

A decorative graphic on the left side of the slide. It features a small, glowing 3D sphere at the top left and a larger, glowing 3D ring or torus shape below it, both with a teal-to-purple gradient and a soft glow effect.

CAPTURING SCREENSHOT USING SELENIUM

HARI HARAN D

Used IDE : Eclipse IDE

1 . Introduction

- Overview of Selenium
- Importance of screenshots in automated testing
- Selenium is an open-source and a portable automated software testing tool for testing web applications. It has capabilities to operate across different browsers and operating systems. Selenium is not just a single tool but a set of tools that helps testers to automate web-based applications more efficiently
- **Importance of screenshots:**
 - Visual validation of test results.
 - Debugging and troubleshooting.
 - Documentation of test cases.

Setting Up Selenium

**This section will show how we can
configure Selenium with Eclipse.**

Pre-requisites:

Download and Install Java

Set up the Environment Variables

Install WebDrivers

Download and Install Eclipse

Selenium code for Capturing Screenshot

This is the test class
Stored in src/test/java

in this class we are creating the object for login page
and sending credentials to login
and calling the Take Screenshot method and gotoHome.

```
1 package finalProject;
2
3 public class TakesScreenShot {
4
5     public static void main(String[] args) throws Exception {
6
7
8         LoginPage log = new LoginPage();
9
10        String uname = "861234567890";
11        String pass = "861234567890";
12
13        log.login(uname, pass);
14        log.takeScreenshot();
15        log.gotoHome();
16
17
18    }
19
20 }
21
```

