

Paso 1.

Creación del archivo de características. Este archivo contiene un escenario de inicio de sesión simple definido en sintaxis Gherkin. El login.feature archivo se encuentra en el src/test/resources directorio:

Feature: User Login

Scenario: Successful login with valid credentials

Given the user is on the login page

When the user enters valid username and password

Then the user should be redirected to the homepage

Paso 3. El archivo de definiciones de pasos relaciona estos métodos Java en el LoginSteps mapa de clases con los pasos en el archivo de características. La LoginSteps clase utiliza Selenium WebDriver para interactuar con la aplicación web durante la prueba:

Pasos de inicio de sesión.java

```
package stepdefinitions;
```

```
import org.openqa.selenium.By;
```

```
import org.openqa.selenium.WebDriver;
```

```
import org.openqa.selenium.WebElement;
```

```
import org.openqa.selenium.chrome.ChromeDriver;
```

```
import io.cucumber.java.After;
```

```
import io.cucumber.java.Before;
```

```
import io.cucumber.java.en.Given;
```

```
import io.cucumber.java.en.Then;
```

```
import io.cucumber.java.en.When;
```

```
public class LoginSteps {
```

```
    private WebDriver driver;
```

```
    private String url = "https://www.saucedemo.com/";
```

```
    // Initialize WebDriver and open the browser
```

@Before

```
public void setup() {
```

```
    System.setProperty("webdriver.chrome.driver", "C:\\Users\\change the path of the  
chromedriver\\drivers\\chromedriver.exe");
```

```
    driver = new ChromeDriver();
```

```
}
```

// Step definition for "Given the user is on the login page"

@Given("the user is on the login page")

```
public void userOnLoginPage() {
```

```
    if (!this.driver.getCurrentUrl().equals(this.url)) {
```

```
        this.driver.get(this.url);
```

```
    }
```

```
}
```

// Step definition for "When the user enters valid username and password"

@When("the user enters valid username and password")

```
public void userEntersValidCredentials() {
```

```
    WebElement userName = this.driver.findElement(By.id("user-name"));
```

```
    userName.sendKeys("standard_user");
```

```
    WebElement passWord = this.driver.findElement(By.id("password"));
```

```
    passWord.sendKeys("secret_sauce");
```

```
    WebElement loginBtn = this.driver.findElement(By.id("login-button"));
```

```
    loginBtn.click();
```

```
}
```

// Step definition for "Then the user should be redirected to the homepage"

@Then("the user should be redirected to the homepage")

```
public void userRedirectedToHomepage() {
```

```

String currentUrl = driver.getCurrentUrl();

assert currentUrl.equals("https://www.saucedemo.com/inventory.html");
}

// Cleanup after each test
@After
public void tearDown() {
    if (driver != null) {
        driver.quit();
    }
}
}

```

Paso 4. Crear la TestRunner clase conecta el archivo de características y las definiciones de pasos mediante la anotación de Cucumber @CucumberOptions. También genera informes HTML tras ejecutar las pruebas.

TestRunner.java

```
package testrunner;
```

```
import io.cucumber.testng.AbstractTestNGCucumberTests;
```

```
import io.cucumber.testng.CucumberOptions;
```

```
@CucumberOptions(
```

```
    features = "C:\\path_to_feature\\login.feature", // Correct path to the feature file
```

```
    glue = "stepdefinitions", // Path to step definitions
```

```
    plugin = {"pretty", "html:target/cucumber-reports"} // Generate HTML report
```

