Rex Jones II

Selenium 4 getRect()
Combines x, y, Width & Height

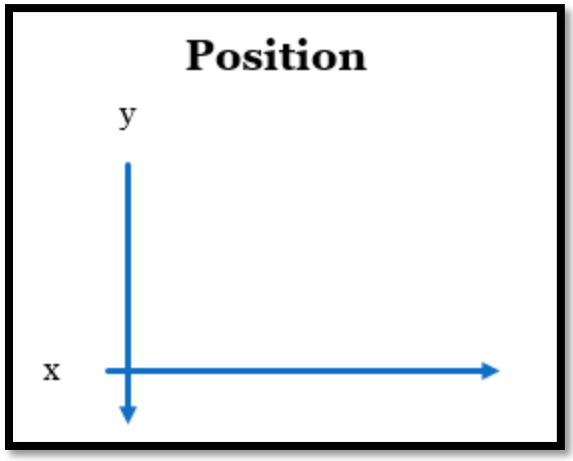
Selenium 4 Playlist
https://www.youtube.com/watch?v=_r79FgJlTc0&list=PLfp-cJ6BH8u_4AMzeLVizVfqn4SCywSTJ&index=10

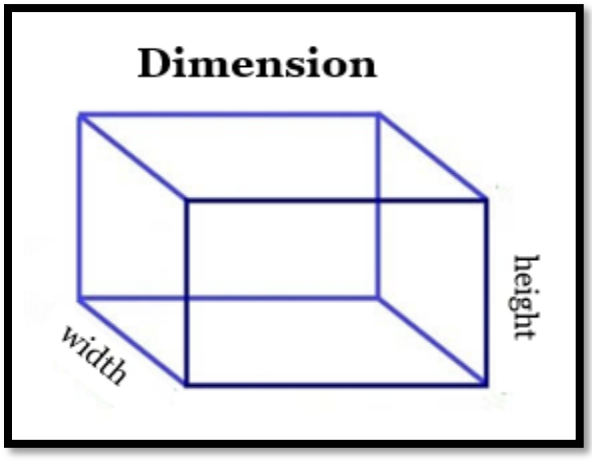
getRect

Introduction

