IT 314 Software Engineering

GUI Testing
(Using WebDriver)

Group 4

1. GUI Testing of Sign-up Page using WebDriver Code:

```java
// Generated by Selenium IDE
import org.junit.Test;
import org.junit.Before;
import org.junit.After;
import static org.junit.Assert.*;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.Dimension;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.openqa.selenium.support.ui.ExpectedConditions;
import java.time.Duration;
import java.util.HashMap;
import java.util.Map;

public class SignupTest {
    private WebDriver driver;
    private Map<String, Object> vars;
    private WebDriverWait wait;

    @Before
    public void setUp() {
        System.out.println("Starting test setup...");
        try {
            System.setProperty("webdriver.chrome.driver", "C:\\Users\\JEET PATEL\\OneDrive\\Desktop\\real estate\\lib\\chromedriver.exe");
            driver = new ChromeDriver();
            vars = new HashMap<>();
            wait = new WebDriverWait(driver, Duration.ofSeconds(10));
            System.out.println("Browser initialized successfully.");
        } catch (Exception e) {
            System.err.println("Setup failed: " + e.getMessage());
            throw e;
        }
    }

    public static void main(String[] args) {
        SignupTest test = new SignupTest();
        boolean testPassed = false;

        try {
            System.out.println("\n=== Starting Sign-Up Test ===");
            test.setUp();
            test.signup();
            testPassed = true;
            System.out.println("\n✅ Sign-up test completed successfully!");
            System.out.println("All test steps passed:");
            System.out.println("  - Browser launched");
            System.out.println("  - Navigated to sign-up page");
            System.out.println("  - Entered sign-up details");
            System.out.println("  - Successfully registered user");
            System.out.println("  - Verified sign-up success");
        } catch (Exception e) {
            System.err.println("\n❌ Sign-up test failed!");
            System.err.println("Error details: " + e.getMessage());
            e.printStackTrace();
        } finally {
            test.tearDown();
            System.out.println("\nTest cleanup completed.");
            System.out.println("Final Result: " + (testPassed ? "PASS " : "FAIL "));
            System.out.println("=== Test Execution Completed ===\n");
        }
    }

    @After
    public void tearDown() {
        if (driver != null) {
            System.out.println("Closing browser...");
            driver.quit();
        }
    }

    @Test
    public void signup() {
        try {
            // Navigate to the website
            System.out.println("Navigating to sign-up page...");
            driver.get("https://estatereal-mf2n.onrender.com/");
            driver.manage().window().setSize(new Dimension(1552, 832));

            // Click the sign-up button
            System.out.println("Clicking sign-up button...");
            wait.until(ExpectedConditions.elementToBeClickable(By.cssSelector(".bg-green-500"))).click();

            // Fill out sign-up form
            System.out.println("Filling out sign-up details...");
            driver.findElement(By.id("username")).sendKeys("jeet");
            driver.findElement(By.id("email2")).sendKeys("pateljeet3004@gmail.com");
```

```
88              driver.findElement(By.id("password2")).sendKeys("#Jeet@3004");
89              driver.findElement(By.id("register")).click();
90
91              // Verify successful sign-up
92              System.out.println("Verifying sign-up success...");
93              String successMessage = wait.until(ExpectedConditions.presenceOfElementLocated(By.id("successMessage"))).getText();
94              assertEquals("Sign-up successful!", successMessage);
95
96              System.out.println("Sign-up completed and verified successfully.");
97
98          } catch (Exception e) {
99              System.err.println("Step failed: " + e.getMessage());
100             throw e;
101         }
102     }
103 }
104
```

Since the code is written using the JUnit framework, it cannot be directly run in VS Code. To make it compatible with VS Code, I have created an entry class method, which allows the code to be compiled and executed properly within the IDE.

