Ryanair Automation Task –Booking

Used IDE:

* Eclipse Luna
* FireFox Browser
* Java 1.8

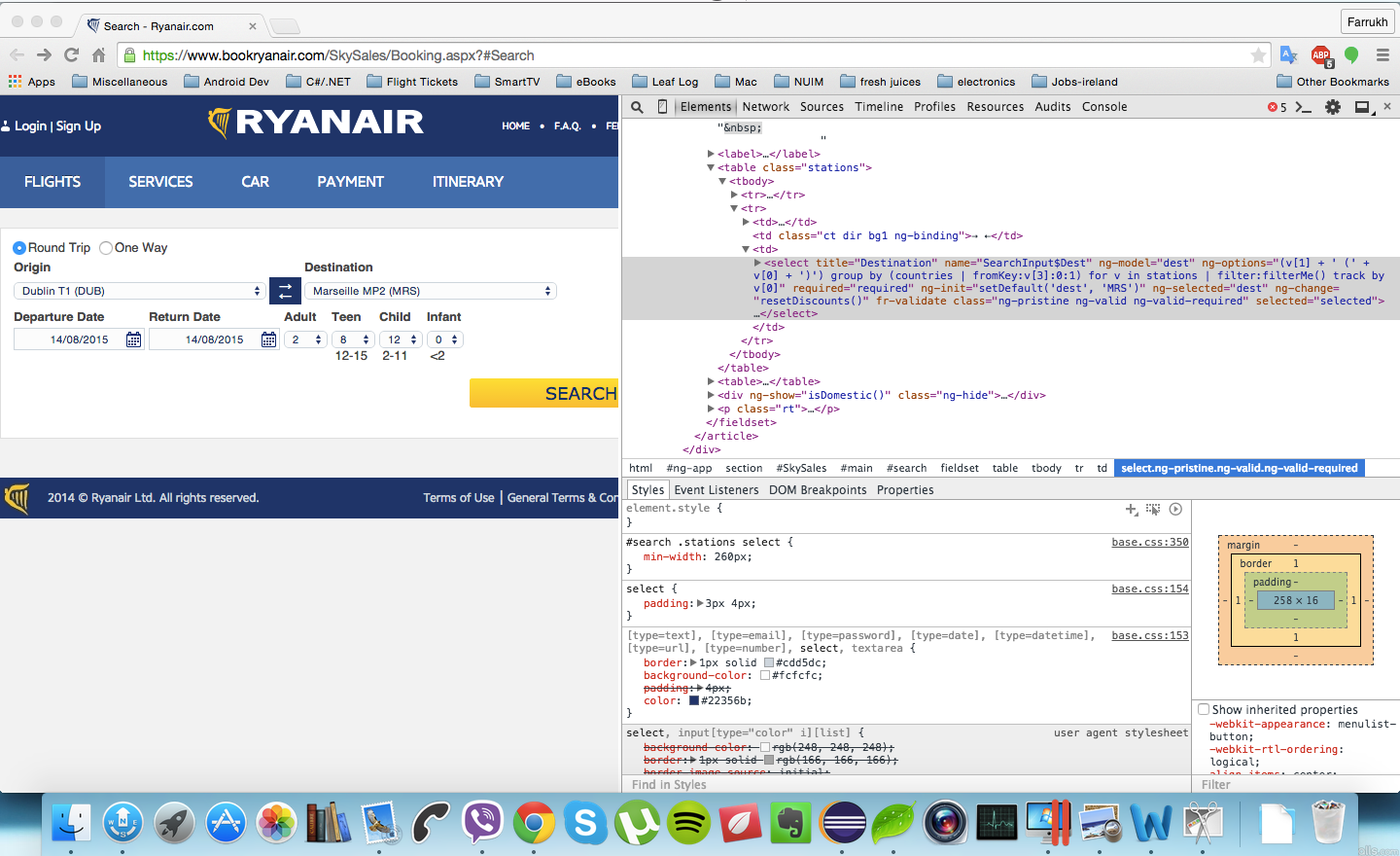
Used External Libraries:

* selenium-server-standalone-2.46.0.jar
* selenium-java-2.46.0.zip Library
* jsoup-1.8.1.jar (Instead of Using Maven Project)

Used Testing Tools:

* JUnit/TestNG
* Selenuim WebDriver

Before Writing a method, which will be sending values to web elements, I used Chrome’s Inspect tool to find names/ids of the web elements.



To Test this web site I created a method which receives 8 parameters of type String (Orig, Destination, Departure date,return date, amouts for Adults, Teenagers, Childrens and Infants).

In the body of the method I am using Firefox webdriver to find elements. After finding element (ComboBox, date, button)

sending values to them from local method’s variables.

