

Automation Code Exercise

Frank Sejas

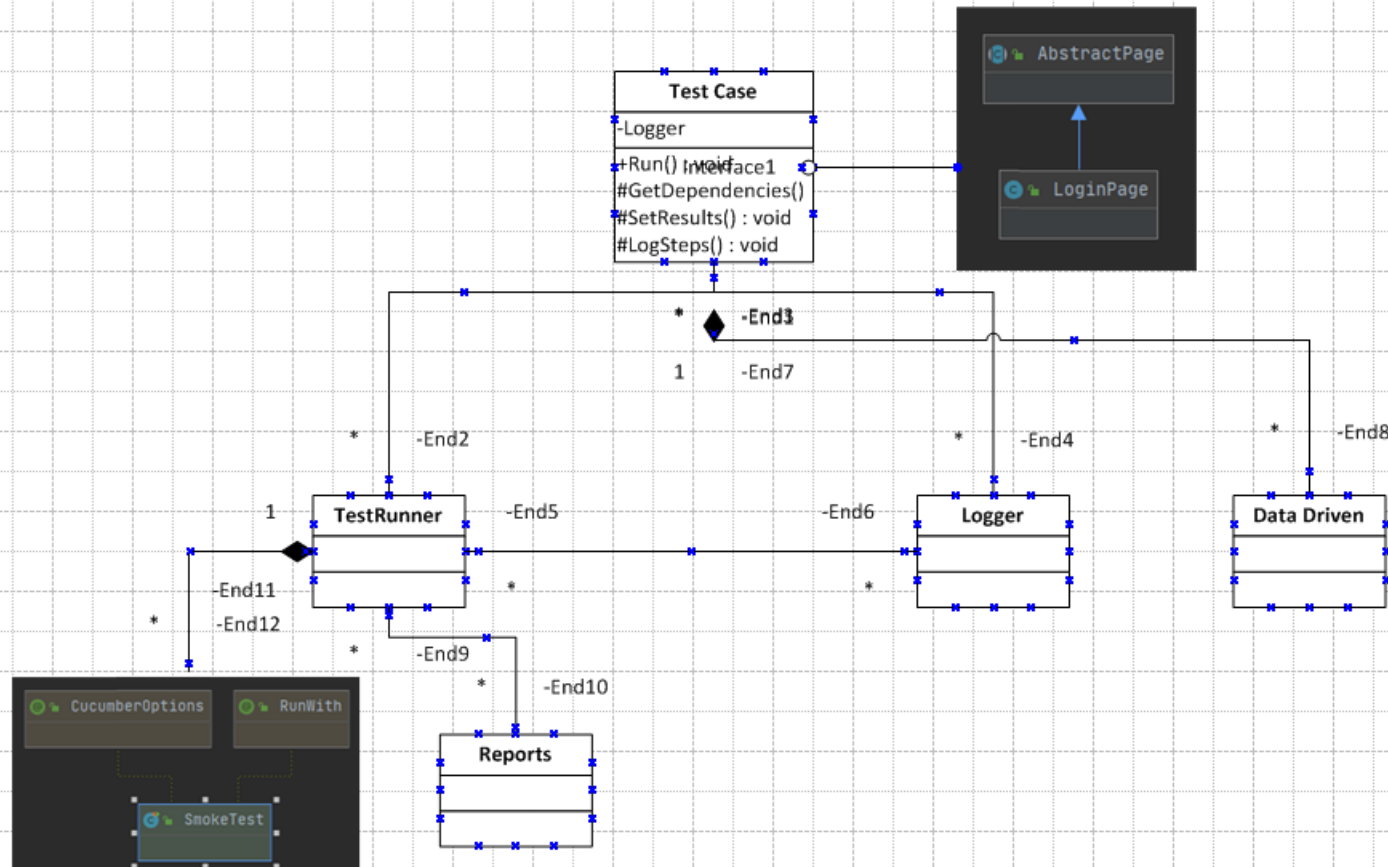
January, 2021

Tool Selected

- **Selenide** is basically a wrapper for Selenium WebDriver.
- Documentation:
<https://selenide.org/documentation.html>

