

# Star Agile Capstone project

**Project name :** Project – Human Resource Management(Automation)

**Project Done By :** Mallika (mallikakpsgvc@gmail.com)

**Date:** 6<sup>th</sup> November 2024

## Modules In Scope for Automation Testing

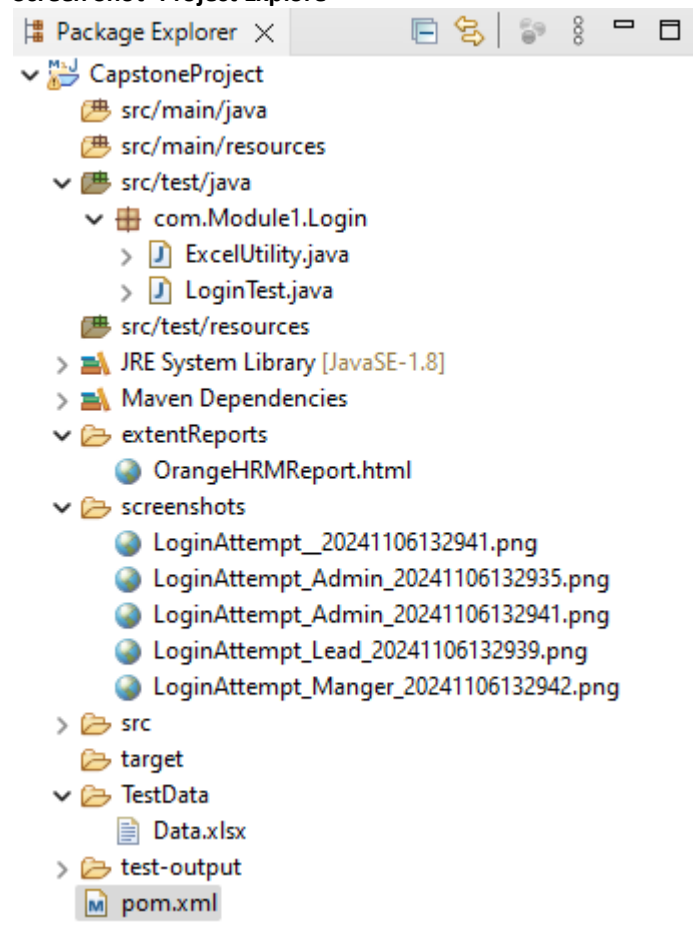
1. Login
2. Admin

### Module 1.Login

**Scenario1:** Automate Orange HRM login and logout with 5 different data set including valid and invalid data set.

1. Save data set in Excel file.
2. Write a script to read data from excel.
3. Prepare script for Login and logout.
4. Perform assertion for valid data set (use Username: Admin and Password:admin123) test case should be pass and invalid data set (other than given data) test case should be fail.
5. Capture Screenshot for every login functionality.
6. Generate Extent Report for the same.

### Screen Shot -Project Explore



Added necessary dependencies(Selenium,Maven,TestNG, Apache poi and Extent Reports).

```

1 CapstoneProject/pom.xml X
2 <modelVersion>4.0.0</modelVersion>
3 <groupId>CapstoneProject</groupId>
4 <artifactId>CapstoneProject</artifactId>
5 <version>0.0.1-SNAPSHOT</version>
6 <dependencies>
7 <!--
8 https://mvnrepository.com/artifact/org.seleniumhq.selenium/selenium-java -->
9 <dependency>
10 <groupId>org.seleniumhq.selenium</groupId>
11 <artifactId>selenium-java</artifactId>
12 <version>4.25.0</version>
13 </dependency>
14 <!-- https://mvnrepository.com/artifact/org.testng/testng -->
15 <dependency>
16 <groupId>org.testng</groupId>
17 <artifactId>testng</artifactId>
18 <version>7.10.2</version>
19 <scope>test</scope>
20 </dependency>
21 <!-- https://mvnrepository.com/artifact/org.apache.poi/poi -->
22 <dependency>
23 <groupId>org.apache.poi</groupId>
24 <artifactId>poi</artifactId>
25 <version>5.3.0</version>
26 </dependency>
27 <!-- https://mvnrepository.com/artifact/org.apache.poi/poi-ooxml -->
28 <dependency>
29 <groupId>org.apache.poi</groupId>
30 <artifactId>poi-ooxml</artifactId>
31 <version>5.3.0</version>
32 </dependency>
33 <!-- https://mvnrepository.com/artifact/com.aventstack/extentreports -->
34 <dependency>
35 <groupId>com.aventstack</groupId>
36 <artifactId>extentreports</artifactId>
37 <version>5.1.2</version>
38 </dependency>

```

[illegible]

## Screen Shot- Excel Utility java class

```
ExcelUtility.java X
2 import java.io.File;
3 import java.io.FileInputStream;
4 import java.io.IOException;
5 import org.apache.poi.xssf.usermodel.XSSFWorkbook;
6 import org.testng.annotations.DataProvider;
7 public class ExcelUtility {
8     private XSSFWorkbook wb;
9     private FileInputStream fs;
10    private File f1;
11    @DataProvider(name="excelLoginData")
12    public Object[][] excelData()
13    {
14        f1=new File(System.getProperty("user.dir")+"//TestData//Data.xlsx");
15        Object arr[][]=null;
16        try {
17            fs = new FileInputStream(f1);
18            //Workbook
19            wb=new XSSFWorkbook(fs);
20            String uName1=wb.getSheet("userData").getRow(1).getCell(0).getStringCellValue();
21            System.out.println("User name: "+uName1);
22            int rows=wb.getSheet("userData").getPhysicalNumberOfRows();
23            System.out.println("No. of Rows: "+rows);
24            int cells=wb.getSheet("userData").getRow(0).getPhysicalNumberOfCells();
25            System.out.println("No. of Columns: "+cells);
26            arr=new Object[rows-1][cells];
27            for(int i=1;i<rows;i++)//to skip heading i starts with 1
28            {
29                for(int j=0;j<cells;j++)
30                {
31                    arr[i-1][j]=wb.getSheet("userData").getRow(i).getCell(j).getStringCellValue();
32                    System.out.print(arr[i-1][j]+" ");
33                }System.out.println();
34            }
35        } catch (IOException e) {
36            e.printStackTrace();
37        }
38        return arr;
39    }
40    public void closeWorkbook() {
41        if (wb != null) {
42            try {
43                wb.close();
44            } catch (IOException e) {
45                e.printStackTrace();
46            }
47        } else {
48            System.out.println("Workbook is already null or was never initialized.");
49        }
50    }
51 }
```

