Exam practicalities and preparation

A short guide for the 180-minute, offline notebook exam

The exam has two parts worth 50 points each. Work in the supplied answer notebook. This guide sets out what you may use, and helps you prepare your laptop and make your work clear to mark. *Shortcuts below are for Windows/Linux; macOS users can find the named VS Code commands through the menus or Command Palette.*

What you may use

Two things are not allowed: anything that needs the internet, and anything built on AI. Everything else is allowed. This is the official exam cover in short: all aids, no internet, no generative AI.

- **Allowed:** your own notes, books and printouts; course materials saved on your laptop (the course ZIP, the lecture PDFs, the cheatsheet); documentation saved beforehand; `help()` and the documentation VS Code shows when you hover over a name; and **VS Code's suggestion list**: type `df.` and it lists `head`, `describe`, `groupby` and the other methods that exist; inside `pd.read_csv(` it shows the arguments.
- **Not allowed:** anything online (web search, Colab, online documentation, the course site, messages to anyone) and anything built on AI: chat assistants such as ChatGPT, and **AI code completion**, the grey suggested code that writes a line or a block for you, or writes code from a comment. GitHub Copilot is the best-known example.

A quick test for an editor feature: picking a name from a list of what already exists is fine; having code written for you is not. For anything else, use your intuition: **if it feels like cheating, it probably is.** If you are unsure about a tool, ask before the exam.

Before the exam

- **Check your local setup.** Follow the local readiness guide. Open the course folder in VS Code, select your exam Python kernel, disconnect from the internet, restart the kernel and Run All in `setup_check.ipynb`. Read the verdict and confirm the plot appears. Installation and package troubleshooting are collected in the setup guide. After a repair, repeat the full check offline.
- **Save your offline materials.** Download the course ZIP and the prepared lecture and cheatsheet PDFs from Exam info. Use these PDFs: printing a live lecture yourself may omit code output. Open saved files with internet off; a bookmark is not an offline copy.
- **Know where your files are.** Make one easy-to-find folder for your saved reference materials. Keep the exam files together when you receive them, and check that the notebook can read its supplied CSV locally. Do not rely on Colab, a remote file, or an installation that needs internet.

Use a mock as a rehearsal

Set aside one uninterrupted **three-hour block**. Start a timer, turn off Wi-Fi, and use VS Code set up as on page 2. Use only what the exam allows, and keep the mock's code-solutions notebook closed; this also shows whether the materials you saved are the ones you need. Work from the paper and answer notebook as you would in the exam. Save your work and rehearse the two-file hand-in at the end. **Only after the timer ends**, check the code solutions, compare your reasoning with the questions, and note what to practise next. The supplied mock code solutions deliberately omit written explanations.

VS Code: switch off AI completion, keep the suggestion list

As page 1 explains, the suggestion list is allowed and AI completion is not. These steps switch off the one and keep the other. Do them in the VS Code profile and folder you will use for the exam.

1. **Switch off the built-in AI.** Press `Ctrl+Shift+P`, choose `Preferences: Open User Settings (JSON)`, and add these two entries *inside* the existing outer braces; put a comma after an existing entry if needed.

   ```
   "chat.disableAIFeatures": true,
   "editor.inlineSuggest.enabled": false
   ```

   The first is the setting `Chat: Disable AI Features`: it hides built-in chat and inline AI suggestions and disables the Copilot extensions. The second stops grey suggested code after the cursor, whichever extension would supply it. Neither affects the suggestion list, the argument hints or the hover documentation.

2. **Disable AI extensions.** Open Extensions with `Ctrl+Shift+X` and search for `@enabled`. Use the gear menu to **Disable** any AI chat, code-generation or completion extension, for example GitHub Copilot, IntelliCode or Tabnine (IntelliCode is AI; IntelliSense, the suggestion list, is not). Keep the Python and Jupyter extensions: they run the notebooks and supply the suggestion list.

3. **Check it, with Wi-Fi on**, so that an AI extension that needs the internet would show itself. Reopen VS Code, and in a notebook code cell type `import pandas as pd`, then `df = pd.DataFrame()`, then `df.` on a new line. A list with `head` should appear. Then type the comment `# average daily return`, press `Enter` and wait a few seconds: **no grey code should appear**. If it does, find the extension that supplies it under `@enabled` and disable it.

4. **Followed an earlier version of this guide?** It also switched off the suggestion list. To turn the list back on, delete the other `editor.` entries it added, keeping `editor.inlineSuggest.enabled`. If you removed the `Ctrl+Space` shortcut, open Keyboard Shortcuts with `Ctrl+K Ctrl+S`, search for `Trigger Suggest`, right-click it and choose `Reset Keybinding`.

VS Code references (checked September 2026): built-in AI switch; IntelliSense settings and shortcuts; extension controls; settings precedence; notebook editor. Configure and test everything before the offline exam.

While you work

- Read each task and its point value. Answer what it asks for: code, a number, a plot, or an explanation. For a plot, give readable axis labels and explain what it shows when asked.
- Use short cells, clear variable names, and simple checks such as shapes, missing-value counts, or a few displayed rows. Save often. Before finishing, restart the kernel and run the completed notebook from top to bottom to catch missing imports and hidden state.
- **If stuck, show your reasoning.** Write the intended calculation or modelling step in a Markdown answer cell, then complete the code you can. Partial work can earn points. If a later part uses an earlier result that may be wrong, state what you are carrying forward and continue; follow-through marking can credit a correct later method.
- Leave enough time to check that every requested answer is in the answer notebook and that it opens correctly. Do not substitute the question paper or a screenshot for a required file.

The two-file hand-in

Submit the **completed answer notebook** and the **supplied blank one-page PDF**. Put all code and written answers in the notebook. The PDF is intentionally blank and takes no answers. Use the exact filenames printed in your exam instructions, and follow the official submission directions.