软件测试上机报告

第二次上机作业

学　　院_____软件学院_____
专　　业_____软件工程_____
姓　　名_____刘坤鑫_____
学　　号_____3017218061_____
年　　级_____17 级_____
班　　级_____1 班_____

Tasks

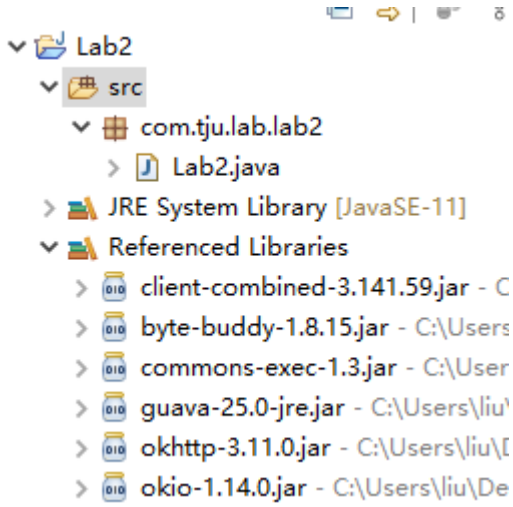
1. Install Selenium with Eclipse.
2. Install Firefox and SeleniumIDE plugin.
3. Try to record and export scripts using SeleniumIDE.
4. Please complete the following task using Selenium Webdriver:

"Selenium Lab.xlsx" contains information about the students, and http://103.120.226.190/selenium-demo/git-repo can view someone's information after logging in (student id as username, git address as password). Please check each record in the excel to make sure that each student's information is consistent with the information on the website.

The Experimental Steps

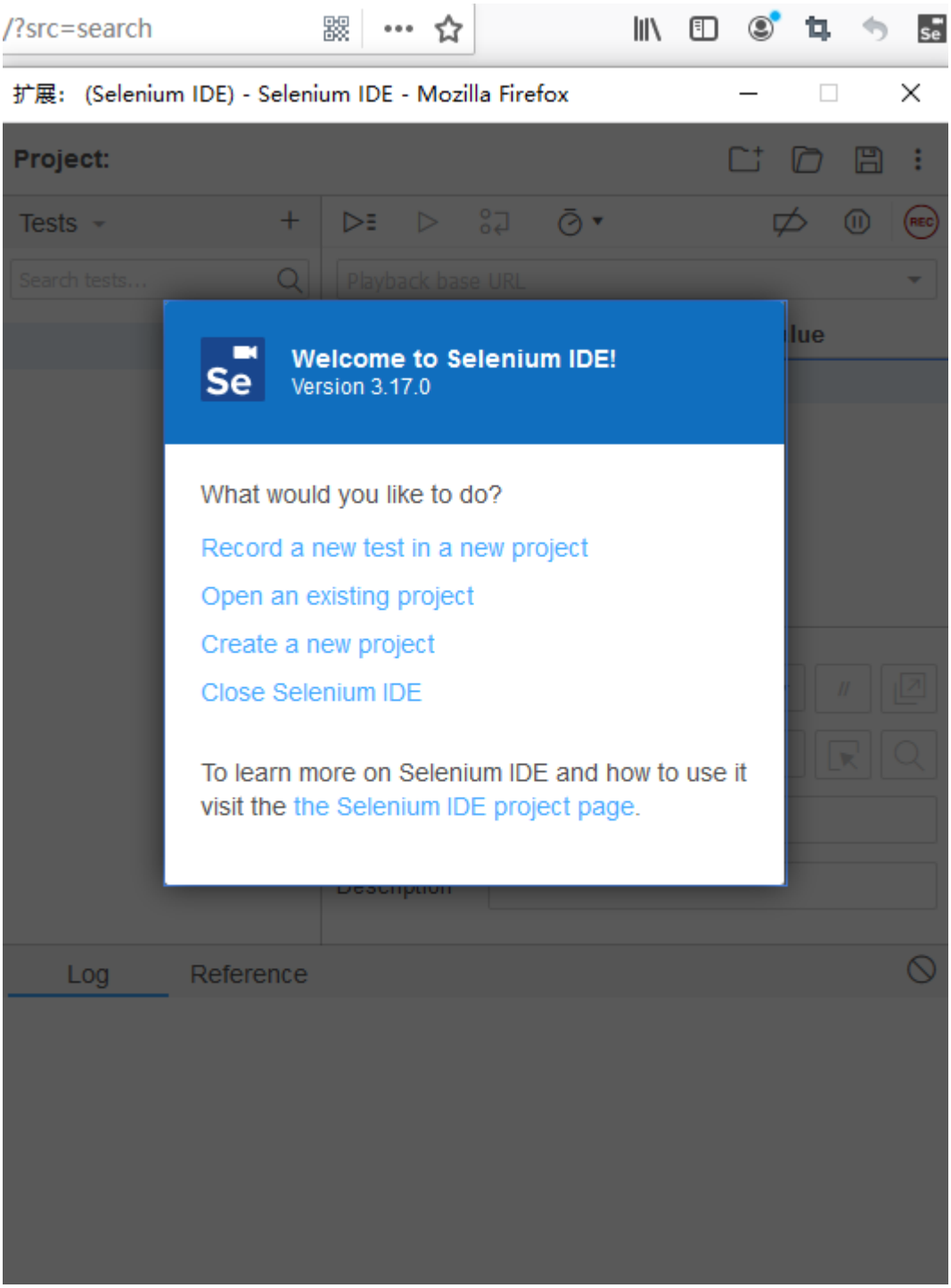
1. Install Selenium, Firefox and SeleniumIDE.