So to compile it in VS Code, write:

```
C:\Users\JEET PATEL\OneDrive\Desktop\real estate>javac -cp ".;lib/*" SignupTest.java

C:\Users\JEET PATEL\OneDrive\Desktop\real estate>java -cp ".;lib/*" SignupTest.java
```

The output which i get is:

```
=== Starting Sign-Up Test ===
Starting test setup...
Browser initialized successfully.
Navigating to sign-up page...
Clicking sign-up button...
Filling out sign-up details...
Verifying sign-up success...
```

```
Closing browser...

Test cleanup completed.
Final Result: PASS
=== Test Execution Completed ===
```

2. GUI testing of login page using WebDriver
Code:

```java
import org.junit.Test;
import org.junit.Before;
import org.junit.After;
import static org.junit.Assert.*;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.Dimension;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.openqa.selenium.support.ui.ExpectedConditions;
import java.time.Duration;
import java.util.HashMap;
import java.util.Map;

public class LoginTest {
    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;
    private WebDriverWait wait;

    @Before
    public void setUp() {
        System.out.println("Starting test setup...");
        try {
            System.setProperty("webdriver.chrome.driver", "C:\\Users\\JEET PATEL\\OneDrive\\Desktop\\real estate\\lib\\chromedriver.exe");
            driver = new ChromeDriver();
            js = (JavascriptExecutor) driver;
            vars = new HashMap<>();
            wait = new WebDriverWait(driver, Duration.ofSeconds(10));
            System.out.println("Browser initialized successfully.");
        } catch (Exception e) {
            System.err.println("Setup failed: " + e.getMessage());
            throw e;
        }
    }

    public static void main(String[] args) {
        LoginTest test = new LoginTest();
        boolean testPassed = false;

        try {
            System.out.println("\n=== Starting Login Test ===");
            test.setUp();
            test.login();
            testPassed = true;
            System.out.println("\n✅ Login test completed successfully!");
            System.out.println("All test steps passed:");
            System.out.println("  - Browser launched");
            System.out.println("  - Navigated to login page");
            System.out.println("  - Entered credentials");
            System.out.println("  - Successfully logged in");
            System.out.println("  - Verified redirect to dashboard");
        } catch (Exception e) {
            System.err.println("\n❌ Login test failed!");
            System.err.println("Error details: " + e.getMessage());
            e.printStackTrace();
        } finally {
            test.tearDown();
            System.out.println("\nTest cleanup completed.");
            System.out.println("Final Result: " + (testPassed ? "PASS " : "FAILS "));
            System.out.println("=== Test Execution Completed ===\n");
        }
    }

    @After
    public void tearDown() {
        if (driver != null) {
            System.out.println("Closing browser...");
            driver.quit();
        }
    }
}
```

```java
74      @Test
75      public void login() {
76          try {
77              // Navigate to the login page
78              System.out.println("Navigating to login page...");
79              driver.get("https://estatereal-mf2n.onrender.com/");
80              driver.manage().window().setSize(new Dimension(1552, 832));
81
82              // Click login button
83              System.out.println("Clicking login button...");
84              wait.until(ExpectedConditions.elementToBeClickable(By.cssSelector(".bg-green-500"))).click();
85
86              // Enter email
87              System.out.println("Entering email...");
88              wait.until(ExpectedConditions.presenceOfElementLocated(By.id("email2"))).click();
89              driver.findElement(By.id("email2")).sendKeys("pmmc@daiict.ac.in");
90
91              // Enter password
92              System.out.println("Entering password...");
93              driver.findElement(By.id("password2")).click();
94              driver.findElement(By.id("password2")).sendKeys("Le nikal photu@5");
95
96              // Submit login form
97              System.out.println("Submitting login form...");
98              driver.findElement(By.cssSelector("button:nth-child(7)")).click();
99
100             // Wait for redirect and verify URL
101             String expectedUrl = "https://estatereal-mf2n.onrender.com/dashboard";
102             wait.until(ExpectedConditions.urlToBe(expectedUrl));
103             assertEquals("Verification failed: Not redirected to dashboard", expectedUrl, driver.getCurrentUrl());
104             System.out.println("Successfully verified redirect to dashboard.");
105
106         } catch (Exception e) {
107             System.err.println("Step failed: " + e.getMessage());
108             throw e;
109         }
110     }
111 }
```

So to compile it in VS code, write:

```
C:\Users\JEET PATEL\OneDrive\Desktop\real estate>javac -cp ".;lib/*" LoginTest.java

C:\Users\JEET PATEL\OneDrive\Desktop\real estate>java -cp ".;lib/*" LoginTest.java
```

