## Identificamos las pruebas

@Test

**public** **void** MuestraLosDetallesDelProductoAlIngresarUnaOrden() {

}

@Test

**public** **void** ConfirmaLaCompraDeUnProducto() {

}

## Planificamos la prueba

@Test

**public** **void** MuestraLosDetallesDelProductoAlIngresarUnaOrden() {

//ARRANGE

//Load Data (Insert product with PRODUCTNAME y PRODUCTPRICE)

//Cargar el driver

//ACT

//Abrir el listado de productos

//Realizar click en un producto .... (nos muestra una nueva pantalla)

//ASSERT

//(en la nueva pantalla)

//Verificar que se muestra un producto con nombre PRODUCTNAME

//Verificar que se muestra un producto con nombre PRODUCTPRICE

//TEARDOWN

//Cerrar el driver

//Cleanup Data

}

## Implementamos la Prueba

**Creamos los Drivers**

**static** WebDriver *driver*;

@BeforeClass

**public** **static** **void** InitDriver(){

*driver*=**new** FirefoxDriver();

}

@AfterClass

**public** **static** **void** CloseDriver(){

*driver*.close();

}

**Completamos el ACT Y ASSERT**

@Test

**public** **void** MuestraLosDetallesDelProductoAlIngresarUnaOrden() {

//ARRANGE

//Load Data (Insert product with PRODUCTNAME "Orange Muffin" y PRODUCTPRICE "6.99")

//ACT

*driver*.get("http://localhost:8080");

*driver*.findElement(By.*xpath*("//h3[contains(text(),'Orange Muffin')]/../.././/a"));

//ASSERT

String productName = *driver*.findElement(By.*tagName*("h1")).getText();

*assertTrue*(productName.contains("Orange Muffin"));

String price = *driver*.findElement(By.*id*("orderPrice")).getText();

*assertTrue*(price.contains("6.99"));

//TEARDOWN

//Cleanup Data

}

**Cargamos y Limpiamos los datos**

@Before

**public** **void** LoadData(){

InsertData(**new** Product("Orange Muffin",**null**,6.99,**null**));

}

Object loadedData;

**public** **void** InsertData(Object object){

Session session = sessionFactory.openSession();

Transaction transaction=session.beginTransaction();

session.save(object);

loadedData=object;

transaction.commit();

session.close();

}

@After

**public** **void** CleanData(){

DeleteData();

}

**public** **void** DeleteData(){

Session session = sessionFactory.openSession();

Transaction transaction=session.beginTransaction();

session.delete(loadedData);

transaction.commit();

session.close();

}

## Page Object

**public** **class** ElegirProductoPage {

**private** WebDriver driver;

**public** ElegirProductoPage(WebDriver driver) {

**this**.driver = driver;

}

**public** **void** Abrir() {

driver.get("http://localhost:6709/");

}

**public** IngresarOrdenPage comprar(String name) {

driver.findElement(By.*xpath*("//h3[contains(text(),'" + name + "')]/../.././/a")).click();

**return** PageFactory.*initElements*(driver, IngresarOrdenPage.**class**);

}

}

**public** **class** IngresarOrdenPage {

**private** WebDriver driver;

@FindBy(how = How.*TAG\_NAME*, using = "h1")

**private** WebElement producto;

@FindBy(how = How.*ID*, using = "orderPrice")

**private** WebElement precio;

**public** IngresarOrdenPage(WebDriver driver) {

**this**.driver = driver;

}

**public** String getProducto() {

**return** producto.getText();

}

**public** String getPrecio() {

**return** precio.getText();

}

}

@Test

**public** **void** MuestraLosDetallesDelProductoAlIngresarUnaOrden() {

ElegirProductoPage elegirProducto=**new** ElegirProductoPage(*driver*);

elegirProducto.Abrir();

IngresarOrdenPage ingresarOrden = elegirProducto.comprar("Orange Muffin");

*assertTrue*(ingresarOrden.getProducto().contains("Orange Muffin"));

*assertTrue*(ingresarOrden.getPrecio().contains("6.99"));

}