

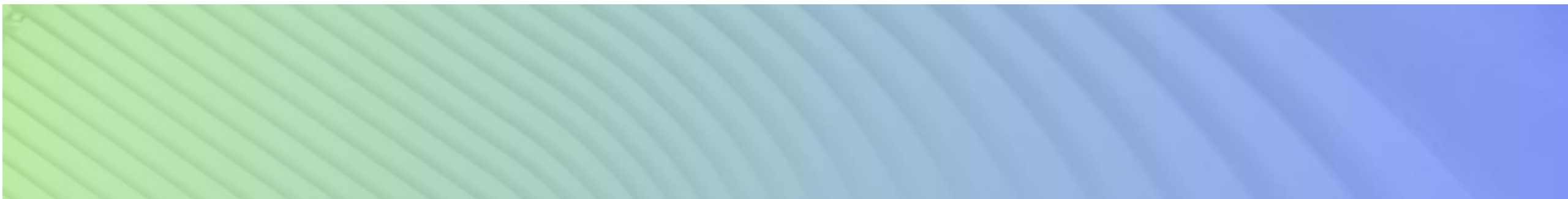
Batch 44

SELENIUM

01

15.01.2022

Selenium giris



Selenium giriş SELENIUM GIRIS

OTOMASYON TEST NEDIR?

SELENIUM PROJESİ OLUSTURMA



SELENIUM NEDIR

- Selenium, web uygulamalarını farklı tarayıcılarda ve platformlarda test etmek için ücretsiz (açık kaynaklı) bir araçtır.
- Selenium yalnızca web tabanlı uygulamaları otomasyon yapmaya odaklanır. Mobil ve Windows testi yapmak için eklentiler selenium'a eklenebilir.
- Selenium jar dosyaları ile kurulabilir. Kurulum sırasında jar

About Selenium

Selenium is a suite of tools for automating web browsers.

The Selenium Browser Automation Project

Selenium is an umbrella project for a range of tools and libraries that enable and support the automation of web browsers.

- Selenium, otomasyon yapmak için kendi sınıflarına veyöntemlerine sahip bir kütüphanedir..
- 2021'de piyasaya sürülen Selenium 4'ü öğreneceğiz.
- Selenium'u çeşitli programlama dilleri ile yazabilirsiniz java, Python, ruby, .Net vb.



IntelliJ IDEA NEDİR

- IntelliJ IDEA 2000 Yılında Kurulmuş Olan JetBrains İsimli Yazılım Firmasına Ait Olan, Popüler Bir Java Editörüdür.. (ide)
- IntelliJ IDEA'nın her yönü, geliştirici üretkenliğini en üst



düzeye çıkarmak için tasarlanmıştır.

- Akıllı kodlama yardımı ve ergonomik tasarım ile, kod yazimini yalnızca verimli değil, aynı zamanda keyifli hale getirir.
- Bircok Framework ve plugin ile calisma imkani verir
- Akilli tamamlama ozelligi ile kod yazimini oldukca kolaylastirir
- IDE ihtiyaclarinizi tahmin eder ve sıkıcı ve tekrarlayan geliştirme görevlerini otomatikleştirir, böylece büyük resme odaklanabilirsiniz.



SOFTWARE **T**ESTING **N**EDİR ?

EXPECTED RESULT (beklenen sonucun), **ACTUAL RESULT** (gerçek sonuca) esit olup olmadigini kontrol etme işlemidir.

- Eger Expected result = Actual result, ise status **PASSED**
(test basarili)
- Eger Expected result !=Actual result, ise status **FAIL**
(test basarisiz)



Sonuç olarak, olması gereken şeylerin olmadığını veya olmaması gereken şeylerin olduğunu kontrol etmek ve ortaya çıkartmak yazılım testinin amacı olmalıdır.

- Her User Strory icin Positive ve Negative Test(ler) yapılmalıdır
- Test, musteri/isletme ihtiyaclarini karsilamak icin yapilir.
- Bir uygulamayi test etmek icin onceden belirlenmis user storyler (kullanici hikayeleri) ve tanimlanmis acceptance criterias (kabul kriterleri) dikkate alinir.



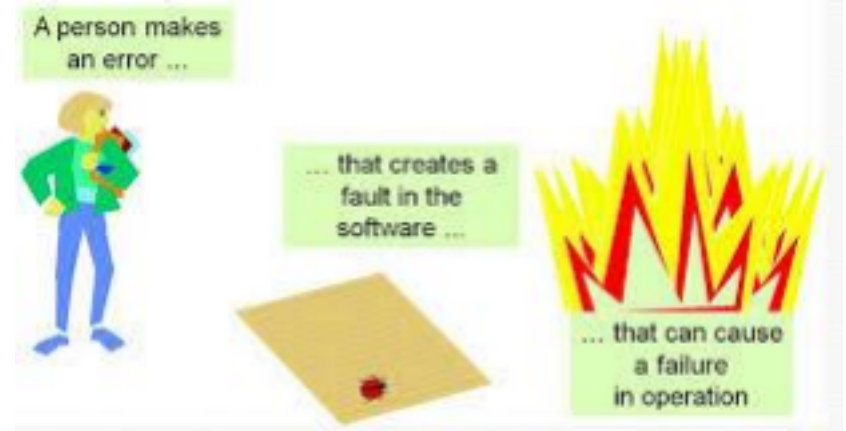
SOFTWARE TESTING NEDEN ÖNEMLİDİR ?

