Week 1 Quiz 1

1. Which of the following are benefits to working with an LLM as part of software testing?

    - LLMs can help brainstorm scenarios for which tests should be developed.

      LLMs can be an excellent tool for helping brainstorm scenarios for testing, especially when provided sufficient context about the project overall and the specific parts of the software that need testing.

    - Code written by LLMs is of a high enough quality to not need to be tested.
    - LLMs can help develop test scripts for a given set of scenarios and prepare them for automated testing frameworks.

      LLMs can be a great tool for preparing test scripts for a set of scenarios either provided by a user or proposed by the LLM itself

    - LLMs replace the need for dedicated software testers.

2. Which of the following is NOT likely to improve the quality of the test cases developed by an LLM?

    - Asking an LLM to convert a set of functional tests into an exploratory testing plan
    - Providing the LLM information on the context of the project, libraries being used, and any test cases already known to be important.
    - Providing the LLM the code to be tested
    - Assigning the LLM a role as an expert software tester

    Typically the order of this process is reversed. Exploratory testing is done to understand the behavior of the software, and the insights from that experience inform the development of functional tests. An LLM can be helpful in this process when done in the expected order.

3. Which of the following statements does NOT accurately describe how LLMs can be used as part of software testing?

    - LLMs can brainstorm and propose test cases for you or testing colleagues to consider.
    - LLMS can develop tests designed for a specified testing framework or package.
    - Test cases developed by LLMs need to be reviewed by developers to ensure they properly align with a project's context and expected behavior.
    - LLMs are useful in functional testing but typically can't help with exploratory testing.

      LLMs can indeed support exploratory testing by suggesting scenarios, edge cases, or new perspectives that a tester might not have considered, thereby enriching the exploratory testing process. Exploratory testing benefits from creativity and insight, which LLMs can provide, even though the primary work of exploration and discovery rests with the human tester.

4. What is the main goal of Exploratory Testing?

   - To automate repetitive testing tasks.
   - To verify application performance under load.
   - To explore the application without predefined test cases and discover bugs.
   - To follow predefined test cases strictly.

   Exploratory testing involves testers using the software as end users would, without predefined test cases, to identify unexpected issues and bugs through natural interaction with the application.

5. How does functional testing differ from exploratory testing?

   - LLMs can only support functional testing, but not exploratory testing
   - Exploratory testing relies on predefined test cases while functional testing is more open-ended
   - Functional testing is more structured and ensures software meets predefined requirements
   - Exploratory testing needs to be updated after code changes while functional tests will remain valid

   Functional testing is more structured than exploratory testing and ensures software meets a set of predefined requirements.