## Screen shot- LoginTest TestNg File

```
3 import java.io.File;
24
25 Run All
26 public class LoginTest {
27     WebDriver driver;
28     ExcelUtility excelData;
29     ExtentReports extent;
30     ExtentTest test;
31
32    @BeforeClass
33    public void setUp() {
34        excelData= new ExcelUtility();
35        driver = new ChromeDriver();
36        driver.get("https://opensource-demo.orangehrmlive.com/web/index.php/auth/login");
37        driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10));
38        driver.manage().window().maximize();
39        ExtentSparkReporter sparkReporter = new ExtentSparkReporter("extentReports/OrangeHRMReport.html");
40        extent = new ExtentReports();
41        extent.attachReporter(sparkReporter);
42    }
43 }
```

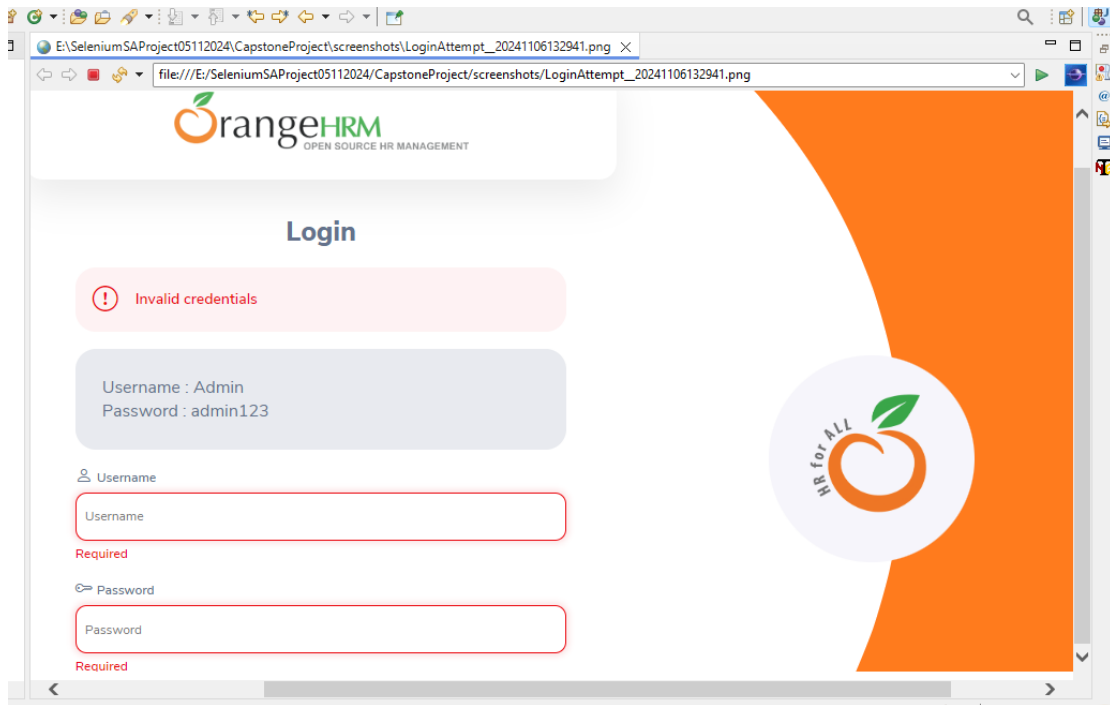
```

44 @Test(dataProvider = "excelLoginData", dataProviderClass = ExcelUtility.class)
45 Run | Debug
46 public void loginTest(String username, String password) {
47     test = extent.createTest("Login Test with username: " + username);
48     driver.findElement(By.name("username")).sendKeys(username);
49     driver.findElement(By.name("password")).sendKeys(password);
50     driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(20));
51     driver.findElement(By.xpath("//button[normalize-space(text()='Login')]")).click();
52     driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(30));
53
54     // Assertion and validation
55     if (username.equals("Admin") && password.equals("admin123")) {
56         Assert.assertEquals(driver.getCurrentUrl(), "https://opensource-demo.orangehrmlive.com/web/index.php/dashboard/index");
57         WebDriverWait wait = new WebDriverWait(driver, Duration.ofSeconds(20));
