

## IMPORTANT

Please ensure the following:

1. That you submit a zip file of the COMPLETE executable application, with corresponding data text files.
2. PDF files of source code will not be marked.
3. DO NOT submit a git bundle. We have discontinued use of the git tool for version control.
4. The submitted work is entirely your own.
5. That you have checked to ensure that you did not upload any sample programs you may have consulted during your programming activity.
6. You did not provide your submitted source code to any of your fellow students.
7. Empty submissions, or submissions that are missing the source code that we need to mark your assignment will be awarded 0.
8. You have 3 days once marks have been released to query your mark and request a review, after that, your query can unfortunately not be considered.
9. Submitting an assignment that does not address the specification will be awarded 0. Please do not submit past assignments with the hope that you will get marks: such assignments will receive 0.

Expectations: please see the assignment instructions in the pre-amble learning unit for expectations.

Result summary:

No source code in submission	Warning: Similar to other submission	Late submission	Wrong submission	Marked
------------------------------	--------------------------------------	-----------------	------------------	--------

1. No source code in submission: you did not submit any source code that can be marked. You are awarded 0.
2. Similar to other submission: your assignment bears a striking resemblance to another person's submission. You have been warned. Should assignment 2 again look similar to another submission, you will be awarded 0, and further steps will be taken.
3. Late submission: your submission is late. Please read the guidelines above regarding late submissions.
4. Wrong submission: Your assignment does not address the requirements of the specification. For example, the assignment requires a program for motor racing club management, but yours is a hotel booking system. You are awarded 0.
5. Marked: your assignment has been been marked, and the result is provided at the bottom of this document.

**Input****[20]**

Assessment for input considers the usability of the system. This means that an end user should be able to easily enter data: no input using `inputbox()` for example, birthdates should be entered using date pickers, and so on.

- (a) [5 marks] Text boxes used to enter textual data. Design and layout is consistent.
- (b) [2 marks] Radio Boxes used for mutually exclusive data. Gender selection, for example.
- (c) [2 marks] DatePicker used (registration/birth dates). No dates are entered using plain text-boxes.
- (d) [3 marks] ListBoxes used (choose members)
- (e) [4 marks] Automatic generation and display of membership numbers.
- (f) [4 marks] Search functionality for members using membership number. Validation checks for entered number.

**Navigation****[6]**

- (a) [3 marks] Menu bar/Buttons used to aid in navigation between forms. (Tabs are also acceptable.)
- (b) [3 marks] Shortcut keys used. (Alt+M for member management form, Ctrl+S to save data, for example.)

**Error handling****[8]**

Marks will be subtracted if code uses Try/Catch as if they are If/Then statements.

- (a) [4 marks] Try/Catch used (one mark per try/catch). Generic catches are only used when no specific exception can be used – code comments clearly explain why this is the case.
- (b) [4 marks] At least one Throw used: proper throw to indicate that some exception has occurred – cannot just be a throw for the sake of it. Exception is caught and handled.
- (c) [2 marks (bonus)] Finally used as part of the Try/Catch. Placement and utility of finally makes sense and is not superfluous.

**Files and Data****[16]**

- (a) [2 marks] Check that files exist before reading/writing from them (or any other method of defensive programming).
- (b) Member File
  - i. [2 marks] Open and matching close when used.
  - ii. [2 marks] System correctly stores data in file (comma separated value, format is record and field based)
  - iii. [2 marks] System correctly reads data from file (and can display it).
  - iv. [2 marks (bonus)] Can delete driver from the system.
- (c) Event and race results file.
  - i. [2 marks] Open and matching close when used.
  - ii. [2 marks] System correctly stores/reads data in file (comma separated value, format is record and field based).
  - iii. [2 marks] System correctly reads data from file (and can display it).
  - iv. [2 marks (bonus)] Can delete event from file (cannot delete race results from file).
- (d) [2 marks] User can choose the location of the driver/race file.

**Meets requirements in specification****[10]**

- (a) [5 marks] This is a fully functional system as specified in the assignment specification. Users can manage drivers, events, and record race results for drivers.
- (b) [5 marks] Membership numbers are generated, and validated correctly.
- (c) [5 marks (bonus)] Cannot delete events or racing driver once race results for a racing driver has been recorded.