

5. Implement 3 and 4 Programs using TestNG Concepts and write it using @Test annotation.

Step 1: Installed TestNG using Eclipse Marketplace.

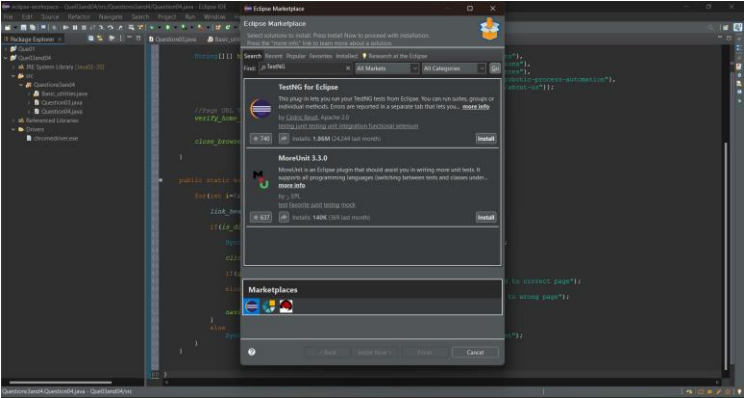
Step 2: Added TestNG libraries.

Step 3: Modified the code with TestNG annotations.

Question3\_TestNG class: used @BeforeClass, @AfterClass and @Test annotations

Question4\_TestNG class: used @DataProvider, @BeforeClass, @AfterClass and @Test annotations

Note: In both classes, I have added assertions. And previously Question4.java had only one method as “verify\_home\_header\_links()”. To add multiple annotations, I have divided that into two methods as “verify\_home\_header\_link\_availability()” and “verify\_home\_header\_links\_navigation()”.



Note: Enter code is available at the end of the document

And add TestNG Attributes – priority, enabled and timeOut to run the programs.

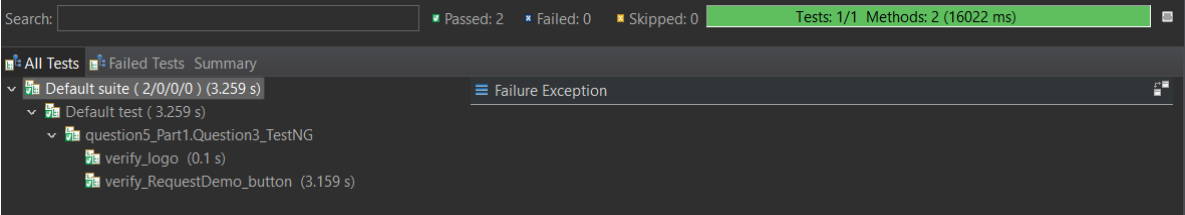
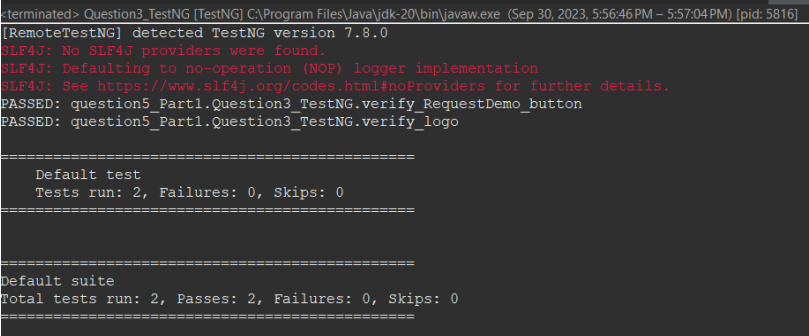
Question3\_TestNG class: used priority and enabled attributes.

Question4\_TestNG class: used dependsOnMethods, dataProvider, priority and timeOut attributes.

Run the program with enabled attribute value as either true or false.

In Question3\_TestNG class, I have added a new test as “verify\_RequestDemo\_button\_navigation()” and using enabled attribute, I skipped that.

```
@Test(enabled=false)
//-----This is not available in the question-----
public void verify_RequestDemo_button_navigation() {
    click(button_RequestDemo);
    Assert.assertEquals(get_current_url(), "https://www.automationanywhere.com/request-live-demo","Request Demo'link navigated to wrong page");
}
```



Run the program with timeOut Attribute values.

