv**) from the ZIP-file before uploading.  You must document your code properly. Use docstrings where needed including but not limited to entire classes and methods. Use inline comments to clarify certain parts of code.  If you use any external resources, you should provide references. |

**Assessment Instrument:**

|  |
| --- |
| Step 1 – Knowledge Question (50-80 words) Research Inter-Process Communication (IPC). Provide at least three different examples of techniques that you can use to implement IPC, and, in your own words, describe how each one of them works. Step 2 – Create new project for the Magic 8-ball In this step, you will write a simple client-server application that uses pipes or sockets for inter-process communication. The program should implement a so-called Magic 8-ball, which you can ask questions. The Magic 8-ball then answers with a specific reply. The replies can be found here: <https://en.wikipedia.org/wiki/Magic_8-Ball>.  The server must somehow use the question to determine an answer, but it is up to you to determine what the reply will be. For instance, you can use the question to calculate a simple hash that determines the answer. You might add a bit of randomness, so it doesn’t always reply with the same answer for a certain question.  You may add your project to GitHub but that is not mandatory for this question. Step 3 – Knowledge Question (30-60 words) Operating System “signals” can be used as a form of sending commands to other processes. Describe, in your own words, what signals are, and which signals are available (this differs for each Operating System). Also, provide at least three examples how signals can be used in real-world applications. Step 4 – Write a signal handler In this step, you will create a very simple Python script that reacts to an operating system signal. You are free to select the signal that your script will handle.  Be aware that Windows only has a small subset of signals. You might, therefore, consider running the script in an online Python environment that is based on Unix.  You must provide evidence of the script reacting correctly to the signal.  You may add your project to GitHub but that is, again, not mandatory for this question. Submitting your work ZIP your entire project into a single file.  Make sure to remove the Python Virtual Environment folder (called **venv** or **.venv**) from the ZIP-file before uploading it into the Blackboard assessments area. |