Stock Management Application

==========

Step 1
Create the test product details

==========

```
package lesson3TDD;
import org.testng.Assert;
import org.testng.annotations.Test;

public class TestStringCalculator {


	// write a test case to

	// that should send a String to the java code
	// java code will calculate the length of the String and give to the user.

	// test if length of String is equal to the length user has given

	@Test(priority='1')

	public void passString()
	{
		// I am assuming that I have a class StringCalculator,

		StringCalculator s1 = new StringCalculator();

		// I am assuming that the above class has method to compute length

		int actuallength = s1.stringlength("testDriven");

		int expectedlenght=10;

		// using testNg assertion I am comparing the length of the string

		Assert.assertEquals(actuallength, expectedlenght);


	}

	// The calculator should be able to add 2 strings

	@Test(priority='2')

	public void TestaddString()
	{

		// I am assuming that I have a class StringCalculator,
```

```
		StringCalculator str = new StringCalculator();

				// I am assuming that the above class has method to concatinate
2 strings

		String actualString= str.addstring("selenium","tool");

		String expectedString = "SELENIUMTOOL";

		Assert.assertEquals(actualString,expectedString);
	}

}
```

==========

Step 3 create the product search page

```
package lesson3TDD;
import java.util.HashMap;
import java.util.Map;
public class ProductSearch {



	public Map<String, String> getproductdetails()
	{

	Map<String, String> productMap = new HashMap<>();
	productMap.put("SeleniumBook", "books");
	productMap.put("coach", "handbag");
	productMap.put("camlin", "pencil");
	productMap.put("mac", "laptop");
	productMap.put("samsung", "mobile");

	return productMap;

	}

	public String getproductType(String product) {

		Map<String, String> productMap = null;

		String producttype = null;

		if(product.isEmpty()){

			throw new NullPointerException("Exception: Product name should not
be empty");

		}

		productMap = getproductdetails();
```

```
                if(!productMap.containsKey(product))
                {
                        throw new NullPointerException("Exception: Product name is invalid.
Details donot exist");
                }
                else {
                 producttype = productMap.get(product);
                }

                return producttype ;
        }

}
```

Output Screenshot

==========