Final Assessment

* **Due** 20 Oct by 14:00

A blue arrow in a white circle

AI-generated content may be incorrect. **Assessment Brief**

This assessment is designed to evaluate your understanding of the core principles and best practices in software testing. As a quality engineer in a start-up environment, your role is critical in ensuring the software being developed adheres to the highest standards of quality. You will be required to demonstrate your ability to apply software testing methodologies effectively, perform requirements analysis, and implement design principles to develop a comprehensive testing strategy.

By completing this assignment, you will:

* Implement complete test coverage for the entire codebase.
* Build a CI/CD pipeline using GitHub Actions.
* Improve the code efficiency where possible.

The GitHub repository you will need to complete this assessment is here: [https://github.com/liuc8/online-bookstore-final-assessmentLinks to an external site.](https://github.com/liuc8/online-bookstore-final-assessment)

**Scenario**

In this scenario, you are working with a development team that has completed the build of an Online Bookstore for an e-commerce website.

**Your task now is to identify any potential bugs in the system and optimize the performance of the code.**

To achieve this, you will need:

1. To construct a series of test cases to assess the system's functionality.
2. Use Python tools such as timeit and cProfile to measure the performance of critical parts of the application, identify inefficiencies, and suggest improvements.

This assessment will not only test your ability to ensure the system’s functionality but also your capability to optimize the code for better performance. By identifying and addressing performance bottlenecks and optimizing the system's response time, you will ensure that the Online Bookstore operates efficiently, providing users with a seamless experience. You will also be expected to explain your rationale for the design decisions made and how they align with the project’s objectives, ensuring a high-quality end product.

**Task**

**Project Requirements:**

You are tasked with developing the necessary test cases using **Python**and integrating them into a CI/CD pipeline with **GitHub Actions**. Additionally, you are expected to identify areas for **code improvement** based on your findings, implementing optimizations where necessary.

As part of this assessment, you will create a comprehensive **report**that outlines the issues you identified, the steps you took to address them, and the improvements made. The report should include detailed **explanations of the fixes**, accompanied by relevant **screenshots**and the **outcomes**of the test cases. The objective is to ensure both the functionality and efficiency of the code, with clear documentation of your testing and optimization process.

**Part 1: Test Code with Automation (Software Artefact)**

In this section, you are required to implement automated test cases for the software using Python. The test cases should be designed to validate the core functionality of the system, ensuring that the application performs as expected under various scenarios. Your implementation should includeunit tests, integration tests, and any other relevant testing strategies.

The test automation process should be integrated with a Continuous Integration/Continuous Deployment (CI/CD) pipeline using GitHub Actions. This integration will automate the execution of the tests whenever new code is pushed to the repository, ensuring that any changes made to the system are continuously validated.

Your test code should be structured, maintainable, and capable of identifying potential issues in the software. The tests must cover a broad range of scenarios, including edge cases and performance considerations. Additionally, you should aim to improve the efficiency of the tests where possible, using Python tools such as timeit and cProfile to identify performance bottlenecks and suggest code optimizations.

The test cases should:

* Ensure comprehensive coverage of all unit tests for the application’s core functionalities.
* Identify inefficiencies in the code and suggest possible improvements.
* Provide clear recommendations for code optimization and demonstrate these improvements through before-and-after comparisons.
* Be fully automated within the GitHub Actions CI/CD pipeline, ensuring that the testing process is triggered automatically with every code push.

Your goal is to deliver a robust and efficient testing solution that not only validates the software’s functionality but also contributes to the continuous improvement of its performance.

**Part 2: Critical Evaluation (1500 words Report)**

Your report should provide a comprehensive and critical evaluation of the test cases you have developed, the improvements you implemented, and the integration of CI/CD automation. This evaluation should address key areas in detail, offering insights into your decision-making process, the effectiveness of your testing strategy, and the impact of the optimizations on both the functionality and performance of the software.

In your report, you should cover following areas:

1. **Test Case Design and Coverage:** Evaluate the design of your test cases, ensuring they cover all necessary functionalities and edge cases. Discuss the reasoning behind the chosen testing strategies (unit tests, integration tests, performance tests, etc.), highlighting any challenges encountered during test creation and how they were addressed.
2. **Effectiveness of the Improvements:**Provide a detailed analysis of the code inefficiencies you identified and the improvements you implemented. Discuss how these changes positively impacted the system’s performance, referencing the before-and-after comparisons where applicable. Explain the specific benefits of each optimization, whether related to execution time, memory usage, or other performance metrics.
3. **CI/CD Automation Integration:**Critically evaluate the integration of your test cases into the CI/CD pipeline using GitHub Actions. Reflect on the process of automating the tests, the benefits of continuous testing, and any challenges you faced while setting up the automation. Additionally, discuss how the automated testing process ensures faster feedback loops and contributes to maintaining a high-quality codebase.
4. **Code Quality and Maintainability:**Assess the overall quality of your test code, including its maintainability, readability, and scalability. Reflect on how the test code is structured, whether it can easily accommodate future changes to the software, and how it aligns with industry best practices for test automation.
5. **Future Considerations and Recommendations:** Offer recommendations for further improvements, both in terms of test case coverage and code optimization. Discuss any areas that may require additional attention or tests that could be added to further enhance the software's quality. You may also provide suggestions on improving the CI/CD pipeline, adding new tools, or using additional frameworks that could benefit the testing process in the future.

