
TestNG

8 ARALIK 2020
DERS 13

TestBase Class
Handle Windows

Mehmet BULUTLUOZ
Elektronik Muh.

TECHPROED

TestNG

Önceki Dersten Aklımızda Kalanlar

- 1- Alerts : basit alert: sadece tamam diyebiliyoruz, confirm (onaylama) alerti: Kabul edebilir veya reddedebiliriz, prompt: kullanıcıdan bilgi almak için kullanılır, dolayısıyla biz de promptta metin yollayabiliriz ve tamam tusuna basabiliriz veya cancel tusuna basabiliriz.
 - 2- kullanacağımız komutlar accept(), dismiss(), sendKeys(), getText()
 - 3- alert ile işlem yapmak istediğimizde her seferinde mutlaka switch to kullanmalıyız
 - 4- bir web sitesi içinde çalışan ve başka bir web sitesinden alinti yapan webelementtir. Genellikle tagları <iframe> olur.
 - 5- iFrame içinde işlem yapmak için ilk önce index, id, web element yöntemlerinden birini kullanarak iframe'e geçiş yapılmalı
 - 6- bir kere iframe'e geçiş yapınca çıkma kodu yazıncaya kadar driver'imiz iframe'in içinde kalır
 - 7- iframe'den çıkmak için switchTo().parentFrame veya defaultFrame yazmalıyız.
-

TestNG

TestBase Class

- TestBase, testlerden önce ve sonra tekrar tekrar kullandığımız kodları içermektedir.
- İçerisindeki metodları kullanabilmemiz için extend yapıyoruz. Bu sayede test class'ımızda sadece test case'ler bulunmaktadır.
- Utilities package'da TestBase'i oluşturuyoruz.
 - setUp method
 - reports (Raporlar)
 - tearDown method
- TestBase class'i abstract yapabiliriz (Olmasa da olabilir) , extends yaparak kullanabiliriz. Ve bu class'da object create edemeyiz.

```
import java.util.concurrent.TimeUnit;

public abstract class TestBase {
    protected WebDriver driver;
    @BeforeMethod
    public void setUp() {
        WebDriverManager.chromedriver().setup();
        driver = new ChromeDriver();
        driver.manage().window().maximize();
        driver.manage().timeouts().pageLoadTimeout(10, TimeUnit.SECONDS);
        driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);
    }

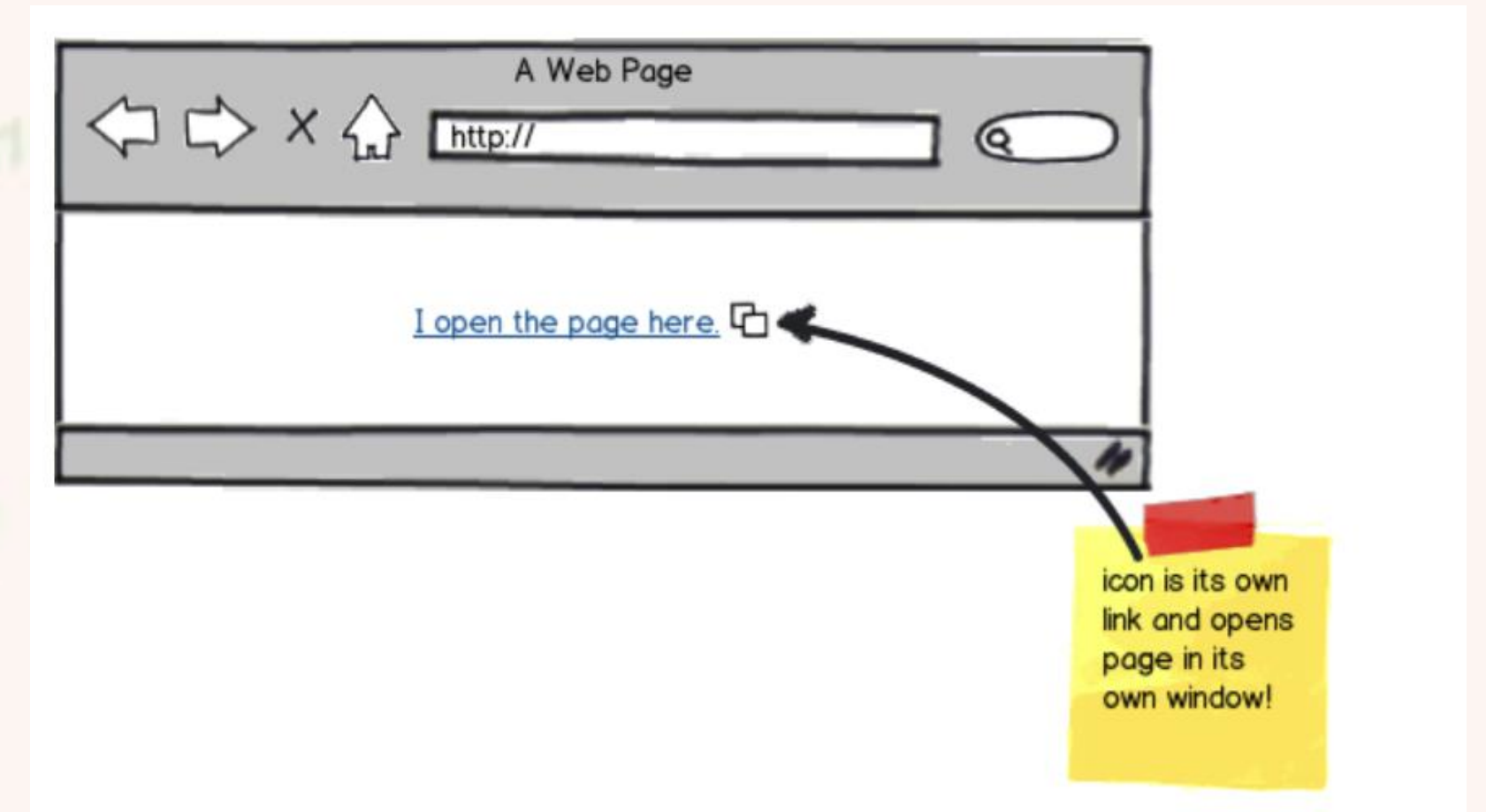
    @AfterMethod
    public void tearDown(){
        driver.quit();
    }
}
```