- İnsanlar hata yaparlar, bu hatalar kodda, yazılımda, sistemde ya da dokümanda defect (Kusur) oluşturur.

- Defect olan kod çalıştırıldığında sistem beklenen fonksiyonları gerçekleştiremez ve başarısız olur.

Bu sebeplerden dolayı;

- Müşteriye sunulmadan önce ürün kalitesinden emin olmak,
- Yeniden çalışma (düzeltme) ve geliştirme masraflarını azaltmak,
- Geliştirme işleminin erken aşamalarında yanlışları saptayarak ileri aşamalara yayılmasını önlemek, böylece zaman ve maliyetten tasarruf sağlamak amaçlarıyla ürün müşteriye sunulmadan önce test edilmesi gerekmektedir.



KISACA : Testing saves money and lives .

Testing para kayiplarini onler ve hayat kurtarir. (Ucak kazalari vs..)



MANUAL(FUNCTIONAL) TESTING NEDİR?

- Manuel test, uygulamayı herhangi bir otomasyon aracı olmadan manuel olarak test etmektir.
- Manuel test kullanıcıları dokümantasyon için sınırlı teknoloji (Excel vb.) kullanır, ancak otomasyon araçları veya dili kullanmazlar.
- Manuel testte insan hatası olabilir.
- Tüm Otomasyon Tester'lar, herhangi bir otomasyon yapmadan önce uygulamayı anlamak için mutlaka manuel test yapmalıdır.



- İyi bir otomasyon tester aynı zamanda iyi bir manuel testerdir.



TEST OTOMATION NEDİR ?



- Bir sistemi bir otomasyon aracı (tool) yardımıyla test etmeye 'Test Otomasyonu' denir.
- Otomasyon test yazılımı test verilerini Test Edilen Sistem'e girebilir, beklenen ve gerçek sonuçları karşılaştırabilir ve ayrıntılı test raporları oluşturabilir.

- Bir test otomasyon tool'u kullanarak, çalıştırılan test paketini kaydetmek ve gerektiğinde yeniden çalıştırmak mümkündür. Test paketi otomatik hale getirildikten sonra hiçbir insan

müdahalesi gerekmesizin programlandigi zamanda calisabilir.

- Giderek daha popüler hale gelmektedir.



MANUAL TESTING vs AUTOMATION TESTING

Asagida yazili metin sizce neyi ifade etmektedir ?

A- Test Case

B- Manuel tester icin test adimlari

C- Otomasyon ile test yapan kodlar



MANUAL TESTING vs AUTOMATION TESTING





EN COK KULLANILAN TEST TOOL'LARI





Basarılı **B**ir **O**tomasyon **T**esti **i**cin **Y**apılması **G**erekenler

- Doğru otomasyon araçlarını(tools) seçin. Genellikle şirketlerin kullandıkları tool'ları vardır, bazı şirketler de sizin isteginize göre tool secebilirler.
- Uygulamanız hakkında iyi bilgi sahibi olun.
- Test senaryolarınızı (test cases) kısa ve bağımsız tutun.
- Test otomasyonlarınızı önem derecesine göre sıralayın.
- Otomasyondan önce test verilerini hazırlayın(id,url,environment)
- Gerekirse, test caselerinizi yönetin(manage) ve bakımını yapın(maintain).Yeni test caselerinizin eski otomasyon test scriptlerini bozmadığından emin olun.



- Testlerinizde hata olabileceğini göz önünde bulundurun. Bir BUG'ı rapor etmeden önce mutlaka testinizi gözden geçirin
- Her zaman test ekibinizle, özellikle team liderleri ile iyi iletişim kurun.



SELENIUM NEDİR ?

Selenium'un dört bileşeni vardır;

- Selenium Integrated Development Environment (IDE)
(Selenyum Entegre Geliştirme Ortamı (IDE))
- Selenium Remote Control (RC)(Selenyum Uzaktan Kumanda (RC))



- WebDriver (Biz Selenium WebDriver kullanacağız)

- Selenium Grid (paralel test için kullanılıyor)



SELENIUM NASIL CALISIR?



- 1) Driver selenium'dan komutları alır.
- 2) Bu komutları tarayıcının API'ında dönüştürür
- 3) Browser'lardan donen sonuclari alır ve



SELENIUM'UN AVANTAJLARI NELERDİR?

1) Ücretsiz ve açık kaynaklıdır.

(Open source)

2) Bir çok programlama dilini destekler

(Java, Python, PHP, C#, Ruby vs.)

3) Çoklu işletim sistemleriyle çalışır.

Multiple operating systems

(Windows, MacOS, Linux)

4) Birden çok tarayıcı ile çalışır.

Multiple browsers

(IE, Safari, Chrome, Firefox vs.)



SELENIUM'UN **D**EZAVANTAJLARI **N**ELERDIR?

- Programlama bilgisi gerektirir
(Biz Java biliyoruz)
- Yalnızca web tabanlı uygulamaları test eder
- Profesyonel desteğe sahip değil
(Ama geniş bir kullanıcı kitlesi var)





SELENIUM'UN YAPAMADIKLARI

- performans testi
- handle captcha
(diğer tüm otomasyon araçları gibi)



TEST AUTOMATION FRAMEWORK NEDİR?

- Test Otomasyon Framework (TAF), yazılım projelerindeki Test Case'lerin oluşturulma ve gözden geçirme süresini azaltmak ve QA ekibimizin isini kolaylaştırmak için tasarlanmış yapılardır.



