Working with Drag, Drop & Mouse Hover

In this lesson, we will discuss the drag, drop, and hovering functionality using Selenium WebDriver.

WE'LL COVER THE FOLLOWING ∧

- Drag & drop
- Mouse hover

The below example code can be downloaded and run locally; it is included as a part of the Sample UI test framework, please refer.

Selenium provides `org.openqa.selenium.interactions.Actions` class that contains methods to build complex user gestures that involve Keyboard and Mouse interactions.

Drag & drop

Drag & drop involves a series of user interactions such as *move to source element → click and holding on the element → move to the target location → release click.*

We have overloaded methods to move by offset from source `WebElement`.

```java
// Create a WebDriver object
WebDriver driver = DriverManager.getWebDriver();

// open the web url
driver.get("http://codetoautomate.com/educative-selenium-demo/");

// scroll to the end of the page
JavaScriptExecutor jsDriver = (JavaScriptExecutor) driver;
jsDriver.executeScript("document.getElementById('div1').scrollIntoViewIfNe
eded()");

WebElement source = driver.findElement(By.id("drag1"));
```

```java
WebElement target = driver.findElement(By.id("div1"));


// using Robot, move the mouse pointer to the target element
// This is required, as the drop happens onto the location where the move
// pointer is currently placed
Robot robot = new Robot();

// get co-ordinates to the center of the target element
int x = target.getLocation().getX() + target.getRect().getWidth() / 2;
int y = target.getLocation().getY() / 2 - target.getRect().getHeight();

// move the mouse pointer to the co-ordinates
robot.mouseMove(x, y);

// perform drag-drop
Actions actions = new Actions(driver);
actions.dragAndDrop(source, target).build().perform();

// an overloaded method to move by offset from source location
// actions.dragAndDropBy(source, 100, 100).build().perform();
```

Mouse hover

Mouse hover is nothing but moving focus to the source element. This can be
achieved by:

```java
// create the WebDriver object
        WebDriver driver = DriverManager.getWebDriver();


        // open the web URL
        driver.get("http://codetoautomate.com/educative-selenium-demo/");


        JavascriptExecutor jsDriver = (JavascriptExecutor) driver;


        // Scroll down to the bottom of the page
        jsDriver.executeScript("document.getElementById('mouseover').scrol
lIntoViewIfNeeded()");


        WebElement source = driver.findElement(By.id("mouseover"));
        Actions action = new Actions(driver);
        action.moveToElement(source).build().perform();


        Assert.assertTrue("Hover Over Text".equalsIgnoreCase(source.getAtt
ribute("title")));
```

That's all for handling the drag, drop and mouse hovering. In the next lesson, you'll learn how to handle a double-click and right-click functionality.