The output which i get is:

```
=== Starting Login Test ===
Starting test setup...
Browser initialized successfully.
Navigating to login page...
Clicking login button...
Entering email...
Entering password...
Submitting login form...
```

```
Closing browser...

Test cleanup completed.
Final Result: PASS
=== Test Execution Completed ===
```

3. GUI testing of Homepage using WebDriver
Code:

```java
1   import org.junit.Test;
2   import org.junit.Before;
3   import org.junit.After;
4   import static org.junit.Assert.*;
5   import org.openqa.selenium.By;
6   import org.openqa.selenium.WebDriver;
7   import org.openqa.selenium.chrome.ChromeDriver;
8   import org.openqa.selenium.Dimension;
9   import org.openqa.selenium.JavascriptExecutor;
10  import org.openqa.selenium.support.ui.WebDriverWait;
11  import org.openqa.selenium.support.ui.ExpectedConditions;
12  import java.time.Duration;
13  import java.util.HashMap;
14  import java.util.Map;
15
16  public class HomepageTest {
17      private WebDriver driver;
18      private Map<String, Object> vars;
19      JavascriptExecutor js;
20      private WebDriverWait wait;
21
22      @Before
23      public void setUp() {
24          System.out.println("Starting test setup...");
25          try {
26              System.setProperty("webdriver.chrome.driver", "C:\\Users\\JEET PATEL\\OneDrive\\Desktop\\real estate\\lib\\chromedriver.exe");
27              driver = new ChromeDriver();
28              js = (JavascriptExecutor) driver;
29              vars = new HashMap<>();
30              wait = new WebDriverWait(driver, Duration.ofSeconds(10));
31              System.out.println("Browser initialized successfully.");
32          } catch (Exception e) {
33              System.err.println("Setup failed: " + e.getMessage());
34              throw e;
35          }
36      }
37
38      public static void main(String[] args) {
39          HomepageTest test = new HomepageTest();
40          boolean testPassed = false;
41
42          try {
43              System.out.println("\n=== Starting Homepage Test ===");
44              test.setUp();
45              test.homepage();
46              testPassed = true;
47              System.out.println("\n✅ Homepage test completed successfully!");
48              System.out.println("All test steps passed:");
49              System.out.println("  - Browser launched");
50              System.out.println("  - Navigated to homepage");
51              System.out.println("  - Searched for location");
52              System.out.println("  - Entered price details");
53              System.out.println("  - Selected property type");
54              System.out.println("  - Finalized selection");
55          } catch (Exception e) {
56              System.err.println("\n❌ Homepage test failed!");
57              System.err.println("Error details: " + e.getMessage());
58              e.printStackTrace();
59          } finally {
60              test.tearDown();
61              System.out.println("\nTest cleanup completed.");
62              System.out.println("Final Result: " + (testPassed ? "PASS " : "FAIL "));
63              System.out.println("=== Test Execution Completed ===\n");
64          }
65      }
66
67      @After
68      public void tearDown() {
69          if (driver != null) {
70              System.out.println("Closing browser...");
71              driver.quit();
72          }
73      }
74
75      @Test
76      public void homepage() {
77          try {
78              // Navigate to the homepage
79              System.out.println("Navigating to homepage...");
80              driver.get("https://estatereal-mf2n.onrender.com/");
81              driver.manage().window().setSize(new Dimension(1552, 832));
82
83              // Search for location
84              System.out.println("Entering location...");
85              wait.until(ExpectedConditions.elementToBeClickable(By.cssSelector("input"))).click();
86              driver.findElement(By.cssSelector("input")).sendKeys("surat");
```

```
87
88            // Submit the search
89            System.out.println("Submitting location search...");
90            driver.findElement(By.cssSelector("button")).click();
91
92            // Enter price details
93            System.out.println("Entering price...");
94            wait.until(ExpectedConditions.elementToBeClickable(By.cssSelector("div:nth-child(1) > div:nth-child(3) > input"))).click();
95            driver.findElement(By.cssSelector("div:nth-child(1) > div:nth-child(3) > input")).sendKeys("55000");
96
97            // Select property type
98            System.out.println("Selecting property type...");
99            driver.findElement(By.name("type")).click();
100           driver.findElement(By.cssSelector("div:nth-child(7) > .gap-x-2:nth-child(4) > input")).click();
101
102           // Finalize selection
103           System.out.println("Finalizing selection...");
104           driver.findElement(By.cssSelector("div:nth-child(8) > .rounded-md")).click();
105
106           // Verify successful interaction (this could be a simple check for a visible element on the next page or success message)
107           System.out.println("Verifying successful interaction...");
108           // Example: assertTrue(driver.findElement(By.id("successMessage")).isDisplayed());
109
110       } catch (Exception e) {
111           System.err.println("Step failed: " + e.getMessage());
112           throw e;
113       }
114   }
115 }
```