(Isinizi rahat yapabilmeniz icin hazirlanmis calisma ortaminizdir)

- Kullanilan Tool'lar

Java,
Eclipse/IntelliJ,
Selenium,
JUnit/
TestNG,
Cucumber

- Test Yapilan Layer'lar

UI , API , DataBase



SELENIUM ve CHROME DRIVER KURULUMU ¹⁾

<https://www.selenium.dev/downloads/> adresine gidelim

2) Selenium Client & WebDriver Language Bindings altinda Java driver'ini

download edin 3) Browsers altinda Chrome documentation linkini tiklayalim

Chrome'un kendi sayfasina gidip Current stable release'i tiklayip size uygun olani download edin * buradaki surum ile bilgisayarinizdaki Chrome surumunun ayni oldugundan emin olun**

4) src altinda resources director'si olusturun

5) Bu klasor altinda drivers ve libraries klasorleri olusturun

6) Indirdigimiz chromedriver'i drivers klasorune, selenium-java dosyasini ise libraries klasorune cikartalim

7) IntelliJ 'de yeni project / package / class olusturalim ve class icinde main method

olusturalim 8) File/Project Structure/Modules/Dependencies kismindan jar dosyalarini yukleyelim



WebDriver Olusturulmasi

1. Yeni bir class oluşturun: class name ⇒

C01_Get

2. main method olusturun

3. Java'dan System.setProperty("", "") method'unu kullanarak webdriver'in turunu ve path'ini belirleyelim.

```
System.setProperty("webdriver.chrome.driver" ,"src/driver/chromedriver"); /MAC
```

```
System.setProperty("webdriver.chrome.driver","src/driver/chromedriver.exe");
```

\\WINDOWS

4. Chrome driver oluşturun

```
WebDriver driver = new ChromeDriver();
```

5. Olusturdugumuz driver objesi ile WebDriver class'indan static method'lari kullanin

Batch 44

SELENIUM 02

17.01.2022

driver method'lari



driver method'ları **d**river.get(); **M**ethod'ları
driver.navigate(); **M**ethod'ları
driver.manage().window(); **M**ethod'ları



Önceki Dersten Aklımızda Kalanlar 1. Selenium : Web sitelerini otomasyon ile test edebilmemize imkan veren bir suite of tool'dur.

2. Artıları

- ücretsiz , açık kaynaktır
- birçok kodlama dili ile çalışır (biz java'yi selenium için öğrendik)
- geniş bir kullanıcı kitlesi var, yaşadığımız sorunlara hızlı cevap bulabileceğimiz sadık bir topluluk var - şu an piyasada en çok kullanılan otomasyon tool'udur

3- Eksi yanlari

- sadece web uygulamalari icin kullanilabilir, mobil uygulamalar icin Appium kullanilarak test yapılabilir. - captcha yi asamaz

4- WebDriver : bir interface'dir, dolayisiyla obje uretemeyiz. Browser'larin driver olusturabilmesi icin bir kalip gibi calisir. Biz hangi browser'ı kullanmak istersek driver olustururken o browser'a ait constructor'ı kullaniriz.

5- driver, insan olarak yaptigimiz veya gordugumuz tum islemleri otomasyon yapmamiza yardimci olur.



driver.get(); Method'lari **driver.get(String Url);** String olarak girilen Url'e gider

driver.getTitle(); Icinde olunan sayfanin basligini String olarak getirir

driver.getCurrentUrl(); Icinde olunan sayfanin Url'ini
String olarak getirir

driver.getPageSource(); Icinde olunan sayfanin kaynak
kodlarini String olarak getirir

driver.getWindowHandle();
driver.getWindowHandles(); Icinde olunan sayfa ve/veya
tab'larin handle degerlerini
getirir



driver.get(); Method'lari Class Work

1. Yeni bir package olusturalim : day01
2. Yeni bir class olusturalim : C03_GetMethods
3. Amazon sayfasina gidelim.
<https://www.amazon.com/>
4. Sayfa basligini(title) yazdirin
5. Sayfa basliginin “Amazon” icerdigini test edin
6. Sayfa adresini(url) yazdirin
7. Sayfa url'inin “amazon” icerdigini test edin.
8. Sayfa handle degerini yazdirin
9. Sayfa HTML kodlarinda “alisveris” kelimesi gectigini test edin
10. Sayfayi kapatın.





driver.navigate(); Method'lari **driver.navigate().to(String Url);** String olarak girilen Url'e gider

driver.navigate().back (); Icinde olunan sayfadan, geldigi onceki sayfaya dondurur

driver.navigate().forward (); Back ile donulen bir sayfadan tekrar ileri gider

driver.navigate().refresh (); Icinde olunan sayfayi yeniler.



Selenium Navigation Methods class Work

1. Yeni bir Class olusturalim.C05_NavigationMethods 2.

Youtube ana sayfasina gidelim .

<https://www.youtube.com/> 3. Amazon soyfasina gidelim.

