Class Notes-25

What is framework ?
Test automation framework is an environment that we write and execute tests. Test automation framework uses different tools, designs in order to efficiently generate, run and report automated tests. In framework we use different packages, folders to make them more efficient,scalable,understadable etc.

java--> our framwork us written in java lang.
maven--> build managamenet tool we use for managing our dependencies and plugins, run from terminal
selenium --> we use for automate the browser

testng --> used to create tests, run tests (with @test annotations) , generate reports (before and after , not only with testng), building smoke/regression suites (with testing xml runner) , assertions, annotations

intellij--> ide taht we used, where we write our code.
extent reports --> used to generate html report with steps, metricts, screenshots
apache poi --> read/write excel files
git/github --> version control
jenkins --> to schedule smoke/regression tests,send email to team etc.

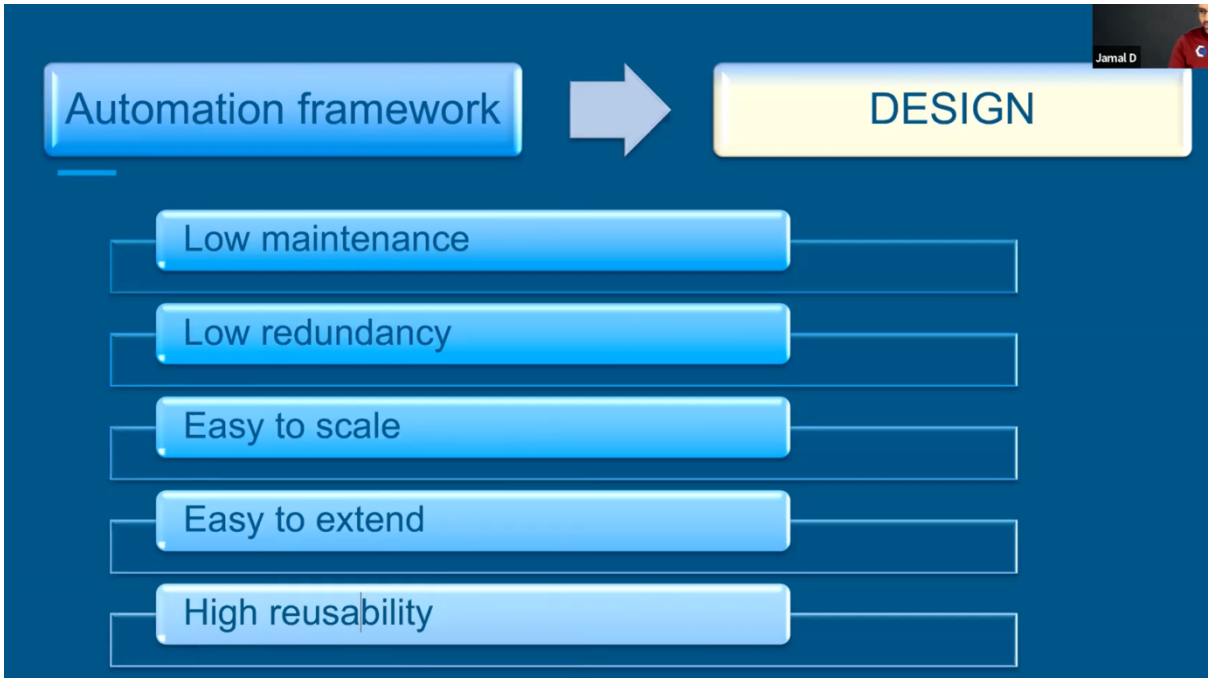
HOW DID WE USE OOP CONCEPT IN OUR FRAMEWORK ?

Inheritance
we use base classes such as test base in our framework. this class contains the properties and methods which are common to all test classes. all test classes extend this testBase class.

Encapsulation
we have a driver class that uses private driver variable and we use the public getter method to access this.

Abstraction
we have and abstract basepage where we put the logic to common all the page object classes. my page object classes extend the base class. when we create object of the any page object, we can access the method from abstract class.

