



Selenium

25 KASIM 2020
DERS 3

LOCATORS

TECHPROED

Mehmet BULUTLUOZ
Elektronik Muh.

Önceki Dersten Aklımızda Kalanlar

1- Basic Navigation komutlarını gördük

`get()`, `navigate().to()`, `navigate().back()`, `navigate().forward()`, `navigate().refresh()`
`manage().window().maximize()`, `Thread.sleep(milisaniye)` (bunu eklediğimizde exception uyarısı koymamız gerekiyor), `close()`, `quit()`

2- Title ve URL testleri yaptık

`getTitle()`, `getCurrentUrl()` bunlarla title ve URL'i alıp if bloğu ile doğruluyorduk (verify)

3- Web Element : Web sayfasında bulunan etkileşimli veya etkileşimsiz her şeye Web Element diyoruz. Bir QA olarak her zaman yapacağımız iş web elementlerini locate edip, testlerimizi gerçekleştirmek.

4- LOCATORS : konum bulucu, yer bulucu. Web elementlerini otamasyonda kullanabilmek için o web elementinin yerini kodla bulmamız lazım. Bu işleme locate etme diyoruz.

Id, name, className, linkText, partialLinkText

tagName

Xpath, css

findElement() Method

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

- Bir elementi bulmak için findElement () yöntemini kullanırız.
- Bu, tek bir web elementini döndürür. Aynı locator ile ulaşılabilen birden fazla web element varsa ilkinin döndürür.
- Driver elementi bulamazsa, runtime exception verir : **NoSuchElementException**.
- **NoSuchElementException**'i gördüğünüzde, locatorı tekrar kontrol etmelisiniz.

findElements() Method

```
List<WebElement> elementName=driver.findElement(By.LocatorStrategy("LocatorValue"));
```

- Web elementlerinin bir listesini döndürür
- Locator stratejisiyle eşleşen web elementi yoksa boş bir liste döndürür.
- **NoSuchElementException** hatası vermez.
- Her Web elementi, tıpkı array'deki gibi 0'dan başlayan bir sayı indeksi alır.

*** **findElements()** ile **findElement()** arasındaki farklar nelerdir ?

WebElement Üzerindeki İşlemler

- Bir WebElement üzerinde eylemler gerçekleştirmek otomasyon tester'ları için çok önemlidir.
- **sendKeys** ("yazdığımız metin"); =>metin kutusuna yazmak için kullanılır.
- **clear()**; => Metin kutusunun içindekileri temizler.
- **submit()**; => **click()**; bir form veya form içindeki bir öğeyi gönderir. Genellikle form veya arama kutusundaki metni göndermek için **sendKeys()** yönteminden sonra kullanılır.
- **isDisplayed()** => Bir web elementinin görünür olup olmadığını kontrol eder
- **getText()** => Web elementine ait yazıyı getirir
- **Keys.ENTER()** => enter'a basar

Class Work: Login Test

1. Bir class oluşturun: LocatorsIntro
 2. Main method oluşturun ve aşağıdaki görevi tamamlayın.
 - a. <http://a.testaddressbook.com> adresine gidiniz.
 - b. Sign in butonuna basin
 - 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) doğrulayin(verify).
 3. Sayfada kac tane link oldugunu bulun.
-

Class Work: Search Test

1. Bir class oluşturun : **GoogleSearchTest**
2. Main method oluşturun ve aşağıdaki görevi tamamlayın.
 - a.google web sayfasına gidin. <https://www.google.com/>
 - b. Search(ara) “city bike”
 - c. Google'da görüntülenen ilgili sonuçların sayısını yazdırın
 - d. “Shopping” e tıklayın.
 - e. Sonra karşınıza çıkan ilk sonucun resmine tıklayın.

TECHPROED