<https://www.amazon.com/> 4. Tekrar YouTube'sayfasina
donelim

5. Yeniden Amazon sayfasina gidelim

6. Sayfayi Refresh(yenile) yapalim



driver.manage().window(); Method'lari **driver.manage().window().maximize();** Browser'i maximize yapar

driver.manage().window().minimize(); Browser'i minimize yapar

driver.manage().window().fullscreen(); Browser'i full screen yapar

driver.manage().window().getPosition(); Browser'in koordinatlarini verir

`driver.manage().window().getSize();` Browser'in olculerini verir



Selenium Navigation Methods Class Work

1. Yeni bir Class olusturalim.C06_ManageWindow

2. Amazon soyfasina gidelim.

<https://www.amazon.com/> 3. Sayfanin konumunu ve boyutlarini yazdirin

4. Sayfayi simge durumuna getirin

5. simge durumunda 3 saniye bekleyip sayfayi maximize yapin

6. Sayfanin konumunu ve boyutlarini maximize durumunda

yazdirin 7. Sayfayi fullscreen yapin

8. Sayfanin konumunu ve boyutlarini fullscreen durumunda

yazdirin 9. Sayfayi kapatın



driver.manage(). Method'lari

driver.manage().window().setPosition(new Point (x:80 , y:0)) ;

Browser'i istenen koordinata tasir

driver.manage().window().setSize(new Dimension(800 , 600));

Browser'i istenen olculere getirir.

driver.manage().timeouts(
).implicitlyWait(Duration.ofSeconds(10)); Istenen sayfa
acilincaya kadar maximum bekleme suresini belirtir

driver.close(); sadece calisilan browser'i kapatir

driver.quit(); acik olan tum browser'lari kapatir



Selenium Navigation Methods **C**lass **W**ork

1. Yeni bir Class olusturalim.C07_ManageWindowSet

2. Amazon soyfasina gidelim.

<https://www.amazon.com/> 3. Sayfanin konumunu ve boyutlarini yazdirin

4. Sayfanin konumunu ve boyutunu istediginiz sekilde ayarlayin 5. Sayfanin sizin istediginiz konum ve boyuta geldigini test edin 8. Sayfayi kapatın



Homework

- 1.Yeni bir class olusturalim (Homework)
- 2.ChromeDriver kullanarak, facebook sayfasina gidin ve sayfa basliginin (title) “facebook” oldugunu dogrulayin (verify), degilse dogru basligi yazdirin.
- 3.Sayfa URL'inin “facebook” kelimesi icerdigini dogrulayin, icermiyorsa “actual” URL'i yazdirin.
- 4.<https://www.walmart.com/> sayfasina gidin.
5. Sayfa basliginin “Walmart.com” icerdigini dogrulayin.
6. Tekrar “facebook” sayfasina donun
7. Sayfayi yenileyin
8. Sayfayi tam sayfa (maximize) yapin
- 9.Browser'i kapatın

Batch 44

SELENIUM 03

18.01.2022

WEBELEMENTS LOCATORS (YER BULUCU-KONUM BELIRLEYICI)



Tekrar Testi

1. Yeni bir class olusturun (TekrarTesti)
2. Youtube web sayfasına gidin ve sayfa başlığının “youtube” olup olmadığını doğrulayın (verify), eğer değilse doğru başlığı(Actual Title) konsolda yazdırın.
3. Sayfa URL'sinin “youtube” içerip içermediğini (contains) doğrulayın, içermiyorsa doğru URL'yi yazdırın.
4. Daha sonra Amazon sayfasına gidin <https://www.amazon.com/>
5. Youtube sayfasına geri donun

6. Sayfayı yenileyin

7. Amazon sayfasına dönün

8. Sayfayı tamsayfa yapın

9. Ardından sayfa başlığının "Amazon" içerip içermediğini (contains) doğrulayın, Yoksa doğru başlığı(Actual Title) yazdırın.

10. Sayfa URL'sinin <https://www.amazon.com/> olup olmadığını doğrulayın, değilse doğru URL'yi yazdırın

11. Sayfayı kapatın



WEBELEMENTS

• Web sayfasında kullanılan etkileşimli olan veya olmayan

herşeye webelement denir

- Button,
- Search box(arama kutusu),
- Text box(metin kutusu),

- Headers(başlıklar),
- Tables(tablolar) vb...
- Farklı türde WebElement tag'ları(etiketleri) vardır.

<html>,<body>,<form>,<label>,<input>,<a> vb.

- Otomasyon için unique(tek) web öğelerini(element) tanımlamak üzere HTML kodunu inceleyeceğiz(inspect).
- Web elementleri birlikte kullanıcı arayüzünde (UI) bir web sayfası oluştururlar.



WEBELEMENTS





Web Sayfalarını İnceleme (**Inspect**)





LOCATORS (YER BULUCU-KONUM BELİRLEYİCİ)

- ❖ Selenium LOCATORS, web sayfasındaki web öğelerini tanımlamak için kullanılır.
- ❖ Selenium'da; metin kutuları, onay kutuları, linkler, radyo butonları, liste kutuları ve **diğer tüm web öğeler** üzerinde eylemler gerçekleştirmek için **LOCATORS**'a ihtiyacımız vardır.
- ❖ Konum belirleyiciler bize nesneleri tanımlamada yardımcı olur.



- ❖ Web Elementlerine ulasmak icin tag veya bazi attribute'ler kullanilir, bunlarla ulasilamayan webelementleri icin ozel olarak tanimlanan **Xpath** ve css locator'lari kullanilir.



LOCATORS (YER BULUCU-KONUM BELIRLEYICI)



Bir

web elementini tanımlamak için 8 tane selenium locator vardır.

1. id

2. name

3. className

4. tagName

5. linkText

6. partialLinkText

7. xpath => xpath yazmanın birden fazla yolu vardır

8. cssSelector => css yazmanın birden fazla yolu vardır



LOCATORS BULMAK ICIN KULLANILAN METHOD'LAR



1) Tanımladığımız web elementin yerini driver'in bulabilmesi için findElement (Locator)

method'unu kullaniriz.