58         wait.until(ExpectedConditions.visibilityOfElementLocated(By.xpath("//h6[text()='Dashboard']")));
59         takeScreenshot("LoginAttempt_" + username);
60         test.pass("Valid credentials - Login successful.");
61         logout();
62     }
63     else {
64         WebElement errorMessage =
65             driver.findElement(By.xpath("//p[text()='Invalid credentials']"));
66         Assert.assertTrue(errorMessage.isDisplayed(), "Invalid credentials");
67         driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(30));
68         takeScreenshot("LoginAttempt_" + username);
69         test.fail("Invalid credentials - Login failed as expected.");
70     }
71 }
72
73 public void logout() {
74     driver.findElement(By.xpath("//span[@class='oxd-userdropdown-tab']")).click();
75     driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10));
76     driver.findElement(By.xpath("//li/a[text()='Logout']")).click();
77     driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10));
78 }
79
80 public void takeScreenshot(String testName) {
81     File screenshot = ((TakesScreenshot) driver).getScreenshotAs(OutputType.FILE);
82     String timestamp = new SimpleDateFormat("yyyyMMddHHmmss").format(new Date());
83     String filePath = "Screenshots/" + testName + "_" + timestamp + ".png";
84     try {
85         FileUtils.copyFile(screenshot, new File(filePath));
86         test.addScreenCaptureFromPath(filePath);
87     } catch (IOException e) {
88         e.printStackTrace();
89     }
90 }
91
92 @AfterClass
93 public void tearDown() {
94     excelData.closeWorkbook();
95     if (driver != null) {
96         driver.quit();
97     }
98     extent.flush();
99 }
100 }

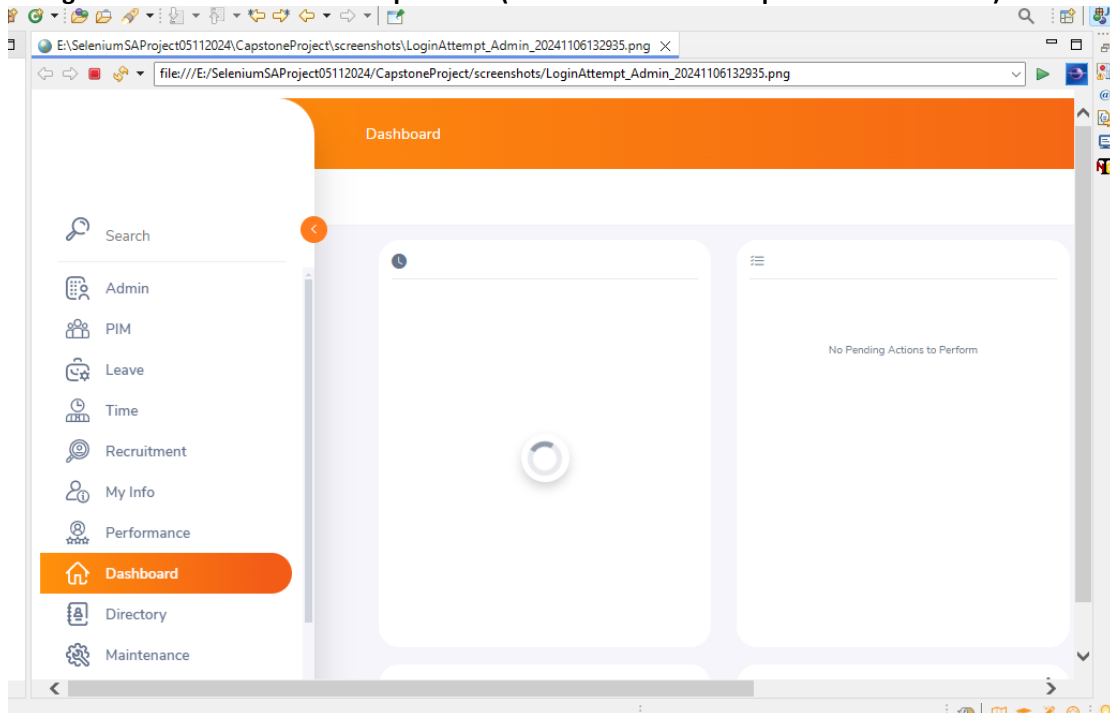
```

