

## Selenium'da Sınıf Seçin Kısmı İçinde Bir Değer Nasıl Seçilir? (dropdown list)

### Selenium'da Select Class'a Giriş

Selenium'da Select sınıfı, HTML SELECT etiketinin uygulanmasını sağlar. Bir Select etiketi, seçme ve seçimi kaldırma seçenekleriyle yardımcı yöntemler sağlar. Select sıradan bir sınıf olduğundan, nesnesi New anahtar sözcüğüyle oluşturulur.

#### Syntax:

- `Select objSelect = new Select();`

Yukarıdaki sözdiziminde, Select'in yapıcısı için bir eleman tipi nesne istediğini, yani select sınıfının bir nesnesini yaratacağını açıkça belirtir.

#### Farklı Select Methodları:

- Aşağıdakiler, bir sorunla başa çıkmak için en sık kullanılan drop-down yöntemleridir.

#### 1. `selectByVisibleText`: `selectByVisibleText(String arg0): void`

Bu yöntem, kutudaki seçeneklerden birini veya çoklu seçim kutuları arasından bir seçeneği seçmek için drop-down kullanılır. Select öğesinin değerlerinden biri olan String parametresini alır ve hiçbir şey döndürmez.

#### Syntax:

`obj.Select.selectByVisibleText("text");`

#### Example:

```
Select objSelect =new Select(driver.findElement(By.id("search-box")));  
objSelect.selectByVisibleText("Automation");
```

- ✓ Bir seferde birden fazla seçime izin veren bir açılır kutu içerir.

page source

```
<select id="fruits" multiple="">  
  <option value="banana">Banana</option>  
  <option value="apple">Apple</option>  
  <option value="orange">Orange</option>  
  <option value="grape">Grape</option>  
</select>
```

HTML page



- ✓ Aşağıdaki kod, `SelectByVisibleText()` yöntemini kullanarak ilk iki seçeneği seçecektir.

```
public static void main(String[] args) {  
    WebDriver driver = new FirefoxDriver();  
    driver.get("http://jsbin.com/osebed/2");  
    Select fruits = new Select(driver.findElement(By.id("fruits")));  
    fruits.selectByVisibleText("Banana");  
    fruits.selectByIndex(1);  
}
```



## 2. `selectByIndex: selectByIndex(int arg0) : void`

Bu yöntem '`selectByVisibleText`'e benzer, ancak buradaki fark, kullanıcının seçenek için metin yerine dizin numarasını sağlaması gerektiridir. `Select` öğesinin indeks değeri olan tamsayı parametresini alır ve hiçbir şey döndürmez.

### Syntax:

```
oSelect.selectByIndex(int);
```

### Example:

```
Select objSelect = new Select(driver.findElement(By.id("Search-box")));  
objSelect.selectByIndex(4);
```

## 3. `selectByValue: selectByValue(String arg0) : void`

Bu yöntem, seçenek metni veya bir dizin yerine istenen seçeneğin değerini sorar. `Select` elementinin değerlerinden biri olan `String` parametresini alır ve hiçbir şey döndürmez.

### Syntax:

```
oSelect.selectByValue("text");
```

### Example:

```
Select objSelect = new Select(driver.findElement(By.id("Search-box")));  
objSelect.selectByValue("Automation Testing");
```

#### 4. getOptions: getOptions( ) : List<WebElement>

Bu yöntem Select etiketine ait tüm seçenekleri alır. Parametre almaz ve List<WebElements> döndürür.

##### Syntax:

```
oSelect.getOptions();
```

##### Example:

```
Select objSelect = new Select(driver.findElement(By.id("Search-box")));  
List <WebElement> elementCount = oSelect.getOptions();  
System.out.println(elementCount.size());
```

#### 5. deselectAll()

Bu yöntem, seçilen tüm girişleri temizler. Bu, yalnızca açılır öğe birden çok seçimi desteklediğinde geçerlidir.

##### Syntax:

```
objSelect.deselectAll();
```

★ Selenium açılır listesinde kullanılan en yaygın yöntemler şunlardır:

Yöntem	Açıklama
<b>selectByVisibleText ()</b> ve <b>deselectByVisibleText ()</b> Örnek: <code>drpCountry.selectByVisibleText ("ANTARCTICA");</code>	<ul style="list-style-type: none"><li>Parametreyle eşleşen metni görüntüleyen seçeneği seçer / seçimi kaldırır.</li><li><b>Parametre</b> : Belirli bir seçeneğin tam olarak görüntülenen metni</li></ul>
<b>selectByValue ()</b> ve <b>deselectByValue ()</b> Örnek: <code>drpCountry.selectByValue ("234");</code>	<ul style="list-style-type: none"><li>"Değer" özniteliği belirtilen parametreyle eşleşen seçeneği seçer / seçimi kaldırır.</li><li><b>Parametre</b> : "değer" özelliğinin değeri</li><li>Tüm açılır seçeneklerinin aşağıdaki örnekte olduğu gibi aynı metne ve "değere" sahip olmadığını unutmayın.</li></ul> <div><pre>&lt;option value="10"&gt;ANGUILLA &lt;/option&gt; &lt;option value="234"&gt;ANTARCTICA &lt;/option&gt; &lt;option value="1"&gt;ANTIGUA AND BARBUDA &lt;/option&gt;</pre></div>
<b>selectByIndex ()</b> ve <b>deselectByIndex ()</b> Örnek: <code>drpCountry.selectByIndex(0);</code>	<ul style="list-style-type: none"><li>Verilen dizindeki seçeneği seçer / seçimi kaldırır.</li><li><b>Parametre</b> : seçilecek seçeneğin dizini.</li></ul>
<b>isMultiple ()</b> Örnek: <pre>if (drpCountry.isMultiple()) {     //do something here }</pre>	<ul style="list-style-type: none"><li>Açılır öğe aynı anda birden çok seçime izin veriyorsa DOĞRU döndürür; Aksi takdirde YANLIŞ.</li><li><b>Parametre gerekmez</b></li></ul>
<b>deselectAll ()</b> Örnek: <code>drpCountry.deselectAll();</code>	<ul style="list-style-type: none"><li>Tüm seçili girdileri temizler. Bu, yalnızca açılır öğe birden çok seçimi desteklediğinde geçerlidir.</li><li><b>Parametre gerekmez</b></li></ul>

