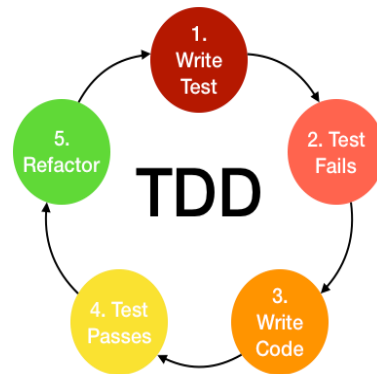


Assignment 1: Create an infographic illustrating the Test-Driven Development (TDD) process. Highlight steps like writing tests before code, benefits such as bug reduction, and how it fosters software reliability.

Test-Driven Development (TDD) Process:



Write Test Cases:

- Developer writes test cases based on requirements before writing any code.
- Tests are designed to fail initially.

Run Tests:

- Execute the test suite to ensure all tests fail as expected.

Write Code:

- Developer writes code to pass the failing tests.
- Focus on writing the simplest code to make the tests pass.

Run Tests Again:

- Execute the test suite again to check if the new code passes all tests.

Refactor Code:

- Improve the code without changing its external behaviour.
- Ensure tests continue to pass after refactoring.

Repeat the steps to check all the test cases.

Benefits of TDD: -

Bug Reduction:

- Early detection and fixing of bugs before they become costly.
- Continuous testing prevents bugs from proliferating.

Increased Reliability:

- Rigorous testing ensures software behaves as expected.
- Confidence in the reliability of the codebase.

Improved Design:

- Forces developers to focus on modular, well-structured code.
- Encourages writing code with testability in mind.

Faster Development:

- Reduces time spent on debugging and fixing issues.
- Prevents regressions, leading to faster development cycles.

Test-Driven Development (TDD) is a disciplined approach to software development that prioritizes writing tests before writing code. By following this process, developers can significantly reduce bugs, enhance software reliability, and improve overall code quality.