Name : Akshay Pardeshi

Mob: +91 8329528875

Loacation: Mumbai

Project Overview:

Our project is a Pathology Lab Management web application that helps manage various aspects of a pathology lab. The application features the following key components:

1. Login Page:

URL: https://gor-pathology.web.app/

Credentials:

- Username: test@kennect.io

- Password: Qwerty@1234

**package** pathologylab;

**import** org.openqa.selenium.By;

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

**import** org.openqa.selenium.chrome.ChromeOptions;

**public** **class** FirstTest {

**public** **static** **void** main(String[] args) **throws** InterruptedException {

System.*setProperty*("webdriver.chrome.driver", "C://Users//HP//Desktop//Selenium//New folder//chromedriver.exe");

// ChromeDriver driver = new ChromeDriver();

ChromeOptions options = **new** ChromeOptions();

options.addArguments("--remote-allow-origins=\*");

ChromeDriver driver = **new** ChromeDriver(options);

driver.get("https://gor-pathology.web.app/");

Thread.*sleep*(2000);

driver.findElement(By.*xpath*("//input[@name='email']")).sendKeys("test@kennect.io");

driver.findElement(By.*xpath*("//input[@name='password']")).sendKeys("Qwerty@1234");

driver.findElement(By.*name*("login")).click();

String exptitle = "Facebook";

String acttitle = driver.getTitle();

**if**(exptitle.equals(acttitle)){

System.***out***.println("test is passed");

}

**else**{

System.***out***.println("test is failed");

}

driver.close();

}

}

OutPut

Test passed

Test failed

2. Home Page:

After successful login, the user will land on the home page, where they can view a list of todos and access the Cost Calculator for blood tests.

This script adds the following steps:

Navigates to the home page after a successful login.

Accesses the list of todos on the home page.

Accesses the Cost Calculator for blood tests on the home page.

**package** pathologylab;

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.WebElement;

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

**import** org.openqa.selenium.chrome.ChromeOptions;

**public** **class** PathologyLabHome {

**public** **static** **void** main(String[] args) **throws** InterruptedException {

System.*setProperty*("webdriver.chrome.driver", "C://Users//HP//Desktop//Selenium//New folder//chromedriver.exe");

// ChromeDriver driver = new ChromeDriver();

ChromeOptions options = **new** ChromeOptions();

options.addArguments("--remote-allow-origins=\*");

ChromeDriver driver = **new** ChromeDriver(options);

//driver.get("https://gor-pathology.web.app/");

**try** {

// Navigate to the login page

driver.get("https://gor-pathology.web.app/");

// Find the username, password input fields, and login button

WebElement usernameInput = driver.findElement(By.*name*("//input[@name='email']"));

WebElement passwordInput = driver.findElement(By.*name*("//input[@name='password']"));

WebElement loginButton = driver.findElement(By.*name*("login"));

// Enter the credentials

usernameInput.sendKeys("test@kennect.io");

passwordInput.sendKeys("Qwerty@1234");

// Click the login button

loginButton.click();

Thread.*sleep*(3000);

// After successful login, navigate to the home page

WebElement homeLink = driver.findElement(By.*linkText*("Home"));

homeLink.click();

Thread.*sleep*(3000);

WebElement todosSection = driver.findElement(By.*id*("todos-section"));

System.***out***.println("List of Todos: " + todosSection.getText());

WebElement costCalculatorLink = driver.findElement(By.*linkText*("Cost Calculator"));

costCalculatorLink.click();

Thread.*sleep*(3000);

// Print the page title to the console

System.***out***.println("Page Title: " + driver.getTitle());

} **finally** {

// Close the browser

driver.quit();

}

}

}

3. Cost Calculator for Blood Test:

On the home page, there is a cost calculator feature that calculates the cost of various blood tests. The user can select the desired tests and apply discounts if applicable.

Navigates to the home page after a successful login.

Accesses the Cost Calculator for blood tests on the home page.

Selects desired blood tests.

Applies discounts if applicable.

Clicks the calculate button.

Prints the calculated cost or any other relevant information.

**package** pathologylab;

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.WebElement;

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

**import** java.util.List;

**public** **class** PathologyLabCostCalculator {

**public** **static** **void** main(String[] args) **throws** InterruptedException {

// Set the path to your ChromeDriver executable

System.*setProperty*("webdriver.chrome.driver", "C://Users//HP//Desktop//Selenium//New folder//chromedriver.exe");

WebDriver driver = **new** ChromeDriver();

**try** {

// Navigate to the login page

driver.get("https://gor-pathology.web.app/");

WebElement usernameInput = driver.findElement(By.*xpath*("//input[@name='email']"));

WebElement passwordInput = driver.findElement(By.*xpath*("//input[@name='password']"));

WebElement loginButton = driver.findElement(By.*name*("login"));

usernameInput.sendKeys("test@kennect.io");

passwordInput.sendKeys("Qwerty@1234");

loginButton.click();

Thread.*sleep*(3000);

WebElement homeLink = driver.findElement(By.*linkText*("Home"));

homeLink.click();

Thread.*sleep*(3000);

WebElement costCalculatorLink = driver.findElement(By.*linkText*("Cost Calculator"));

costCalculatorLink.click();