## Screen Shot- Captured Screenshot for every login functionality

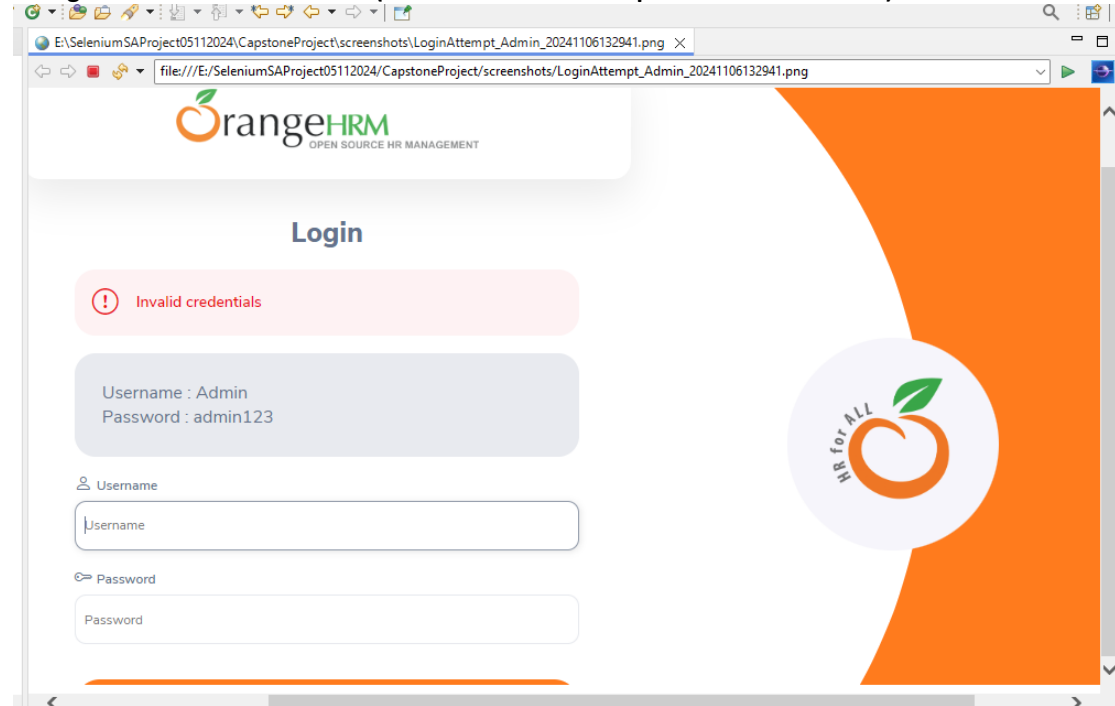
### 1. Login with blank data



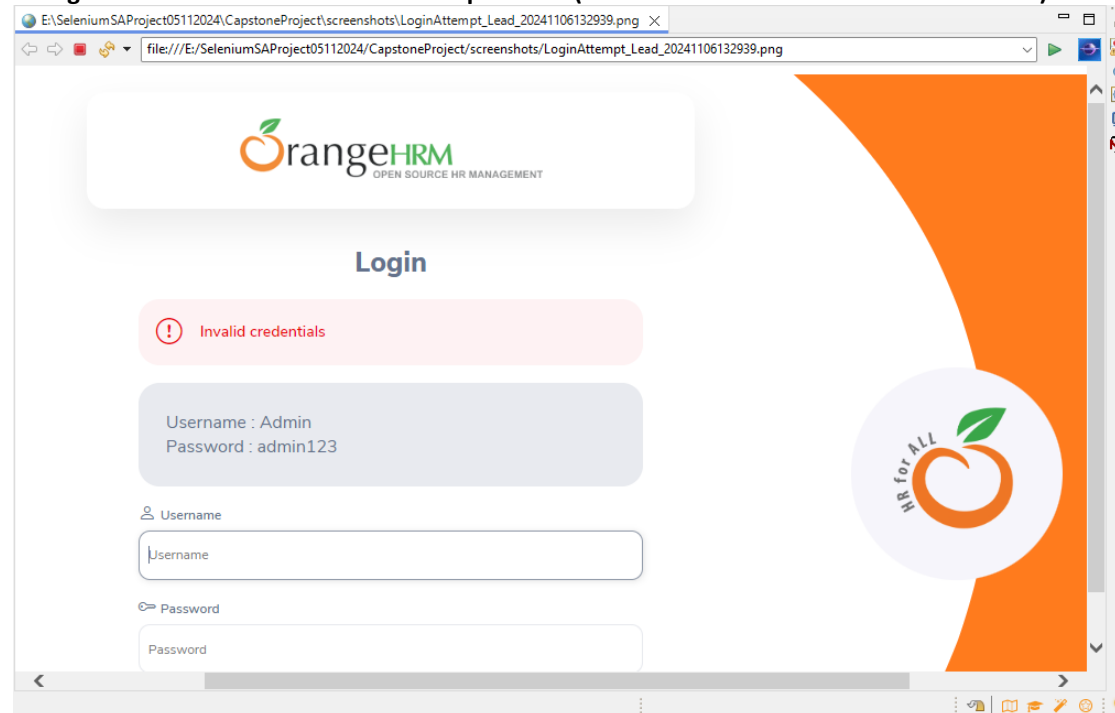
## 2. Login with Valid User name and password(Username: Admin and password: admin123)



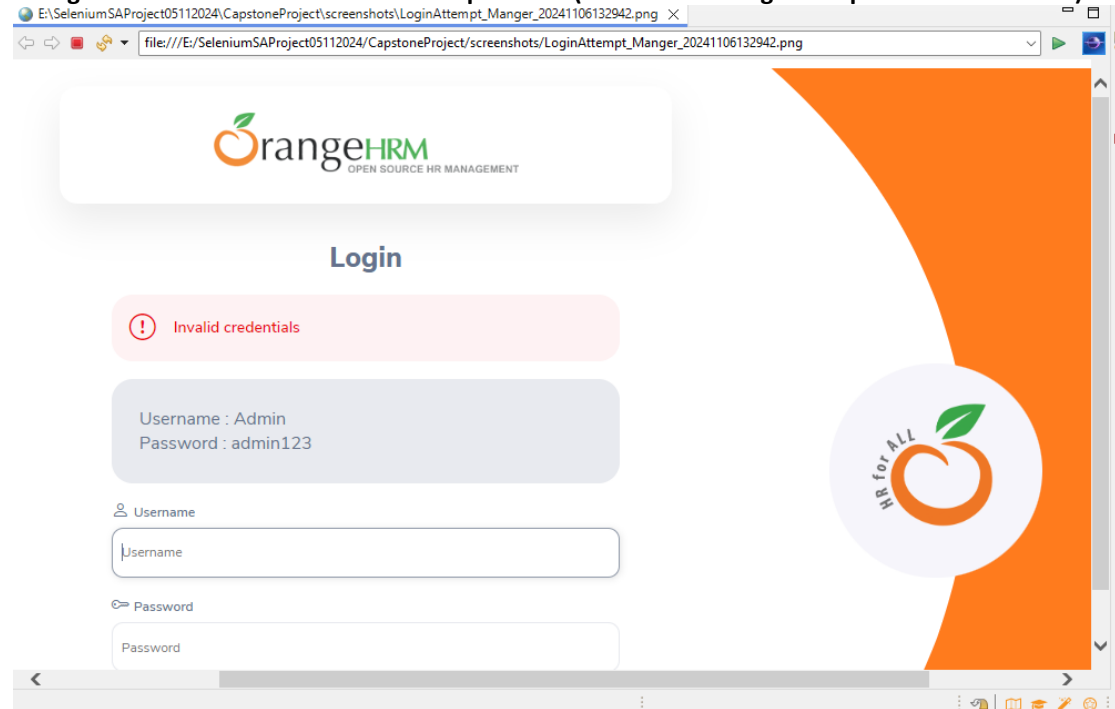
### 3. Login with Invalid Credentials (Username: Admin and password:member123)