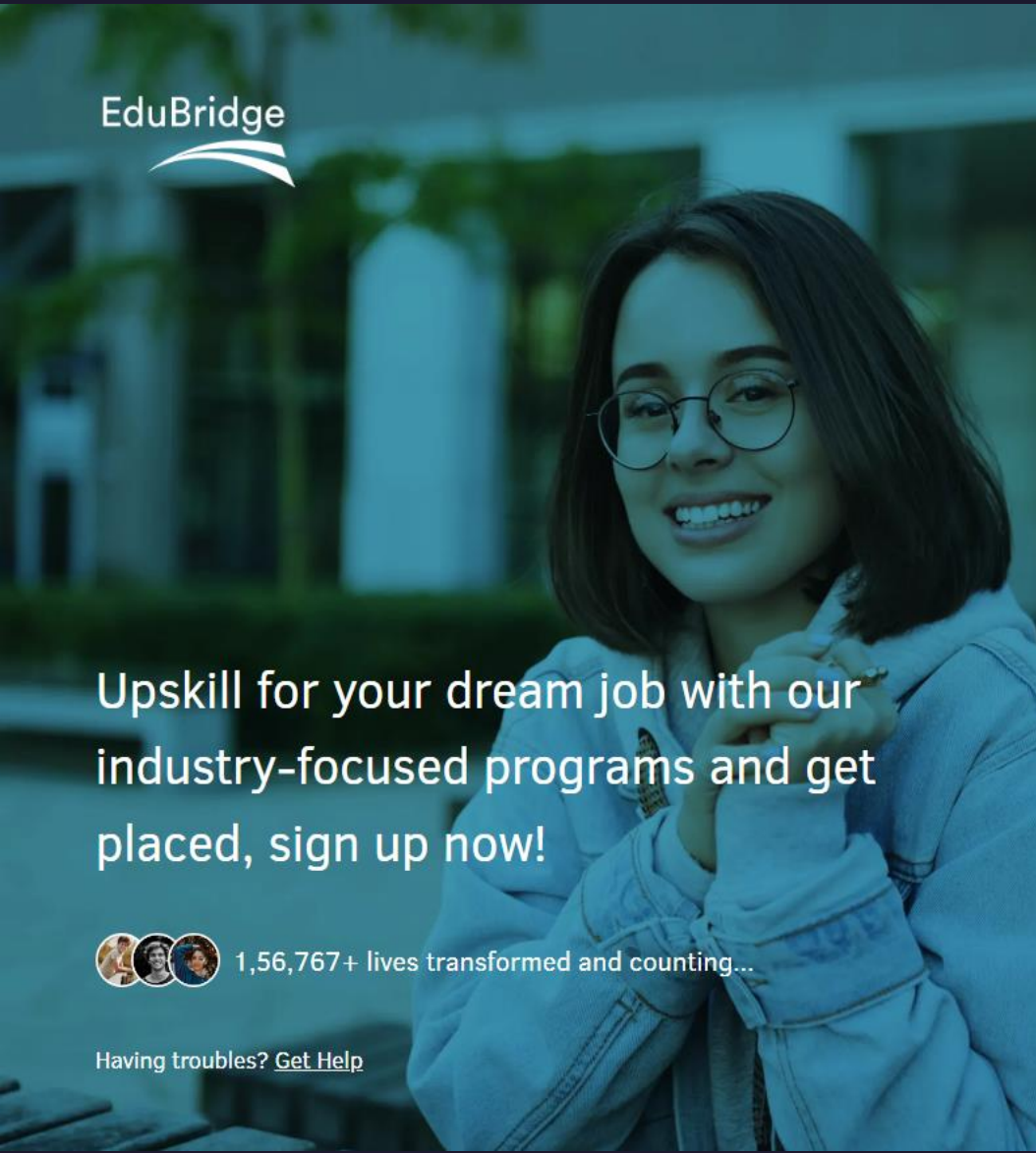


Selenium Code
for capturing Screenshot of Login
page of Edubridge login site

Stored in src/main/java

```
9 import org.openqa.selenium.WebDriver;
10 import org.openqa.selenium.chrome.ChromeDriver;
11
12 public class LoginPage {
13
14     WebDriver driver;
15     File ScreenShotFile;
16
17     LoginPage()
18     {
19         driver = new ChromeDriver();
20         driver.manage().window().maximize();
21         driver.get("https://www.edubridgeindia.com/login");
22     }
23
24
25     By email = By.id("login_mobile");
26     By pwd = By.id("login_password");
27
28     By btnSubmit = By.id("btnLoginSubmit");
29
30
31     public void login(String mobile,String password) {
32         driver.findElement(email).sendKeys(mobile);
33         driver.findElement(pwd).sendKeys(password);
34
35         driver.findElement(btnSubmit).click();
36     }
37
38     public void takeScreenshot() throws Exception
39     {
40         // Taking ScreenShot of Login Page
41         ScreenShotFile = ((TakesScreenshot) driver) .getScreenshotAs(OutputType.FILE);
42         FileUtils.copyFile(ScreenShotFile,new File("../screenshot/Login.png"));
43     }
44
45
46     public void gotoHome() throws Exception
47     {
48         HomePage page = new HomePage(driver);
49         page.handleAlert();
50         page.takeHomeScreenshot();
51         page.logout();
52         page.teardown();
53
54
55     }
56
57 }
58
59
```

Output Image of Login page



Upskill for your dream job with our industry-focused programs and get placed, sign up now!



1,56,767+ lives transformed and counting...

Having troubles? [Get Help](#)

[← Back](#)

Not a member yet? [Sign up](#)

Login to EduBridge

 (+91)

[Use email instead](#)



Enter Password

••••••••



[Forgot Password?](#)

LOGIN



or

 Login with Google

 Login with Facebook




Stored in src/main/java

```


1 package finalProject;
2
3 import java.io.File;
4
5 import org.apache.commons.io.FileUtils;
6 import org.openqa.selenium.By;
7 import org.openqa.selenium.OutputType;
8 import org.openqa.selenium.TakesScreenshot;
9 import org.openqa.selenium.WebDriver;
10 import org.openqa.selenium.WebElement;
11
12 public class HomePage {
13     WebDriver driver;
14
15     HomePage(WebDriver driver)
16     {
17         this.driver=driver;
18     }
19
20     public void handleAlert()
21     {
22         driver.findElement(By.className("close")).click();
23         driver.findElement(By.id("bd-versions")).click();
24     }
25
26     public void takeHomeScreenshot() throws Exception
27     {
28         // Taking ScreenShot of Home Page
29         File ScreenshotFile1 = ((TakesScreenshot) driver).getScreenshotAs(OutputType.FILE);
30         FileUtils.copyFile(ScreenshotFile1,new File("../screenshot/Home.png"));
31     }
32
33     public void logout()
34     {
35         WebElement out = driver.findElement(By.xpath("/html/body/div[1]/header/div[2]/div[2]/div/div[6]/div/div/div[1]"));
36         out.click();
37     }
38
39     public void teardown()
40     {
41         driver.quit();
42     }
43 }

```

Output image of Home page



[Home](#)

[My Courses](#) 


[All Courses](#)

[In Progress](#)

[Completed](#)


[My Calendar](#)


[My Resume](#)



Your Feedback Matters
[Click Here](#)



[Refer, Earn with ELC](#)


[1800 120 1193](#)





 Welcome back, Hari Haran


 Search course catalog

 [Hari Haran](#) 

 [My Profile](#)


 [My Purchases](#)

 [My Certificates](#)

 [My Documents](#)

[Log out](#)


All Courses



Not Started Yet

Aptitude Test Preparation for Job Seekers - II


[View Course](#)




