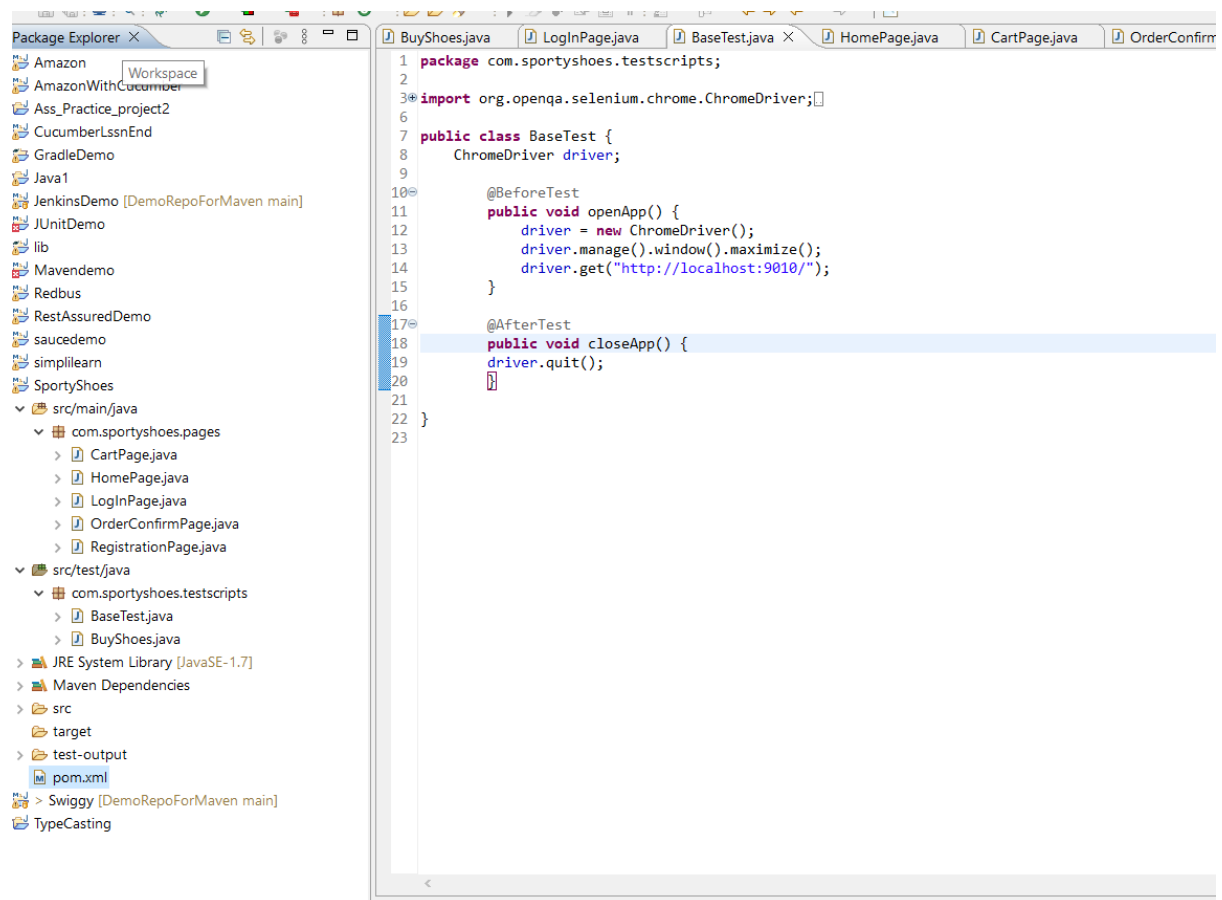


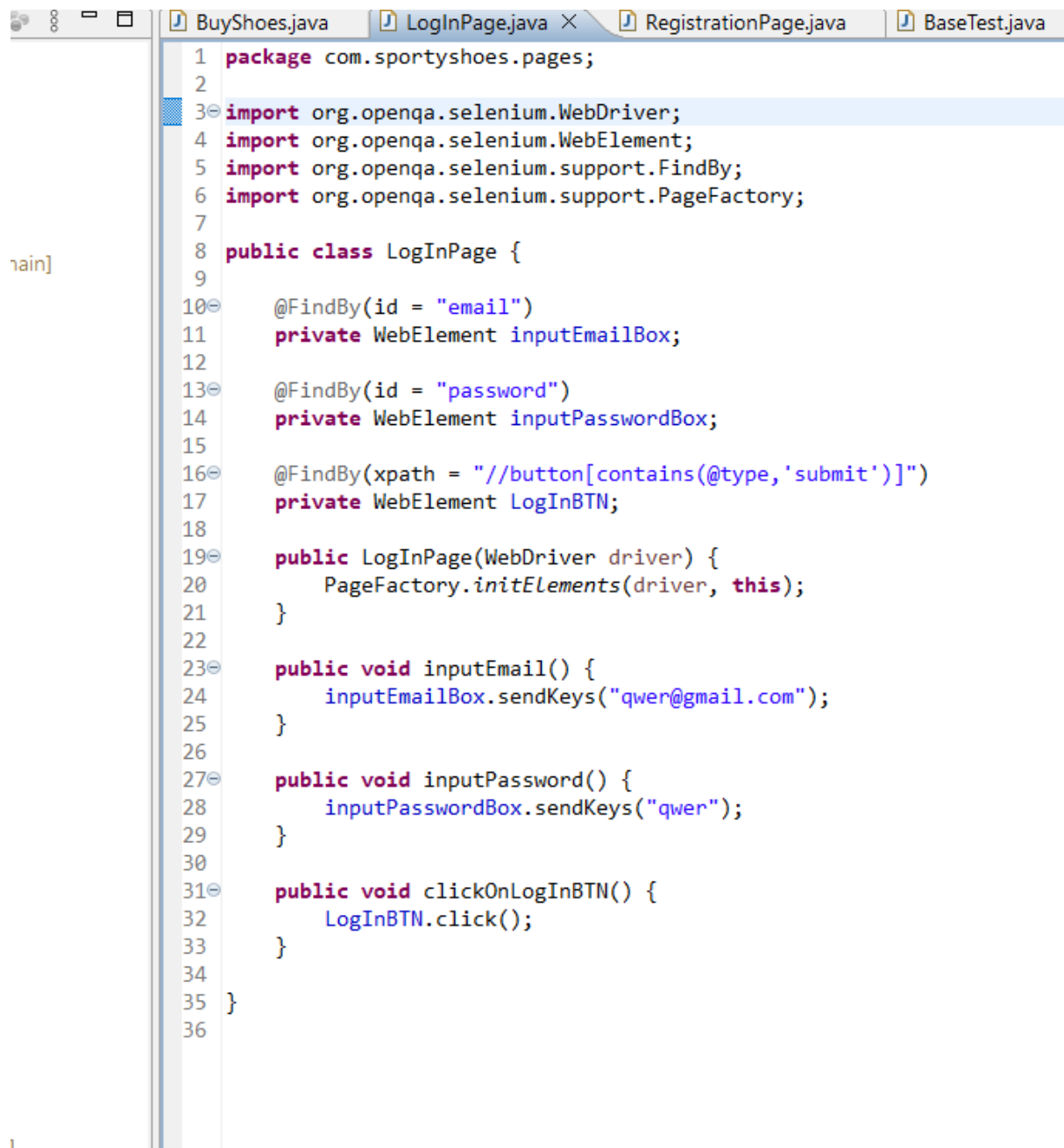
BASETEST:



REGISTRATION PAGE:

```
2
3 import org.openqa.selenium.WebDriver;
7
8 public class RegistrationPage {
9
10     @FindBy(linkText = "New User? Register Here")
11     private WebElement regBTN;
12
13     @FindBy(id = "name")
14     private WebElement inputNameBox;
15
16     @FindBy(id = "email")
17     private WebElement inputEmailBox;
18
19     @FindBy(id = "password")
20     private WebElement inputPasswordBox;
21
22     @FindBy(xpath = "//button[contains(@type,'submit')]")
23     private WebElement regiBTN;
24
25     public RegistrationPage(WebDriver driver) {
26         PageFactory.initElements(driver, this);
27     }
28
29     public void regPage() {
30         regBTN.click();
31     }
32
33     public void inputName() {
34         inputNameBox.sendKeys("sfEFWsfzfarfa");
35     }
36
37     public void inputEmail() {
38         inputEmailBox.sendKeys("qwer@gmail.com");
39     }
40
41     public void inputPassword() {
42         inputPasswordBox.sendKeys("qwer");
43     }
44
45     public void clickOnRegBTN() {
46         regiBTN.click();
47     }
48
49 }
```

