

Source Codes

```
package com.flyawayrunner;

import org.junit.runner.RunWith;
import io.cucumber.junit.Cucumber;
import io.cucumber.junit.CucumberOptions;

@RunWith(Cucumber.class)
@CucumberOptions(
    features = "src/test/java/Features",
    glue = "StepDefn",
    tags = "@tagBooking or @tagregister"
)
public class FlightRunner {

}
```

```
package StepDefn;

import java.util.List;

import org.junit.Assert;
import org.openqa.selenium.By;
import org.openqa.selenium.TimeoutException;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.Select;
import org.openqa.selenium.support.ui.WebDriverWait;

import io.cucumber.java.en.Given;
import io.cucumber.java.en.Then;
import io.cucumber.java.en.When;

public class FlyawayBookingsteps {

    public static WebDriver driver;

    @Given("Sam is on Registration page")
    public void sam_is_on_registration_page() {
        // Write code here that turns the phrase above into concrete actions
        System.setProperty("webdriver.chrome.driver",

"C:\\Users\\jsahani\\install\\Selenium\\chromedriver\\chromedriver.exe");
        driver = new ChromeDriver();
        driver.get("http://localhost:8080/FlyAway/signup");
        driver.manage().window().maximize();
        System.out.println("Sam is on the SignUp page");
    }
}
```

```
}
```

```
@When("Sam enters {string},{string}, {string}, {string}, {string} and {string}")
```

```
public void sam_enters_and(String emailid, String pwd, String pwd2, String name,  
String address, String city) {
```

```
    // Write code here that turns the phrase above into concrete actions
```

```
    WebElement email_id1 = driver.findElement(By.name("email_id"));
```

```
    email_id1.sendKeys(emailid);
```

```
    WebElement password = driver.findElement(By.name("pwd"));
```

```
    password.sendKeys(pwd);
```

```
    WebElement password2 = driver.findElement(By.name("pwd2"));
```

```
    password2.sendKeys(pwd2);
```

```
    WebElement name1 = driver.findElement(By.name("name"));
```

```
    name1.sendKeys(name);
```

```
    WebElement address1 = driver.findElement(By.name("address"));
```

```
    address1.sendKeys(address);
```

```
    WebElement city1 = driver.findElement(By.name("city"));
```

```
    city1.sendKeys(city);
```

```
    System.out.println("Sam Enters Registrations Details");
```

```
}
```

```
@When("Sam clicks on Signup")
```

```
public void sam_clicks_on_signup() {
```

```
    // Write code here that turns the phrase above into concrete actions
```

```
    System.out.println("Sam clicks on Signup");
```

```
    WebElement signup = driver.findElement(By.tagName("button"));
```

```
    signup.click();
```

```
}
```

```
@When("Sam should be able to check {string}")
```

```
public void sam_should_be_able_to_check(String string) {
```

```

        // Write code here that turns the phrase above into concrete actions
        System.out.println("Sam is able to Register successfully");

        List<WebElement> listLogin =
driver.findElements(By.linkText("Login/Signup"));

        Assert.assertTrue(listLogin.size() > 0);
    }

    @Then("Sam should be navigated to booking")
    public void sam_should_be_navigated_to_booking() {
        // Write code here that turns the phrase above into concrete actions
        System.out.println("Sam is navigated to home page");
        String strTitle = driver.getTitle();
        System.out.println(strTitle);
    }

    @Given("Sam is on the login page")
    public void sam_is_on_the_login_page() {
        // Write code here that turns the phrase above into concrete actions

        System.setProperty("webdriver.chrome.driver", "C:\\\\Users\\jsahani\\install\\Seleniu
m\\chromedriver\\chromedriver.exe");

        driver = new ChromeDriver();
        driver.get("http://localhost:8080/FlyAway/login");
        driver.manage().window().maximize();
        System.out.println("Sam is on the login page");
    }

    @When("Sam enters {string} and {string}")
    public void sam_enters_and(String un, String pwd) {
        // Write code here that turns the phrase above into concrete actions
        System.out.println("Sam Enters UN and PWD");
        WebElement username = driver.findElement(By.name("email_id"));
        username.sendKeys(un);
        WebElement password = driver.findElement(By.name("pwd"));
        password.sendKeys(pwd);
    }

```

```

}

@When("Sam clicks on Login")
public void sam_clicks_on_login() {
    // Write code here that turns the phrase above into concrete actions
    System.out.println("Sam clicks on Login");
    WebElement login = driver.findElement(By.tagName("button"));
    login.click();
}

@When("Sam should be able to validate {string}")
public void sam_should_be_able_to_validate(String string) {
    // Write code here that turns the phrase above into concrete actions
    System.out.println("Sam should be able to login successfully");
    List<WebElement> listLogout = driver.findElements(By.linkText("Logout"));
    Assert.assertTrue(listLogout.size() > 0);
}

@Then("Sam should be navigated to homepage")
public void sam_should_be_navigated_to_homepage() {
    // Write code here that turns the phrase above into concrete actions
    System.out.println("Sam should navigate to home page");
    String strTitle = driver.getTitle();
    System.out.println(strTitle);
}

@Given("Sam has login with {string} and {string}")
public void sam_has_login_with_and(String string, String string2) {
    // Write code here that turns the phrase above into concrete actions

    System.setProperty("webdriver.chrome.driver", "C:\\\\Users\\jsahani\\install\\Seleniu
m\\chromedriver\\chromedriver.exe");

    driver = new ChromeDriver();
    driver.get("http://localhost:8080/FlyAway/login");
    driver.manage().window().maximize();
}