Polymorphism
we use method overloading in our framework in multiple instances.
we have overloaded utilites methods which can take different types of arguments. sometimes we can pass webelements, or by locators.



Low redundancy →less repetition --fazlaliklardan kacinma , ayni codelari surekli olarak yazip isi uzatmama , ornegin class extends TestBase , before and after I bir yere koyuyoruz ,sonra butun test case lerde yazmaya gerek kalmiyor

Page object model ile we can centralize our locaters and if they are changed , we can change it in one place. You don't need to go ,find them in the test case

bASEpAGE VE tESTbASE I KARISTIRMAYALIM

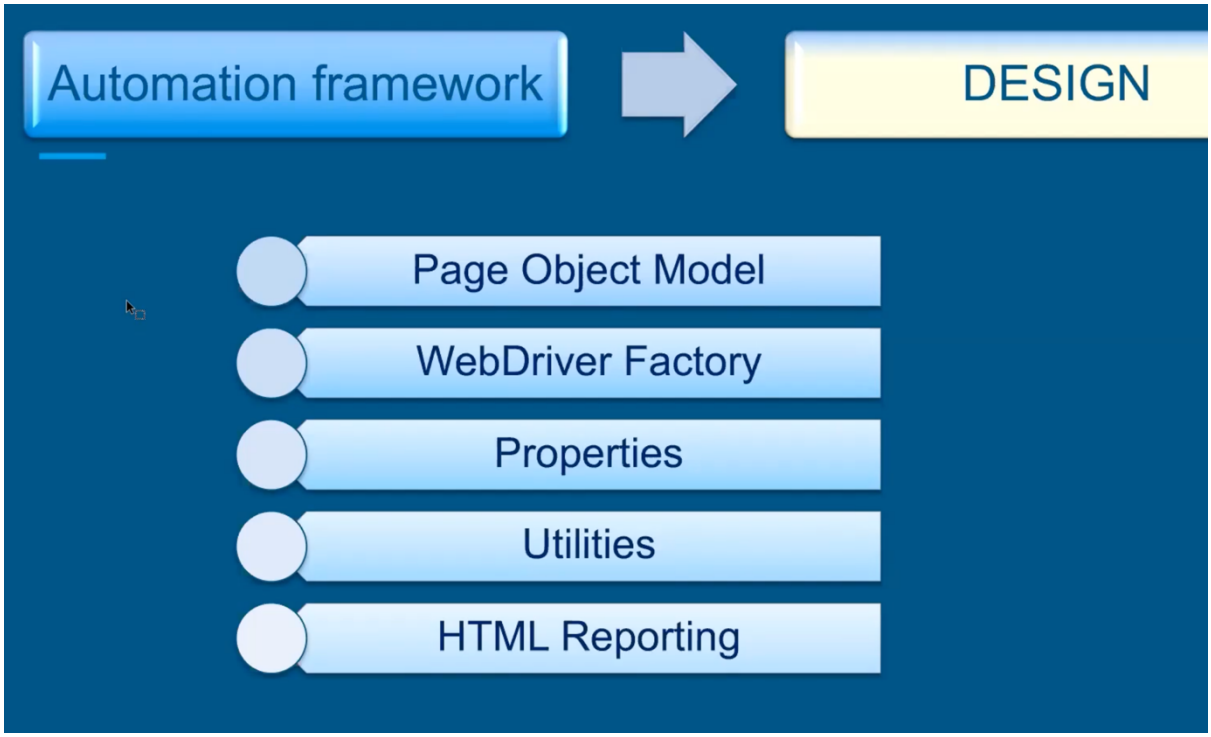
BasePage is part of page object model , testBase is where we keep our testing annotations

