**Q1. What is Test-Driven Development (TDD) and Behavior-Driven Development (BDD); and explain difference between them.**

**Answer:** Test-Driven Development (TDD) and Behavior-Driven Development (BDD) are two software development methodologies that focus on writing automated tests as a primary activity during the development process.

1. **Test-Driven Development (TDD):**

TDD is a development approach in which tests are written before any code is developed. In TDD, the developer writes a test case that defines the desired behavior of the code. The developer then writes code to satisfy that test case. Once the code is written, the developer runs the test case to verify that the code works as expected. If the test passes, the developer writes another test case and repeats the process. If the test fails, the developer modifies the code and re-runs the test case until the test passes.

The key benefits of TDD include:

* **Higher code quality:** By writing tests before writing code, developers ensure that the code they write meets the desired behavior and is free from bugs.
* **Faster feedback:** TDD allows developers to quickly detect and fix issues in the code, leading to faster development and fewer bugs.
* **Improved design:** TDD encourages developers to write clean and modular code that is easy to maintain and extend.

1. **Behavior-Driven Development (BDD):**

BDD is a software development methodology that emphasizes collaboration between developers, testers, and business stakeholders to define and validate the behavior of the system. In BDD, the desired behavior of the system is described in terms of user stories and scenarios written in a natural language format that is easily understandable by non-technical stakeholders.

BDD tests are often written using a specialized testing framework, such as Cucumber or SpecFlow, which allows stakeholders to read and understand the tests without needing to know how to write code.

The key benefits of BDD include:

* **Better collaboration:** BDD promotes collaboration between developers, testers, and business stakeholders, leading to better communication and shared understanding of the system's behavior.
* **Clear requirements:** BDD tests are written in a natural language format, which makes it easier to communicate and understand the system's behavior.
* **Reduced rework:** By defining and validating the system's behavior upfront, BDD reduces the likelihood of rework and ensures that the system meets the desired behavior.

1. **Difference between TDD and BDD:**

The main difference between TDD and BDD is the level of abstraction at which tests are written. TDD tests are written at a lower level of abstraction and focus on the behavior of individual code units or modules, while BDD tests are written at a higher level of abstraction and focus on the behavior of the system as a whole.

In TDD, tests are typically written using programming languages and frameworks, while BDD tests are often written using natural language formats that are easier to read and understand by non-technical stakeholders.

Overall, TDD and BDD are complementary methodologies that can be used together to improve the quality and behavior of software systems.