

Intro To Selenium Actions

Table of Contents

Introduction	2
Demo	2

Introduction

Hello and Welcome, My name is Rex Jones II. This video is an Introduction To The Actions Class In Selenium.

We are going to start with the difference between Action without an "s" and Actions with an "s". Both are part of Selenium's interaction package.

Action is an interface representing 1 user-interaction. It only has 1 method and that method is perform. The Actions class has a lot of methods like build, dragAndDrop, dragAndDropBy, keyDown, keyUp, moveToElement, and it also has perform. This perform() method is different from the one in the Action Interface. The description shows perform is a convenience method for performing the actions without calling build() first. The purpose of build() is to collect a sequence of actions and store all of those actions into 1 action. That's why the description states "Generates a composite action containing all actions."

Let's walkthrough an example to help us see the difference between Selenium's Action Interface and Actions Class.

Demo

We are going to enter all capital letters TEST into the Search box. Inspect the search box and we see q as the value for name. Now, let's go to Eclipse. In this method, we are going to enter the Actions Class and Action Interface. Find the element by writing `WebElement searchBox = driver.findElement(By.name("q"));`

Instantiate the Actions class and act as the object = `new Actions`. We need the driver in parenthesis to instantiate the class. Next, we write the sequence of actions to make all capital letters. Capital Letters are created when we press down the Caps Lock key or press down the Shift Key. Let's press down the Shift Key by writing `act.keyDown`. The WebElement is searchBox then `Keys.SHIFT`).

The next action is to enter text in the search box `act.sendKeys`. We can select anyone of the sendKeys methods. Let's select the one with WebElement. searchBox and the sequence is "with build". It does not matter if we send the keys with lowercase, mixed case, or uppercase letters. We keyed down the Shift Key, now we key up the Shift key. `act.keyUp(searchBox, Keys.SHIFT);`

Now, generate a composite of all 3 actions by building the actions. `act.build()` and assign it to Action interface. This is how we include the Action Interface. The last step is perform all of these actions by writing `action.perform`. Let's Run. We see WITH BUILD in all caps.

Recall perform in the Actions class. It said perform actions without calling build first. Now, let's execute these same actions without calling the build method. Copy, Paste, add s to Action, and change with to without. We don't need the build() method because perform() in the Actions class will automatically build the actions and perform the actions. Therefore, we write `act.perform()`.

Now, we have 2 perform() methods but the one from the Action Interface is no longer necessary. Let's remove the build and perform methods. There's 1 more concept. All of these actions can be combined into 1 statement line.

```
act.keyDown(searchBox, Keys.SHIFT).sendKeys(" Works Too").keyUp(searchBox, Keys.SHIFT).perform();
```

Let's run. WITH BUILD / WITHOUT BUILD WORKS TOO

Thanks for watching Introduction to Selenium's Actions class – What Is The Difference Between Selenium Action Interface and Action Class. Next, we are going to drag and drop target.