LOGIN PAGE:



```
1 package com.sportyshoes.pages;
2
3 import org.openqa.selenium.WebDriver;
4 import org.openqa.selenium.WebElement;
5 import org.openqa.selenium.support.FindBy;
6 import org.openqa.selenium.support.PageFactory;
7
8 public class LogInPage {
9
10     @FindBy(id = "email")
11     private WebElement inputEmailBox;
12
13     @FindBy(id = "password")
14     private WebElement inputPasswordBox;
15
16     @FindBy(xpath = "//button[contains(@type,'submit')]")
17     private WebElement LogInBTN;
18
19     public LogInPage(WebDriver driver) {
20         PageFactory.initElements(driver, this);
21     }
22
23     public void inputEmail() {
24         inputEmailBox.sendKeys("qwer@gmail.com");
25     }
26
27     public void inputPassword() {
28         inputPasswordBox.sendKeys("qwer");
29     }
30
31     public void clickOnLogInBTN() {
32         LogInBTN.click();
33     }
34
35 }
36
```

HOMEPAGE:

```

1 package com.sportyshoes.pages;
2
3 import java.time.Duration;
4
11 public class HomePage {
12
13     @FindBy(xpath = "//a[contains(@class,'btn btn-primary')][1]")
14     private WebElement firstItem;
15
16
17     @FindBy(xpath = "//a[contains(@class,'btn btn-primary')][2]")
18     private WebElement secItem;
19
20     @FindBy(linkText = "Logout")
21     private WebElement logoutBTN;
22
23     JavascriptExecutor js;
24     WebDriverWait wait;
25     Actions action;
26
27
28     public HomePage(WebDriver driver) {
29         js = (JavascriptExecutor) driver;
30         PageFactory.initElements(driver, this);
31         action = new Actions(driver);
32         wait = new WebDriverWait(driver, Duration.ofSeconds(60));
33     }
34
35     public void addFirstItem() throws InterruptedException {
36         Thread.sleep(2000);
37         js.executeScript("arguments[0].scrollIntoView();", firstItem);
38         js.executeScript("arguments[0].click()", firstItem);
39     }
40
41     public void addSecondItem() {
42         action.moveToElement(secItem).click().perform();
43     }
44
45     public void clickOnLogoutBTN() {
46         logoutBTN.click();
47     }
48
49 }
50

```

CARTPAGE:

```

1 package com.sportyshoes.pages;
2
3 import java.time.Duration;
4
12 public class CartPage {
13
14     @FindBy(xpath = "//a[contains(@class,'nav-link')][1]")
15     private WebElement cart;
16
17     @FindBy(xpath = "//a[contains(@class,'btn btn-primary')][1]")
18     private WebElement placeOrdBTN;
19
20     @FindBy(xpath = "//a[contains(@class,'nav-link')][1]")
21     private WebElement homeBTN;
22
23     JavascriptExecutor js;
24     WebDriverWait wait;
25     Actions action;
26
27     public CartPage(WebDriver driver) {
28         js = (JavascriptExecutor) driver;
29         PageFactory.initElements(driver, this);
30         action = new Actions(driver);
31         wait = new WebDriverWait(driver, Duration.ofSeconds(60));
32     }
33
34     public void goToCart() {
35         cart.click();
36     }
37
38     public void clickOnPlaceOrdBTN() throws InterruptedException {
39         Thread.sleep(2000);
40         js.executeScript("arguments[0].scrollIntoView();", placeOrdBTN);
41         js.executeScript("arguments[0].click()", placeOrdBTN);
42     }
43
44     public void clickOnHomeBTN() {
45         homeBTN.click();
46     }
47
48 }
49
50

```

ORDERCONFIRMATION PAGE:

n]

```
1 package com.sportyshoes.pages;
2
3 import org.openqa.selenium.WebDriver;
4 import org.openqa.selenium.WebElement;
5 import org.openqa.selenium.support.FindBy;
6 import org.openqa.selenium.support.PageFactory;
7
8 public class OrderConfirmPage {
9
10     @FindBy(xpath = "/html/body/div[3]/div/p")
11     private WebElement orderPlacedMsg;
12
13     @FindBy(xpath = "/html/body/div[3]/div/p")
14     private WebElement itemAddedMsg;
15
16     public OrderConfirmPage(WebDriver driver) {
17         PageFactory.initElements(driver, this);
18     }
19
20     public String getMsg() {
21         return orderPlacedMsg.getText();
22     }
23
24     public String getMsg1() {
25         return itemAddedMsg.getText();
26     }
27
28
29
30 }
31
```

CONSOLE OUTPUT:

Console ×Results of running class BuyShoes

<terminated> BuyShoes [TestNG] C:\Program Files\Java\jdk-20\bin\javaw.exe (24-Aug-2023, 4:48:04 pm – 4:48:14 pm) [pid: 2376]
[RemoteTestNG] detected TestNG version 7.7.1
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
Aug 24, 2023 4:48:08 PM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
WARNING: Unable to find an exact match for CDP version 116, so returning the closest version found: 115
PASSED: com.sportyshoes.testscripts.BuyShoes.buyShoes

=====
Default test
Tests run: 1, Failures: 0, Skips: 0
=====

ages

age.java
e.java

estscripts

ivaSE-1.7]

rMaven main]

TESTNG REPORT:

Test	# Passed	# Skipped	# Retried	# Failed	Time (ms)	Included Groups	Excluded Groups
Default suite							
Default test	1	0	0	0	7,384		

Class	Method	Start	Time (ms)
Default suite			
Default test — passed			
com.sportyshoes.testscripts.BuyShoes	buyShoes	1692875888784	4955

Default test

com.sportyshoes.testscripts.BuyShoes#buyShoes

[back to summary](#)

API TESTING IN POSTMAN:

1. GET ALL PRODUCT DETAILS:

← → Home Workspaces API Network Explore Search Postman

workspace1 New Import Overview POST postProd GET getAllProd + ... OpenWeather API

OpenWeather
Reques
SportyShoes
GET getAllProd
GET getAllUsers
POST postProd

SportyShoes / getAllProd

GET http://localhost:9010/get-shoes Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

Key	Value	Description	Bulk Edit
Key	Value	Description	

Body Cookies Headers (5) Test Results Status: 200 OK Time: 28 ms Size: 1.55 KB Save as Example

Pretty Raw Preview Visualize JSON

```
1 {
2   "code": 101,
3   "message": "12 Shoes Fetched Successfully.",
4   "shoes": [
5     {
6       "id": 101,
7       "image": "1.png",
8       "name": "BlueWave Running Shoes",
9       "category": "Sports",
10      "sizes": "7, 8, 9",
11      "price": 2000
12    },
13    {
14      "id": 201,
15      "image": "2.png",
16      "name": "Elegant Leather Loafers",
17      "category": "Formal",
18      "sizes": "7, 8, 9",
19      "price": 3000
20    },
21    {
22      "id": 301,
23      "image": "3.png",
24      "name": "NeoFlex Athletic Shoes".
```

2. GET ALL USERS DETAILS

workspace1 New Import Overview POST postProd GET getAllProd GET getAllUsers + ... OpenWeather API

OpenWeather
Reques
SportyShoes
GET getAllProd
GET getAllUsers
POST postProd

SportyShoes / getAllUsers

GET http://localhost:9010/get-users Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

Key	Value	Description	Bulk Edit
Key	Value	Description	

Body Cookies Headers (5) Test Results Status: 200 OK Time: 8 ms Size: 440 B Save as Example

Pretty Raw Preview Visualize JSON

```
1 {
2   "code": 101,
3   "message": "3 Users Fetched Successfully.",
4   "users": [
5     {
6       "name": "John Watson",
7       "email": "john@example.com",
8       "password": "john123"
9     },
10    {
11      "name": "Fionna Flynn",
12      "email": "fionna@example.com",
13      "password": "fionna123"
14    },
15    {
16      "name": "Sia Sen",
17      "email": "sia@example.com",
18      "password": "sia123"
19    }
20  ]
21 }
```

3. POST ONE PRODUCT:

The screenshot shows a REST client interface for a project named 'SportyShoes'. The selected endpoint is 'postProd' with a POST method. The URL is 'http://localhost:9010/add-shoe?id=101&image=image_url&name=SampleShoe&category=Running&sizes=9&price=1000'. The 'Query Params' tab is active, displaying a table of parameters:

Key	Value	Description
id	101	
image	image_url	
name	SampleShoe	
category	Running	
sizes	9	
price	1000	

The 'Body' tab shows the response in JSON format:

```

1  {
2    "code": 101,
3    "message": "SampleShoe Added Successfully.",
4    "shoe": {
5      "id": 101,
6      "image": "image_url",
7      "name": "SampleShoe",
8      "category": "Running",
9      "sizes": "9",
10     "price": 1000
11   }
12 }

```

Additional details: Status: 200 OK, Time: 23 ms, Size: 398 B.

