Add Test cases ( Java or Ruby lanagauage which is derived from requirement)

----> See Test fail ( As code is not fully developed) ---> Write code to

Pass ------> Run Tests ---> Refactor code-----This loop will continie

TDD : Per functionality basis

2> Develop Centric

3> There is no role of customer

4. Good Enough for simple cases.

Example: Junit, TestNG and NUnit Framewordks

#### ########TDD (Is the code correct? ############

**Step 1: Add a test**

The first step is to take a requirement and turn it into a test that is clear so that the developer can fully understand the feature specifications.

**Step 2: Run all tests and see if the new test fails**

Next step is to run the test to make sure that it fails. If the new test passes, that means that the required behavior already exists and new code is not needed or that the new test is flawed and needs to be modified.

**Step 3: Write the code**

Third, the developer writes code that would make the test pass. Code quality is not as important at this stage; the developer’s goal is to write code that will make the test pass.

**Step 4: Run Tests**

The fourth step is to run all the tests. If all tests pass, we are satisfied that the code meets the requirements and that it did not affect existing features. If not all tests pass, the developer must continue to adjust.

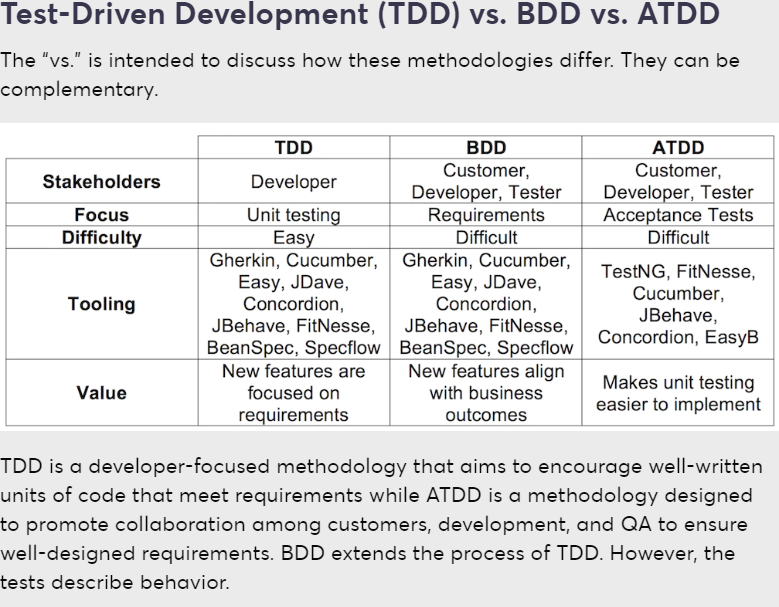
**Step 5: Refactor code**

refactoring a code i.e. changing/adding some amount of code to the existing code without affecting the behavior of the code.

Now that we passed all the tests, the new code must be cleaned up to meet the hygiene standards.

**Step 5: Repeat**

“Repeat” is not necessarily a next step but a reminder that the developer needs to repeat the process if there are new requirements or wants to improve the functionality of the software.



####################

TDD isn’t about testing

Test-driven development focuses on guiding developers

The most popular BDD frameworks used by effective developers worldwide are

Cucumber, JBehave, Easyb, Concordion.

TDD (Test-driven development) is a software development technique

The most tricky thing about TDD for several developers is the fact you have

to write your tests before writing code.

TDD vs. BDD

· Unlike Test-driven development test cases written in programming

languages like Java, Ruby, etc., Behaviour-driven development is in a more

readable format by each stake holder since it is in simple English.

######################

TDD Unit-Testing - Unit testing is a part of the test-driven development

(TDD) methodology that requires developers to first write failing unit

tests. Then they write code in order to change the application until the

test passes.

BDD Unit-Testing - Nowadays, unit testing is done with BDD(Behavior Driven

Development) which is extension of TDD where user or tester create the

automated specifications in plain english language and developer write the

underlying code that implements the test.

##############

The "process" starting creating the scenarios of your feature(BDD). For

each scenario, start coding using TDD. Repeate the cycle until you have no

more scenarios left.

#######################################

Basically, in both TDD and BDD the programmers write unit tests.

In BDD, the programmers write the code and then the test.

In TDD it’s the other way around - the programmers write the test and then

the code. It sounds a bit strange at the beginning, but once you get used

to it, it really improves the development process.

#########################