So to compile it in VS Code, write:

```
C:\Users\JEET PATEL\OneDrive\Desktop\real estate>javac -cp ".;lib/*" HomepageTest.java

C:\Users\JEET PATEL\OneDrive\Desktop\real estate>java -cp ".;lib/*" HomepageTest.java
```

The output which i get is:

```
=== Starting Homepage Test ===
Starting test setup...
Browser initialized successfully.
Navigating to homepage...
Entering location...
Submitting location search...
Entering price...
Selecting property type...
Finalizing selection...
Verifying successful interaction...
```

```
Test cleanup completed.
Final Result: PASS
=== Test Execution Completed ===
```

4. GUI Testing of Profile Page using WebDriver
Code:

```java
1   import org.junit.Test;
2   import org.junit.Before;
3   import org.junit.After;
4   import static org.junit.Assert.*;
5   import org.openqa.selenium.By;
6   import org.openqa.selenium.WebDriver;
7   import org.openqa.selenium.chrome.ChromeDriver;
8   import org.openqa.selenium.Dimension;
9   import org.openqa.selenium.JavascriptExecutor;
10  import org.openqa.selenium.support.ui.WebDriverWait;
11  import org.openqa.selenium.support.ui.ExpectedConditions;
12  import java.time.Duration;
13  import java.util.HashMap;
14  import java.util.Map;
15
16  public class ProfileTest {
17      private WebDriver driver;
18      private Map<String, Object> vars;
19      JavascriptExecutor js;
20      private WebDriverWait wait;
21
22      @Before
23      public void setUp() {
24          System.out.println("Starting test setup...");
25          try {
26              System.setProperty("webdriver.chrome.driver", "C:\\Users\\JEET PATEL\\OneDrive\\Desktop\\real estate\\lib\\chromedriver.exe");
27              driver = new ChromeDriver();
28              js = (JavascriptExecutor) driver;
29              vars = new HashMap<>();
30              wait = new WebDriverWait(driver, Duration.ofSeconds(10));
31              System.out.println("Browser initialized successfully.");
32          } catch (Exception e) {
33              System.err.println("Setup failed: " + e.getMessage());
34              throw e;
35          }
36      }
37
38      public static void main(String[] args) {
39          ProfileTest test = new ProfileTest();
40          boolean testPassed = false;
41
42          try {
43              System.out.println("\n=== Starting Profile Test ===");
44              test.setUp();
45              test.profile();
```

```java
45              test.profile();
46              testPassed = true;
47              System.out.println("\n✅ Profile test completed successfully!");
48              System.out.println("All test steps passed:");
49              System.out.println("  - Browser launched");
50              System.out.println("  - Navigated to profile page");
51              System.out.println("  - Updated mobile number");
52              System.out.println("  - Changed password");
53              System.out.println("  - Saved profile changes");
54          } catch (Exception e) {
55              System.err.println("\n❌ Profile test failed!");
56              System.err.println("Error details: " + e.getMessage());
57              e.printStackTrace();
58          } finally {
59              test.tearDown();
60              System.out.println("\nTest cleanup completed.");
61              System.out.println("Final Result: " + (testPassed ? "PASS " : "FAIL "));
62              System.out.println("=== Test Execution Completed ===\n");
63          }
64      }
65
66      @After
67      public void tearDown() {
68          if (driver != null) {
69              System.out.println("Closing browser...");
70              driver.quit();
71          }
72      }
73
74      @Test
75      public void profile() {
76          try {
77              // Navigate to the profile page
78              System.out.println("Navigating to profile page...");
79              driver.get("https://estatereal-mf2n.onrender.com/");
80              driver.manage().window().setSize(new Dimension(1552, 832));
81
82              // Click on profile icon
83              System.out.println("Clicking on profile icon...");
84              wait.until(ExpectedConditions.elementToBeClickable(By.cssSelector(".rounded-full"))).click();
85
86              // Open profile settings
```

```
87              System.out.println("Opening profile settings...");
88              driver.findElement(By.cssSelector(".transition-all")).click();
89
90              // Update mobile number
91              System.out.println("Updating mobile number...");
92              wait.until(ExpectedConditions.presenceOfElementLocated(By.id("mobile"))).click();
93              driver.findElement(By.id("mobile")).sendKeys("9426311639");
94
95              // Enter current password
96              System.out.println("Entering current password...");
97              driver.findElement(By.id("currentPassword")).click();
98              driver.findElement(By.id("currentPassword")).sendKeys("#Jeet@3004");
99
100             // Enter new password
101             System.out.println("Entering new password...");
102             driver.findElement(By.id("newPassword")).click();
103             driver.findElement(By.id("newPassword")).sendKeys("#JEet@3004");
104
105             // Save the profile changes
106             System.out.println("Saving profile changes...");
107             driver.findElement(By.cssSelector(".px-6:nth-child(1)")).click();
108
109             // Verify successful profile update (this can be a check for a success message)
110             System.out.println("Verifying successful profile update...");
111             // Example: assertTrue(driver.findElement(By.id("successMessage")).isDisplayed());
112
113         } catch (Exception e) {
114             System.err.println("Step failed: " + e.getMessage());
115             throw e;
116         }
117     }
118 }
```