```

```

        driver.findElement(By.name("email_id")).sendKeys(string);
        driver.findElement(By.name("pwd")).sendKeys(string2);
        driver.findElement(By.tagName("button")).click();

        WebDriverWait wait = new WebDriverWait(driver, 10);
        try {

            wait.until(ExpectedConditions.invisibilityOfElementLocated(By.linkText("Logout")));
        } catch (TimeoutException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }

        int listLogout = driver.findElements(By.linkText("Logout")).size();
        Assert.assertTrue(listLogout > 0);
    }

    @Given("Same has open home page")
    public void same_has_open_home_page() {
        // Write code here that turns the phrase above into concrete actions
        driver.findElement(By.linkText("Home")).click();
    }

    @When("Sam selects {string} as source and {string} as destination")
    public void sam_selects_as_source_and_as_destination(String Source, String
Destination) {
        // Write code here that turns the phrase above into concrete actions
        Select selectsrc = new Select(driver.findElement(By.name("source")));
        Select selectDest = new Select(driver.findElement(By.name("destination")));
        selectsrc.selectByVisibleText(Source);
        selectDest.selectByVisibleText(Destination);
    }

    @When("Sam clicks on submit")
    public void sam_clicks_on_submit() {

```

```

        // Write code here that turns the phrase above into concrete actions
        driver.findElement(By.tagName("button")).click();
    }

    @Then("Sam is able to validate for flight id {string} with msg {string}")
    public void sam_is_able_to_validate_for_flight_id_with_msg(String flightNo, String
expMessage) {
        // Write code here that turns the phrase above into concrete actions
        List<WebElement> lstFlights = driver.findElements(By.linkText("Book Flight"));
        String actMsg = "Flight Not Present";
        for (WebElement flight : lstFlights) {
            System.out.println(flight.getAttribute("href"));
            System.out.println("Sam is able to validate for flight id");
            if (flight.getAttribute("href").contains("bookflight?id=" + flightNo)) {
                actMsg = "Flight Present";
                break;
            }
        }
    }
}

```

@tagregister

Feature: Register

I want to use this template for my feature file

@tagr

Scenario: Sam is able to register successfully

Given Sam is on Registration page

When Sam enters "<email_id>", "<password>", "<password>", "<Name>", "<Address>" and "<City>"

And Sam clicks on Signup

And Sam should be able to check "<signupstatus>"

Then Sam should be navigated to booking

Examples:

	email_id	password	password	Name	Address
City	signupstatus				
	tejasnaik@user.com	user	user	Jashwant	
Maharashtra	Navi Mumbai	RegisterPass			

@tagl

Feature: Login

I want to use this template for test Login functionality

@tagLogin

Scenario Outline: Sam is able to login successfully

Given Sam is on the login page

When Sam enters "<username>" and "<password>"

And Sam clicks on Login

And Sam should be able to validate "<loginstatus>"

Then Sam should be navigated to homepage

Examples:

username	password	loginstatus
user@user.com	user	LoginPass

@tagBooking

Feature: Booking

I want to use this template for test Flight Booking functionality

@tagSelectFlight

Scenario: Sam is able to select a flight

Given Sam has login with "<username>" and "<password>"

And Same has open home page

When Sam selects "<Source>" as source and "<Destination>" as destination

And Sam clicks on submit

Then Sam is able to validate for flight id "<FlightNo>" with msg "<ExpResult>"

Examples:

	username	password	Source	Destination
FlightNo	ExpResult			
5	user@user.com	user	Bangalore	Mumbai
	Flight Present			
10	user@user.com	user	Bangalore	Ahmedabad
	Flight NOT Present			

```
package com.listeners;
```

```
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.annotations.Test;

public class FlightBooking {
    WebDriver driver;
    @Test
    public void Registrationpage() {
        System.setProperty("webdriver.chrome.driver",
"C:\\\\Users\\tejasna\\install\\Selenium\\chromedriver\\chromedriver.exe");
        driver = new ChromeDriver();
        driver.get("http://localhost:8080/FlyAway/signup");
        driver.manage().window().maximize();
    }
    @Test(dependsOnMethods = "Registrationpage")
    public void Clicks_signup() {
        WebElement email_id1 =
driver.findElement(By.name("email_id"));
        email_id1.sendKeys("tejasnaik@user.com");
        WebElement password = driver.findElement(By.name("pwd"));
        password.sendKeys("user");
        WebElement password2 =
driver.findElement(By.name("pwd2"));
        password2.sendKeys("user");
        WebElement name1 = driver.findElement(By.name("name"));
        name1.sendKeys("tejas");
        WebElement address1 =
driver.findElement(By.name("address"));
        address1.sendKeys("Nerul4");
        WebElement city1 = driver.findElement(By.name("city"));
        city1.sendKeys("Navi-Mumbai");

        driver.findElement(By.xpath("//button[text()='Signup']")).click();
    }
    @Test(dependsOnMethods = "Clicks_signup")
```

```

        public void homepage() {
            driver.findElement(By.xpath("//a[contains(text(), 'Login
to continue checking flights')]")).click();

        }
        @Test(dependsOnMethods = "homepage")
        public void login_successfully() {

            driver.findElement(By.name("email_id")).sendKeys("user@user.com");
            driver.findElement(By.name("pwd")).sendKeys("user");

            driver.findElement(By.xpath("//button[text()='Login']")).click();
        }
        @Test(dependsOnMethods = "login_successfully")
        public void Flight_Booking_Details() {

            driver.findElement(By.xpath("//a[text()='Home']")).click();

            driver.findElement(By.xpath("//html/body/table/tbody/tr[3]/td[6]/a")
).click();
            driver.findElement(By.xpath("//a[text()='Click to
complete booking']")).click();
            driver.findElement(By.xpath("//a[text()='See Your
Bookings']")).click();

        }
    }
}

```

Testng

PASSED: Clicks_signup
 PASSED: Flight_Booking_Details
 PASSED: login_successfully
 PASSED: Registrationpage
 PASSED: homepage

```

=====
Default test
Tests run: 5, Failures: 0, Skips: 0
=====

```

```

=====
Default suite
Total tests run: 5, Passes: 5, Failures: 0, Skips: 0
=====

```

Testing.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<suite parallel="false" name="Suite">
  <test name="Test">
    <classes>
      <class name="com.listeners.FlightBooking"/>
    </classes>
  </test> <!-- Test -->
</suite> <!-- Suite -->
```

Pom.xml

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
    https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>FlyAwayCapstoneProj</groupId>
  <artifactId>FlyAwayCapstoneProj</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <dependencies>
  <dependency>
    <groupId>org.seleniumhq.selenium</groupId>
    <artifactId>selenium-java</artifactId>
    <version>3.141.59</version>
  </dependency>
  <!-- https://mvnrepository.com/artifact/org.testng/testng -->
  <dependency>
    <groupId>org.testng</groupId>
    <artifactId>testng</artifactId>
    <version>7.5</version>
    <scope>test</scope>
  </dependency>
  <!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-
java -->
  <dependency>
    <groupId>io.cucumber</groupId>
    <artifactId>cucumber-java</artifactId>
    <version>7.2.3</version>
  </dependency>

  <!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-
junit -->
  <dependency>
    <groupId>io.cucumber</groupId>
    <artifactId>cucumber-junit</artifactId>
    <version>7.2.3</version>
    <scope>test</scope>
  </dependency>
  <!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-
core -->
  <dependency>
    <groupId>io.cucumber</groupId>
    <artifactId>cucumber-core</artifactId>
    <version>7.2.3</version>
  </dependency>

</dependencies>
</project>
```

Jenkins

```
Started by user admin
Running as SYSTEM
Building in workspace C:\Users\tejasna\.jenkins\workspace\RunPostman
[RunPostman] $ cmd /c call
C:\Users\tejasna\AppData\Local\Temp\jenkins7378680367986986415.bat

C:\Users\tejasna\.jenkins\workspace\RunPostman>cd C:\Users\jsahani\Pracs
Automation\Postman\Collections

C:\Users\tejasna\Pracs Automation\Postman\Collections>newman run
Flyaway.postman_collection.json
newman

Flyaway

→ Signup
  POST
  http://localhost:8080/FlyAway/signupaction?email_id=user4@user.com&pwd=user&pwd2=user&name=user1&address=user&city=user [200 OK, 1.4kB, 59ms]

→ Login
  POST http://localhost:8080/FlyAway/loginaction?email_id=user4@user.com&pwd=user
  [200 OK, 591B, 27ms]

→ SelectFlight
  GET http://localhost:8080/FlyAway/home?source=1&destination=2
  [200 OK, 2.43kB, 28ms]

→ BookFlight
  GET http://localhost:8080/FlyAway/bookflight?id=5 [200 OK, 984B, 33ms]

→ ConfirmFlight
  GET http://localhost:8080/FlyAway/completepurchase [200 OK, 805B, 28ms]
```

	executed	failed
iterations	1	0
requests	5	0
test-scripts	5	0
prerequest-scripts	5	0
assertions	0	0
total run duration: 573ms		
total data received: 5.18kB (approx)		
average response time: 35ms [min: 27ms, max: 59ms, s.d.: 12ms]		

Finished: SUCCESS