Framework Design

Automation Framework Design

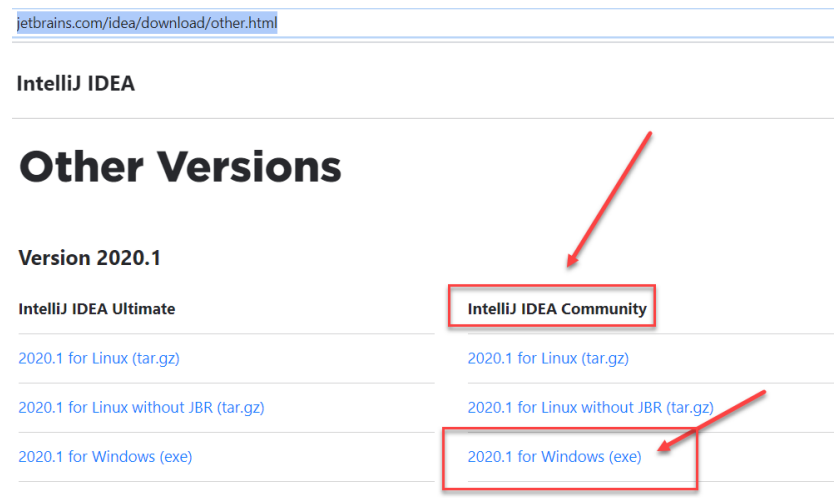


Setup

- Install Java JDK from <https://www.oracle.com/java/technologies/javase-jdk8-downloads.html>

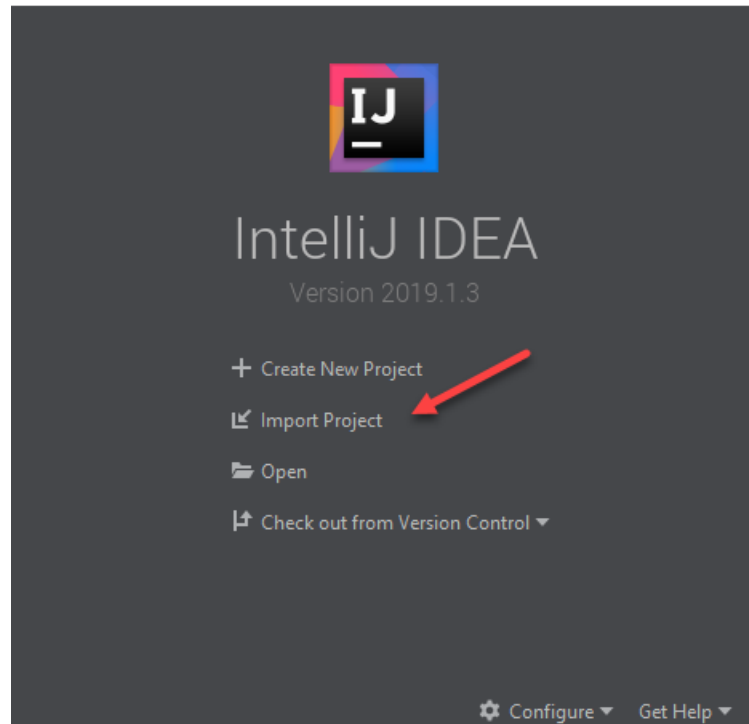
Setup

- Download IntelliJ from <https://www.jetbrains.com/idea/download/other.html>
- Windows version or the platform needed.



Setup

- Import the project from IntelliJ



Setup

- Select the following options:

Import Project

- 1
- ☐ Create project from existing sources
 - ☒ Import project from external model

Android Gradle
Eclipse
Gradle
Maven

2

Root directory: C:\CodeExercise

☐ Search for projects recursively

Project format: .idea (directory based)

☐ Keep project files in

☒ Import Maven projects automatically

☒ Detect compiler automatically

Import Project

Please enter a name to create a new IntelliJ IDEA project.