### 4. Login with Invalid username and valid password (Username: Lead and Password: admin123)



## 5. Login with Invalid username and valid password (Username: Manger and password: admin123)



## Screen shot-Console Output:

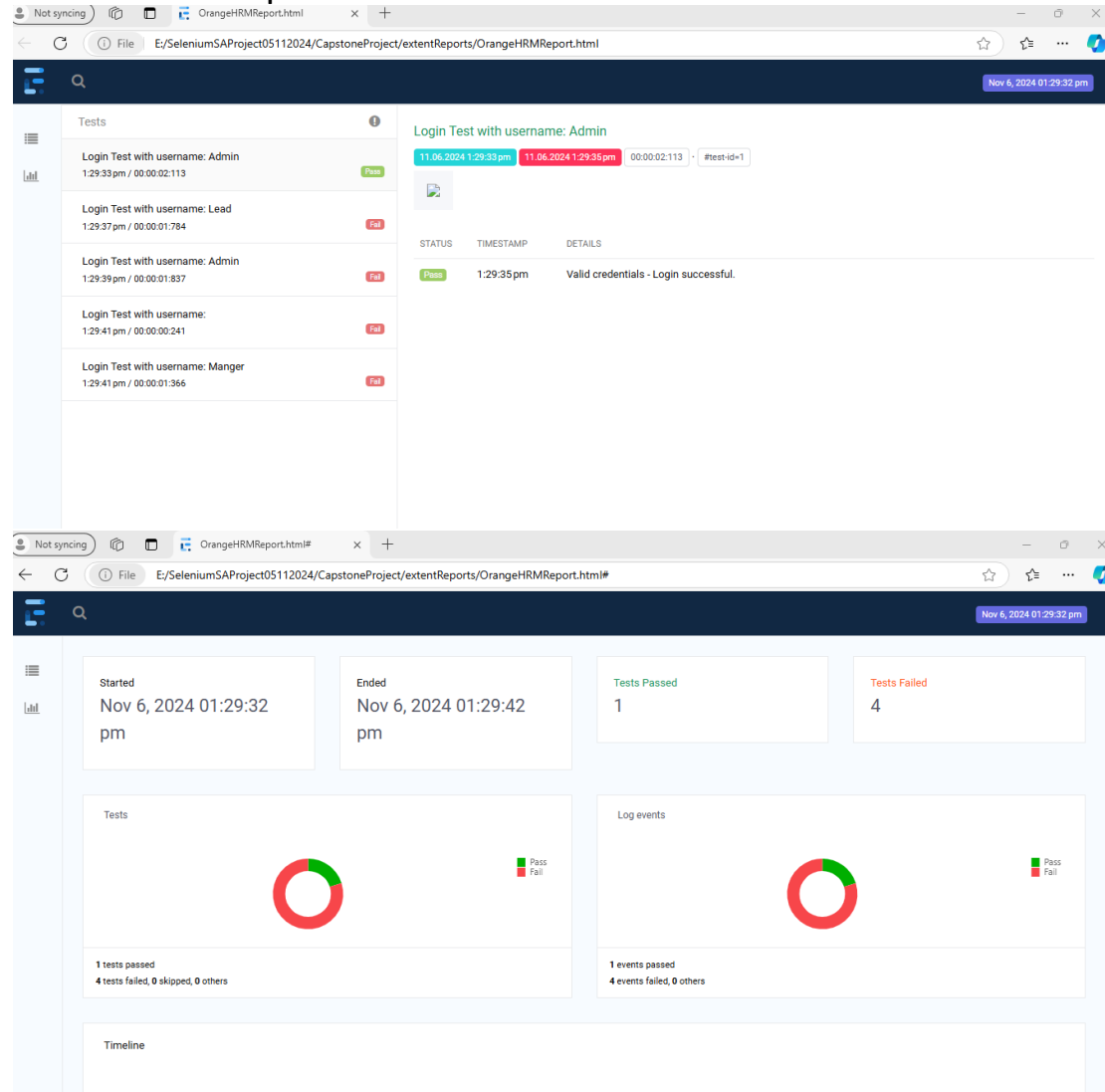
```
Problems Javadoc Declaration Console X Results of running class LoginTest
<terminated> LoginTest [TestNG] C:\Program Files\Java\jdk-21\bin\javaw.exe (06-Nov-2024, 1:29:24 pm - 1:29:43 pm) [pid: 33184]
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
[RemoteTestNG] detected TestNG version 7.10.2
Nov 06, 2024 1:29:28 PM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
WARNING: Unable to find an exact match for CDP version 130, returning the closest version; found: 129; Please update to a Selenium version that supports CDP version 130
2024-11-06T07:59:32.346043200Z main ERROR Log4j2 could not find a logging implementation. Please add log4j-core to the classpath. Using SimpleLogger to log to the console...
User name: Admin
No. of Rows: 6
No. of Columns: 2
Admin admin123
Lead lead123
Admin member123

Manger admin123
Workbook is already null or was never initialized.
PASSED: com.Module1.Login.LoginTest.loginTest("Lead", "lead123")
PASSED: com.Module1.Login.LoginTest.loginTest("Admin", "admin123")
PASSED: com.Module1.Login.LoginTest.loginTest("", "")
PASSED: com.Module1.Login.LoginTest.loginTest("Manger", "admin123")
PASSED: com.Module1.Login.LoginTest.loginTest("Admin", "member123")

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

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

## Screen shot - Extent Reports



\*\*\*\*\*

## Module-2

\*\*\*\*\*

### Module2: Admin

**Scenario1:** Create a Page Object Model for two pages LoginPage and Admin Page and write automation script for Login functionality and Admin search feature Create a Page and Class for LoginPage and automate the functionality of OrangeHRM login for Valid Test Data: (username: Admin and Password: admin123).

**Test cases:**Create a Page and class for Admin where you can prepare 4 important

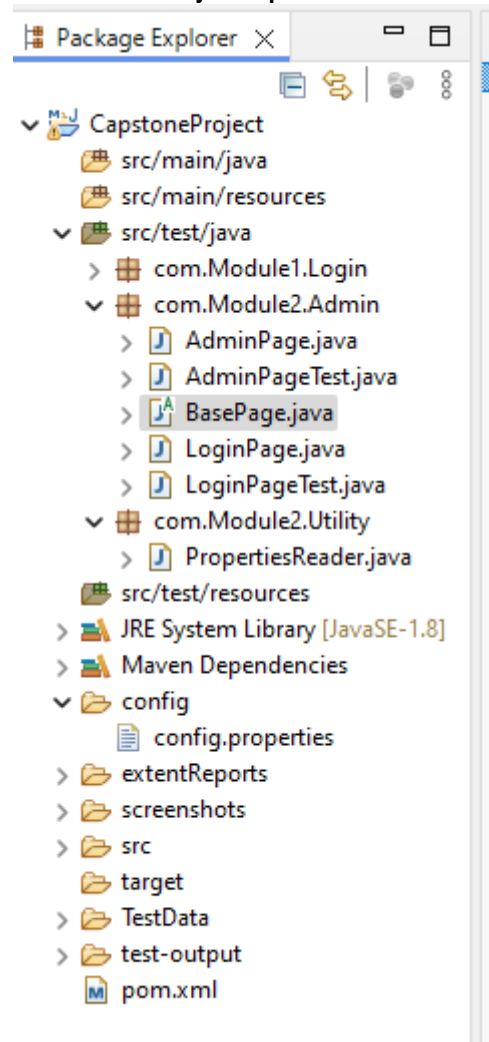
1. Create test case to get all 12 options from left side menu and print the count which should be 12 from that list click on Admin then Admin page get open.



2. Create test case for search For Existing Employee searchByUsername() : here send username Admin to username text box and click on search button and display total record found and refresh page.
3. Create test case for search For Existing Employee searchByUserRole() : here automate dropdown and select Role Admin and click on search button and display total record found and refresh page.
4. Create test case for search For Existing Employee searchByUserStatus() : here automate dropdown and select status Enabled or Disabled then click on search button and display total record found.

**Note:** In the Second scenario use Selenium +TestNG and Design Page Object Model where you can implement Object Oriented Principle.

#### Screen Shot -Project Explore



Screen Shot -Pom.xml file:

```

1 <?xml version="1.0" encoding="UTF-8" ?>
2 <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
3 <modelVersion>4.0.0</modelVersion>
4 <groupId>CapstoneProject</groupId>
5 <artifactId>CapstoneProject</artifactId>
6 <version>0.0.1-SNAPSHOT</version>
7 <dependencies>
8 <!-- https://mvnrepository.com/artifact/org.seleniumhq.selenium/selenium-java -->
9 <dependency>
10 <groupId>org.seleniumhq.selenium</groupId>
11 <artifactId>selenium-java</artifactId>
12 <version>4.25.0</version>
13 </dependency>
14 <!-- https://mvnrepository.com/artifact/org.testng/testng -->
15 <dependency>
16 <groupId>org.testng</groupId>
17 <artifactId>testng</artifactId>
18 <version>7.10.2</version>
19 <scope>test</scope>
20 </dependency>
21 <!-- https://mvnrepository.com/artifact/org.apache.poi/poi -->
22 <dependency>
23 <groupId>org.apache.poi</groupId>
24 <artifactId>poi</artifactId>
25 <version>5.3.0</version>
26 </dependency>
27 <!-- https://mvnrepository.com/artifact/org.apache.poi/poi-ooxml -->
28 <dependency>
29 <groupId>org.apache.poi</groupId>
30 <artifactId>poi-ooxml</artifactId>
31 <version>5.3.0</version>
32 </dependency>
33 <!-- https://mvnrepository.com/artifact/com.aventstack/extentreports -->
34 <dependency>
35 <groupId>com.aventstack</groupId>
36 <artifactId>extentreports</artifactId>
37 <version>5.1.2</version>
38 </dependency>
39 </dependencies>
40 </project>

```

### Screen Shot -config properties file

```

1 url=https://opensource-demo.orangehrmlive.com/web/index.php/auth/login
2 username=Admin
3 password=admin123

```

### Screen Shot -Properties reader class:

```

1 package com.Module2.Utility;
2
3 import java.io.File;
4 import java.io.FileInputStream;
5 import java.io.FileNotFoundException;
6 import java.io.IOException;
7 import java.util.Properties;
8
9 public class PropertiesReader {
10     Properties prop;
11     public PropertiesReader()
12     {
13         prop=new Properties();
14         File f1=new File(System.getProperty("user.dir")+"//config//config.properties");
15
16         try {
17             FileInputStream fs=new FileInputStream(f1);
18             prop.load(fs);
19         } catch (FileNotFoundException e) {
20             e.printStackTrace();
21         } catch (IOException e) {
22             e.printStackTrace();
23         }
24     }
25
26     public String getData(String key)
27     {
28         return prop.getProperty(key);
29     }
30 }
31

```

### Screen Shot -Base Page class:

```

1 package com.Module2.Admin;
2
3 import org.openqa.selenium.WebDriver;
4 import org.openqa.selenium.WebElement;
5 import org.openqa.selenium.support.PageFactory;
6
7 public abstract class BasePage {
8     protected WebDriver driver;
9
10    public BasePage(WebDriver driver) {
11        this.driver = driver;
12        PageFactory.initElements(driver, this);
13    }
14
15    // Encapsulation principle WebDriver actions
16    public abstract void clickElement(WebElement ele);
17
18    protected void enterText(WebElement ele1, String text) {
19        ele1.sendKeys(text);
20    }
21
22    public void addWait() {
23        try {
24            Thread.sleep(1500);
25        } catch (InterruptedException e) {
26            e.printStackTrace();
27        }
28    }
29 }
30 }

```

Screen Shot -Login Page class for locators and action methods:

```

1 package com.Module2.Admin;
2
3 import org.openqa.selenium.WebDriver;
4 import org.openqa.selenium.WebElement;
5 import org.openqa.selenium.support.FindBy;
6 import com.Module2.Utility.PropertiesReader;
7
8 public class LoginPage extends BasePage {
9     PropertiesReader prop;
10
11    // Login page constructor to initialize
12    public LoginPage(WebDriver driver) {
13        super(driver);
14    }
15
16    /*
17     * Locators of login page
18     */
19    @FindBy(name = "username")
20    WebElement uName;
21
22    @FindBy(name = "password")
23    WebElement pswrd;
24
25    @FindBy(xpath = "//button[text()=' Login ']")
26    WebElement loginBtn;
27 }

```

```

28-  /*
29-   * Action Methods
30-   */
31-
32-  public void login(String username, String password) {
33-      enterText(uName, username);
34-      enterText(pswrd, password);
35-      clickElement(loginBtn);
36-      addWait();
37-  }
38-
39-  @Override
40-  // Abstract method
41-  public void clickElement(WebElement we) {
42-      we.click();
43-  }
44-
45-  public void refreshPage() {
46-      driver.navigate().refresh();
47-  }
48-
49-  public String appUrl() {
50-      String url = driver.getCurrentUrl();
51-      return url;
52-  }
53-
54-  public String appTitle() {
55-      String title = driver.getTitle();
56-      return title;
57-  }
58-
59-  public void addWait() {
60-      try {
61-          Thread.sleep(1500);
62-      } catch (InterruptedException e) {
63-          e.printStackTrace();
64-      }
65-  }

```

### Screen Shot -Login Page Test