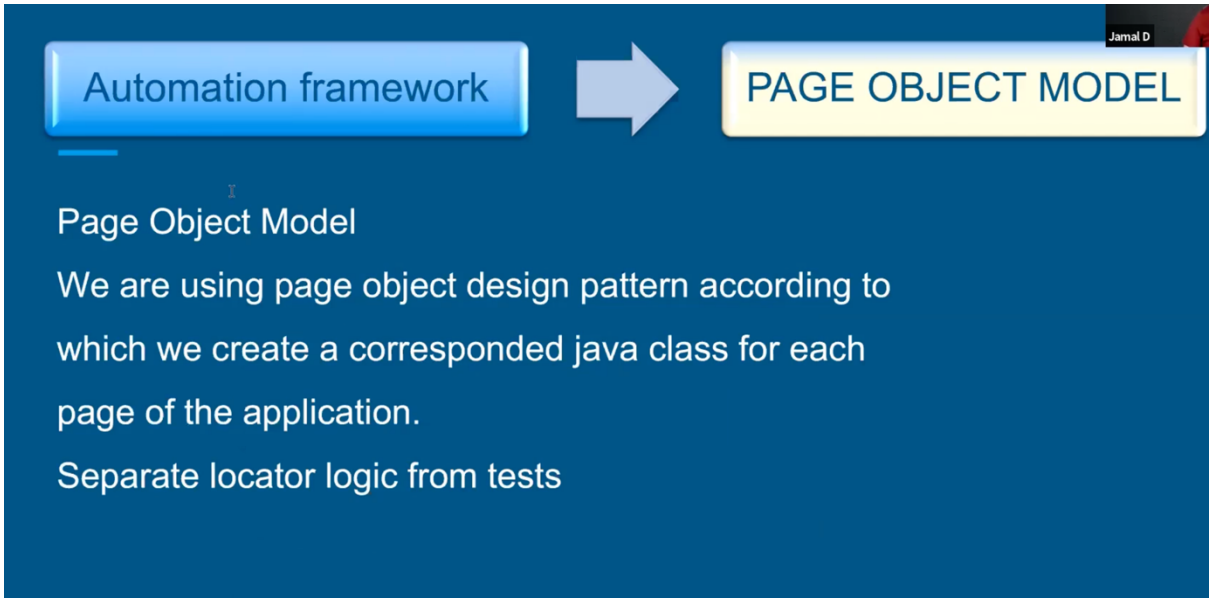
You might have testBase even if you are not implementing to page object model, for example you are working as a tester and you don't implement page object model , you keep your locaters inside your test cases bu durumda bile yinede kullanabilecegimiz bir class