`driver.findElement(Locator);`

2) `findElement (Locator)` method'unun icine parameter olarak yazacagimiz `Locator`'in 8 `locator`'dan hangisi oldugunu belirtmek icin de `By.LocatorTuru("LocateBilgisi");` kullanilir



3) Locate ettigimiz Web Elementini kullanabilmek icin bir variable'a atama yapariz



LOCATORS 1) By.id();



`WebElement aramaKutusu = driver.findElement(By.id ("twotabsearchtextbox"));`

- Web ögesini tanımlamanın en popüler yolu id kullanmaktır.
- id en güvenli ve en hızlı locator seçeneği olarak kabul edilir ve her zaman birden çok locator arasında ilk öncelik olmalıdır.
- Eğer yanlış id locate edilirse; **NoSuchElementException** hatası oluşur.

*** **NoSuchElementException** gordugumuzde hata veren satirdaki locator gozden gecirilmedir





LOCATORS 2) By.name();



WebElement passwordTextBox

=driver.findElement(By.name("field-keywords")); ❖ Name ve value unique

ise bu metodu da kullanabilirsiniz.





LOCATORS 3) By.className();



```
WebElement passwordTextBox =driver.findElement(By.className("form-control"));
```

- ❖ Class attribute'u olduğunda kullanılabilir.
- ❖ Class ve value unique ise, bu metodu da kullanabilirsiniz, ancak genelde class attribute aynı işlevi yapan bir grup Web Elementi için kullanılır
- ❖ Class attribute'nun değeri boşluk içeriyorsa genelde By.classname() ile yapılan locator'lar sağlıklı çalışmaz



LOCATORS 4) By.linkText();

```
WebElement passwordTextBox = driver.findElement(By.linkText("Addresses"));
```

- ❖ Bu yalnızca HTML bağlantılarını(link) tanımlamak için kullanılabilir.
- ❖ HTML link elementleri, bir web sayfasında bağlantı etiketi(tag) kısaltması olan <a> etiketi(tag) kullanılarak temsil edilir.
- ❖ Kullanıcı arayüzündeki(UI) hyperlinkleri kolayca tanıyabilir ve sonra bu

yöntemi kullanabilirsiniz

❖ Büyük / küçük harfe duyarlıdır (case sensitive) ve bağlantı(link) metniyle eşleşmelidir



LOCATORS 5) By.partialLinkText();

```
WebElement passwordTextBox = driver.findElement(By.partialLinkText("dresses"));
```

❖ linkText () yöntemine benzer.

❖ Tek fark, tam metin vermek zorunda kalmamanızdır.

- ❖ Metnin yalnızca belirli bir bölümünü verebilirsiniz.
- ❖ Metnin tamamını verdiğinizde de kabul eder.



LOCATORS 6) By.tagName();



```
WebElement passwordTextBox =driver.findElement(By.tagName("input"));
```

- ❖ Bu, diğer konum belirleyicilerden biraz farklıdır.

- ❖ `<div>`, `<a>`, `<input>`, ... gibi belirli bir etiketi ilettiğinizde, birden fazla aynı ad etiketine sahip olabileceğiniz için birden çok öğeyi döndürür.

Çoğunlukla öğelerin bir listesini almak için kullanılır. Bu nedenle `findElements()` yöntemiyle kullanılması önerilir.

Örneğin, kullandığımız bir sayfadaki tüm linkleri döndürmek için **`By.tagName("a")`** kullanılabilir



`findElement()` Method WebElement

`elementName=driver.findElement(By.LocatorStrategy("LocatorValue"));`

- ❖ Driver'in bir elementi bulması için `findElement()` yöntemini kullanırız.

❖ Bu, tek bir web elementini döndürür. Aynı locator ile ulaşılabilen birden fazla web element varsa ilkinin döndürür

❖ Driver elementini bulamazsa, runtime exception verir :

NoSuchElementException. ❖ **NoSuchElementException**'i gördüğünüzde,

locatorı tekrar kontrol etmelisiniz.



findElements() Method List<WebElement>

elementName=driver.**findElements**(By.**LocatorStrategy**("LocatorValue"));

- ❖ Locator degerine uygun Web elementlerinin listesini döndürür
- ❖ Locator stratejisiyle eşleşen web elementi yoksa boş bir liste döndürür.
- ❖ **NoSuchElementException** hatası vermez.
- ❖ Listedeki her Web elementi, 0'dan başlayan bir indeks alır.



findElements() ile findElement() arasındaki farklar

	findElement()	findElements()

websayfasinda birden fazla Web Element Locator ile uyusursa	Ilk elemani dondurur	Tum elemanlari ondurur
websayfasinda hicbir Web Element Locator ile uyusmazsa	NoSuchElementException firlatir	Exception firlatmaz, bos bir liste dondurur
Return Type	WebElement	List<WebElement>
Elemana erisim	Direk ulasilabilir	Liste'den index veya iterator ile ulasilabilir





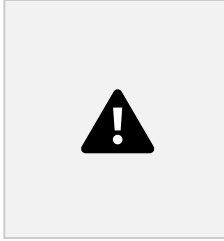
WebElement Method'ları Bir WebElement üzerinde eylemler gerçekleştirmek otomasyon tester'ları için çok önemlidir.

webElement.click(); Web Element'e click yapar

webElement.sendKeys("Metin"); Parametre olarak yazılan metni Web Elemente gönderir

webElement.submit(); Web element ile işlem yaparken Enter tusuna basma görevi yapar

webElement.sendKeys("Metin" + Keys.ENTER); İstedigimiz metni yollayıp, sonra ENTER'a basar.