API TESTING WITH REST-ASSURED:

```

1 package in.amazon.RestAssuredDemo;
2
3 import org.testng.annotations.Test;
4
5
6
7 public class Get_Post_On_SportyShoes extends TestBase{
8
9     @Test(priority=0)
10    public void getAllProd() {
11        RestAssured
12            .when()
13            .get("http://localhost:9010/get-shoes")
14            .then()
15            .assertThat()
16            .statusCode(200)
17            .log().all();
18
19        logger.info("All Product Details fetched Successfully");
20    }
21
22    @Test(priority=1)
23    public void getAllUsers() {
24        RestAssured
25            .when()
26            .get("http://localhost:9010/get-users")
27            .then()
28            .assertThat()
29            .statusCode(200)
30            .log().all();
31
32        logger.info("All Users Details fetched Successfully");
33    }
34
35    @Test(priority=2)
36    public void addProd() {
37        RestAssured
38            .when()
39            .post("http://localhost:9010/add-shoe?id=101&image=image_url&name=SampleShoe&category=Running&sizes=9&price=1000")
40            .then()
41            .assertThat()
42            .statusCode(200)
43            .log().all();
44
45        logger.info("Add one Product Successfully");
46    }
47 }

```

OUTPUT:

```

Console X  Results of running class Get_Post_On_SportyShoes
<terminated> Get_Post_On_SportyShoes [TestNG] C:\Program Files\Java\jdk-20\bin\javaw.exe (24-Aug-2023, 4:59:16 pm – 4:59:19 pm) [pid: 13540]
Content-Type: application/json
Transfer-Encoding: chunked
Date: Thu, 24 Aug 2023 11:29:19 GMT
Keep-Alive: timeout=60
Connection: keep-alive

{
  "code": 101,
  "message": "SampleShoe Added Successfully.",
  "shoe": {
    "id": 101,
    "image": "image_url",
    "name": "SampleShoe",
    "category": "Running",
    "sizes": "9",
    "price": 1000
  }
}
INFO    2023-08-24 16:59:19,643 103    LogDemo [main]  Add one Product Successfully
PASSED: getAllProd
PASSED: getAllUsers
PASSED: addProd

=====
      Default test
      Tests run: 3, Failures: 0, Skips: 0
=====

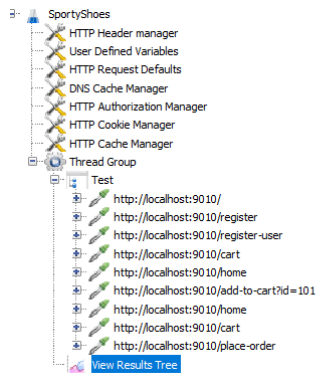
=====
Default suite
Total tests run: 3, Passes: 3, Failures: 0, Skips: 0
=====

```

LOAD TESTING WITH JMETER:

(WITH 50 USERS)

The screenshot displays the Apache JMeter GUI. On the left, a tree view shows the test plan structure for 'SportyShoes', including HTTP Header manager, User Defined Variables, HTTP Request Defaults, DNS Cache Manager, HTTP Authorization Manager, HTTP Cookie Manager, HTTP Cache Manager, and a Thread Group. The Thread Group is expanded, showing a Test element with a sequence of HTTP requests: http://localhost:9010/, http://localhost:9010/register, http://localhost:9010/register-user, http://localhost:9010/cart, http://localhost:9010/home, http://localhost:9010/add-to-cart?id=101, http://localhost:9010/home, http://localhost:9010/cart, and http://localhost:9010/place-order. On the right, the 'Thread Group' configuration panel is visible. It shows the Name as 'Thread Group' and the Number of Threads (users) set to 50. Other settings include Ramp-up period (seconds) set to 2, Loop Count set to 1, and the 'Same user on each iteration' checkbox checked. The 'Duration (seconds)' and 'Startup delay (seconds)' are both set to 0.



View Results Tree

Name: View Results Tree

Comments:

Write results to file / Read from file

Filename

Search: ☐ Case sensitive ☐ Regular exp.

Text

Sampler result
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/register
✓ http://localhost:9010/register
✓ http://localhost:9010/register
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/
✓ http://localhost:9010/register
✓ http://localhost:9010/
✓ http://localhost:9010/register
✓ http://localhost:9010/
✓ http://localhost:9010/register
✓ http://localhost:9010/register
✓ http://localhost:9010/register
✓ http://localhost:9010/
✓ http://localhost:9010/register
✓ http://localhost:9010/register
✓ http://localhost:9010/
✓ http://localhost:9010/register
✓ http://localhost:9010/register
✓ http://localhost:9010/

☐ Scroll automatically

Raw Parsed