```

LoginPageTest.java X
1 package com.Module2.Admin;
2
3- import org.openqa.selenium.WebDriver;
4 import org.openqa.selenium.chrome.ChromeDriver;
5 import org.testng.Assert;
6 import org.testng.annotations.AfterClass;
7 import org.testng.annotations.BeforeTest;
8 import org.testng.annotations.Test;
9 import com.Module2.Utility.PropertiesReader;
10
11 Run ALL
12 public class LoginPageTest
13 {
14     WebDriver driver;
15     LoginPage lp=new LoginPage(driver);
16     PropertiesReader prop;
17     AdminPage ap=new AdminPage(driver);
18-
19     @BeforeTest
20     public void setUp()
21     {
22         prop =new PropertiesReader();
23         driver= new ChromeDriver();
24         driver.get(prop.getData("url"));
25         lp.addWait();
26         driver.manage().window().maximize();
27         lp=new LoginPage(driver);
28     }

```

```

28
29 @Test(priority = 1)
    Run | Debug
30 public void loginTest()
31 {
32     lp.login(prop.getData("username"), prop.getData("password"));
33 }
34
35 @Test(priority = 2)
    Run | Debug
36 public void validateUrl()
37 {
38     String url=lp.appUrl();
39     Assert.assertEquals(url, "https://opensource-demo.orangehrmlive.com/web/index.php/dashboard/index","Url not matched.Test Fail!");
40     System.out.println("Url Matched launch success.Test Passed");
41 }
42
43 @Test(priority = 3)
    Run | Debug
44 public void validateTitle()
45 {
46     System.out.println("Page Title: "+lp.appTitle());
47     Assert.assertEquals(lp.appTitle().contains("OrangeHRM"),true,"Title not matched.Test Failed");
48     System.out.println("Title Matched launch success.Test Passed");
49 }
50
51 @AfterClass
52 public void tearDown()
53 {
54     if(driver!=null)
55         driver.quit();
56     else
57         System.out.println("Driver is null");
58 }
59 }
60

```

## Console Output:

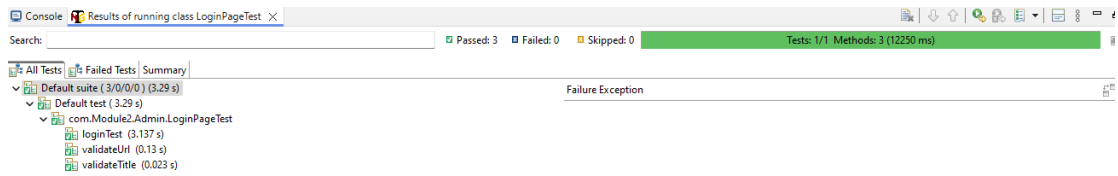
```

Console X Results of running class LoginPageTest
<terminated> LoginPageTest [TestNG] C:\Program Files\Java\jdk-21\bin\javaw.exe (11-Nov-2024, 6:44:48 pm – 6:45:02 pm) [pid: 36760]
[RemoteTestNG] detected TestNG version 7.10.2
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.
Nov 11, 2024 6:44:53 PM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
WARNING: Unable to find an exact match for CDP version 130, returning the closest version; found: 129; Please upc
Url Matched launch success.Test Passed
Page Title: OrangeHRM
Title Matched launch success.Test Passed
PASSED: com.Module2.Admin.LoginPageTest.validateUrl
PASSED: com.Module2.Admin.LoginPageTest.loginTest
PASSED: com.Module2.Admin.LoginPageTest.validateTitle

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

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

```



\*\*\*\*\*

## Admin Page

\*\*\*\*\*

Screen Shot -Admin Page class(Locators and action methods)

AdminPage.java X

```
1 package com.Module2.Admin;
2
3 import java.util.List;
4 import org.openqa.selenium.WebDriver;
5 import org.openqa.selenium.WebElement;
6 import org.openqa.selenium.support.FindBy;
7 import org.openqa.selenium.support.PageFactory;
8
9 public class AdminPage extends BasePage
10 {
11     WebDriver driver;
12     // Initialize page factory
13     public AdminPage(WebDriver driver)
14     {
15         super(driver);
16         this.driver = driver;
17         PageFactory.initElements(driver, this);
18     }
19     /*
20     * Dashboard and Admin Page Locators
21     */
22
23     // Un order left side list
24     @FindBy(xpath = "//li[@class='oxd-main-menu-item-wrapper']")
25     List menuOptions;
26
27     // Admin option WebElement
28     @FindBy(xpath = "//span[text()='Admin']")
29     WebElement adminOption;
30
31     // Admin page heading
32     @FindBy(xpath = "//h6[text()='Admin']")
33     WebElement adminHeader;
34
35     /*
36     * Admin Page Locators
37     */
38
39     // Search User name text box locator
40     @FindBy(xpath = "//input[@class='oxd-input oxd-input--active']")[2]
41     WebElement searchTextBox;
42
43     // Search button locator
44     @FindBy(xpath = "//button[text()=' Search ']")
45     WebElement searchBtn;
46
47     // No. of records in user grid after search
48     @FindBy(xpath = "//div[@class='oxd-table-body']")
49     List totalRecords;
50
51     // record found header after search
52     @FindBy(xpath = "//span[@class='oxd-text oxd-text--span']")[1]
53     WebElement recordFoundHeader;
54
55     // User role drop down
56     @FindBy(xpath = "(//div[@class='oxd-select-text-input'])[1]")
57     WebElement roleDrpDwn;
58
59     // Admin option from drop down
60     @FindBy(xpath = "//div[contains(@role,'listbox')]")
61     WebElement adminDropdownOption;
62
63     // User status drop down
64     @FindBy(xpath = "(//i[@class='oxd-icon bi-caret-down-fill oxd-select-text--arrow'])[2]")
65     WebElement statusDrpDwn;
66
67     // Status drop down option
68     @FindBy(xpath = "//span[text()='Enabled']")
69     WebElement statusDropdownOption;
70
```