Project name: uittests

Project file location: C:\CodeExercise

3

Import Project

Select Maven projects to import

☒ code.automation:uittests:1.0-SNAPSHOT

4

5

Import Project

Please select project SDK.
This SDK will be used by default by all project modules.

+ -

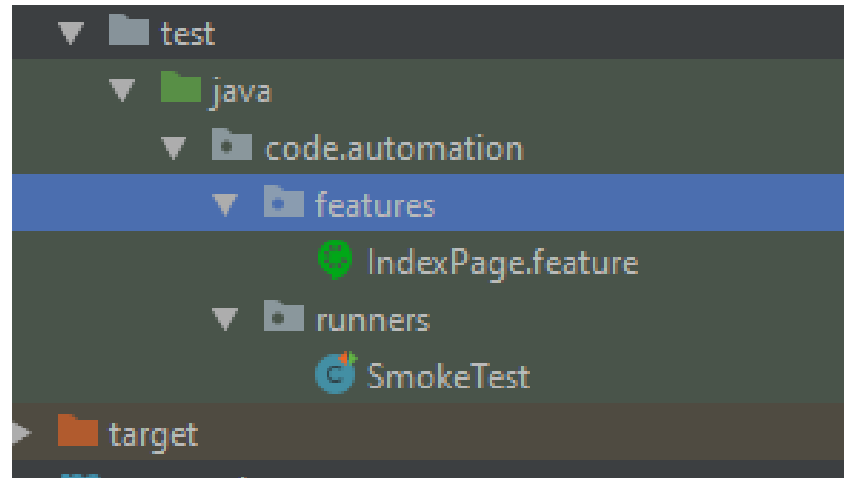
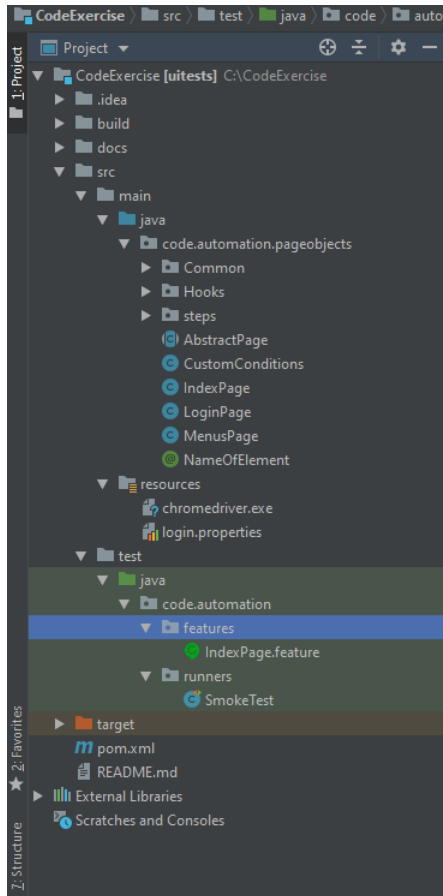
1.8