WebElement Method'leri Class **Work: Login Test**

1. Bir class oluşturun: LoginTest
2. Main method oluşturun ve aşağıdaki görevi tamamlayın.
 - a. <http://a.testaddressbook.com> adresine gidiniz.
 - b. Sign in butonuna basın
 - c. email textbox,password textbox, and signin button elementlerini locate ediniz..
 - d. Kullanıcı adını ve şifreyi aşağıya girin ve oturum aç (sign in)buttonunu tıklayın: i. Username : testtechproed@gmail.com
ii. Password : Test1234!
 - e. Expected user id nin testtechproed@gmail.com oldugunu dogrulayin(verify).

f. “Addresses” ve “Sign Out” textlerinin görüntülendiğini(displayed) doğrulayın(verify).

3. Sayfada kaç tane link olduğunu bulun.



WebElement Get Method'ları `webElement.getText();` Web Element

üzerindeki yazıyı getirir

`webElement.getAttribute(“Att.ismi”);` Ismi girilen attribute'un
değerini getirir

`webElement.tagName();` Web elementin tag ismini
getirir

Not : Web element ile ilgili bu method'ların dışında size, istenen CSS özelliği, Location, Rect.geometrik özellikler, DOM değeri vb.. Bir çok method vardır ama Automation Test için kullanmıyoruz



WebElement Is Method'ları

webElement.isEnabled(); Web Element erişilebilir ise true
yoksa false döner

webElement.isDisplayed(); Web Element görünür ise true
yoksa false döner

webElement.isSelected(); Web Element secili ise true
yoksa false doner

Not : Web element ile ilgili bu method'larin disinda size, istenen CSS ozelligi,
Location, Rect.geometrik ozellikler, DOM degeri vb.. Bir cok method vardır ama
Automation Test için kullanmıyoruz

Batch 44

SELENIUM

04

19.01.2022

XPath

CssSelector

Relative Locators



WebElement Method'leri Tekrar Testi

1. Bir class oluşturun : AmazonSearchTest
2. Main method oluşturun ve aşağıdaki görevi tamamlayın.
 - a.google web sayfasına gidin. <https://www.amazon.com/>
 - b. Search(ara) “city bike”
 - c. Amazon'da görüntülenen ilgili sonuçların sayısını yazdırın
 - d. “Shopping” e tıklayın.

e. Sonra karşınıza çıkan ilk sonucun resmine tıklayın.



HTML Parent-Child-Sibling Terimleri Nedir?



Tags: < > şeklinde

görülen komutlara etiket (tag) adı verilir.

parent-child-sibling relationship(Ebeveyn-çocuk-kardeş ilişkisi) hakkında konuştuğumuzda, yalnızca tag adları önemlidir

<table>, <div> in çocuğudur(child) ve <tbody>, <tr> in ebeveyndir (parent)

<td> taglari ise siblings (kardes)'dir



LOCATORS XPath 7) By.xpath();

Bir WebElement'i locate etmek için kullanabileceğimiz en etkin

method'dur. WebElement passwordTextBox

```
=driver.findElement(By.xpath("xpath"));
```

Method'un yazımı açısından diğer 6 yöntem ile aynı olmakla beraber xpath'i diğerlerinden ayıran çok önemli bir fark vardır

Diğer 6 method HTML kod'una bağlıdır. Web element'in kodunda id yoksa By.id() method'unu, web element link değilse By.linkText() method'unu kullanamazsınız.

Xpath ise dinamiktir. Her türlü web element için mutlaka bir xpath yazılabilir

path

2 çeşit Xpath yazılabilir

1.Absolute xpath (mutlak)

2.Relative xpath (bağıl)



LOCATORS XPath 1.Absolute XPath()

// div/ table/ tbody



```
// tbody / tr / td[3] // span
```

```
// tbody / tr / td[3] / a / span
```

Absolute xpath yazmak için en basa // sonraki her adımda / yazarak hedef web element'e kadar tüm tag'lar yazılır.

Eğer aynı path'e sahip birden fazla element varsa index kullanılabilir. [2] gibi

Eğer bir parent'in grand child'lari içinde unique bir tag varsa parent // grand child yazılabilir





LOCATORS XPath 2. Relative XPath()



Bir web element'te temel olarak 3 bileşen vardır.

1) tag name input

2) attribute type, id , value, name, autocomplete, placeholder, class 3

) attribute value type → 'text', id→ 'twotabsearchtextbox' , value → ''

Bu 3 bileşeni birlikte kullanarak her bir web element için unique sonuç veren bir çok xpath yazılabilir

Relative xpath yazmak için bu 3 bileşen aşağıdaki gibi bir araya getirilir, unique sonuç veren her relative xpath kullanılabilir.

//tagName[@attributeismi='attributeValue']



LOCATORS XPath 2. Relative XPath()



❖ Genelde 3 bileşen de kullanılır, Ancak bazen daha az bileşen yazmak da yeterli olabilir.