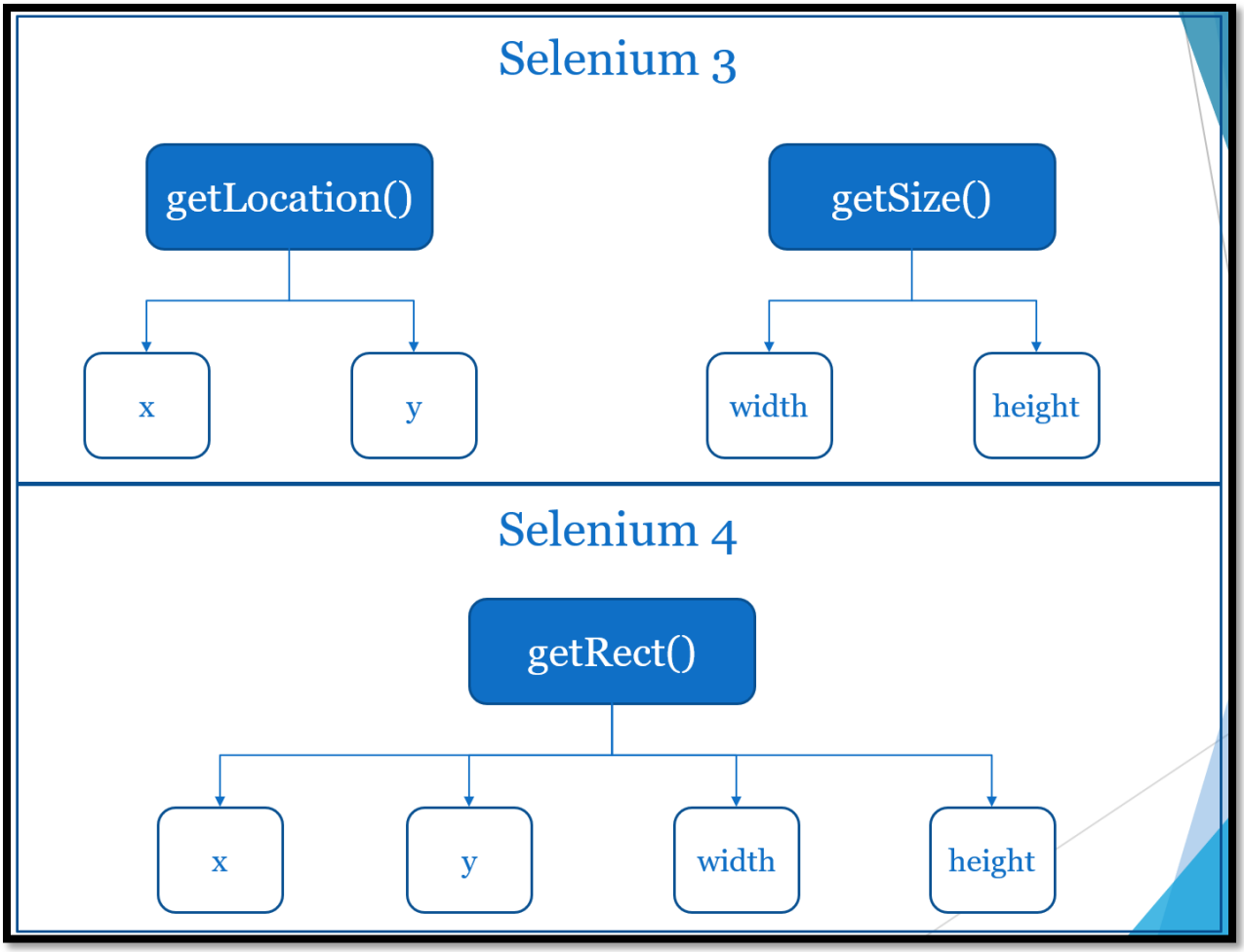
In our automation program, there are scenarios that requires us to get the position and/or dimension of a WebElement. The positions are x and y coordinates while the dimensions are width and height. Selenium 4 has the getRect method that allows to get both positions and dimensions.