So to compile it in VS Code, write:

```
C:\Users\JEET PATEL\OneDrive\Desktop\real estate>javac -cp ".;lib/*" ProfileTest.java

C:\Users\JEET PATEL\OneDrive\Desktop\real estate>java -cp ".;lib/*" ProfileTest.java
```

The output which i get is:

```
=== Starting Profile Test ===
Starting test setup...
Browser initialized successfully.
Navigating to profile page...
Clicking on profile icon...
Opening profile settings...
```

```
Test cleanup completed.
Final Result: PASS
=== Test Execution Completed ===
```

5. GUI Testing of Add Property page using WebDriver Code:

```java
import org.junit.Test;
import org.junit.Before;
import org.junit.After;
import static org.junit.Assert.*;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.Dimension;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.interactions.Actions;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.openqa.selenium.JavascriptExecutor;
import java.time.Duration;
import java.util.HashMap;
import java.util.Map;

public class AddPropertyTest {
    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;
    private WebDriverWait wait;

    @Before
    public void setUp() {
        System.out.println("Starting test setup...");
        System.setProperty("webdriver.chrome.driver", "C:\\Users\\JEET PATEL\\OneDrive\\Desktop\\real estate\\lib\\chromedriver.exe");
        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<>();
        wait = new WebDriverWait(driver, Duration.ofSeconds(10));
        System.out.println("Browser initialized successfully.");
    }
    public static void main(String[] args) {
        AddPropertyTest test = new AddPropertyTest();
        boolean testPassed = false;

        try {
            System.out.println("\n=== Starting Add Property Test ===");
            test.setUp();
            test.addProperty();
            testPassed = true;
            System.out.println("\n✅ Add Property test completed successfully!");
            System.out.println("All test steps passed:");
            System.out.println("  - Browser launched");
            System.out.println("  - Logged into the website");
            System.out.println("  - Navigated to Add Property page");
            System.out.println("  - Filled in property details");
            System.out.println("  - Submitted property successfully");
        } catch (Exception e) {
            System.err.println("\n❌ Add Property test failed!");
            System.err.println("Error details: " + e.getMessage());
            e.printStackTrace();
        } finally {
            test.tearDown();
            System.out.println("\nTest cleanup completed.");
            System.out.println("Final Result: " + (testPassed ? "PASS" : "FAIL"));
            System.out.println("=== Test Execution Completed ===\n");
        }
    }


    @After
    public void tearDown() {
        if (driver != null) {
            System.out.println("Closing browser...");
            driver.quit();
        }
    }

    @Test
    public void addProperty() {
        try {
            // Navigate to the website
            System.out.println("Navigating to login page...");
            driver.get("https://estatereal-mf2n.onrender.com/");
            driver.manage().window().setSize(new Dimension(1552, 832));

            // Perform login
            System.out.println("Clicking login button...");
            wait.until(ExpectedConditions.elementToBeClickable(By.cssSelector(".bg-green-500"))).click();
            System.out.println("Entering email...");
            wait.until(ExpectedConditions.presenceOfElementLocated(By.id("email2"))).sendKeys("pmmc@daiict.ac.in");
            System.out.println("Entering password...");
            driver.findElement(By.id("password2")).sendKeys("Le nikal photu@5");
            System.out.println("Submitting login form...");
            driver.findElement(By.cssSelector("button:nth-child(7)")).click();