Easy to scale→easy to add new test cases

Easy to extend→ it is easy to add new features

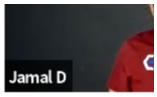
High reusability→ methods



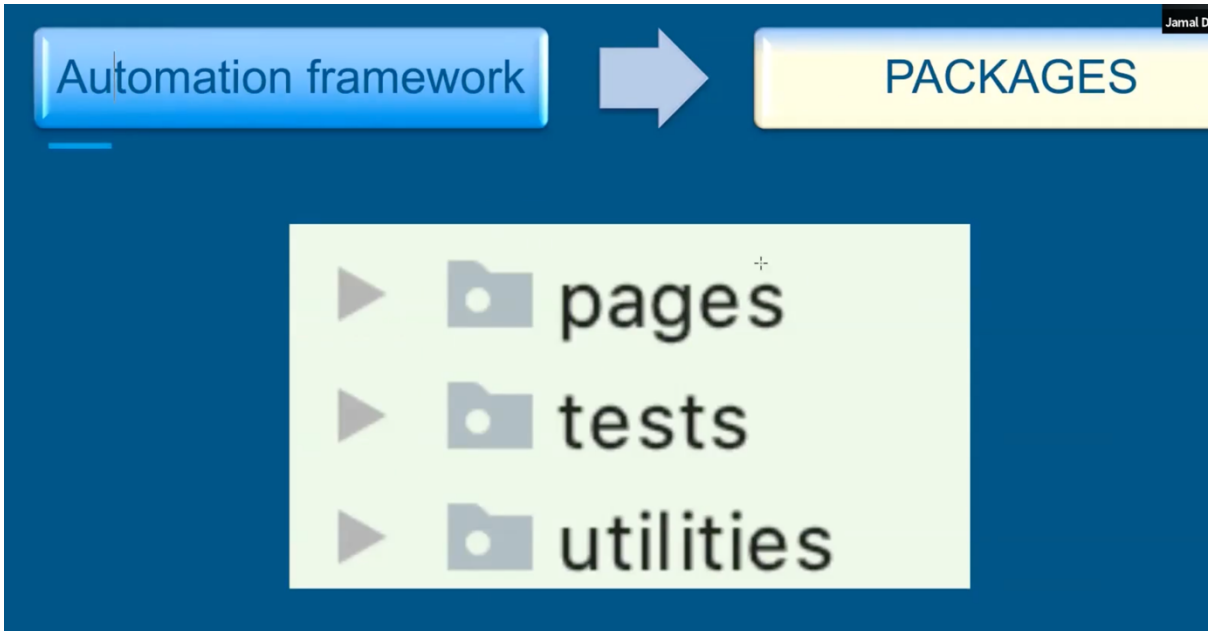


Why page object model? Diye baslarsak ve guzel aciklarsak etkilenirler

There are some companies they don't implement page object model , they locate the web elments in regular way , like below. One day if locater changes , you need to update it. We can use page object model to centralize the locaters , if there is a change, we just change it in one place , and it will update all the related test cases. Ornegin we created a pages package .By the way , page object model is completely unique to the project , you cannot copy paste , maybe copy the structure but you cannot copy the any locator or any method



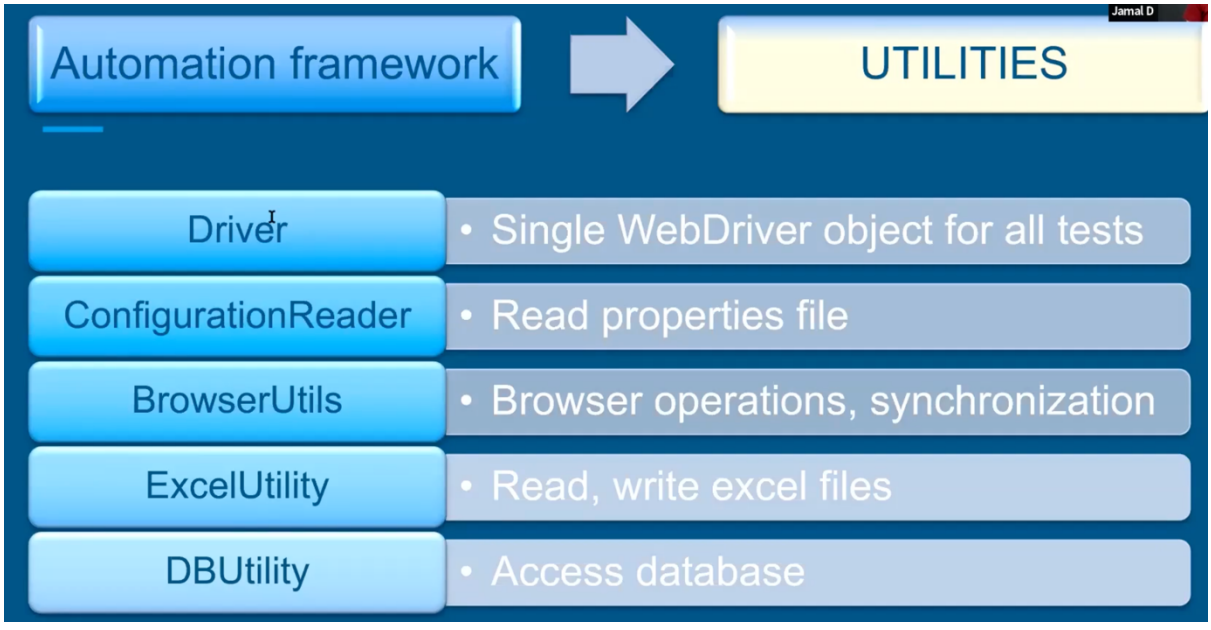

```java
@Test
public void test1() throws InterruptedException {
    WebDriver driver = WebDriverFactory.getDriver( browserType: "Chrome");
    driver.get("http://practice.cybertekschool.com/checkboxes");

    WebElement checkbox1 = driver.findElement(By.xpath("//input[1]"));
    WebElement checkbox2 = driver.findElement(By.xpath("//input[2]"));

    //How to verify checkbox is selected or not ?
    System.out.println("checkbox1.isSelected() = " + checkbox1.isSelected());
    System.out.println("checkbox2.isSelected() = " + checkbox2.isSelected());
    //verify checkbox 1 is not selected, 2 is selected
    Assert.assertFalse(checkbox1.isSelected(), message: "verify checkbox 1 is NOT selected")
    Assert.assertTrue(checkbox2.isSelected(), message: "verify checkbox 2 is selected");

    //how to check checkboxes?
    //just like a radio button we use click() method
    Thread.sleep( millis: 2000);
    checkbox1.click();

    //verify after click
    Assert.assertTrue(checkbox1.isSelected(), message: "verify checkbox 1 is selected");
    Assert.assertTrue(checkbox2.isSelected(), message: "verify checkbox 2 is selected");

    Thread.sleep( millis: 2000);
    driver.quit();
```