Name: 1.8

JDK home path: C:\Program Files\Java\jdk1.8.0_161

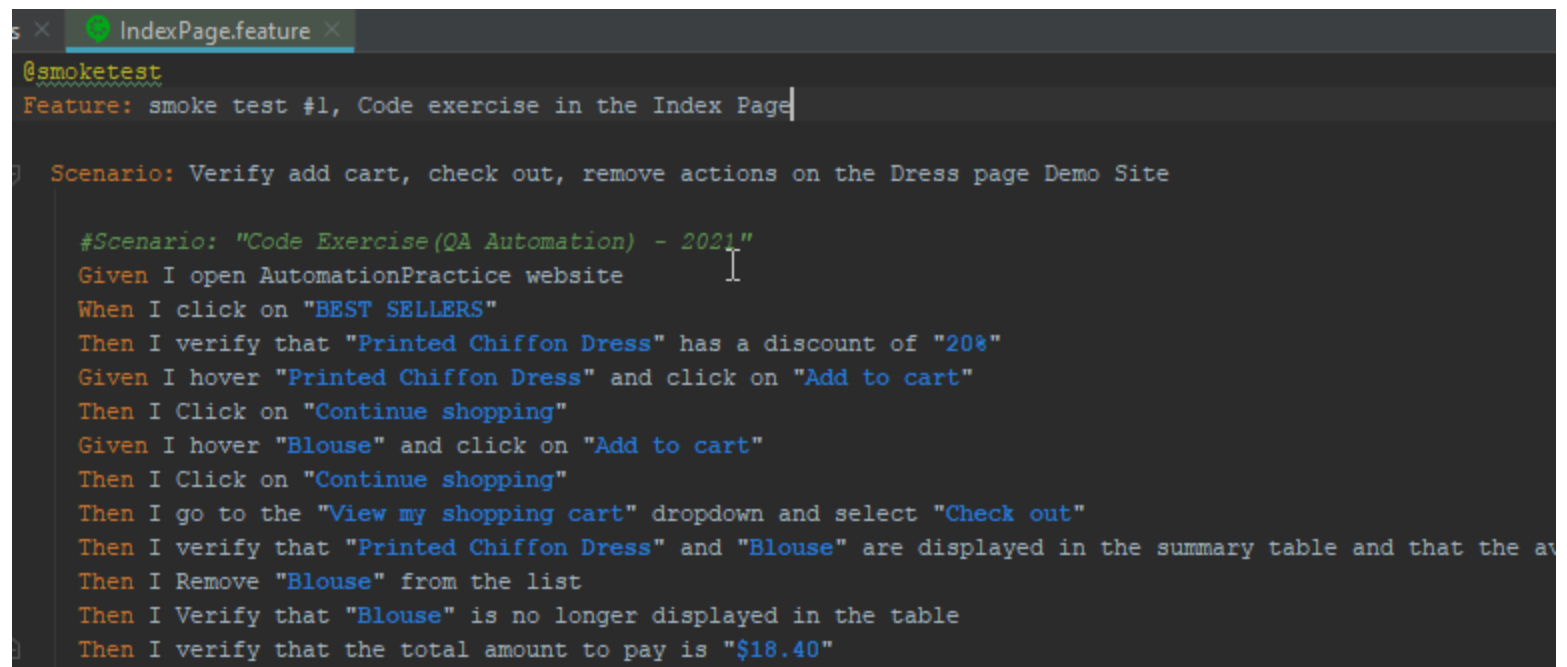
Classpath	Sourcepath	Annotations	Documentation Paths
C:\Program Files\Java\jdk1.8.0_161\jre\lib\charsets.jar			
C:\Program Files\Java\jdk1.8.0_161\jre\lib\deploy.jar			

Basic Framework Structure



Features

- Index Smoke Test



The screenshot shows a code editor with a file named 'IndexPage.feature'. The code is written in Gherkin syntax for a smoke test. It includes a scenario titled 'Verify add cart, check out, remove actions on the Dress page Demo Site'. The steps in the scenario involve opening the 'AutomationPractice' website, clicking on 'BEST SELLERS', verifying a 20% discount on 'Printed Chiffon Dress', adding it to the cart, clicking 'Continue shopping', adding a 'Blouse' to the cart, clicking 'Continue shopping', selecting 'Check out' from the shopping cart dropdown, verifying both items are in the summary table, removing the 'Blouse', verifying it is no longer in the table, and finally verifying the total amount is '\$18.40'.

```
IndexPage.feature
@smoketest
Feature: smoke test #1, Code exercise in the Index Page

Scenario: Verify add cart, check out, remove actions on the Dress page Demo Site

    #Scenario: "Code Exercise (QA Automation) - 2021"
    Given I open AutomationPractice website
    When I click on "BEST SELLERS"
    Then I verify that "Printed Chiffon Dress" has a discount of "20%"
    Given I hover "Printed Chiffon Dress" and click on "Add to cart"
    Then I Click on "Continue shopping"
    Given I hover "Blouse" and click on "Add to cart"
    Then I Click on "Continue shopping"
    Then I go to the "View my shopping cart" dropdown and select "Check out"
    Then I verify that "Printed Chiffon Dress" and "Blouse" are displayed in the summary table and that the av
    Then I Remove "Blouse" from the list
    Then I Verify that "Blouse" is no longer displayed in the table
    Then I verify that the total amount to pay is "$18.40"
```