In Question4\_TestNG class, I have added a timeOut attribute to the method “verify\_home\_header\_link\_availability()”. If it is failed, method “verify\_home\_header\_links\_navigation()” will be skipped. For Testing purpose, I have added less time(10 ms).

```
@Test (dataProvider ="test_data", timeout = 1000)
public void verify_home_header_link_availability(String link_name, String url ) {

    link_headerOption =By.xpath("//a[text()='"+link_name+"']");
    Assert.assertTrue(is_displayed(link_headerOption), ""+link_headerOption+" is not present");
}

@Test (dependsOnMethods = { "verify_home_header_link_availability" }, dataProvider ="test_data", priority=1)
public void verify_home_header_links_navigation(String link_name, String url ) {

    link_headerOption =By.xpath("//a[text()='"+link_name+"']");

    click(link_headerOption);

    Assert.assertEquals(get_current_url(), url,""+link_name+" link navigated to wrong page");

    navigate_back();
}
```

Search:

Passed: 0Failed: 5Skipped: 1

Tests: 1/1Methods: 6 (8329 ms)

All TestsFailed TestsSummary

Default suite (0/5/1/0) (0.235 s)

question5\_Part1.Question4\_TestNG

verify\_home\_header\_link\_availability (0.049 s)

Products", "https://www.automationanywhere.com/products" (0.049 s)

verify\_home\_header\_link\_availability (0.034 s)

Solutions", "https://www.automationanywhere.com/solutions" (0.034 s)

verify\_home\_header\_link\_availability (0.039 s)

Resources", "https://www.automationanywhere.com/resources" (0.039 s)

verify\_home\_header\_link\_availability (0.055 s)

Beyond RPA", "https://www.automationanywhere.com/rpa/robotic-process-automation" (0.05 s)

verify\_home\_header\_link\_availability (0.058 s)

Company", "https://www.automationanywhere.com/company/about-us" (0.058 s)

verify\_home\_header\_links\_navigation (0 s)

Failure Exception

org.testng.internal.thread.ThreadTimeoutException: Method question5\_Part1.Question4\_TestNG.verify\_home\_header\_link\_availability() didn't finish within the time-out 10  
at java.base/jdk.internal.misc.Unsafe.park(Native Method)  
at java.base/java.util.concurrent.locks.LockSupport.parkNanos(LockSupport.java:269)  
at java.base/java.util.concurrent.CompletableFuture\$Signaller.block(CompletableFuture.java:1866)  
at java.base/java.util.concurrent.ForkJoinPool.unmanagedBlock(ForkJoinPool.java:3780)  
at java.base/java.util.concurrent.ForkJoinPool.managedBlock(ForkJoinPool.java:3725)  
at java.base/java.util.concurrent.CompletableFuture.timedGet(CompletableFuture.java:1939)  
at java.base/java.util.concurrent.CompletableFuture.get(CompletableFuture.java:2095)  
at org.asynchttpclient.netty.NettyResponseFuture.get(NettyResponseFuture.java:206)  
at org.openqa.selenium.remote.http.netty.NettyHttpHandler.makeCall(NettyHttpHandler.java:59)  
at org.openqa.selenium.remote.http.AddSeleniumUserAgent.lambda\$apply\$0(AddSeleniumUserAgent.java:42)  
at org.openqa.selenium.remote.http.Filter.lambda\$andFinally\$1(Filter.java:55)  
at org.openqa.selenium.remote.http.netty.NettyHttpHandler.execute(NettyHttpHandler.java:48)  
at org.openqa.selenium.remote.http.netty.NettyClient.execute(NettyClient.java:96)  
at org.openqa.selenium.remote.HttpCommandExecutor.execute(HttpCommandExecutor.java:188)  
at org.openqa.selenium.remote.service.DriverCommandExecutor.invokeExecute(DriverCommandExecutor.java:196)  
at org.openqa.selenium.remote.service.DriverCommandExecutor.execute(DriverCommandExecutor.java:171)  
at org.openqa.selenium.remote.RemoteWebDriver.execute(RemoteWebDriver.java:602)  
at org.openqa.selenium.remote.ElementLocation\$ElementFinder\$2.findElement(ElementLocation.java:165)  
at org.openqa.selenium.remote.ElementLocation.findElement(ElementLocation.java:59)  
at org.openqa.selenium.remote.RemoteWebDriver.findElement(RemoteWebDriver.java:359)  
at org.openqa.selenium.remote.RemoteWebDriver.findElement(RemoteWebDriver.java:353)  
at question5\_Part1.Basic\_utilities\_TestNG.is\_displayed(Basic\_utilities\_TestNG.java:43)  
at question5\_Part1.Question4\_TestNG.verify\_home\_header\_link\_availability(Question4\_TestNG.java:35)  
at java.base/jdk.internal.reflect.DirectMethodHandleAccessor.invoke(DirectMethodHandleAccessor.java:104)  
at java.base/java.util.concurrent.FutureTask.run(FutureTask.java:317)  
at java.base/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1144)  
at java.base/java.util.concurrent.ThreadPoolExecutor\$Worker.run(ThreadPoolExecutor.java:642)

And run 3 and 4 Programs from TestNG.xml file and verify the generated test reports.