Testlerimizi tests packagenin icine koyuyoruz

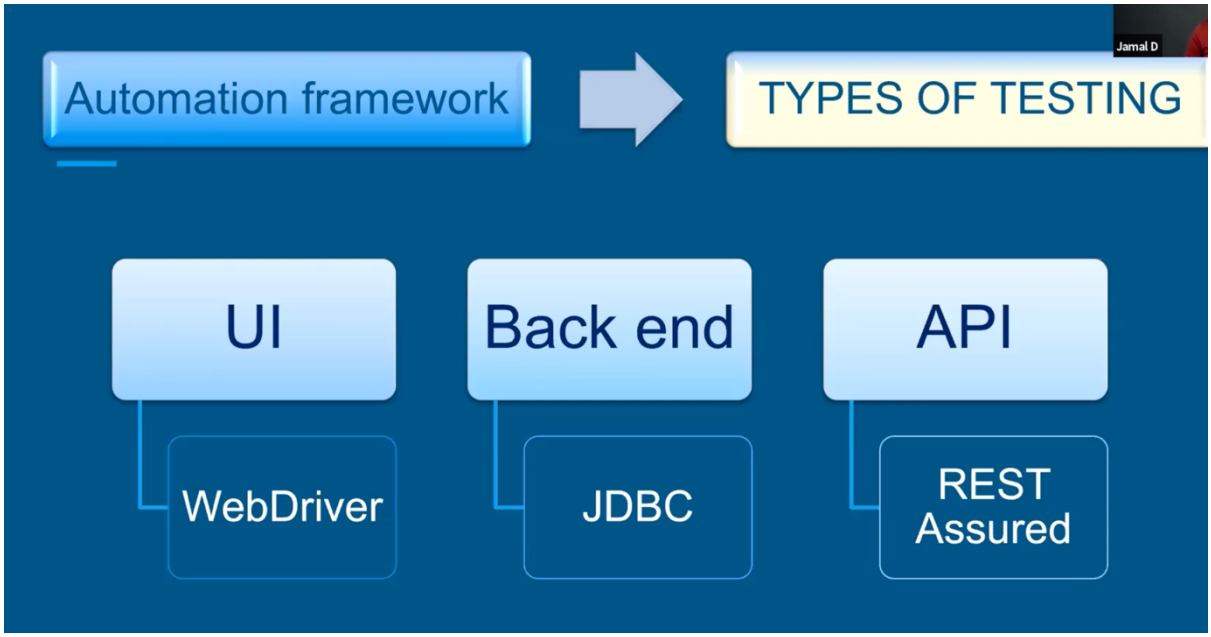
Pages--- page objects

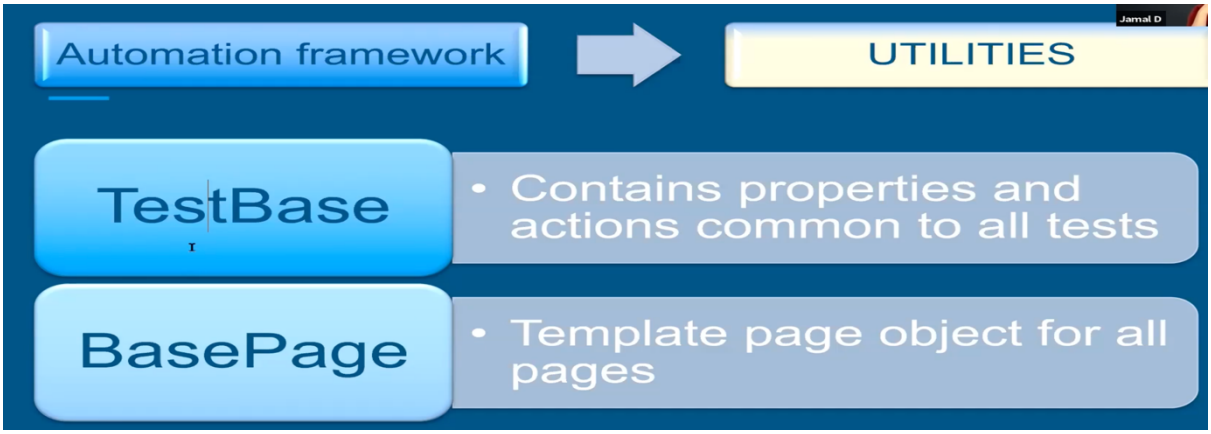
Utilities---Webdriver, file,browser,base

