

TestNG annotations example

```
Public class java
{

@BeforeClass                                // runs only once at the start of the class
Public void method1()

@BeforeMethod                               // it runs before test1
Public void method2()                       // it runs before test2

@Test1(dependsonmethod = test3)             // test1
Public void method3()

@Test2(Timeout = 20 seconds)                // test2
Public void method4()

@Test3()                                    // test3
Public void method7()

@AfterMathod                               // it runs after test1
Public void method5()                       // it runs after test2

@AfterClass                                // runs only once at the end of the class
Public void method6()

}
```

=====

Framework Implementation

1. Demo site for automation > <https://demo.guru99.com/>
2. Enter you email id and get the user id and password for the demo site

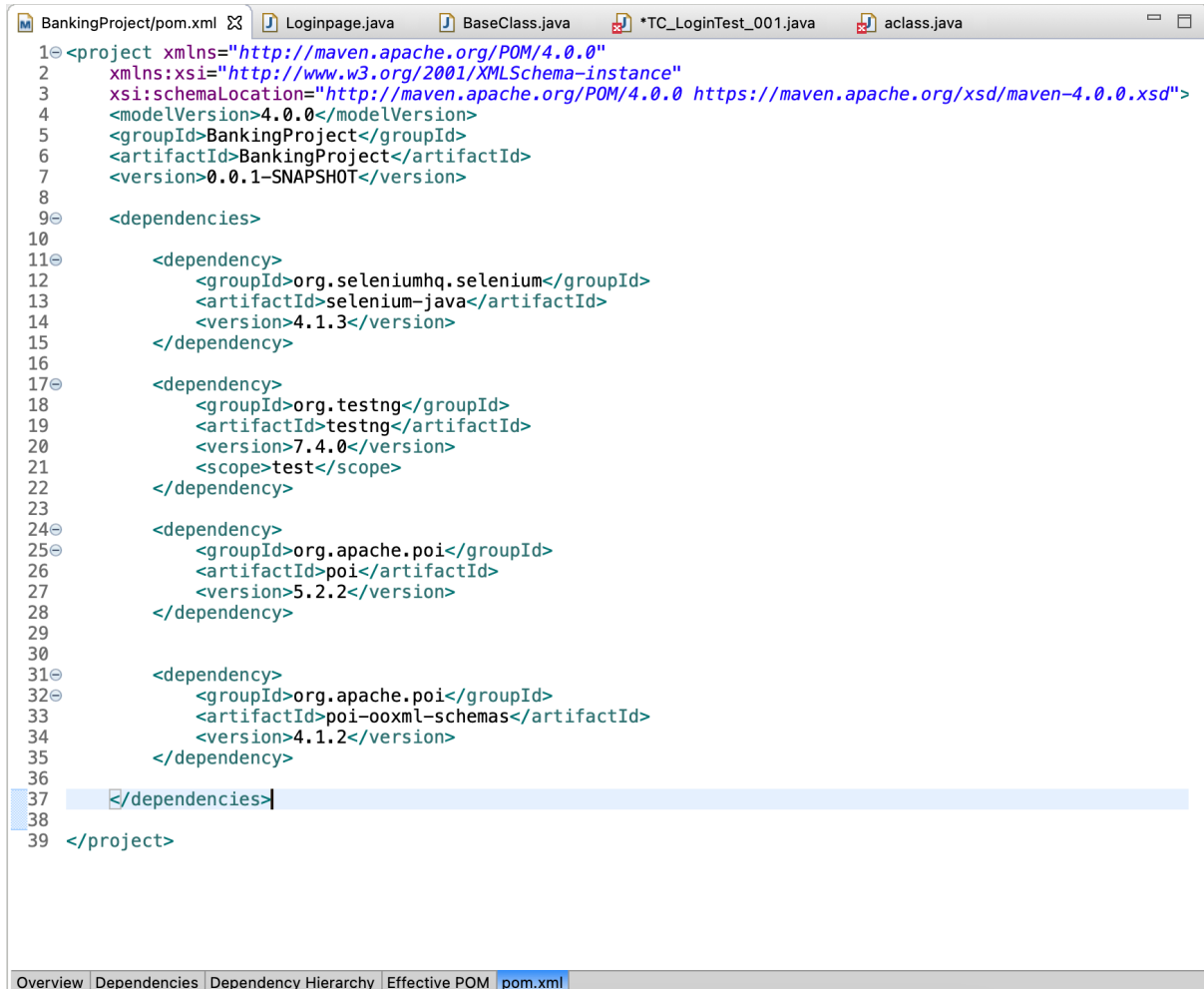
Implementation

1. Create a maven project

- Go to File > New > Other > Maven
- Inside maven folder > select maven project
- Click on create simple project
- Enter Group id and Artifact id
- Click on finish

2. Add dependencies in pom.xml file

- Go to <https://mvnrepository.com/>
- Search for selenium, testng, apache poi and add the dependencies into the pom.xml file.