- Şimdi, Select komutuyla birden çok öğenin nasıl seçileceğini anlamak için daha derine inelim.

### Select komutuyla birden çok öğe nasıl seçilir?

Çoklu seçim özelliği bir boole ifadesidir. Bu yöntem, aynı anda birden fazla seçeneğin seçilebileceğini belirtir. Bu seçenekler farklı işletim sistemleri ve tarayıcılar için değişiklik gösterir.

Windows için: Birden fazla seçenek belirlemek için kontrol (ctrl) düğmesine basın.

Mac için: Birden çok seçenek belirlemek için komut düğmesini basılı tutun.

- Birden çok komut seçmek için isMultiple yöntemini kullanın.

**isMultiple(): boolean** – Bu yöntem, Select öğesinin aynı anda birden çok seçim seçeneğini destekleyip desteklemediğini bildirir. Bu yöntem hiçbir şeyi kabul etmez ve bir boole değeri (doğru/yanlış) döndürür.

### Syntax:

```
objSelect.isMultiple();
```

### Example:

```
Select objSelect = new Select(driver.findElement(By.id(Element_ID)));
objSelect.selectByIndex(index);
objSelect.selectByIndex(index);
// Or can be used as
objSelect.selectByVisibleText(text);
objSelect.selectByVisibleText(text);
// Or can be used as
objSelect.selectByValue(value);
objSelect.selectByValue(value);
```

### Özet:

Eleman	Komut	Açıklama
Açılır Kutu	selectByVisibleText () / deselectByVisibleText ()	görüntülenen metne göre bir seçeneği seçer / seçimi kaldırır
	selectByValue () / deselectByValue ()	"value" özniteliğinin değerine göre bir seçeneği seçer / seçimi kaldırır
	selectByIndex () / deselectByIndex ()	indeksine göre bir seçeneği seçer / seçimi kaldırır
	isMultiple ()	açılır öğe bir seferde birden fazla seçime izin veriyorsa, DOĞRU döndürür; Aksi takdirde YANLIŞ
	Hiçbirini seçme()	önceden seçilen tüm seçeneklerin seçimini kaldırır

- Şimdi select komutunun gerçek zamanlı bir örneğine bakalım.

Selenium'da Select sınıfının kullanımını gösteren bu örnekte, Selenium'daki Select sınıfını kullanarak açılır listeden bir değerin nasıl seçileceğini keşfedelim.

```
import org.junit.Test;
import org.openqa.selenium.By;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.support.ui.Select;

public class SelectExample {
    @Test
    public static void main(String[] args) throws InterruptedException {
        System.setProperty("webdriver.chrome.driver", "Path_of_Chrome_Driver");
        WebDriver driver = new ChromeDriver();
        driver.get("https://www.browserstack.com/");
        driver.manage().window().maximize();
        WebElement developers_dropdown = driver.findElement(By.id("developers-menu-toggle"));
        Select objSelect = new Select(developers_dropdown);
        objSelect.selectByIndex(2);
        Thread.sleep(3000);
        driver.navigate("https://www.browserstack.com/");
        WebElement solutions_dropdown = driver.findElement(By.id("solutions-menu-dropdown"));
        Select solutions = new Select(solutions_dropdown);
        solutions.selectByValue("4000");
        Thread.sleep(3000);
        WebElement solutions_d = driver.findElement(By.id("solutions-menu-dropdown"));
        Select s1 = new Select(solutions_d);
        s1.selectByVisibleText("Geolocation Testing");
    }
}
```

### ★ Hap bilgi:

#### DropDown Menü (Açılır Menü):

- ★ dropdown bir webelement'den birden fazla option kullanabilmemizi sağlar.
- ★ dropdown ile select tag'ı birbirinin ayrılmaz parçasıdır.
- ★ Bir HTML ögesi select tag'ı kullanıyorsa çok büyük bir ihtimalle dropdown'dır.
- ★ Dropdown'da asıl amaç, webelement'i locate etmek değil, dropdown'dan istediğimiz option'u seçebilmektir.

Seçme işlemi 3 adımda yapılır:

1. - Dropdown webelementini locate ederiz.
2. - Select class'ından bir obje oluşturup, parametre olarak dropdown webElementini yazarız.
3. - Dropdown'dan option seçmek için 3 seçenekten birini kullanırız.
  - ✓ - selectByIndex
  - ✓ - selectByVisibleText
  - ✓ - selectByValue

#### Kaynak:

- <https://www.browserstack.com/guide/select-class-in-selenium>
- <https://tr.csstricks.net/8222913-how-to-select-value-from-dropdown-using-selenium-webdriver>
- <https://techproeducation.com/>