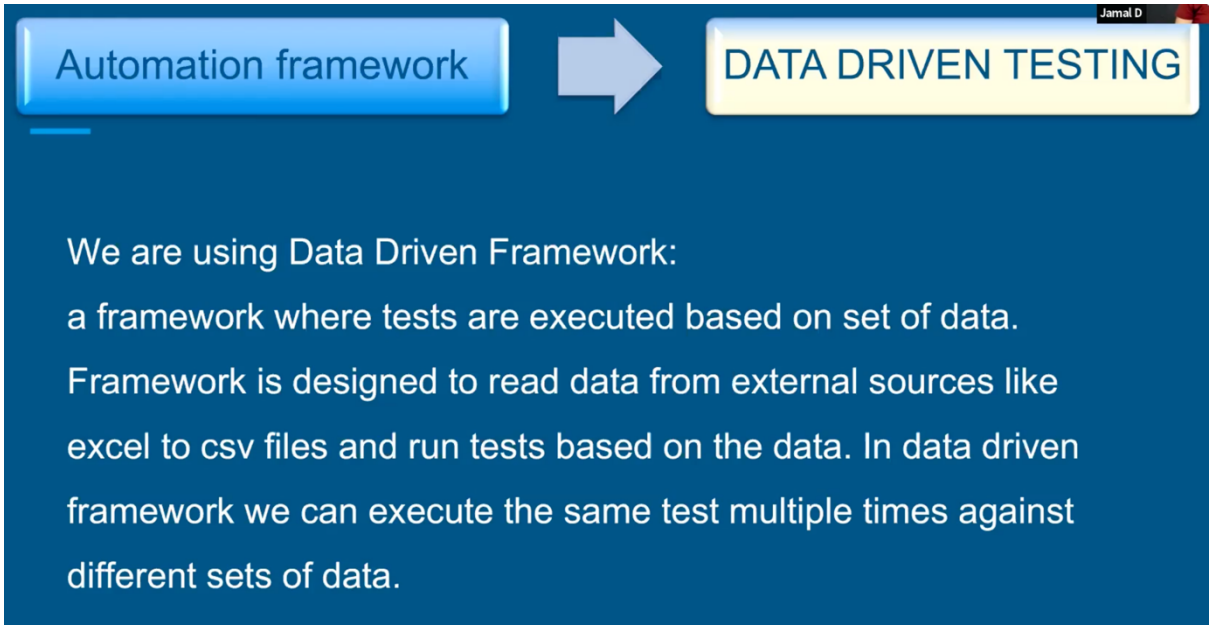


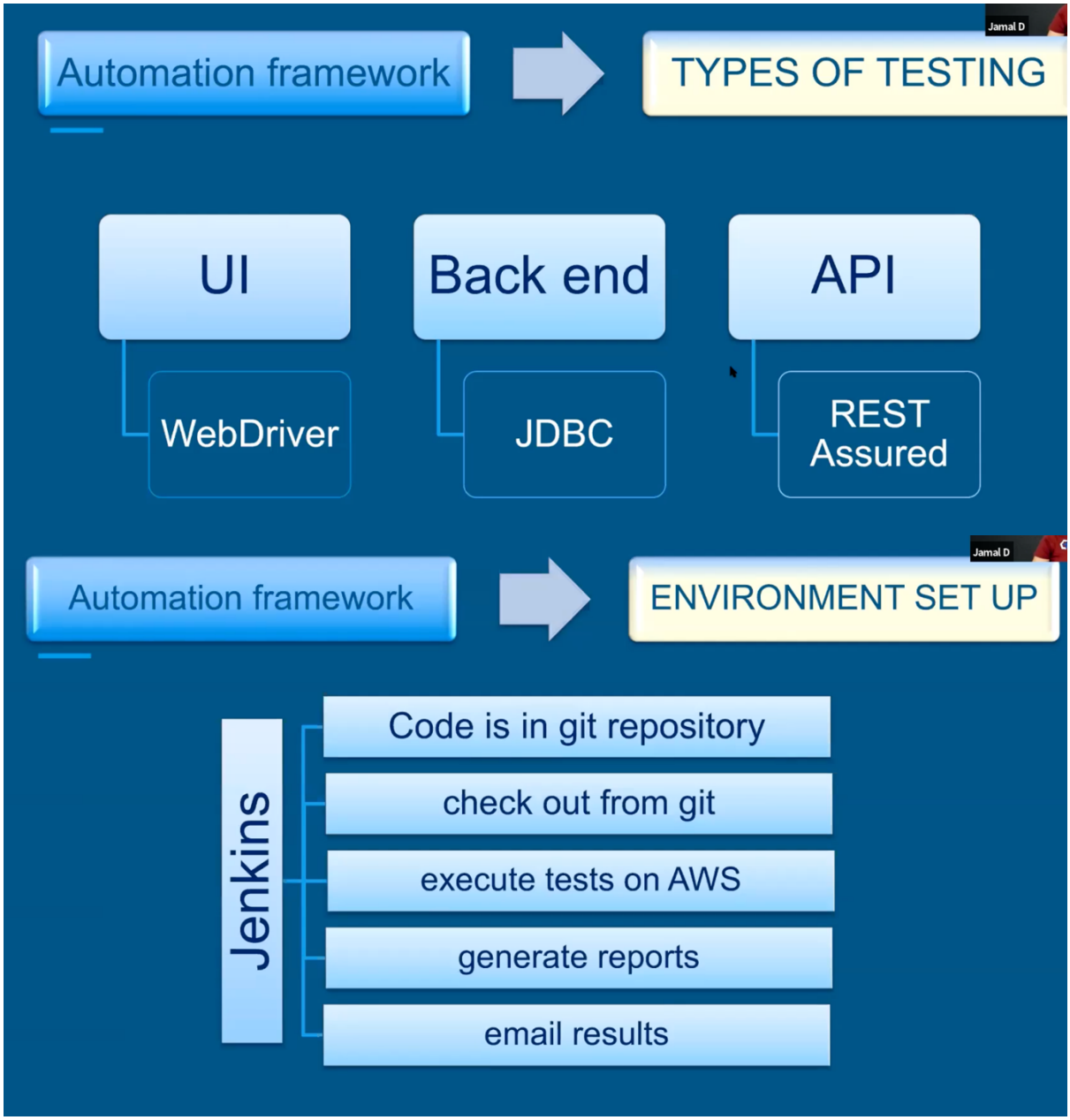
Driver is singleton, what is its benefit? Same object all the time
Driver.get() → will point the same object during the execution time → browserutil classinda switchtowindow methodunu incele

