

Once in a while we came across the situation where Management or engineering Mangers asked us to bring the best development practices in the team and we are not sure what those are and wished we knew it so that we can add more value to the team.  
Not only this but if you apply for a Scrum Master or Agile Coach job , In JD you will find one criteria like good Knowledge on best engineering practice like - TDD , BDD , Pair Programming , Unit Testing , Code refactoring and many more..   
  
So today I am going to share the details about engineering practices TDD, BDD, Unit Testing ,Code Refactoring. I hope it will give a fair understanding.  
  
So let's Start …  
  
What is TDD?   
TDD (Test Driven Development) is an iterative development process where developers write the test cases first for each small functionality before the code is written.   
  
How does the TDD work?  
Simple steps:   
🔹Developer read a requirement/ User story and write the test.  
🔹He run the test and it will fail since no code is written for it.   
🔹Then he’ll write the code to fix the failed test cases.  
🔹He run the test again and it will pass now.  
🔹Repeat above loop for every new functionality.  
  
Still not clear   
let’s understand with real world example -  
  
🔹You have written a function to add two numbers:  
int Add(int a, int b) { return a+b; }  
  
🔹A unit test code would look something like this:  
void TestAdd1() { Assert.IsEqual(Add(5, 10), 15) }  
  
The above unit test “asserts” that 5 + 10 is equal to 15. If the Add function returns anything else, it will show error and the test case will fail.Then developer corrects the code to pass the unit test case.  
  
Tools for TDD –   
🔹Junit - Designed for the Java  
🔹 JMeter -Dynamic Web applications (Mainly for Load/Performance testing)  
🔹Mockito – Used for REST API testing.   
🔹Watir – Used to automate the browser based tests  
  
Benefits of TDD –   
🔹By doing tests first reduce the defect in the beginning itself.   
🔹Avoid duplicates of code.   
🔹TDD drive the code design and approach simple, cleaner, bug free.  
  
Bonus Point - Above which you read is nothing but “Unit Testing” .   
  
Now Let’s Talk about BDD   
  
What is BDD ?   
BDD -Behaviour Driven Development is a practice where we write the test cases in simple text which end users or business people can understand.   
  
Lets understand with an example –   
For logging in with invalid credential   
  
🔹Given I am at the login page  
🔹When I fill the email with the value “[incorrectaluev@email.com](mailto:incorrectaluev@email.com)”  
🔹And I fill the password with the value “IncorrectPassword”  
🔹And I click on Login  
🔹Then a text “Wrong Email & Password”   
  
Now write the code to pass the failed testcases.   
  
Tools for BDD:   
  
🔹Cucumber (Ruby framework)  
🔹SpecFlow (.NET framework)  
🔹Behave (Python framework)  
🔹JBehave Web (Java framework)  
🔹Behat (PHP framework)  
  
The benefits of BDD are that it's easy to understand by business users.