

# Quick Wins Checklist

- **A list of data structures (variables, lists) that you use with an explanation as to how they will be used in your solution (remember that your documentation is supposed to be prior to programming and is therefore in future tense!). For example:**

*guessCount – this variable will be used to store the number of attempts the user takes to complete the quiz. It is a global variable which means that it can be access anywhere inside the program and not just within a single procedure. The data store in the variable can then be called at any point and output to the screen for the user’s convenience.*

- **A separate section discussing the choice of programming techniques used. Use the template in shared (called Controlled Assessment Portfolio Template) to guide you, but ensure you explain why you chose one approach over another – no matter how obvious it may be – and how it allows the solution to meet the user’s needs.**

*E.g. Using a loop instead of IF statements to ensure every question was answered twice, a user requirement. Without a loop the program would not have been able to load a new question each time...*

- **Show screen shots of your solution in order of the tasks set in the assessment brief and explain how each screen meets the task and user needs.**
- **Completed solution Testing - testing is worth up to 6 marks.**

Give each test a number and test each section of your program – not just that it loads but also with data you would expect (or not expect!) from a user. Explain what you expect to happen, what actually happens and any action required from you to fix it.

You should also take screen shots of your program as you test it and fix it to back up what you are saying. Insert the screen shots into your work with an annotation to explain what they show. You could also give the image a reference and refer to it in your test table?

Test Plan & Evidence of Tests

| Item No. | Description        | Test                | Input Data/User Action | Expected Outcome                                     | Actual Result                                   | Action Required                          |
|----------|--------------------|---------------------|------------------------|--|---|--|
| 1.1      | Opening menu works | Menu choice 1 works | 1                      | Course number is presented                           | As expected                                     | No                                       |
| 1.2      |                    | Menu choice 2 works | 2                      | The two client sub menus are displayed               | As expected                                     | No                                       |
| 1.3      |                    | Menu choice 3 works | 3                      | Months of the year are displayed (number then month) | All displayed although not in the obvious order | Re-order months by numbers (1 before 11) |
| 1.4      |                    | Menu choice 4 works | 4                      | Closing message displayed and program exits          | As expected                                     | No                                       |
| 2.1      | Search by          | Accepts             |                        |  |   |  |

- **Annotated Pseudo and Python code – worth up to 6 marks in total (3 each) if FULLY annotated!!**

This is something you could easily do at home? Put comments in different colour and print with colour showing.

- **Evaluate your solution – up to 3 marks.**

Describe how your solution meets the needs of the user. Spelling, punctuation and grammar can also gain you marks in this section. Aim to use specialist vocabulary (efficiency, robustness, global variables, iteration and selection) wherever possible.