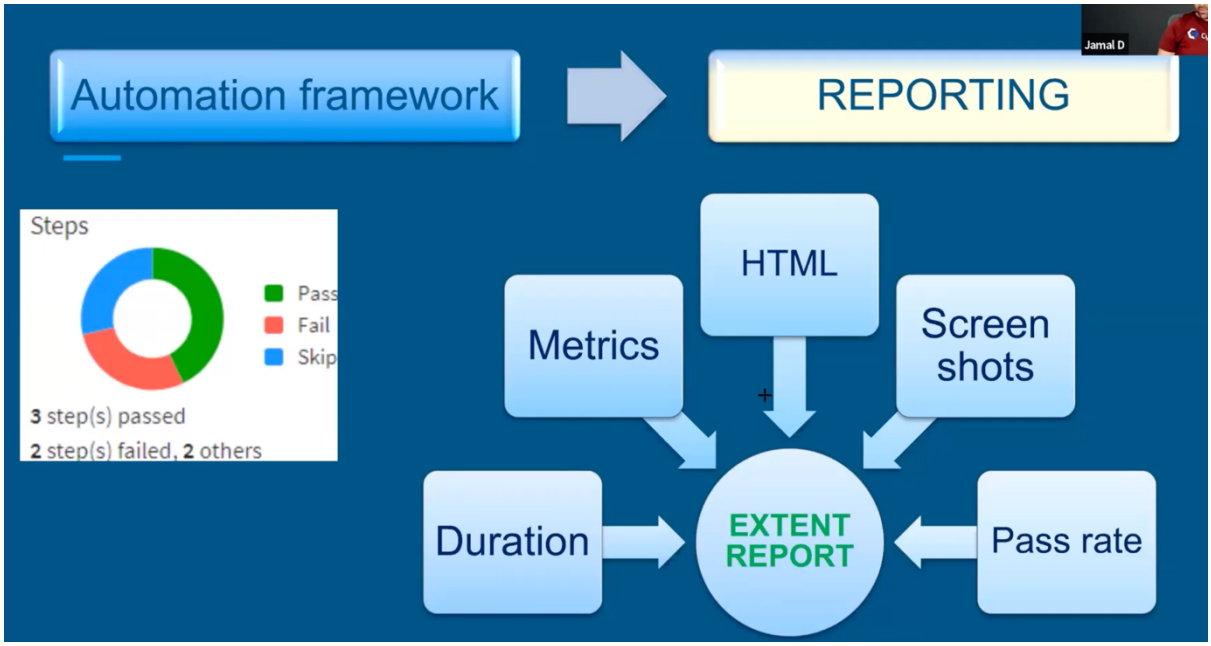
ConfigurationReader- reads configuration.properties -baglanti saglar properties ilr


Jamal D
Automation framework
TYPES OF TESTING
UI
Back end
API
WebDriver
JDBC
REST Assured


Jamal D
Automation framework
UTILITIES
TestBase
• Contains properties and actions common to all tests
BasePage
• Template page object for all pages


Jamal D
Automation framework
DATA DRIVEN TESTING
We are using Data Driven Framework:
a framework where tests are executed based on set of data.
Framework is designed to read data from external sources like excel to csv files and run tests based on the data. In data driven framework we can execute the same test multiple times against different sets of data.


Jamal D
Automation framework
TYPES OF TESTING
UI
Back end
API
WebDriver
JDBC
REST Assured
Jamal D
Automation framework
ENVIRONMENT SET UP
Jenkins
Code is in git repository
check out from git
execute tests on AWS
generate reports
email results


Jamal D
Automation framework
REPORTING
Steps
Pass
Fail
Skip
3 step(s) passed
2 step(s) failed, 2 others
HTML
Metrics
Screen shots
Duration
EXTENT REPORT
Pass rate