Step Definitions

```
feature x
  @IndexSteps.java x

  public void clickOnLink(String link)
  {
    //Switch case has an issue reported here: https://www.programmersought.com/article/71624700632/
    //For this reason was use the if selection case to extend more cases and re use this method
    if (link.contains(link))
    {
      indexPage.get(link).waitUntil(Condition.appears, 30000);
      indexPage.get(link).click();
      log.log(Level.INFO, msg: "Click on: " + link);
    }
  }

  @When("^I hover \"([^\"]*)\" and click on \"([^\"]*)\"$")
  public void hoverAndClickOnAddCart(String dress, String button)
  {
    //Using the parameter values to interact with the method
    indexPage.HoverOverDress(dress);
    indexPage.ClickOnAddCartButton(button, dress).click();
    log.log(Level.INFO, msg: "Hover on: " + dress + " and click on: " + button);
  }

  @When("^I Click on \"([^\"]*)\"$")
  public void clickOnContinueShopping(String button)
  {
    indexPage.ClickOnContinueButton(button).waitUntil(Condition.appears, 30000);
    indexPage.ClickOnContinueButton(button).click();
    log.log(Level.INFO, msg: "Click on: " + button);
  }

  @When("^I go to the \"([^\"]*)\" dropdown and select \"([^\"]*)\"$")
  public void goToTheCartDropDownAndSelectCheckout(String cart, String button)
  {
    //Switch case has an issue reported here: https://www.programmersought.com/article/71624700632/
    //For this reason was use the if selection case to extend more cases and re use this method
    if (cart.contains("View my shopping cart"))
    {
      indexPage.HoverDropDownAndClickOnButton(cart, button).waitUntil(Condition.appears, 30000);
      indexPage.HoverDropDownAndClickOnButton(cart, button).click();
      log.log(Level.INFO, msg: "Hover on: " + cart + " and click on: " + button);
    }
  }
```

Mapped Elements

```
page.feature x IndexSteps.java x IndexPage.java x
import ...

public class IndexPage extends AbstractPage
{

    @NameOfElement("BEST SELLERS")
    @FindBy(xpath = ".//*[@href='#blockbestsellers']")
    public SelenideElement BestSeller;

    @NameOfElement("Printed Chiffon Dress")
    @FindBy(xpath = ".//*[@id='blockbestsellers']//*[title='Printed Chiffon Dress']")
    public SelenideElement PrintedChiffonDress;

    @NameOfElement("Printed Chiffon Dress percent display")
    @FindBy(xpath = ".//ul[@id='blockbestsellers']//a[@title='Printed Summer Dress']//div")
    public SelenideElement PrintedChiffonDressPercentDisplay;

    @NameOfElement("Printed Chiffon Dress Image")
    @FindBy(xpath = ".//ul[@id='blockbestsellers']//a[@title='Printed Summer Dress']//div")
    public SelenideElement PrintedChiffonDressImage;

    @NameOfElement("Add to cart button")
    @FindBy(xpath = ".//ul[@id='blockbestsellers']//a[@title='Add to cart']//div[@class='add-to-cart-button']")
    public SelenideElement AddToCartButton;

    @NameOfElement("Summary Table")
    @FindBy(xpath = ".//table[@id='cart_summary']/tbody")
    public SelenideElement SummaryTable;

    @NameOfElement("Total Price")
    @FindBy(xpath = ".//td[@id='total_price_container']/span[@id='total_price']")
    public SelenideElement TotalPriceText;

    public SelenideElement HoverOverDress(String dress)
    {
        return $(byXpath(".//ul[@id='blockbestsellers']//a[@title='"+ dress +"']")).hover();
    }
}
```