Sadece tag name kullanarak xpath yazmak için

```
driver.findElement(By.xpath("//input" ));
```

Tag name farketmeksizin attribute ismi ve attribute value kullanarak xpath

```
yazmak için driver.findElement(By.xpath("// * [@type='text']" ));
```

Attribute name farketmeksizin tag name ve attribute value kullanarak xpath

yazmak için `driver.findElement(By.xpath("//input[@*='text']"))`;

Attribute value farketmeksizin tag name ve attribute ismi kullanarak xpath

yazmak için `driver.findElement(By.xpath("//input[@type]"))`;



LOCATORS XPath

Class Work: Add Remove Element

1- https://the-internet.herokuapp.com/add_remove_elements/ adresine

gidin 2- Add Element butonuna basin

3- Delete butonu'nun gorunur oldugunu test edin

4- Delete tusuna basin

5- “Add/Remove Elements” yazisinin görünür olduğunu test edin



LOCATORS XPath 2. Relative Xpath()

❖ Bazen de attribute’e bağlı olmadan sadece web element icinde bulunan text kullanılabilir.

Exact Text(Belirli bir text) ile element bulma:

```
driver.findElement(By.xpath("//tagname[.='text name']" ));
```

```
driver.findElement(By.xpath("//*[.='text name'] " ));
```

```
driver.findElement(By.xpath("//*[text()='exact text with extra space and all'] " ));
```

Belirli bir metni içeren bir öğeyi bulmak için şunları kullanabiliriz:

```
driver.findElement(By.xpath("//*[contains(text(),'piece of text')] " ));
```

❖ Tek attribute ile unique bir sonuca ulaşamazsak birden fazla attribute yazabiliriz

```
driver.findElement(By.xpath("//div[@id='logo' or class='flex-col  
logo' ] ' '));
```

```
driver.findElement(By.xpath("//div[@id='logo' and class='flex-col logo' ] ' '));
```



LOCATORS XPath

Class Work: Add Remove Element

Asagidaki testi text'leri kullanarak locate edin

1- https://the-internet.herokuapp.com/add_remove_elements/ adresine

gidin 2- Add Element butonuna basin

3- Delete butonu'nun gorunur oldugunu test edin

4- Delete tusuna basin

5- "Add/Remove Elements" yazisinin gorunur oldugunu test edin



LOCATORS **CssSelector** 8- **By.cssSelector() Method**



Css selector xpath'e benzer. Üç ana tip kullanılır

1) **css = tagName[attribute name= 'value'];**


```
driver.findElement(By.cssSelector("input[name='session[password]']"));
```

2) `css="tagName#idValue"` veya sadece `css="#idValue"` =>yalnızca id value ile çalışır `driver.findElement(By.cssSelector("input#session_password"));`

3) `css="tagName.classValue"` veya sadece `css=".classValue"` =>yalnızca class value ile çalışır `driver.findElement(By.cssSelector(".form-control"));`



LOCATORS **CssSelector** Home Work: Log in Test Using **Css**

1) Bir class oluşturun : Locators_css

2) Main method oluşturun ve aşağıdaki görevi tamamlayın.

- a. Verilen web sayfasına gidin.
http://a.testaddressbook.com/sign_in
- b. Locate email textbox
- c. Locate password textbox ve
- d. Locate signin button
- e. Asagidaki kullanıcı adını ve şifreyi girin ve sign in düğmesini tıklayın
 - i. Username : testtechproed@gmail.com
 - ii. Password : Test1234!

NOT: cssSelector kullanarak elementleri locate ediniz.

Batch 44

SELENIUM 05

20.01.2022

Relative Locators

Maven Projesi Olusturma



LOCATORS XPathTekrar **T**esti

1-C01_TekrarTesti isimli bir class olusturun

2- <https://www.amazon.com/> adresine

gidin 3- Browseri tam sayfa yapin

4-Sayfayi “refresh” yapin

5- Sayfa basliginin “Spend less” ifadesi icerdigini test

edin 6- Gift Cards sekmesine basin

7- Birthday butonuna basin

8- Best Seller bolumunden ilk urunu
tiklayin 9- Gift card details'den 25 \$'i
secin

10-Urun ucretinin 25\$ oldugunu test edin

10-Sayfayi kapatın



LOCATORS **Relative Locators** **Relative Locators** nedir ?

- ❖ Selenium 4 ile gelen yeniliklerden biri de bagli locator'lardir.
- ❖ Bir web elementi direk locate edemedigimiz durumlarda gunluk hayatimizda kullandigimiz sekilde o web elementi etrafindaki web



elementlerin referansi ile tarif edebiliriz.

- ❖ Örneğin yandaki resimde Berlin için bir çok relative locator tanımlayabiliriz.
 - Boston'ın sağında , Sailor'ın üstünde
 - NYC'nin altında, Bay Area'nin solunda - Boston yakınlarında Bay Area'nin solunda ve Toronto'nun sağında vb..

<https://www.diemol.com/selenium-4-demo/relative-locators-demo.html>



LOCATORS **Relative Locators** **Class Work: Relative Locators**





LOCATORS **Relative** Locators

Class Work: Relative Locators

- 1) <https://www.diemol.com/selenium-4-demo/relative-locators-demo.html> adresine gidin
- 2) Berlin'i 3 farkli relative locator ile locate edin
- 3) Relative locator'larin dogru calistigini test edin





Bir projeyi geliştirmek kadar onu anlaşılır kılmak, tekrar eden kodlardan kaçınmak, update'leri tek merkezden kolayca yapabilmek ve projenin sürdürülebilirliğini sağlamak (maintenance) da önemlidir.

- ❖ **Apache Maven bir yazılım proje yönetimi ve anlama aracıdır.**
- ❖ **Maven, en iyi uygulamaların geliştirilmesi için güncel ilkeleri bir araya getirmeyi ve bir projeye bu yönde rehberlik etmeyi kolaylaştırmayı amaçlamaktadır.**
- ❖ **Maven geliştiricileri birçok ayrıntıdan**
- ❖ **Proje nesne modeli (POM Project Object Model) konseptine dayalıdır.**



korur.