x comes before y and width comes before height. The x coordinate is measured horizontally starting from the left to the right. The y coordinate is measured vertically starting from the top to the bottom. Width and height are ways to measure the size of an element. Notice, there are 2 differences between Selenium and how we normally measure in Math. On a web page, the length is not available as a Dimension and y is not measured starting from the bottom.



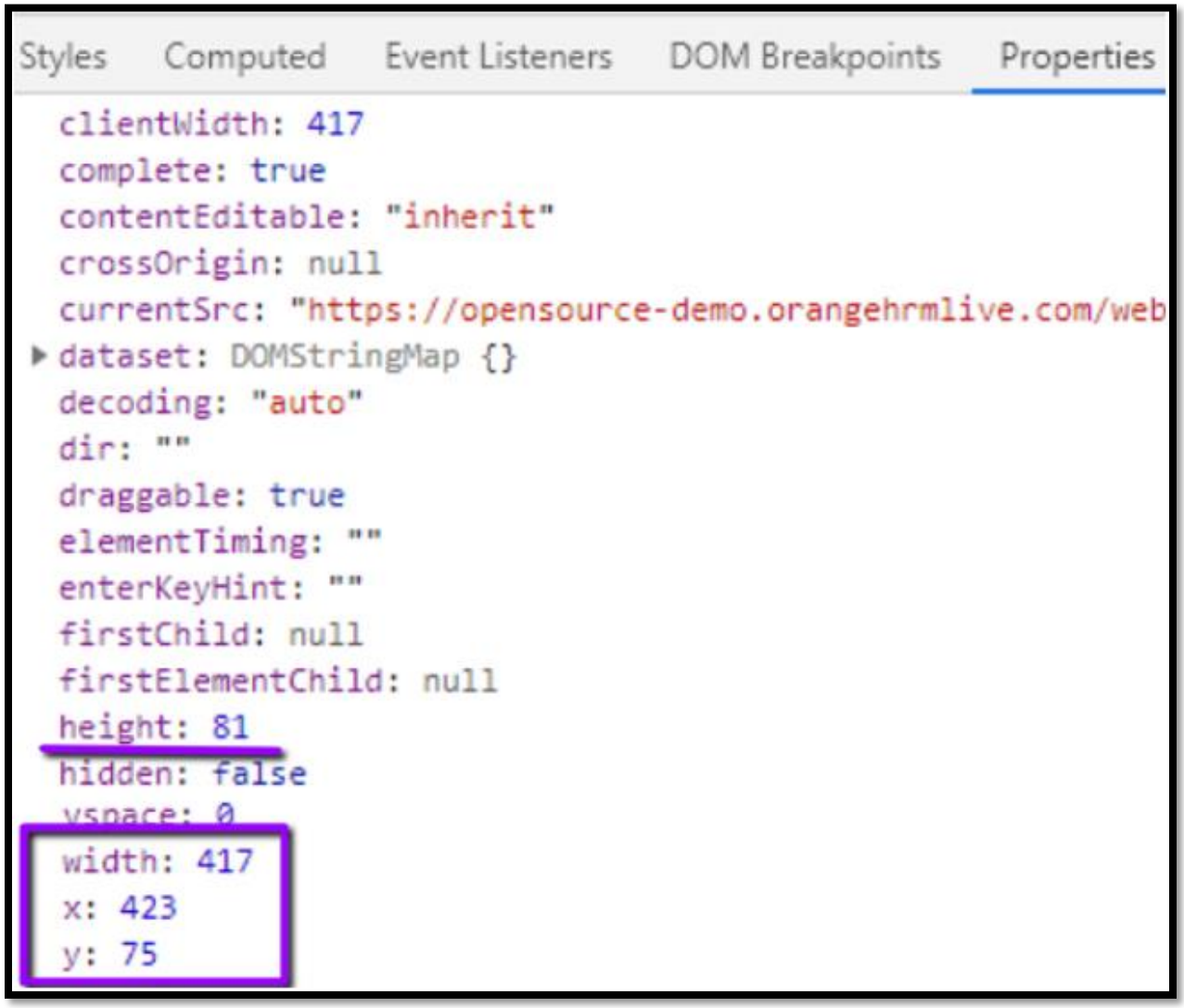


With Selenium 3, we have getLocation and getSize. The getLocation method returns x and it returns y while the getSize method is used to fetch the width and height. With Selenium 4, the new getRect method is a combination that gets the x coordinate, y coordinate, width, and height. In this video, I am going to show you all 3 methods: getRect, getLocation, and getSize then compare getRect to see if it returns the same value as getLocation and getSize.



If you are interested in programming and automation videos, subscribe to my YouTube channel then click the bell icon. Also, connect with me on LinkedIn, Facebook, and follow me on Twitter. The code, presentation, and transcript document will get placed on GitHub.

We are going to use this Orange HRM logo as our WebElement. When I inspect the logo, we see img as the tag name and src as the attribute but no id attribute. However, the parent has an id value of divLogo. There are different ways to get this value. This time, I'm going to use CSS. CTRL + F to bring up this Find by string, selector, or XPath then write hash tag # then divLogo > img. Bingo. Copy this value. Also, let's look at the size and location. Go to Properties, select img and we see height as 81, width is 417, x is 423, and y is 75.



getRect

Let's go to our Test Script and start with the getRect method to make sure the measurements are the same. @Test / public void getPositionDimension () {}

The WebElement is logoOrangeHRM = driver.findElement(By.cssSelector("#divLogo > img")); Paste the value. Now that we have the logoOrangeHRM. WebElement, we can access the getRect method. Do you see Rectangle? That means the getRect method returns a Rectangle. The description shows it returns the location and size of the rendered element. Since it returns a Rectangle, we assign the location and size to Rectangle with an object name like rectLogo =. That's all we need to get the x and y coordinates plus the width and height. Import the Rectangle class from Selenium package. Also import the @Test annotation from TestNG.