TestNG

Tabs(Sekmeler):

Birden çok pencere arasında geiş yapma(Switch).

- Bazen bir butona tıkladığımızda, başka bir sekmede(tab) yeni bir pencere açılır.
- Birden fazla pencereyle uğraşırken driver'ı pencereler arasında değiştirmemiz gerekir.
- Selenium WebDriver, WebDriver objesi başlatıldığında her pencereye bir alfanumerik kimlik atar.Bu benzersiz(unique) alfanumerik kimliğe pencere tanıtıcısı(window handle) denir. Selenium, birkaç pencere arasında kontrolü değiştirmek için bu benzersiz kimliği kullanır.



TestNG

Tabs(**S**ekmeler):

Birden çok pencere arasında geçiş yapma(**S**witch).

- **switchTo()** => pencereler arasında geçiş yapma(switch) metodu
- **getWindowHandle()** => WebDriver'in o anda kontrol ettiği sayfanın window handle'ini almak için kullanılır. String return eder.
- **getWindowHandles()** => Açık olan tüm pencerelerin window handle'lerini almak için kullanılır. Set return eder.
- Syntax: **driver.switchTo().window(driver.getWindowHandle());**

<https://the-internet.herokuapp.com/iframe>



Handle Windows Class Work

- Tests package'ında yeni bir class oluşturun: D13_WindowHandle2
- <https://the-internet.herokuapp.com/windows> adresine gidin.
- Sayfadaki textin “Opening a new window” olduğunu doğrulayın.
- Sayfa başlığının(title) “The Internet” olduğunu doğrulayın.
- Click Here butonuna basın.
- Acılan yeni pencerenin sayfa başlığının (title) “New Window” olduğunu doğrulayın.
- Sayfadaki textin “New Window” olduğunu doğrulayın.
- Bir önceki pencereye geri döndükten sonra sayfa başlığının “The Internet” olduğunu doğrulayın.