Your report should be written in a clear, **structured format**, with well-supported **arguments**and relevant **references**to demonstrate your knowledge and understanding of software testing, automation, and performance optimization. Ensure that your evaluation provides a balanced perspective, addressing both successes and areas for further development.

**Guidelines**

You should use the assessment submission link on the module’s Virtual Learning Environment page to submit the following files:

* **A PDF file for your report.**
* **A ZIP file of your software artefact. This should contain your Python files.**

This assignment should be submitted no later than **14:00 UK Time on 20th October 2025**.  You should ensure that you make a timely submission by the stated deadline.

Your work should be double-spaced in 12 point standard font (Calibri or Arial). You should use the Harvard referencing style as set out on [Cite Them Right](https://www.citethemrightonline.com/category-list?docid=CTRHarvard). Please make sure that you correctly cite and reference all secondary sources you use, and include a reference list. The reference list will not be included in your final word count.

You must include your 8-digit student number on the front of your submission

Don't forget that your orientation module contains several useful resources including:

* Academic Skills support
* How to avoid plagiarism and academic misconduct
* Extenuating Circumstances if your personal circumstances are affecting your studies

**The assessment must be completed individually. You must not share, in part or whole, your assessment with another party other than the module convenor and for the purpose of submission to the university. You must ensure that the University’s academic misconduct guidelines are followed in their entirety.**

**Grading**

This activity will be graded and you will receive feedback on it. The following rubric is provided to help you understand how you will be graded: [Final Assessment Rubric](https://hepuk.instructure.com/courses/3574/files/191418?wrap=1)[Download Final Assessment Rubric](https://hepuk.instructure.com/courses/3574/files/191418/download?download_frd=1)

**By submitting this assignment:**

I confirm that I have completed this Assignment to the best of my abilities. It includes all requirements, and the correct file has been uploaded to this assignment. In keeping with the University’s Academic Integrity & Plagiarism Policy, I declare that I have worked with integrity and that the materials included are my own. I accept responsibility for my work and have not used the materials of others or utilised any unauthorised collaborations including artificial intelligence for my assignment.

**Support Resources**

Before you undertake any assessment, please make sure you are familiar with the rules relating to academic integrity and the support available to help you improve your assessment technique. These can all be found in the **Orientation module** (available on your Canvas dashboard).

This include several really important and useful resources:

* Academic writing
* Academic integrity
* Referencing

There is also a useful reading list of study skills resources you can access through the library. These can be found [here](https://stmarys.alma.exlibrisgroup.com/leganto/readinglist/lists/3288982040007896?institute=44SMUN_INST&auth=SAML).

**Assessment Policy**

The University Assessment policies can be found be selecting the following link: [Assessment and Feedback](https://www.stmarys.ac.uk/ctess/Learning-and-Teaching/assessment-and-feedback/overview.aspx).

Please note that this includes a copy of the generic, University-wide assessment criteria. More specific, tailored assessment criteria will be provided for each separate piece of coursework within this module. Please also note appendix section 4 (Student's Responsibilities), which begins on page 24. This sets out the specific responsibilities of students, including the responsibility to ensure that work is submitted successfully and that it should be submitted in advance of the advertised deadline. Please note appendix section 8 "Late Submission," which states that "work submitted after the deadline without extenuating circumstances will be marked as zero."

If you are unable to submit work by the advertised deadline due to personal or professional circumstances you should apply for extenuating circumstances as soon as foreseeable. Please note: your course tutor cannot extend your submission deadline. You must submit an extenuating circumstance claim. Advice and guidance on how to submit an extenuating circumstances claim can be found by selecting this link: [Extenuating Circumstances](https://www.stmarys.ac.uk/registry/extenuating-circumstances.aspx).

If you have a 7-day extension, your deadline will be 27th October 2025.   
If you have mitigated your assessment, your deadline will be 5th December 2025.  
If you have failed a component, this must be resubmitted to the mitigated date of 5th December 2025.

Please note that in terms of mitigation or module fails, only outstanding or failed components should be submitted.

It is your responsibility to ensure that you submit your work on time to the relevant assessment page of your module.

**Feedback: You should receive feedback from your tutor in approximately 3 weeks.**