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.

Infographic: Test-Driven Development (TDD) - Building Reliable Software

Center Text: Test-Driven Development *Building Reliable Software One Test at a Time*

Left Side (Red Circle): Failing Test

- Illustration: A red X mark with a frowning code snippet

Description:

- Start with a failing test. This test defines a desired feature or functionality that isn't yet implemented.

Right Side (Green Circle): Minimal Code

- Illustration: A green checkmark with a small code snippet

Description:

- Write the minimum amount of code to make the failing test pass. This focuses on the essential functionality.

Center Arrow (Double Headed): Refactor

Description:

- Refactor the code to improve its design, readability, and maintainability without affecting functionality.

Bottom Section: Benefits of TDD

- **Reduced Bugs:** Early and frequent testing catches errors early in the development cycle.
- **Improved Design:** Focus on passing tests encourages clean and maintainable code.
- **Higher Code Coverage:** Tests ensure all areas of code are properly exercised.
- **Enhanced Confidence:** Reliable tests provide assurance that the code functions as intended.
- **Agile Development:** Supports iterative development with frequent feedback loops.

Bottom Center: Reliable Software

- Illustration: A checkmark shield icon

Text at Bottom: TDD leads to robust and dependable software that meets user requirements.

Assignment 2: Produce a comparative infographic of TDD, BDD, and FDD methodologies. Illustrate their unique approaches, benefits, and suitability for different software development contexts. Use visuals to enhance understanding.

Infographic: Development Methodology Showdown: TDD vs. BDD vs. FDD

Center Text: Building Software the Agile Way: Choose Your Weapon!

Top Section - Three Sections with Icons

- **Left Section (Blue): Test-Driven Development (TDD)**
    - Icon: Microscope hovering over lines of code
    - Text: **Focus:** Writing unit tests before code (red-green-refactor cycle)
- **Center Section (Green): Behavior-Driven Development (BDD)**
    - Icon: Puzzle pieces with gears and speech bubbles
    - Text: **Focus:** Collaboration on user stories and scenarios to drive development
- **Right Section (Orange): Feature-Driven Development (FDD)**
    - Icon: Checklist with cascading timelines
    - Text: **Focus:** Breaking down projects into features and iterative development cycles

Middle Section - Strengths and Applications

- **Sections Divided by Arrows for Easy Comparison**
- **TDD:**
    - Strengths (Icon): Checkmark next to a bug
        - Text: Reduces Bugs, Improves Code Design
    - Applications (Icon): Screwdriver and Code Snippet
        - Text: Unit-level testing, Core functionalities, APIs, Libraries
- **BDD:**
    - Strengths (Icon): Lightbulb with gears
        - Text: Clear Communication, Faster Feedback Loops
    - Applications (Icon): Diverse Team at a table
        - Text: Collaborative projects, User-centric development, Integration testing
- **FDD:**
    - Strengths (Icon): Calendar with puzzle pieces
        - Text: Manages Complex Projects, Streamlines Development
    - Applications (Icon): Large building under construction
        - Text: Enterprise applications, Mission-critical systems, Long-term projects

Bottom Section:

- **Text:** The optimal methodology depends on your project's specific needs. Consider team size, project complexity, and stakeholder involvement.