

## Handle Alerts

### Alert Nedir?

Alert kullanıcıya bir tür bilgi vermek veya belirli bir işlemi gerçekleştirme izni istemek için ekran bildirimi görüntüleyen küçük bir mesaj kutusudur. Uyarı amacıyla da kullanılabilir.

### HTML Alerts

Bir alert çıktığında sağ click ile inspect yapabiliyorsak html alert'dir ve extra bir işleme gerek yoktur.

### Js Alerts

Js alerts inspect yapılamaz, ekstra işleme ihtiyaç vardır.

#### 1.Simple Alert :

Bu basit alert ekranda bazı bilgiler veya uyarılar görüntüler. Ok denilerek kapatılır.

#### 2. Confirmation Alert :

Bu onay uyarısı bir tür işlem yapma izni ister. Alert onaylanıyorsa OK, onaylanmıyorsa Cancel butonuna basılır.

#### 3.Prompt Alert :

Bu Prompt Uyarısı kullanıcıdan bazı girdilerin girilmesini ister ve selenium webdriver metni sendkeys ("input....") kullanarak girebilir.

### Handle Allert Methods

- `accept()` => Bir uyarıda(alert) OK'ı tıklamakla aynı.

```
driver.switchTo().alert().accept();
```

- `dismiss()` => Bir uyarıda(alert) Cancel'ı tıklamakla aynı.

```
driver.switchTo().alert().dismiss();
```

- `getText()` => Uyarıdaki(alert) mesajı almak için.

```
driver.switchTo().alert().getText();
```

- `sendKeys("Text")` => Uyarı(alert) text kutusuna text göndermek için

```
driver.switchTo().alert().sendKeys("Text");
```

### Dip Not:

**Alerts** : Web sitelerinde iki tür alert vardır.

**1. HTML alerts** : sağ click yapıp incelenebilir, dolayısıyla diğer web elementler gibidir

**2. JS alerts**: bunlar sağ click yapıp incelenemez, dolayısıyla da diğer WebElementler'den farklıdır.

Bunlarla çalışabilmek için "`driver.switchTo().alert()`" method'u ile alert'e geçip, ondan sonra `accept`, `dismiss`, `sendKeys` veya `getText` method'larından uygun olanı kullanılır.

Art arda bile olsa alert ile işlem yapmak istediğimiz her seferde `switchTo` kullanmalıyız.

## Selenium'da Allert nedir?

Bir Selenyum içinde Uyarısı kullanıcıya bazı bilgiler veya bildirimi vermek ekranda görüntülenir küçük bir mesaj kutusudur. Kullanıcıya bazı özel bilgiler veya hatalar bildirir, belirli görevleri gerçekleştirmek için izin ister ve ayrıca uyarı mesajları verir.

### Selenium türlerinde birkaç uyarı:

#### 1) Basit Uyarı

Selenium'daki basit uyarı sınıfı, ekranda bazı bilgiler veya uyarılar görüntüler.

#### 2) İstemi Uyarısı.

Bu İstem Uyarısı, kullanıcıdan bir miktar girdi ister ve Selenium web sürücüsü, gönderme anahtarlarını kullanarak metni girebilir ("giriş.... ").

#### 3) Onay Uyarısı.

Bu onay uyarısı, bir tür işlem yapmak için izin ister.

Selenium WebDriver'da Uyarı nasıl işlenir

Uyarı arabirimi, Selenium Webdriver'da yaygın olarak kullanılan aşağıdaki birkaç yöntemi sağlar.

1) void dismiss () // Uyarının 'İptal' düğmesine tıklamak için.

driver.switchTo (). alert (). dismiss ();

2) void accept () // Uyarının 'Tamam' düğmesine tıklamak için.

driver.switchTo (). alert (). accept ();

3) String getText () // Uyarı mesajını yakalamak için.

driver.switchTo (). alert (). getText ();

4) void sendKeys (String stringToSend) // Bazı verileri uyarı kutusuna göndermek için.

driver.switchTo (). alert (). sendKeys ("Metin");

Eclipse tarafından önerilen aşağıdaki ekranda gösterildiği gibi bir dizi Uyarı yönteminin görüntülendiğini görebilirsiniz.

