

2) Developer creates a test that specifies the desired behavior. Initial test fails as there's no corresponding code.

3) Developer writes the simplest code to make the test pass.

4) Developer ensures all previous tests pass, including the new one. Improve code without changing its behavior.

**Benefits of TDD :**

Fewer Mistakes: TDD catches mistakes early, making them easier to fix.

Faster Development: Despite writing tests first, TDD speeds up development by reducing time spent on fixing bugs later.

Clearer Code: TDD encourages writing code in smaller, easier-to-understand pieces.

Confidence in Changes: With good tests, developers can make changes knowing that any issues will be caught.

Improved Collaboration: TDD fosters collaboration between developers and testers, leading to better communication.

Better Understanding: Writing tests first helps developers understand what the code should do.

**More Reliable Software: TDD leads to more reliable software by catching bugs early and encouraging clear code.**