

Deselect Values From A Drop Down

Deselect Values

In this session, we are going to deselect values from a drop-down. Let's use W3 Schools multiple select box. Just like the previous session, we see a [multiple attribute](#). First, let's find the element. [Inspect](#) the drop-down and we see [View Frame Source](#). View Frame Source shows this element is located in a frame. Therefore, we must switch to the frame before selecting an option then deselecting an option. Scroll up the DOM and we see `iframe`. [iframeResult](#) is the value for `id` and `name`. You can watch [Videos 29 – 32](#) where I introduce frames then show 3 ways to switch to a frame. The `select` tag has `cars` as the value for `name`.

In our Test Script, let's use all 4 deselect methods. The methods are `deselectByIndex`, `deselectByValue`, `deselectByVisibleText`, and `deselectAll`. We see 4 options in this drop-down. The values are the [same words as visible text but have a different capitalization](#). Let's go to Eclipse.

Switch to the drop-down that's located in a frame by writing
`driver.switchTo().frame(driver.findElement(By.id("iframeResult")));`

Next, is Selenium's `Select` class
`Select select = new Select` find the `WebElement` (`driver.findElement(By.name("cars"))`);

In this Test Method, let's look at 3 deselect methods after selecting 3 options.

```
select.selectByIndex(0); // Volvo
select.selectByValue("saab");
select.selectByVisibleText("Opel");
```

We deselect by writing `select dot` and there are 4 deselect methods:

1. `deselectAll`
2. `deselectByIndex`
3. `deselectByValue`
4. `deselectByVisibleText`

Let's start with `deselectByIndex`. It [deselect the option at the given index](#).
`select.deselectByIndex(0); // Volvo`

Next is `deselectByValue`.

`select.deselectByValue("saab");` This method **Deselect all options that have a value matching the argument.**

Next is `deselectByVisibleText`. It **Deselect all options that display text matching the argument.**

`select.deselectByVisibleText("Opel");`

Let's **Run**. Did you see how 3 options were selected then deselected? The same concept applies for `deselectAll`. **Copy and Paste** all of the select statements. Let's add one more

`select.selectByVisibleText("Audi");` then write `select.deselectAll();`

The `Select` class also has a method to check if the drop-down is a multiple select option:

`select.isMultiple` It returns **whether this select element support selecting multiple options at the same time.** Let's go ahead and print to validate the drop-down supports multiple.

`sysout System.out.println("Drop Down Supports Multiple Selections? " + select.isMultiple());` Let's **Run**.

All 4 options were selected and deselected. That's it for deselecting values.