Que01Que03and04

JRE System Library [JavaSE-20]

src

question5\_Part1

Basic\_utilities\_TestNG.java

Question3\_TestNG.java

Question4\_TestNG.java

Questions3and4

Basic\_utilities.java

Question03.java

Question04.java

Referenced Libraries

TestNG

Drivers

testng.xml

https://testng.org/testng-1.0.dtd (doctype)

1 <?xml version="1.0" encoding="UTF-8"?>

2 <!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">

3 <suite name="Testing www.automationanywhere.com home page">

4 <test name="Regression">

5 <classes>

6 <class name="question5\_Part1.Question3\_TestNG"></class>

7 <class name="question5\_Part1.Question4\_TestNG"></class>

8 </classes>

9 </test> <!-- Test -->

10 </suite> <!-- Suite -->

11

<terminated> Que03and04\_testng.xml [TestNG] C:\Program Files\Java\jdk-20\bin\javaw.exe (Sep 30, 2023, 6:04:11 PM – 6:05:05 PM) [pid: 29464]

[RemoteTestNG] detected TestNG version 7.8.0

SLF4J: No SLF4J providers were found.

SLF4J: Defaulting to no-operation (NOP) logger implementation

SLF4J: See <https://www.slf4j.org/codes.html#noProviders> for further details.

=====

Testing www.automationanywhere.com home page

Total tests run: 12, Passes: 12, Failures: 0, Skips: 0

=====

Search:

Passed: 12Failed: 0Skipped: 0

Tests: 1/1Methods: 12 (52510 ms)

All TestsFailed TestsSummary

Testing www.automationanywhere.com home page ( 12/0/0/0 ) (39.18 s)

Failure Exception

Regression ( 39.18 s)

question5\_Part1.Question3\_TestNG

verify\_logo (0.069 s)

verify\_RequestDemo\_button (3.185 s)

question5\_Part1.Question4\_TestNG

verify\_home\_header\_link\_availability (0.028 s)

"Products","https://www.automationanywhere.com/products"

verify\_home\_header\_link\_availability (0.026 s)

"Solutions","https://www.automationanywhere.com/solutions"

verify\_home\_header\_link\_availability (0.034 s)

"Resources","https://www.automationanywhere.com/resources"

verify\_home\_header\_link\_availability (0.043 s)

"Beyond RPA","https://www.automationanywhere.com/rpa/rot"

verify\_home\_header\_link\_availability (0.029 s)

"Company","https://www.automationanywhere.com/company/"

verify\_home\_header\_links\_navigation (11.733 s)

"Products","https://www.automationanywhere.com/products"

verify\_home\_header\_links\_navigation (5.289 s)

"Solutions","https://www.automationanywhere.com/solutions"

verify\_home\_header\_links\_navigation (6.077 s)

"Resources","https://www.automationanywhere.com/resources"

verify\_home\_header\_links\_navigation (7.262 s)

"Beyond RPA","https://www.automationanywhere.com/rpa/rot"

verify\_home\_header\_links\_navigation (5.405 s)

"Company","https://www.automationanywhere.com/company/"

## Entire Code:

Basic\_utilities\_TestNG class:

```
package question5_Part1;

import java.util.concurrent.TimeUnit;
import org.openqa.selenium.By;
import org.openqa.selenium.ElementNotInteractableException;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class Basic_utilities_TestNG {

    public static WebDriver driver;

    public static void open_driver(String browserName ){
        if(browserName.equals("chrome")) {
            System.setProperty("webdriver.chrome.driver",".//Drivers//chromedriver.exe");
            driver = new ChromeDriver();
        }
        driver.manage().window().maximize();
        driver.manage().timeouts().implicitlyWait(60, TimeUnit.SECONDS);
    }

    public static void close_browser() {
        if(driver!=null)
            driver.quit();
    }

    public static void navigate_to_URL(String url) {
        driver.get(url);
    }

    public static void navigate_back() {
        driver.navigate().back();
    }

    public static boolean is_displayed(By element) {
        try {
```

```

        return driver.findElement(element).isDisplayed();
    }
    catch(Exception e) {
        return false;
    }
}

public static boolean is_enabled(By element) {
    try {
        return driver.findElement(element).isEnabled();
    }
    catch(Exception e) {
        return false;
    }
}

public static void scroll_browser(int horizontal, int vertical) {
    try {
        ((JavascriptExecutor) driver).executeScript("window.scrollBy("+horizontal+", "+vertical+")", "");
        Thread.sleep(3000);
    } catch(Exception e) {
        e.printStackTrace();
    }
}

public static void scroll_to_element(By element) {
    try {
        ((JavascriptExecutor) driver).executeScript("arguments[0].scrollIntoView(true);", driver.findElement(element));
    } catch(Exception e) {
        e.printStackTrace();
    }
}

public static void click(By element) {
    try {
        driver.findElement(element).click();
    } catch(ElementNotInteractableException e) {

        scroll_to_element(element);
    }
}

