Manual Testing

- Manual Testing is a software testing approach where testing activities are carried out by human testers.
- who follow test cases, execute test scenarios, and verify the software's behavior manually, without the use of automation tools.
- In manual testing, testers act as end users and interact with the software to find defects, ensure it meets requirements, and evaluate its usability.

1. Benefits of Manual Testing:

- **Early Testing:** Manual testing can be initiated even before the software reaches a stable state, allowing early defect detection.

- **Exploratory Testing:** Testers can explore the software and adapt their testing strategies based on what they discover during testing, making it suitable for finding unexpected issues.

- **User-Centric:** Testers can assess the software's user-friendliness and overall user experience, which can be challenging to automate.

- **Cost-Effective for Small Projects:** Manual testing requires minimal upfront investment in automation tools and scripts, making it cost-effective for small-scale projects or one-time testing efforts.

- Testers can apply critical thinking, domain knowledge, and intuition to identify potential defects and assess the software's business logic.

2. Drawbacks of Manual Testing:

- **Time-Consuming:** Manual testing can be time-intensive, especially for repetitive or extensive test cases, and may slow down the software development process.

- **Resource-Intensive:** It requires a team of skilled testers, which can be costly, and it may not be feasible for large-scale, continuous integration projects.

- **Human Error:** Testers can make mistakes or overlook defects due to fatigue, inconsistency, or misunderstanding of requirements.

- **Limited Test Coverage:** Manual testing may not adequately cover all test cases, edge cases, or scenarios, leading to incomplete test coverage.

- **Difficulties in Regression Testing:** Repeatedly executing test cases manually during regression testing can be error-prone and time-consuming.

Manual Testing

- **Examples of Manual Testing:**

1. **Exploratory Testing:** Testers explore the software without predefined test scripts, focusing on finding defects, understanding functionality, and assessing usability.

2. **Usability Testing:** Testers assess the software's user-friendliness, navigation, and overall user experience, reporting any usability issues.

3. **Ad-Hoc Testing:** Testers perform informal testing without predefined test cases, often to quickly validate a new feature or component.

4. **User Acceptance Testing (UAT):** End users validate that the software meets their requirements and expectations, often before the software's release.

5. **Smoke Testing:** A quick, high-level test to verify that the software's critical functionalities are working after a build or release.

6. **Localization Testing:** Ensuring that the software functions correctly in various languages, cultures, and regions.

7. **Compatibility Testing:** Manually verifying the software's compatibility with different browsers, devices, or operating systems.

8. **Accessibility Testing:** Evaluating the software's accessibility features for people with disabilities, such as screen readers and keyboard navigation.

9. **Manual Regression Testing:** Repeating a subset of test cases to ensure that new code changes did not introduce defects or regressions.