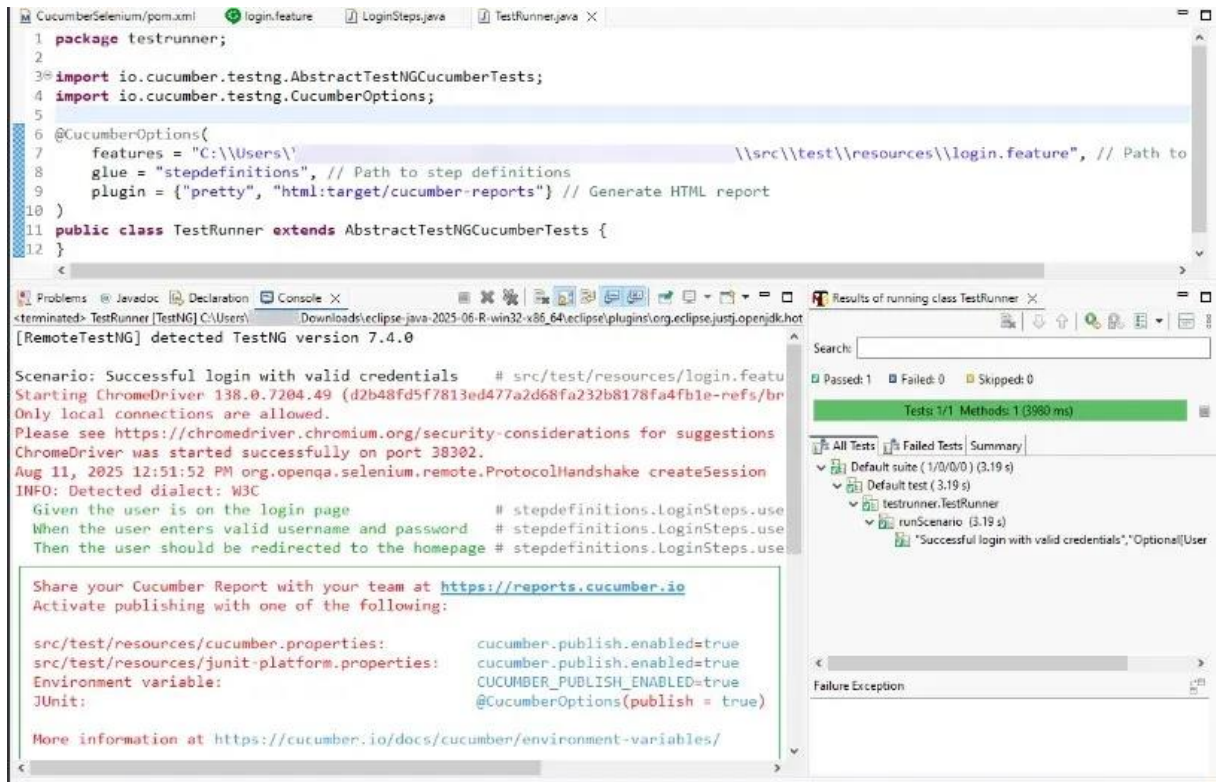
```
)
```

```
public class TestRunner extends AbstractTestNGCucumberTests {
```

```
}
```

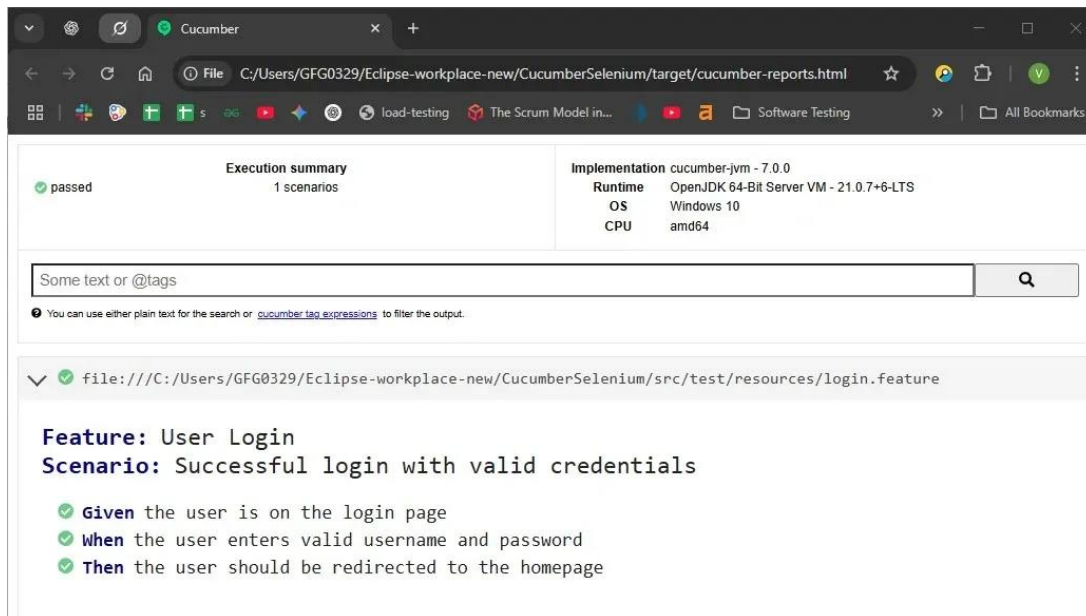
Paso 5. Después de configurar los archivos de características, las definiciones de pasos y el ejecutor de pruebas, puede ejecutar las pruebas con TestNG. Haga clic derecho en la clase TestRunner y seleccione Ejecutar como → Prueba de TestNG.

Producción:



Salida de Cucumber Framework

Paso 6. Verifique el directorio de la carpeta en target/cucumber-report, cambie la extensión del archivo a .html para visualización web y luego abra el archivo en cualquier navegador.



Salida de informes de Cucumber

- **Resumen de la función :** una descripción general de la función y sus escenarios.
- **Resultados del escenario :** una lista de todos los escenarios con su estado de aprobación o rechazo.

- **Resultados del paso** : información detallada sobre cada paso, incluso si se aprobó o no.
- **Tiempo de ejecución** : cuánto tiempo tardó en ejecutarse cada escenario de prueba.