Let's print these values and start with x. sysout("x: " + rectLogo.get) and we see getHeight, getWidth, getX, and getY. Select getX then perform this same process for each method. sysout("y: " + rectLogo.getY()); sysout("Width: " + rectLogo.getWidth()); sysout("Height: " + rectLogo.getHeight());

```
@Test
Run | Debug
public void getPositionDimension () {
  WebElement logoOrangeHRM = driver.findElement(By.cssSelector("#divLogo > img"));
  Rectangle rectLogo = logoOrangeHRM.getRect();
  System.out.println("x: " + rectLogo.getX());
  System.out.println("y: " + rectLogo.getY());
  System.out.println("Width: " + rectLogo.getWidth());
  System.out.println("Height: " + rectLogo.getHeight());
}
```

Let's Run. It's the same values. x = 423, y = 75, Width = 417, Height = 81

```
INFO: Detected dialect: W3C
x: 423
y: 75
Width: 417
Height: 81
PASSED: getPositionDimension
```

Let me hover over this Rectangle class then open the Declaration, we see 2 constructors. The 1st constructor has 4 int parameters: x, y, height, and width. However, the 2nd constructor has a Point parameter and a Dimension. With Selenium 3, we had to use the Point and Dimension classes to help us get the location and size.

```java
public class Rectangle {

  public int x;
  public int y;
  public int height;
  public int width;

  public Rectangle(int x, int y, int height, int width) {
    this.x = x;
    this.y = y;
    this.height = height;
    this.width = width;
  }

  public Rectangle(Point p, Dimension d) {
    x = p.x;
    y = p.y;
    height = d.height;
    width = d.width;
  }
```

Both classes are still available in Selenium 3. Let's compare getRect with getLocation and getSize.

getLocation & getSize

Go back to our Test Script and Erase these print statements then start from scratch. For location, we write logoOrangeHRM.getLocation. and we see Point as the Return Type. After selecting getLocation, we assign it to Point with any name like pointLogo =. Import the class from Selenium. Look at this, we write pointLogo., we see x, y, getX, and getY but notice, we do not have width and height. Width and height are not available because this is the Point class.

For width and height, we write the WebElement logoOrangeHRM.getSize. This time, we see Dimension as the return type. So, we select getSize then assign it to Dimension with dimLogo = as the object. Import Dimension from the Selenium class. Now, dimLogo. have height, width, getHeight, and getWidth.

Let's print the values to verify we get the same values. Let's add some headers. sysout("\t Selenium 3 \t Selenium 4"); Now, let' add some hyphens. sysout("----------") All of the methods came from Selenium 4 but I put Selenium 3 in the print statement. Selenium 3 will represent the old way of getting the location and getting the size. Let's add some print statements to get x, y, width, and height. Start with x.
sysout("x \t\t " + pointLogo.getX() + "\t\t" + rectLogo.getX());
sysout("y \t\t " + pointLogo.getY() + "\t\t" + rectLogo.getY());
sysout("Width \t\t " + dimLogo.getWidth() + "\t\t" + rectLogo.getWidth);
sysout("Height \t\t " + dimLogo.getHeight() + "\t\t" + rectLogo.getHeight);

```
@Test
Run | Debug
public void getPositionDimension () {
  WebElement logoOrangeHRM = driver.findElement(By.cssSelector("#divLogo > img"));
  Rectangle rectLogo = logoOrangeHRM.getRect();
  Point pointLogo = logoOrangeHRM.getLocation();
  Dimension dimLogo = logoOrangeHRM.getSize();
  System.out.println("\t    Selenium 3 \t   Selenium 4");
  System.out.println("-----------------------------------------------");
  System.out.println("x \t\t" + pointLogo.getX() + "\t\t" + rectLogo.getX());
  System.out.println("y \t\t" + pointLogo.getY() + "\t\t" + rectLogo.getY());
  System.out.println("Width \t\t" + dimLogo.getWidth() + "\t\t" + rectLogo.getWidth());
  System.out.println("Height \t\t" + dimLogo.getHeight() + "\t\t" + rectLogo.getHeight());
}
```

Let's Run. We see the same exact value for x, y, Width, and Height.

```
INFO: Detected dialect: W3C
            Selenium 3     Selenium 4
-----------------------------------------------
x               423             423
y               75              75
Width           417             417
Height          81              81
PASSED: getPositionDimension
```

The benefit of Selenium 4's Rectangle class is that it has all of the methods. That's it and Thanks for watching and I'll see you in the next session.

Rex Jones II

Contact

- ✔ YouTube https://www.youtube.com/c/RexJonesII/videos
- ✔ Facebook http://facebook.com/JonesRexII
- ✔ Twitter https://twitter.com/RexJonesII
- ✔ GitHub https://github.com/RexJonesII/Free-Videos
- ✔ LinkedIn https://www.linkedin.com/in/rexjones34/