

PopUpDialogTest

The screenshot shows a web browser window displaying the Adactin Hotel booking page. The page includes a search bar, a list of hotel details (Hotel Name, Location, Room Type, Number of Rooms, Total Days, Price per Night, Total Price, GST, Final Billed Price), and a booking form with fields for First Name, Last Name, Billing Address, Credit Card No., Credit Card Type, Expiry Date, and CVV Number. A "Book Now" button is visible.

The IDE window shows the test results for the `PopUpDialogTest` class. The test results indicate that the test passed successfully. The test runner shows the following details:

- Test Name: `tests.PopUpDialogTest`
- Test Method: `testMyFirstWebDriver`
- Test Parameters: `(0) [1.Sydney,Hotel Creek,Standard,2,10/04/2019,11/04/2019,2,1,125,135 (testMyFirstWebDriver)]`
- Test Results: 1/1 Run, 0 Errors, 0 Failures

The screenshot shows the source code for the `PopUpDialogTest` class in the IDE. The code is written in Java and uses Selenium WebDriver to interact with the Adactin Hotel website. The test method `testMyFirstWebDriver` is annotated with `@Test` and `@Parameters`. It uses `WebDriver` to navigate to the website, perform a login, and book a hotel. The test results show that the test passed successfully.

```

29
30
31 @Before
32 public void setUp() throws Exception {
33     System.setProperty("webdriver.chrome.driver", "C:/ChromeDriver/chromedriver.exe");
34     driver = new ChromeDriver();
35
36     prop = new Properties();
37     prop.load(new FileInputStream("./Login/Login.properties"));
38     username = prop.getProperty("username");
39     password = prop.getProperty("password");
40
41     prop.load(new FileInputStream("./Configuration/HA_Configuration.properties"));
42     sAppURL = prop.getProperty("AppURL");
43     sSharedUIPath = prop.getProperty("SharedUIPath");
44
45     prop.load(new FileInputStream(sSharedUIPath));
46     driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);
47
48 }
49
50 @Test
51 @Parameters("src/Excel/Search Hotel1.csv")
52 public void testMyFirstWebDriver(int testCaseNo, String location, String hotelName, String roomType,
53     String checkOut, String noAdults, String noChild, double ExpPriceNight, double totalPrice)
54     throws Exception {
55     HA_BF_Login(driver, username, password);
56     searchHotel(driver, location, hotelName, roomType, numberRooms, checkIn, checkOut, noAdults, no
57     driver.findElement(By.id(prop.getProperty("Rad_SelectHotel_RadioButton_0"))).click();
58     driver.findElement(By.id(prop.getProperty("Btn_SelectHotel_RadioButton_0"))).click();
59     bookHotel(driver, "Dsouza", "Nigel", "1225 S Pecan St\nThe Heights on Pecan, Arlington", "1001759
60     Thread.sleep(5000);
61     Alert alertPopUp = driver.switchTo().alert();
62     assertEquals("Expiry year cannot be in Past", alertPopUp.getText());
63     System.out.println(alertPopUp.getText());
64 }

```

The test results show that the test passed successfully. The test runner shows the following details:

- Test Name: `tests.PopUpDialogTest`
- Test Method: `testMyFirstWebDriver`
- Test Parameters: `(0) [1.Sydney,Hotel Creek,Standard,2,10/04/2019,11/04/2019,2,1,125,135 (testMyFirstWebDriver)]`
- Test Results: 1/1 Run, 0 Errors, 0 Failures

DynamicUIObjectTest

The screenshot displays a web browser window on the left and an Eclipse IDE window on the right. The browser window shows the Adactin hotel booking website. The page title is "Adactin Driving Innovation". The URL is "adactinhotelapp.com/BookedItinerary.php". The page content includes a "Booked Itinerary" section with a table of booked hotels. The table has columns: Order Id, Cancel, Hotel Name, Location, Rooms, First Name, Last Name, Arrival Date, Departure Date, No. of Days, Rooms Type, Price per Night, and Total Price (Including GST). The table contains one row with the following data: Order Id: 86HX6, Cancel: Cancel 86HX67X969, Hotel Name: Hotel C, Location: Sydney, Rooms: 2 Rooms, First Name: Nigel, Last Name: Dsouza, Arrival Date: 10/04/20, Departure Date: 11/04/2019, No. of Days: 1 Day, Rooms Type: Standard, Price per Night: AUD \$ 100, and Total Price (Including GST): AUD \$ 100. The browser window also shows a "Welcome to Adactin Group of Hotels" message and a "Hello nigel101221 Search Hotel | Booked Itinerary | Change Password | Logout" message. The Eclipse IDE window on the right shows the test results for "testDynamicUIObjectTestPartialMatch". The test results show that the test passed successfully. The test is named "testDynamicUIObjectTestPartialMatch" and is located in the "TestMyFirstWebDriver" class. The test results show that the test passed successfully. The test is named "testDynamicUIObjectTestPartialMatch" and is located in the "TestMyFirstWebDriver" class. The test results show that the test passed successfully. The test is named "testDynamicUIObjectTestPartialMatch" and is located in the "TestMyFirstWebDriver" class.

The screenshot displays the Eclipse IDE window showing the source code for the "testDynamicUIObjectTestPartialMatch" test. The code is written in Java and uses Selenium WebDriver to interact with the Adactin website. The code includes the following elements:

- Imports: `import org.openqa.selenium.*; import org.openqa.selenium.chrome.ChromeDriver; import org.openqa.selenium.support.ui.ExpectedConditions; import org.openqa.selenium.support.ui.WebDriverWait; import org.testng.annotations.Test; import org.testng.annotations.BeforeMethod; import org.testng.annotations.AfterMethod; import org.testng.annotations.BeforeClass; import org.testng.annotations.AfterClass; import org.testng.annotations.BeforeTest; import org.testng.annotations.AfterTest; import org.testng.annotations.BeforeSuite; import org.testng.annotations.AfterSuite; import org.testng.annotations.BeforeClass; import org.testng.annotations.AfterClass; import org.testng.annotations.BeforeTest; import org.testng.annotations.AfterTest; import org.testng.annotations.BeforeSuite; import org.testng.annotations.AfterSuite;`
- Test Method: `@Test public void testDynamicUIObjectTestPartialMatch() throws InterruptedException { // ... };`
- Test Results: The test results show that the test passed successfully. The test is named "testDynamicUIObjectTestPartialMatch" and is located in the "TestMyFirstWebDriver" class. The test results show that the test passed successfully. The test is named "testDynamicUIObjectTestPartialMatch" and is located in the "TestMyFirstWebDriver" class. The test results show that the test passed successfully. The test is named "testDynamicUIObjectTestPartialMatch" and is located in the "TestMyFirstWebDriver" class.

Exercise 11- DynamicUIObjectTestPartialMatch