```

72  /*
73   * Actions Methods
74   */
75  @Override
76  public void clickElement(WebElement ele) {
77      ele.click();
78  }
79  }
80  public int getOptionsCount()
81  {
82      return menuOptions.size();
83  }
84  }
85  // Get the no of records
86  public int getRecordsCount()
87  {
88      return totalRecords.size();
89  }
90  }
91  // Click on role drop down and select admin role
92  public void searchWithAdminRole() {
93      roledrpdwn.click();
94      addWait();
95      clickElement(adminDropdownOption);
96      addWait();
97      clickElement(searchBtn);
98      addWait();
99  }
100
101  // Click on status drop down and select Enabled status
102  public void searchWithStatus()
103  {
104      clickElement(statusdrpdwn);
105      addWait();
106      clickElement(statusDropdownOption);
107      addWait();
108      clickElement(searchBtn);
109      addWait();
110  }
111  }

```

## Screen Shot -Admin Page Test

```

AdminPageTest.java X
1  package com.Module2.Admin;
2
3  import org.openqa.selenium.WebDriver;
4  import org.openqa.selenium.chrome.ChromeDriver;
5  import org.testng.Assert;
6  import org.testng.annotations.AfterClass;
7  import org.testng.annotations.BeforeClass;
8  import org.testng.annotations.BeforeTest;
9  import org.testng.annotations.Test;
10 import com.Module2.Utility.PropertiesReader;
11
12 Run All
13 public class AdminPageTest {
14     PropertiesReader prop;
15     WebDriver driver;
16     LoginPage lp = new LoginPage(driver);
17     AdminPage adminPage = new AdminPage(driver);
18
19     @BeforeClass
20     public void login() {
21         lp.login(prop.getData("username"), prop.getData("password"));
22     }
23
24     @BeforeTest
25     public void setUp() {
26         prop = new PropertiesReader();
27         driver = new ChromeDriver();
28         driver.get(prop.getData("url"));
29         lp.addWait();
30         driver.manage().window().maximize();
31         lp = new LoginPage(driver);
32         adminPage = new AdminPage(driver);
33     }

```



```

33
34 @Test(priority = 1)
    Run | Debug
35 public void getLeftSideMenuOptions() {
36     System.out.println("Left Side Menu Options");
37     lp.addWait();
38     Assert.assertEquals(adminPage.getOptionsCount(), 12, "Count Not matched. Test Failed!");
39     System.out.println("Left side menu count matched to 12.Test Success");
40     lp.addWait();
41     System.out.println("Click on Admin Option from Left side Menu");
42     adminPage.clickElement(adminPage.adminOption);
43     lp.addWait();
44     System.out.println("Clicked on Admin Option");
45     Assert.assertEquals(adminPage.adminHeader.getText(), "Admin", "Failed to launch Admin page!.Test Failed");
46     System.out.println("Launched to Admin page Success!.Test Success");
47 }
48
49 @Test(priority = 2)
    Run | Debug
50 public void searchByUserName() {
51     adminPage.serachTextBox.sendKeys("Admin");
52     lp.addWait();
53     System.out.println("Search with Employee : Admin");
54     adminPage.clickElement(adminPage.searchBtn);
55     ;
56     lp.addWait();
57     String output = adminPage.recordFoundHeader.getText();
58     System.out.println(output);
59     lp.refreshPage();
60     lp.addWait();
61 }
62
63 @Test(priority = 3)
    Run | Debug
64 public void searchByUserRole() {
65     System.out.println("Search Employee with Role :Admin");
66     adminPage.searchWithAdminRole();
67     System.out.println(adminPage.recordFoundHeader.getText());
68     lp.addWait();
69     lp.refreshPage();
70     lp.addWait();
71 }
72
73 @Test(priority = 4)
    Run | Debug
74 public void searchByUserStatus() {
75     System.out.println("Search Employee with status :Enabled");
76     adminPage.searchWithStatus();
77     System.out.println(adminPage.recordFoundHeader.getText());
78     lp.addWait();
79     lp.refreshPage();
80     lp.addWait();
81 }
82
83 @AfterClass
84 public void tearDown() {
85     driver.quit();
86 }
87 }
RR

```

Screen Shot -Console output

<terminated> AdminPageTest [TestNG] C:\Program Files\Java\jdk-21\bin\javaw.exe (11-Nov-2024, 6:58:19 pm – 6:59:06 pm) [pid: 30216]

[RemoteTestNG] detected TestNG version 7.10.2

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.

Nov 11, 2024 6:58:23 PM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch

WARNING: Unable to find an exact match for CDP version 130, returning the closest version; found: 129; Please update

Left Side Menu Options

Left side menu count matched to 12.Test Success

Click on Admin Option from Left side Menu

Clicked on Admin Option

Launched to Admin page Success!.Test Success

Search with Employee : Admin

(1) Record Found

Search Employee with Role :Admin

(6) Records Found

Search Employee with status :Enabled

(21) Records Found

PASSED: com.Module2.Admin.AdminPageTest.searchByUserRole

PASSED: com.Module2.Admin.AdminPageTest.getLeftSideMenuOptions

PASSED: com.Module2.Admin.AdminPageTest.searchByUserName

PASSED: com.Module2.Admin.AdminPageTest.searchByUserStatus

=====

Default test

Tests run: 4, Failures: 0, Skips: 0

=====

=====

Default suite

Total tests run: 4, Passes: 4, Failures: 0, Skips: 0

=====

Console

Results of running class AdminPageTest

Search:

Passed: 4 Failed: 0 Skipped: 0

Tests: 1/1 Methods: 4 (45287 ms)

All Tests

Failed Tests

Summary

Default suite (4/0/0) (29.438 s)

Default test (29.438 s)

com.Module2.Admin.AdminPageTest

getLeftSideMenuOptions (6.68 s)

searchByUserName (5.502 s)

searchByUserRole (8.729 s)

searchByUserStatus (8.527 s)

Failure Exception

Search:

Passed: 4 Failed: 0 Skipped: 0

Tests

Test name	Time (seconds)	Class count	Method count
Default test	29.438	1	4