Reports

Cucumber-jvm Report			Features	Tags	Steps
---------------------	--	--	----------	------	-------

Project	Number	Date
Code Exercise Automation Results	0	24 Jan 2021, 17:40

Feature Statistics

The following graph shows passing and failing statistics for features



Feature	Scenarios			Steps							Duration	Status
	Total	Passed	Failed	Total	Passed	Failed	Skipped	Pending	Undefined	Missing		
smoke test #1, go through the Login page	1	1	0	12	12	0	0	0	0	0	17s 586ms	Passed
1	1	1	0	12	12	0	0	0	0	0	17s 586ms	Totals

Reports

Cucumber-jvm Report			Features	Tags	Steps
Code Exercise Automation Results	0	24 Jan 2021, 17:40			

Step Statistics

The following graph shows step statistics for this build. Below list is based on results. step does not provide information about result then is not listed below. Additionally @Before and @After are not counted because they are part of the scenarios, not steps.

Implementation	Occurrences	Duration	Average	Ratio
IndexSteps.openAutomationWebApp()	1	06s 805ms	06s 805ms	100.0%
IndexSteps.clickOnLink(String)	1	03s 178ms	03s 178ms	100.0%
IndexSteps.verifyThatDressIsNotLongerDisplayedInTheTable(String)	1	03s 005ms	03s 005ms	100.0%
IndexSteps.clickOnContinueShopping(String)	2	01s 579ms	789ms	100.0%
IndexSteps.verifyThatItemsAreDisplayed(String,String,String)	1	01s 338ms	01s 338ms	100.0%
Browser.tearDown()	1	966ms	966ms	100.0%
IndexSteps.goToTheCartDropDownAndSelectCheckout(String,String)	1	665ms	665ms	100.0%
IndexSteps.hoverAndClickOnAddCart(String,String)	2	496ms	248ms	100.0%
IndexSteps.verifyThatTheTotalAmountToPay(String)	1	262ms	262ms	100.0%
IndexSteps.removeItemFromTheList(String)	1	170ms	170ms	100.0%
IndexSteps.verifyDiscount(String,String)	1	083ms	083ms	100.0%
Browser.setUp()	1	000ms	000ms	100.0%

Reports

Cucumber-jvm Report			Features	Tags	Steps
Project	Number	Date			
Code Exercise Automation Results	0	24 Jan 2021, 17:40			

Feature Report

Feature	Scenarios			Steps							Duration	Status
	Total	Passed	Failed	Total	Passed	Failed	Skipped	Pending	Undefined	Missing		
smoke test #1, go through the Login page	1	1	0	12	12	0	0	0	0	0	17s 586ms	Passed

@smoketest

Feature: smoke test #1, go through the Login page

Before *Browser.setUp()*

Scenario: Verify Add Patient was created successful

Given I open AutomationPractice website06s 805ms

When I click on "BEST SELLERS"03s 178ms

Then I verify that "Printed Chiffon Dress" has a discount of "20%"083ms

Given I hover "Printed Chiffon Dress" and click on "Add to cart"138ms

Then I Click on "Continue shopping"748ms

Given I hover "Blouse" and click on "Add to cart"358ms

Then I Click on "Continue shopping"830ms

Then I go to the "View my shopping cart" dropdown and select "Check out"665ms

Then I verify that "Printed Chiffon Dress" and "Blouse" are displayed in the summary table and that the availability displays "In stock"01s 338ms

Then I Remove "Blouse" from the list170ms

Then I Verify that "Blouse" is no longer displayed in the table03s 005ms

Then I verify that the total amount to pay is "\$18.40"262ms

After *Browser.tearDown()*

5:42 PM

BRACE YOURSELF

AUTOMATION IS COMING