<https://maven.apache.org/>



❖ **Maven bir Java derleme aracıdır (build tool). Maven proje otomasyon ve yönetim aracıdır (automation and management tool).**

❖ **Maven,konfigrasyon için pom.xml dosyasını kullanır.**

**Bu dosya projenin insasi , raporlaması ve dokümantasyonu için gerekli bütün bilgileri içerir
(dependencies , plugins)**



- ❖ Bir Maven projesine aşina olduğunuzda, tüm Maven projelerinin nasıl inşa edildiğini bilirsiniz. Bu, birçok projede gezinirken zaman kazandırır.



Neden Maven?

- ❖ Tekrarlanabilir derlemeler / yürütmeleri yapmak kolaylaştırır.(Repeatable builds/executions.)
- ❖ Proje yönetimini kolaylaştırır.
- ❖ Maven, dependencylerle ilgili mevcut jar dosyalarını otomatik olarak indirir.

❖ Birden fazla IDE (intelliJ, eclipse, vb.) ve araçlarla(tools) çalışır.

❖ Open source

❖ Geniş kullanıcı tabanı



Maven PROJESINI OLUSTURMA



1. Create Project: File -> New -> Project



2. Select Maven -> click next
Java versiyonunun bilgisayarinizdaki version ile ayni oldugunu control edin



3. Name: com.Batch44Maven -> finish -> click on (EnableAutoImport)

4. Package olusturun, name : day04 -> Right click on java create the package
5. Class olusturun, name : FirstMavenClass



POM XML

✓ Pom.xml dosyası nedir?

Maven projesindeki en önemli dosyadır. Dependencies'i POM dosyasına ekler ve POM dosyasından yönetiriz.

✓ Maven ve pom arasındaki fark nedir?

Maven bir araç (tool), pom bir xml dosyasıdır. Pom.xml maven'in bir parçasıdır ve pom dependencies'i yönetir.

✓ Projenizde pom'ı nasıl kullanırız? Neden kullanırız?

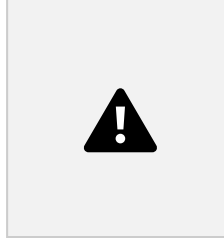
Dependencies'i yönetmek için . Tüm projeyi yönetmeme yardımcı oldu. Pom ayrıca artifact id, group id, ve version gibi proje bilgilerine de sahiptir.

✓ Dependency nedir? Nasıl kullanıyorsunuz? Neden kullanıyorsunuz? Sürücüler (drivers) kurmak, sürücüler oluşturmak için dependencies gereklidir. Gerekli araçları içe aktarır(import).

✓ Pom dosyanızı ve dependencies'i nasıl güncellersiniz ?

Mvnrepository.com adresinden gerekli dosyaları bulur ve pom dosyamıza ekleriz





POM XML DOSYASINA DEPENDENCIES EKLEME

- 1) Mvnrepository.com adresine gidelim
- 2) WebDriverManager aratalim (En guncel ve en cok kullanılan optimum versiyonu alalım)
- 3) Asagida Maven yazan bolumdeki kodlari kopyalayalım
- 4) pom XML dosyasina gidelim
- 5) <properties> satirinin altina <dependencies> </dependencies> tag'i olusturalım
- 6) Kopyaladigimiz kodlari <dependencies> bolumune yapistiralım
- 7) Ayni islemleri Selenium Java dependency icin de yapalım
- 8) Pom XML dosyasinin sag tarafinda **Maven** yazan bolumu tiklayalım

9) Yenile butonuna tiklayip asagida Dependencies bolumunun olustugunu ve altinda ekledigimiz kutuphanelerin gorundugunden emin olun



POM XML DOSYASINA DEPENDENCIES EKLEME





CLASS AYARLARI



File
Project Structure
Modules
Sources
Language level : min **8** yapalim,
bilgisayarınızda kurulu java 8 veya ustu ise
java versiyonu ayni olmalı



File
Settings
Build,Execution,Deployment
Compiler
Java Compiler
Target bytecode version : min **8** yapalim,
bilgisayarınızda kurulu java 8 veya ustu ise
java versiyonu ayni olmalı



CLASS WebDriver AYARLARI

```
WebDriverManager.chromedriver( ).setup( );  
WebDriver driver = new ChromeDriver( );
```

Class Work Amazon Search Test

- 1- <https://www.amazon.com/> sayfasina gidelim
- 2- arama kutusunu locate edelim
- 3- “Samsung headphones” ile arama yapalim
- 4- Bulunan sonuc sayisini yazdiralim

5- İlk ürünü tıklayalım

6- Sayfadaki tüm başlıkları yazdıralım



Soru 1

- 1. <http://zero.webappsecurity.com> sayfasına gidin**
- 2. Signin buttonuna tıklayın**
- 3. Login alanine “username” yazdırin**
- 4. Password alanine “password” yazdırin**
- 5. Sign in buttonuna tıklayın**
- 6. Pay Bills sayfasına gidin**

7. amount kısmına yatırmak istediğiniz herhangi bir miktarı yazın
8. tarih kısmına “2020-09-10” yazdırın
9. Pay buttonuna tıklayın
10. “The payment was successfully submitted.” mesajının çıktığını kontrol edin

Batch 44

SELENIUM

06

21.01.2022

Junit

Annotations