

Candidate number: CANDIDATENUM

## Summary

Mark /40	Total %
TOTALMARK	PERCENTMARK

## Feedback

FEEDBACKTEXT

### Mark breakdown:

Covid Data Updates	S1	/10
Covid News	S2	/10
User interface	S3	/10
Project delivery	S4	/20
<b>Total</b>	<b>TOTAL</b>	<b>/50</b>

## Marking criteria:

The assignment as a whole has a potential mark of 50. The first three sections are out of 10 split into 5 marks for function and 5 marks for style. Functionality marks are assigned cumulatively so the first criteria must pass in order to get marks for the second criteria and so on. Style is assessed based on the style criteria and will be supported by the output of pylint evaluation of the code.

- **Covid Data Updates** [10 marks]

5 marks for functionality. Functionality marks accumulate and will be checked by running the code with tests:

- Is the sample data ingested correctly with (test\_parse\_csv\_data) - 1 mark
- Is the sample data interpreted correctly with (test\_process\_covid\_csv\_data) - 2 marks
- Can structured data accessed from the covid API (test\_covid\_API\_request) - 3 marks
- Can the data updates be scheduled (test\_schedule\_covid\_updates) - 4 marks
- Does the code make good use of data structures and indexing of data (avoiding hard coded logic steps where possible) - 5 marks

Up to 5 marks for style. Style marks are assigned individually and checked by python pylint and manually checking the source code against the criteria listed below. The code must cover all functionality to be eligible for full marks for style.

- 1 mark for layout including consistent indentation, line length, etc.
- 1 mark for effective identifier naming, appropriate variable scoping and appropriate loop nesting.
- 1 mark for type hinting and using default values where appropriate.
- 2 marks for doc strings and use of structured commenting.

- **Covid news** [10 marks]

5 marks for functionality. Functionality marks accumulate and will be checked by running the code and test code:

- Are there a functions called news\_API\_request and update\_news that takes the right arguments (test\_news\_API\_request) - 1 mark
- Does the function return and filter relevant covid data (test\_news\_API\_request) - 2 marks
- Can news updates be scheduled independently of covid data updates? - 3 marks
- Can news items be removed? - 4 marks
- Is there an appropriate method and data structure for news items to be removed? - 5 marks

Up to 5 marks for style. Style marks are assigned individually and checked by python lint and manually checking the source code against the criteria listed below.

- 1 mark for layout including consistent indentation, line length, etc.

- 1 mark for effective identifier naming, appropriate variable scoping and appropriate loop nesting.
- 1 mark for type hinting and using default values where appropriate.
- 2 marks for doc strings and use of structured commenting.

- **User interface** [10 marks]

5 marks for functionality. Functionality marks accumulate and will be checked by running the code:

- Does the interface open from a browser and embed basic covid data? - 1 mark
- Does the interface embed the news and updates data structures? - 2 marks
- Can updates be added from the form? - 3 marks
- Do updates complete correctly and update the news and covid data - 5 marks

Up to 5 marks for style. Style marks are assigned individually and checked by python lint and manually checking the source code against the criteria listed below.

- 1 mark for layout including consistent indentation, line length, etc.
- 1 mark for effective identifier naming, appropriate variable scoping and appropriate loop nesting.
- 1 mark for type hinting and using default values where appropriate.
- 2 marks for doc strings and use of structured commenting.

- **Project delivery** [20 marks]

Project delivery marks will be awarded for the features included in your overall solution.

The marks for project delivery are assigned for:

- Is a configuration file used and does it contain all possible details to customise the application - upto 5 marks
- Is there a log file and does it provide a thorough history of the app events with clear formatting and effective use of severity levels - upto 5 marks
- Is there integration of the provided tests into the final project and further test functions and modules to cover individual functions that work with pytest - upto 5 marks
- Is there clear and structured documentation (README.txt) that works as a user manual to instruct new users how to deploy and use the application. Is the code and project also available to access for further development? - upto 5 marks