```

```

        driver.findElement(element).click();
    }
}

public static String get_current_url() {
    return driver.getCurrentUrl();
}
}

```

Question3\_TestNG class:

```

package question5_Part1;

import org.testng.annotations.Test;
import org.testng.asserts.SoftAssert;
import org.openqa.selenium.By;
import org.testng.Assert;
import org.testng.annotations.AfterClass;
import org.testng.annotations.BeforeClass;

public class Question3_TestNG extends Basic_utilities_TestNG {

    // page elements
    public By logo=By.xpath("//img[@class='coh-image coh-image-responsive-xl']"); //Relative XPath
    public By button_RequestDemo = By.xpath("/html/body/div[1]/header/div/div/div/div/div[2]/div/div/div[1]/div[2]/a"); //Absolute XPath
    public By button_accept_all_cookie = By.xpath("//*[@id='onetrust-accept-btn-handler']");

    @Test(priority=2)
    //-----Question 03-b-----
    public void verify_RequestDemo_button() {

        SoftAssert soft_assert=new SoftAssert(); //Used soft assert-Without scrolling button is not visible

        soft_assert.assertTrue(is_displayed(button_RequestDemo), "RequestDemo button is not present");

        //Scroll down the page by 200px
        scroll_browser(0,200);

        soft_assert.assertTrue(is_displayed(button_RequestDemo), "RequestDemo button is not present");
        soft_assert.assertTrue(is_enabled(button_RequestDemo), "RequestDemo button is not enabled");
    }
}

```

```

}

@Test(priority=1)
//-----Question 03-a-----
public void verify_logo() {
    Assert.assertTrue(is_displayed(logo), "logo is not present");
}

@Test(enabled=false)
//-----This is not available in the question-----
public void verify_RequestDemo_button_navigation() {
    click(button_RequestDemo);
    Assert.assertEquals(get_current_url(), "https://www.automationanywhere.com/request-live-demo","Request Demo'link navigated to wrong page");
}

@BeforeClass
public void set_up() {
    open_driver("chrome");
    navigate_to_URL("https://www.automationanywhere.com/");
    click(button_accept_all_cookie);
}

@AfterClass
public void tear_down() {
    close_browser();
}

}

```

Question4\_TestNG class:

```

package question5_Part1;

import org.testng.annotations.Test;
import org.testng.annotations.BeforeClass;

```

```

import org.testng.annotations.DataProvider;
import org.openqa.selenium.By;
import org.testng.Assert;
import org.testng.annotations.AfterClass;
import org.testng.annotations.AfterTest;

public class Question4_TestNG extends Basic_utilities_TestNG {

    // page elements
    public By button_accept_all_cookie = By.xpath("//*[@id='onetrust-accept-btn-handler']");
    public By link_headerOption;

    @DataProvider(name="test_data")
    public String[][] header_link_data_provider(){

        String[][] headerLinks = {{ "Products", "https://www.automationanywhere.com/products"},
                                    {"Solutions", "https://www.automationanywhere.com/solutions"},
                                    {"Resources", "https://www.automationanywhere.com/resources"},
                                    {"Beyond RPA", "https://www.automationanywhere.com/rpa/robotic-process-automation"},
                                    {"Company", "https://www.automationanywhere.com/company/about-us"} };

        return headerLinks;
    }

    @Test (dataProvider = "test_data", timeOut = 1000)
    public void verify_home_header_link_availability(String link_name, String url ) {

        link_headerOption = By.xpath("//a[text()='"+link_name+"']");
        Assert.assertTrue(is_displayed(link_headerOption), ""+link_headerOption+" is not present");
    }

    @Test (dependsOnMethods = { "verify_home_header_link_availability" }, dataProvider = "test_data", priority=1)
    public void verify_home_header_links_navigation(String link_name, String url ) {

```



```

        link_headerOption =By.xpath("//a[text()='"+link_name+"'"]");

        click(link_headerOption);

        Assert.assertEquals(get_current_url(), url," '"+link_name+"' link navigated to wrong page");

        navigate_back();
    }

    @BeforeClass
    public void set_up() {
        open_driver("chrome");
        navigate_to_URL("https://www.automationanywhere.com/");
        click(button_accept_all_cookie);
    }

    @AfterClass
    public void tear_down() {
        close_browser();
    }
}

```

TestNG XML:

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">
<suite name="Testing www.automationanywhere.com home page">
    <test name="Regression">
        <classes>

```

```
        <class name="question5_Part1.Question3_TestNG"></class>
        <class name="question5_Part1.Question4_TestNG"></class>
    </classes>
</test> <!-- Test -->
</suite> <!-- Suite -->
```