Thread.*sleep*(3000);

List<WebElement> bloodTests = driver.findElements(By.*cssSelector*(".blood-test-checkbox"));

**for** (WebElement test : bloodTests) {

test.click();

}

WebElement discountInput = driver.findElement(By.*id*("discountInput"));

discountInput.sendKeys("10");

WebElement calculateButton = driver.findElement(By.*id*("calculateButton"));

calculateButton.click();

Thread.*sleep*(3000);

WebElement resultElement = driver.findElement(By.*id*("result"));

System.***out***.println("Calculated Cost: " + resultElement.getText());

} **finally** {

// Close the browser

driver.quit();

}

}

}

-----------------------------------------------------------------

4. Adding Patients and Creating Tests:

The application allows the user to add patient details and fill up a form to create test requests for patients. After adding a test, it will be reflected in the list of todos on the home page.

Navigates to the "Add Patient" page.

Fills in the patient details form (patient name, test type, etc.).

Clicks the "Add Test" button.

Navigates back to the home page.

Prints the list of todos on the home page.

**package** pathologylab;

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.WebElement;

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

**public** **class** PathologyLabAddPatient {

**public** **static** **void** main(String[] args) **throws** InterruptedException {

// Set the path to your ChromeDriver executable

System.*setProperty*("webdriver.chrome.driver", "C://Users//HP//Desktop//Selenium//New folder//chromedriver.exe");

// Create a new instance of the ChromeDriver

WebDriver driver = **new** ChromeDriver();

**try** {

// Navigate to the login page

driver.get("https://gor-pathology.web.app/");

// Find the username, password input fields, and login button

WebElement usernameInput = driver.findElement(By.*name*("//input[@name='email']"));

WebElement passwordInput = driver.findElement(By.*name*("//input[@name='password']"));

WebElement loginButton = driver.findElement(By.*name*("login"));

// Enter the credentials

usernameInput.sendKeys("test@kennect.io");

passwordInput.sendKeys("Qwerty@1234");

// Click the login button

loginButton.click();

Thread.*sleep*(3000);

// After successful login, navigate to the "Add Patient" page

WebElement addPatientLink = driver.findElement(By.*linkText*("Add Patient"));

addPatientLink.click();

// Wait for a moment to observe the result (you might want to use explicit waits in real-world scenarios)

Thread.*sleep*(3000);

WebElement patientNameInput = driver.findElement(By.*name*("patientName"));

WebElement testTypeInput = driver.findElement(By.*name*("testType"));

WebElement addTestButton = driver.findElement(By.*name*("addTestButton"));

patientNameInput.sendKeys("Akshay Pardeshi");

testTypeInput.sendKeys("Blood Test");

addTestButton.click();

Thread.*sleep*(3000);

driver.navigate().back();

Thread.*sleep*(3000);

WebElement todosSection = driver.findElement(By.*id*("todos-section"));

System.***out***.println("List of Todos: " + todosSection.getText());

} **finally** {

// Close the browser

driver.quit();

}

}

}

Test Scenarios:

1. Login Page:

Scenario 1: Successful Login

Test Case 1: Enter valid username and password, click login, and verify that the user is redirected to the home page.

Scenario 2: Invalid Credentials

Test Case 2: Enter invalid username and valid password, click login, and verify the error message.

Test Case 3: Enter valid username and invalid password, click login, and verify the error message.

Test Case 4: Leave both username and password fields empty, click login, and verify the error message.

2. Home Page:

Scenario 1: Display of Todos

Test Case 5: After successful login, check if the list of todos is displayed on the home page.

Scenario 2: Navigation to Cost Calculator

Test Case 6: Click on the Cost Calculator link and verify if the user is redirected to the Cost Calculator page.

3. Cost Calculator for Blood Test:

Scenario 1: Calculation of Blood Test Cost

Test Case 7: Select various blood tests, apply discounts, and verify if the calculated cost is accurate.

Scenario 2: Invalid Input Handling

Test Case 8: Attempt to calculate cost without selecting any test and verify the appropriate error message.

Test Case 9: Enter a negative discount and verify the appropriate error message.

4. Adding Patients and Creating Tests:

Scenario 1: Patient and Test Creation

Test Case 10: Navigate to the "Add Patient" page, fill in patient details, create a test request, and verify it reflects in the list of todos on the home page.

Scenario 2: Form Validation

Test Case 11: Submit the form with mandatory fields empty and verify the appropriate error messages.

Test Case 12: Verify that the created test request includes accurate patient information.

---------------------------------------------------------------

Test Execution Documentation

Prerequisites:

Browser: Ensure you have Google Chrome installed on your machine.

WebDriver: Download the appropriate version of ChromeDriver and set the executable path in the test script.

est Output:

After executing the tests, the output will be displayed in the terminal/console.

Examine the output for any failures or errors. The test framework or tool should provide detailed information.

Troubleshooting:

If there are issues with WebDriver or other dependencies, double-check the setup steps and paths.

Ensure the WebDriver version matches your Chrome browser version.

Custom Configuration:

If your test script allows for custom configurations, document how to modify configurations for different environments or scenarios.

The Above All Test Scrits are Without Using any framework just a simple java project