Test-Driven Development (TDD) Infographic

Test-Driven Development (TDD) Process

1. Write a Test

- Define a test for a new function or feature.
- Ensure it specifies the desired behavior.

2. Run the Test

- Execute the test to see it fail.
- Confirm the test correctly identifies the missing feature.

3. Write Code

- Develop the minimum code needed to pass the test.

4. Run All Tests

- Execute all tests to ensure the new code passes and doesn't break existing functionality.

5. Refactor Code

- Optimize and clean up the code.
- Ensure no functionality is altered during refactoring.

6. Repeat

- Continue the cycle for new features and improvements.

Benefits of TDD

- **Bug Reduction**: Early detection and resolution of issues.
- **Software Reliability**: Ensures code meets requirements and works as expected.
- **Improved Design**: Encourages simpler, cleaner code.
- **Documentation**: Tests serve as live documentation of system behavior.

Visual Layout

Top Section: Title - "Test-Driven Development (TDD) Process"

Middle Section: Steps in a Circular Flow

1. Write a Test
2. Run the Test
3. Write Code
4. Run All Tests
5. Refactor Code
6. Repeat

Bottom Section: Benefits

- Bullet points highlighting bug reduction, software reliability, improved design, and documentation.