Firstly, create a new java project. Then we add the "client-combined-3.141.59.jar": Right click on the project→Build Path→Configure Build path…→Libraries→Add External JARS→Add "client-combined-3.141.59.jar" and other libs downloaded from official website.



Secondly, we install Firefox from official website.

Thirdly, we install SeleniumIDE. Open Firefox→click right top bottom→Web 开发者→获取更多工具→Search Selenium IDE→Add to Firefox.



2. Recording and Exporting Scripts.

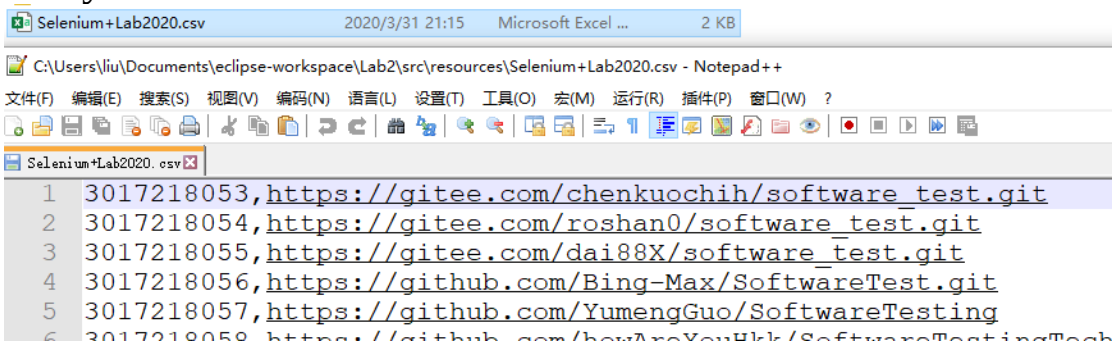
1) Open Selenium IDE.
2) Record a new test in a new project.
3) Enter project name: Lab2.
4) Base URL: http://103.120.226.190/selenium-demo/git-repo.
5) In User Number: Enter something.
6) In Password: Enter something.
7) Click Query.

8) Close browser.
9) Stop Recording. Enter test name: Lab2.
Export as an Java JUnit, which result is as flowering.

```
// Generated by Selenium IDE
import org.junit.Test;
import org.junit.Before;
import org.junit.After;
import static org.junit.Assert.*;
import static org.hamcrest.CoreMatchers.is;
import static org.hamcrest.core.IsNot.not;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.remote.RemoteWebDriver;
import org.openqa.selenium.remote.DesiredCapabilities;
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 org.openqa.selenium.Alert;
import org.openqa.selenium.Keys;
import java.util.*;
import java.net.MalformedURLException;
import java.net.URL;
public class Lab2Test {
  private WebDriver driver;
  private Map<String, Object> vars;
  JavascriptExecutor js;
  @Before
  public void setUp() {
    driver = new FirefoxDriver();
    js = (JavascriptExecutor) driver;
    vars = new HashMap<String, Object>();
  }
  @After
  public void tearDown() {
    driver.quit();
  }
  @Test
  public void lab2() {
    driver.get("http://103.120.226.190/selenium-demo/git-repo");
    driver.findElement(By.name("user_number")).click();
    driver.findElement(By.name("user_number")).sendKeys("admin");
    driver.findElement(By.name("password")).click();
    driver.findElement(By.name("password")).sendKeys("admin");
    driver.findElement(By.cssSelector(".btn")).click();
    driver.close();
  }
}
```

3. Test the Website using Selenium.

Firstly, convert the excel file to a csv file so that we can easily read the file.



Secondly, write code as flowering.

```
package com.tju.lab.lab2;
import org.junit.Test;
import org.junit.Before;
import org.junit.After;
import static org.junit.Assert.*;
import static org.hamcrest.CoreMatchers.is;
import static org.hamcrest.core.IsNot.not;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.remote.RemoteWebDriver;
import org.openqa.selenium.remote.DesiredCapabilities;
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 org.openqa.selenium.Alert;
import org.openqa.selenium.Keys;
import java.util.*;
import java.net.MalformedURLException;
import java.net.URL;
import java.io.*;


public class Lab2 {
    public static void main(String args[]) {
        String    driverPath    =    System.getProperty("user.dir")    +
"\\src\\resources\\driver\\geckodriver.exe";
        System.setProperty("webdriver.gecko.driver", driverPath);
        WebDriver driver = new FirefoxDriver();

        driver.get("http://103.120.226.190/selenium-demo/git-repo");

        String    testListPath    =    System.getProperty("user.dir")    +
"\\src\\resources\\Selenium+Lab2020.csv";
```

```
        try {
            FileReader fileReader = new FileReader(testListPath);
            BufferedReader bufferedReader = new
BufferedReader(fileReader);
            String line;
            while ((line = bufferedReader.readLine()) != null) {
                String context[] = line.split(",");
                WebElement userNumber =
driver.findElement(By.name("user_number"));
                WebElement password =
driver.findElement(By.name("password"));
                WebElement submit =
driver.findElement(By.cssSelector(".btn"));
                userNumber.clear();
                userNumber.sendKeys(context[0]);
                password.clear();
                password.sendKeys(context[1]);
                submit.click();
                String result = driver.findElement(By.cssSelector(".mb-
2:nth-child(6) > code")).getText();
                assert(result == context[1]);
            }
            bufferedReader.close();
            fileReader.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
        driver.close();
    }
}
```

The result is as flowering.