```

```
 89        // Verify successful login by checking the profile page URL
 90        String expectedProfileUrl = "https://estatereal-mf2n.onrender.com/profile";
 91        wait.until(ExpectedConditions.urlToBe(expectedProfileUrl));
 92        assertEquals("Verification failed: Not redirected to profile page", expectedProfileUrl, driver.getCurrentUrl());
 93        System.out.println("Successfully logged in and redirected to profile page.");
 94
 95        // Navigate to Add Property page
 96        System.out.println("Navigating to Add Property page...");
 97        driver.findElement(By.cssSelector(".rounded-full")).click(); // Open menu (if required)
 98        driver.findElement(By.linkText("Add Property")).click();
 99
100        // Verify successful navigation to Add Property page
101        String expectedAddPropertyUrl = "https://estatereal-mf2n.onrender.com/create-listing";
102        wait.until(ExpectedConditions.urlToBe(expectedAddPropertyUrl));
103        assertEquals("Verification failed: Not redirected to Add Property page", expectedAddPropertyUrl, driver.getCurrentUrl());
104        System.out.println("Successfully navigated to Add Property page.");
105
106        // Fill out the Add Property form
107        System.out.println("Filling out Add Property form...");
108        driver.findElement(By.id("name")).sendKeys("The Real Estate Villa");
109        driver.findElement(By.id("description")).sendKeys("This is the best villa in this price range");
110        driver.findElement(By.id("address")).sendKeys("Infocity");
111        driver.findElement(By.id("city")).sendKeys("Gandhinagar");
112        driver.findElement(By.id("pinCode")).sendKeys("363999");
113        driver.findElement(By.id("bedrooms")).sendKeys("2");
114        driver.findElement(By.id("images")).sendKeys("C:\\Users\\JEET PATEL\\Downloads\\photo.jpeg");
115        driver.findElement(By.cssSelector(".upload-button")).click();
116
117        // Select visit slots
118        System.out.println("Selecting visit slots...");
119        WebElement dropdown = driver.findElement(By.id("visitSlots"));
120        dropdown.findElement(By.xpath("//option[. = '7:00 AM - 8:00 AM']")).click();
121        dropdown.findElement(By.xpath("//option[. = '8:00 AM - 9:00 AM']")).click();
122
123        // Submit the form
124        System.out.println("Submitting the Add Property form...");
125        driver.findElement(By.cssSelector(".submit-button")).click();
126        js.executeScript("window.scrollTo(0, 0)");
127
128        // Verify success (you can add an assertion based on expected behavior or URL)
129        System.out.println("Property added successfully!");
130
131    } catch (Exception e) {
```

```
132            System.err.println("Test failed: " + e.getMessage());
133            throw e;
134        }
135    }
136 }
137
```

So to Compile this in VS Code, write:

```
C:\Users\JEET PATEL\OneDrive\Desktop\real estate>javac -cp ".;lib/*" AddPropertyTest.java

C:\Users\JEET PATEL\OneDrive\Desktop\real estate>java -cp ".;lib/*" AddPropertyTest.java
```

The output which i get is:

```
=== Starting Add Property Test ===
Starting test setup...
Browser initialized successfully.
Navigating to login page...
Clicking login button...
Entering email...
Entering password...
Submitting login form...
Successfully logged in and redirected to profile page.
Navigating to Add Property page...
Successfully navigated to Add Property page.
Filling out Add Property form...
Selecting visit slots...
Submitting the Add Property form...
Property added successfully!
```

```
? Add Property test completed successfully!
All test steps passed:
   - Browser launched
   - Logged into the website
   - Navigated to Add Property page
   - Filled in property details
   - Submitted property successfully
Closing browser...

Test cleanup completed.
Final Result: PASS
=== Test Execution Completed ===
```