Selenium'un .switchTo () yöntemini kullanarak ana pencereden kolayca uyarıya geçebiliriz.

### Örnek Alerts Çalışması:

```
static WebDriver driver;
```

```
@BeforeClass
```

```
public static void setup() {
```

```
    WebDriverManager.chromedriver().setup();
```

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

```
    driver.manage().window().maximize();
```

```
    driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(15));
```

```
//● https://the-internet.herokuapp.com/javascript_alerts adresine gidin.
driver.get("https://the-internet.herokuapp.com/javascript_alerts");
}
@AfterClass
public static void tearDown() {
    driver.close();
}
```

```
@Test
public void acceptAlert(){
```

```
// ● Bir metod oluşturun: acceptAlert
// ○ 1. butona tıklayın, uyarıdaki OK butonuna tıklayın ve result mesajının
// “You successfully clicked an alert” olduğunu test edin.
```

```
driver.findElement(By.xpath("//*[text()='Click for JS Alert']")).click();
driver.switchTo().alert().accept(); //ok butonuna basabilmek için yaptık
WebElement mesaj = driver.findElement(By.xpath("//*[@id='result']"));
String actualMesaj = mesaj.getText();
String expectedMesaj = "You successfully clicked an alert";
Assert.assertEquals(expectedMesaj,actualMesaj);
```

```
//Assert.assertTrue(driver.findElement(By.xpath("//*[text()='You successfully clicked an
alert']")).isDisplayed());
//Kısa yollu çözüm de doğru
```

```
}
```

```
@Test
public void dismissAlert() throws InterruptedException {
```

```
//● Bir metod oluşturun: dismissAlert
// ○ 2. butona tıklayın, uyarıdaki Cancel butonuna tıklayın ve result mesajının
// “successfully” içermediğini test edin.
```

```
driver.findElement(By.xpath("//*[text()='Click for JS Confirm']")).click();
```

```
//Alert üzerindeki mesajı ekleyin
```

```
System.out.println("2. butonunun alert mesajı = " + driver.switchTo().alert().getText()); //2.
butonun alert mesajını yazdırdık
```

```
Thread.sleep(2000);
//önce mesajı gördük şimdi iptal yapıyoruz
driver.switchTo().alert().dismiss();
```

```

String actualMesaj = driver.findElement(By.xpath("//*[@id='result']")).getText();
String expectedMesaj = "successfully";
Assert.assertFalse(actualMesaj.contains(expectedMesaj));

/*
Bu yöntemle de yapılabilirdi
WebElement mesaj = driver.findElement(By.xpath("//*[@id='result']"));
String actualMesaj = mesaj.getText();
String expectedMesaj = "successfully";
Assert.assertEquals(expectedMesaj,actualMesaj);
*/
}

@Test
public void sendKeysAlert() throws InterruptedException {
    //● Bir metod oluşturun: sendKeysAlert
    // ○ 3. butona tıklayın, uyarıdaki metin kutusuna isminizi yazın, OK butonuna
    // tıklayın ve result mesajında isminizin görüntülendiğini doğrulayın.

    driver.findElement(By.xpath("//*[text()='Click for JS Prompt']")).click();

    System.out.println("*****");

    //Alert üzerindeki mesajı ekleyin
    System.out.println("3. butonunun alert mesajı = " + driver.switchTo().alert().getText()); //3.
    butonun alert mesajını yazdırdık

    Thread.sleep(2000);

    //uyarıdaki metin kutusuna isminizi yazın, OK butonuna
    driver.switchTo().alert().sendKeys("Zeliha");
    driver.switchTo().alert().accept();

    Thread.sleep(2000);

    //tıklayın ve result mesajında isminizin görüntülendiğini doğrulayın
    WebElement isim=driver.findElement(By.xpath("//*[text()='You entered: Zeliha']"));
    String expectedisim="Zeliha";
    String actualMessage=isim.getText();
    Assert.assertTrue(actualMessage.contains(expectedisim));
}

```

#### Kaynak:

- <https://www.browserstack.com/guide/alerts-and-popups-in-selenium>
- <https://techproeducation.com/> (Ahmet Bulutluoz)