Not Started Yet

Aptitude Test Preparation for Job Seekers - I

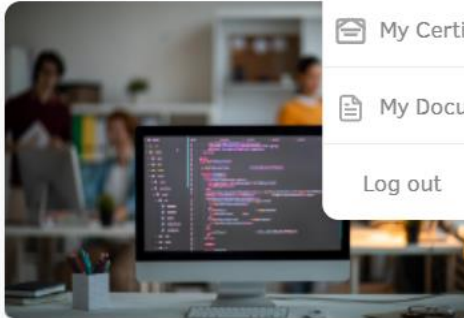
[View Course](#)



85% Completed  56 h remaining


Advanced Certification Program in Software Testing in collaboration with IBM


[View Course](#)



Capgemini - Certified Java Full Stack Professional

[View Course](#)



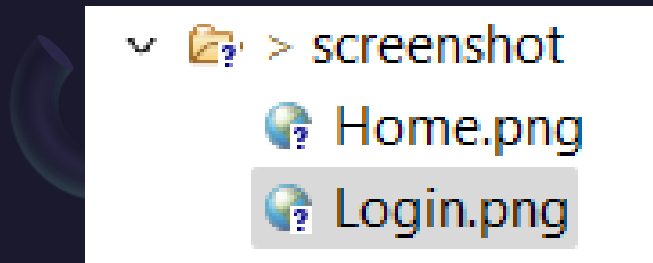
 [Help](#)

Capturing Screenshots And Store

- This is the method is used to take Screenshot

```
public void takeScreenshot() throws Exception
{
    // Taking ScreenShot of Login Page
    ScreenShotFile = ((TakesScreenshot) driver) .getScreenshotAs(OutputType.FILE);
    FileUtils.copyFile(ScreenShotFile,new File("../screenshot/Login.png"));
}
```

- The output of the Screenshot image will stored in folder



This is the method is used to loguot from the site and close the window

```
}  
public void logout()  
{  
    WebElement out = driver.findElement(By.xpath("/html/body/div[1]/header/div[2]/div[2]/div/div[6]/div/div/div[1]/div[2]/a[6]"));  
    out.click();  
}  
public void teardown()  
{  
    driver.quit();  
}
```



Advantages of Selenium:

Cross-Browser Compatibility:

Selenium supports multiple browsers like Chrome, Firefox, Safari, and Internet Explorer, making it easier to test web applications on different platforms.

Platform Independence: Selenium is a cross-platform tool, which means it can be used on different operating systems like Windows, Linux, and macOS.

Support for Multiple Programming Languages:

Selenium supports multiple programming languages, including Java, C#, Python, Ruby, and others. This allows developers and testers to choose the language they are most comfortable with.

Large Community and Resources:

Selenium has a large and active community. This means there is a wealth of documentation, tutorials, and forums available, making it easier for users to find help and solutions to common problems.

Integration with Other Tools:

Selenium can be easily integrated with other tools and frameworks, such as TestNG, JUnit, Maven, and Jenkins, enhancing its capabilities and making it suitable for various testing needs.

Flexibility and Extensibility:

Selenium can be extended for various functionalities through its APIs, making it adaptable to different testing scenarios.

Support for Parallel Execution:

Selenium allows for the parallel execution of test scripts, which can significantly reduce the overall test execution time.





Disadvantages of Selenium:

Limited Support for Desktop Applications: Selenium is primarily designed for web application testing and has limited support for testing desktop applications.

No Built-in Reporting: Selenium lacks built-in reporting capabilities. Test results need to be managed using external tools or by integrating it with other reporting frameworks.

Learning Curve: For beginners, there may be a learning curve associated with Selenium, especially if they are not familiar with the programming languages it supports.


Handling Dynamic Elements: Selenium can sometimes face challenges when dealing with dynamic web elements that change frequently, requiring additional efforts in script maintenance.

No Image Testing: Selenium cannot directly test images, making it challenging to verify certain visual aspects of a web application.

No Support for CAPTCHA and Barcode Readers: Selenium cannot automate tests that involve CAPTCHA or barcode readers directly, as these features are designed to prevent automated interactions.



Conclusion

- 
- Selenium is used to automate testing across several web browsers, including Chrome, Mozilla, Firefox, Safari, and Internet Explorer. Pros of Selenium include its popularity, support for various systems, browsers, and languages, integration with CI/CD platforms, and the ability to test mobile devices. Cons of Selenium include its support for web applications only, lack of built-in image comparison functionality, absence of basic functionality like reports, and lack of technical support. In the final analysis, even if Selenium has a lot to offer in terms of automating web applications, maximizing its value in a testing environment requires careful consideration of its limitations and necessary resources.

THANK YOU ..