The screenshot shows an IDE with several tabs: BankingProject/pom.xml, Loginpage.java, BaseClass.java, *TC_LoginTest_001.java, and aclass.java. The pom.xml file is open, showing the following XML structure:

```
1 <project xmlns="http://maven.apache.org/POM/4.0.0"
2   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3   xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
4   <modelVersion>4.0.0</modelVersion>
5   <groupId>BankingProject</groupId>
6   <artifactId>BankingProject</artifactId>
7   <version>0.0.1-SNAPSHOT</version>
8
9   <dependencies>
10
11     <dependency>
12       <groupId>org.seleniumhq.selenium</groupId>
13       <artifactId>selenium-java</artifactId>
14       <version>4.1.3</version>
15     </dependency>
16
17     <dependency>
18       <groupId>org.testng</groupId>
19       <artifactId>testng</artifactId>
20       <version>7.4.0</version>
21       <scope>test</scope>
22     </dependency>
23
24     <dependency>
25       <groupId>org.apache.poi</groupId>
26       <artifactId>poi</artifactId>
27       <version>5.2.2</version>
28     </dependency>
29
30
31     <dependency>
32       <groupId>org.apache.poi</groupId>
33       <artifactId>poi-ooxml-schemas</artifactId>
34       <version>4.1.2</version>
35     </dependency>
36
37   </dependencies>
38
39 </project>
```

The bottom of the IDE shows a tab bar with the following tabs: Overview, Dependencies, Dependency Hierarchy, Effective POM, and pom.xml (which is currently selected).

3. Create multiple packages naming as follows

- com.bankingproject.pageobjectclass
- com.bankingproject.testclass
- com.bankingproject.testdata
- com.bankingproject.utilities

4. Create various folders

- To store the screenshot
- To store the reports
- To store the browser driver

5. Create page object class under package “com.bankingproject.pageobjectclass”

E.x

```
package com.bankingProject.pageObjectClass;

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.FindBy;
import org.openqa.selenium.support.PageFactory;

public class LoginPage {

    WebDriver ldriver;

    @FindBy(xpath = "//input[@name='uid']")           // elements
    WebElement userID;

    @FindBy(xpath = "//input[@name='password']")
    WebElement password;

    @FindBy(xpath = "//input[@name='btnLogin']")
    WebElement loginButton;

    @FindBy(xpath = "//input[@name='btnReset']")
    WebElement resetButton;

    @FindBy(xpath = "//a[text()='Bank Project']")
    WebElement bankProjectLink;

    public LoginPage(WebDriver rdriver)              // constructor
    {
        ldriver = rdriver;
        PageFactory.initElements(rdriver, this);
    }

    public void setUsername(String name)             // method 1
    {
        userID.sendKeys(name);
    }

    public void setPassword(String pass)            // method 2
    {
        password.sendKeys(pass);
    }

    public void clickOnLoginButton()                // method 3
}
```

```

    {
        loginButton.click();
    }

    public void clickOnResetButton() // method 4
    {
        resetButton.click();
    }

    public void clickOnBankProjectLink()
    {
        bankProjectLink.click();
    }
}

```

=====

6. Create base class under package “com.bankingproject.pageobjectclass”

```

package com.bankingProject.pageObjectClass;

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.annotations.AfterClass;
import org.testng.annotations.BeforeClass;

public class BaseClass {

    public static WebDriver driver;

    @BeforeClass
    public void setup()
    {
        System.setProperty("webdriver.chrome.driver",
"/Users/ amitparate/Downloads/chromedriver");
        driver = new ChromeDriver();
        driver.get("https://demo.guru99.com/V1/index.php");
    }

    @AfterClass
    public void closeBrowser()
    {
        driver.quit();
    }
}

```

=====

7. Create base class under package “com.bankingproject.pageobjectclass”

```
package com.bankingProject.testClass;

import org.testng.annotations.Test;

import com.bankingProject.pageObjectClass.BaseClass;
import com.bankingProject.pageObjectClass.LoginPage;

public class TC_001_loginTest extends BaseClass
{
    @Test
    public void loginTest() throws InterruptedException
    {
        LoginPage login = new LoginPage(driver);

        Thread.sleep(2000);
        //login.clickOnBankProjectLink();
        Thread.sleep(2000);
        login.setUserName("mngr423480");
        Thread.sleep(2000);
        login.setPassword("AdYreva");
        Thread.sleep(2000);
        login.clickOnLoginButton();

        String pageTitle = "GTPL Bank Manager HomePage";

        if(driver.getTitle().equals(pageTitle))
        {
            System.out.println("Title is correct");
        }else {
            System.out.println("Title is incorrect");
        }
    }
}
```

=====