# Directions

1. Complete the following steps using JUnit and the Selenium WebDriver.
2. Screenshot where directed.
3. Submit to Blackboard.

# Test 1

Use a JUnit test structure to create a simple test on <http://cottrell.lee.stuweb.ptcollege.edu/carpetPostPersistent.php>.

Connect the ChromeDriver to the website

1. Create WebElements for all the form objects.
2. Use the WebElements to enter the following data on the site.
   1. Room 602, Width 45 feet, length = 60 feet, carpet cost = 4.50 per sq feet
3. Submit using either the submit() on a form element, or .click() the button.
4. Check the answer against this expected answer 13,000.50. – You will have to use xpath for the div the number is contained within. AssertTrue combined with a .contains test will work.
   1. Screenshot your JUnit pass/fail test
5. Change your expected value to 19450.
   1. Screenshot your JUnit pass/fail output.
6. Submit your code and the pass/fail screenshots

# Test 2

Use JUnit and Selenium to test the contents of the PTCollege website, <https://www.ptcollege.edu>. I strongly suggest separate @Test functions for each line.

Connect the ChromeDriver to the website.

1. Find and click the link to Student Logins. Verify that the title that opens is ‘Current Student Logins’.

There is a problem with this. The page is built using a parallax view. This hides the link until you scroll down. To force the link to be visible, you need to first find the link, and then use JavaScript to scroll down to the object.

WebElement link = driver.findElement(By.linkText("Student Logins"));

((JavascriptExecutor) driver).executeScript("arguments[0].scrollIntoView(true);", link);

<https://stackoverflow.com/questions/3401343/scroll-element-into-view-with-selenium>

1. Find and click the 5th link inside the site-inner using XPath. Verify that the title is Campus Alert System. This test will likely fail.
2. Dr. Scarpino claims that there are 300 links (<a> tags) on the website. I think there are only 237. Write a test to determine who is correct.
   1. Hint, the List class includes a .size() method that returns the number of elements in the list. You can use this in an assertEquals().

Submit the code and the pass/fail screenshots