*public void test\_booking(****String Orig, String Dest, String DeptDate, String RetDate,***

***String Adult, String Teen, String Child, String Infant****){*

*//Looking for Elements By their Name & Passing Values them.*

*driver.findElement( By.name("SearchInput$Orig")).sendKeys(Orig);*

*driver.findElement( By.name("SearchInput$Dest")).sendKeys(Dest);*

*//DEPARTURE and RETURN ELEMENTS*

*driver.findElement( By.name("SearchInput$DeptDate")).clear();*

*driver.findElement( By.name("SearchInput$DeptDate")).sendKeys(DeptDate);*

*driver.findElement( By.name("SearchInput$RetDate")).clear();*

*driver.findElement( By.name("SearchInput$RetDate")).sendKeys(RetDate);*

*//ADULT, TEEN, CHILD, INFANT ELEMENTS (Comboboxes)*

*Select adult = new Select(driver.findElement( By.name("SearchInput$PaxTypeADT")));*

*adult.selectByValue(Adult);*

*Select teen = new Select(driver.findElement( By.name("SearchInput$PaxTypeTEEN")));*

*teen.selectByValue(Teen);*

*Select child = new Select(driver.findElement( By.name("SearchInput$PaxTypeCHD")));*

*child.selectByValue(Child);*

*Select inf = new Select(driver.findElement( By.name("SearchInput$PaxTypeINFANT")));*

*inf.selectByValue(Infant);*

*//Click() on SEARCH Button*

*driver.findElement( By.name("SearchInput$ButtonSubmit")).click();*

*driver.findElement( By.name("SearchInput$ButtonSubmit")).click();*

*//wait untill next page shows*

*wait.until(ExpectedConditions.presenceOfElementLocated(By.className("flr")));*

*}*

After, I am CALLing this method MULTIPLE TIMES with different TEST CASES. Each method Call contains different variant of INPUT DATA.

Using JUnit’s @Test operator I am testing them. To Run Tests see on pic below:

//TEST CASES/DATA

*@Test*

*public void test1() {*

*test\_booking( "DUB", "BRE", "11/06/2015", "16/06/2015", "1","2","3","1");*

*}*

*@Test*

*public void test2() {*

*test\_booking( "DUB", "BRE", "11/06/2015", "16/06/2015", "1","2","3","1");*

*}*

*@Test*

*public void test3() {*

*test\_booking( "DUB", "BRE", "11/06/2015", "16/06/2015", "1","2","3","1");*

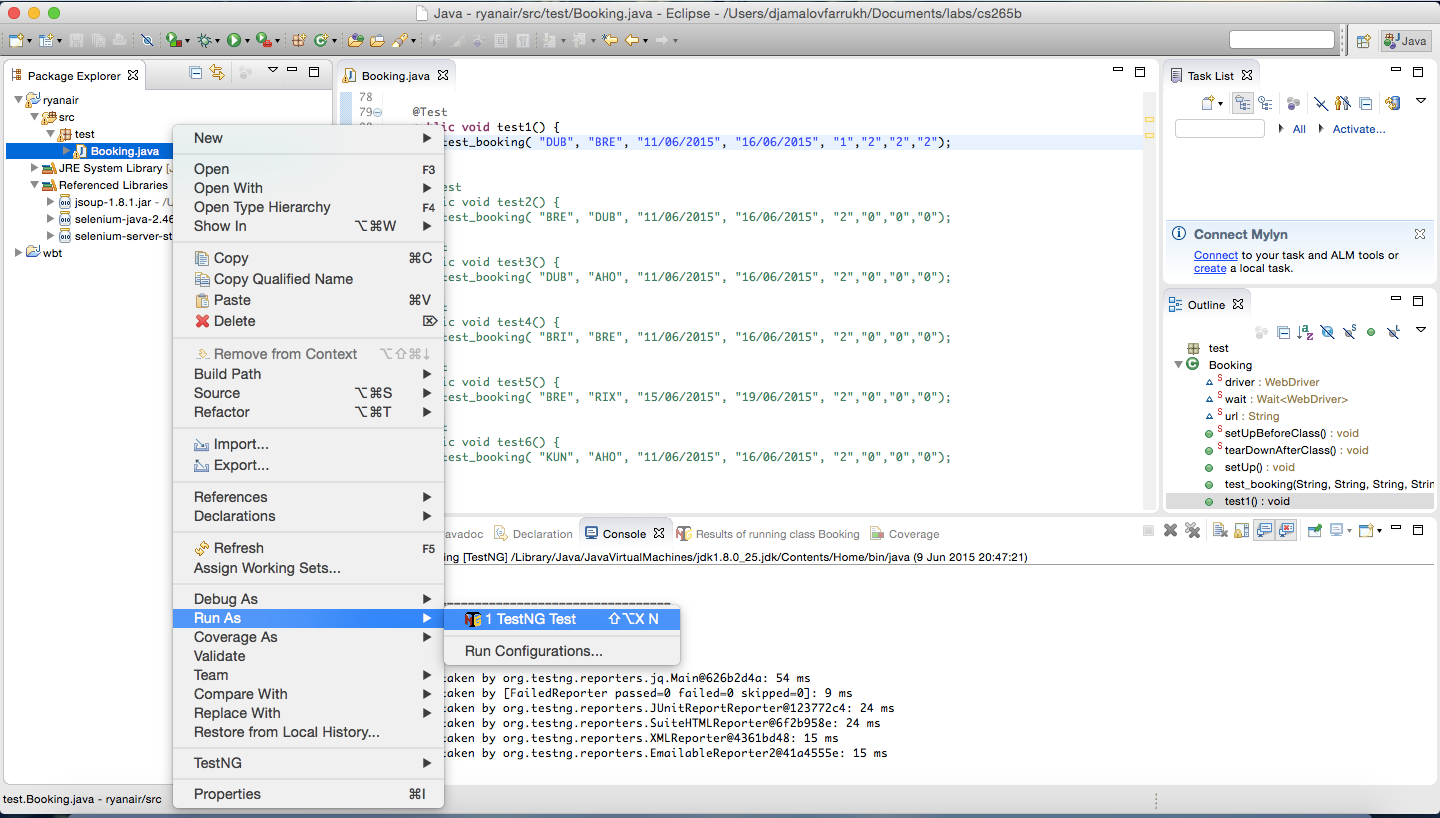
*}*

*@Test*

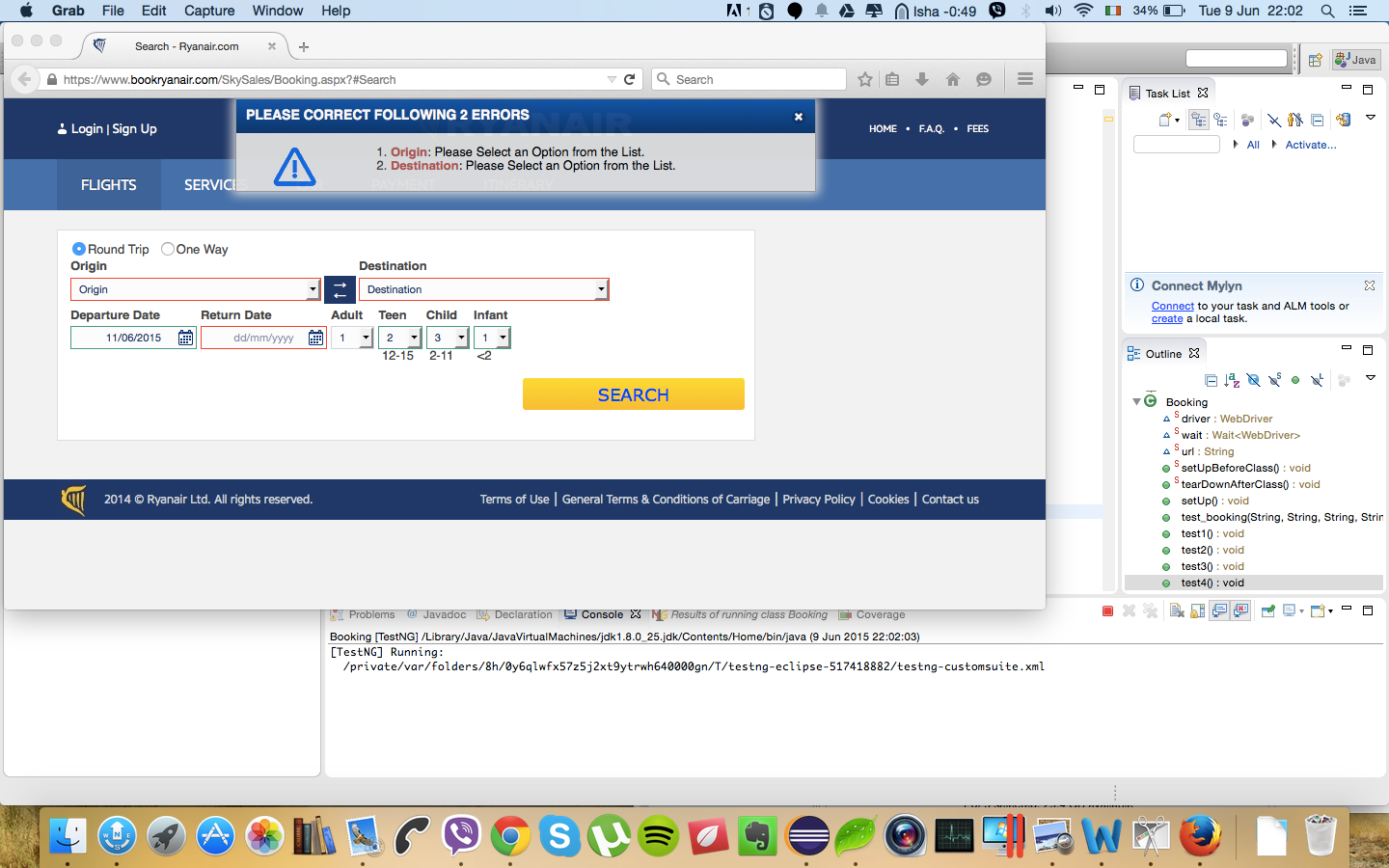
*public void test4() {*

*test\_booking( "DUB", "BRE", "11/06/2015", "16/06/2015", "1","2","3","1");*

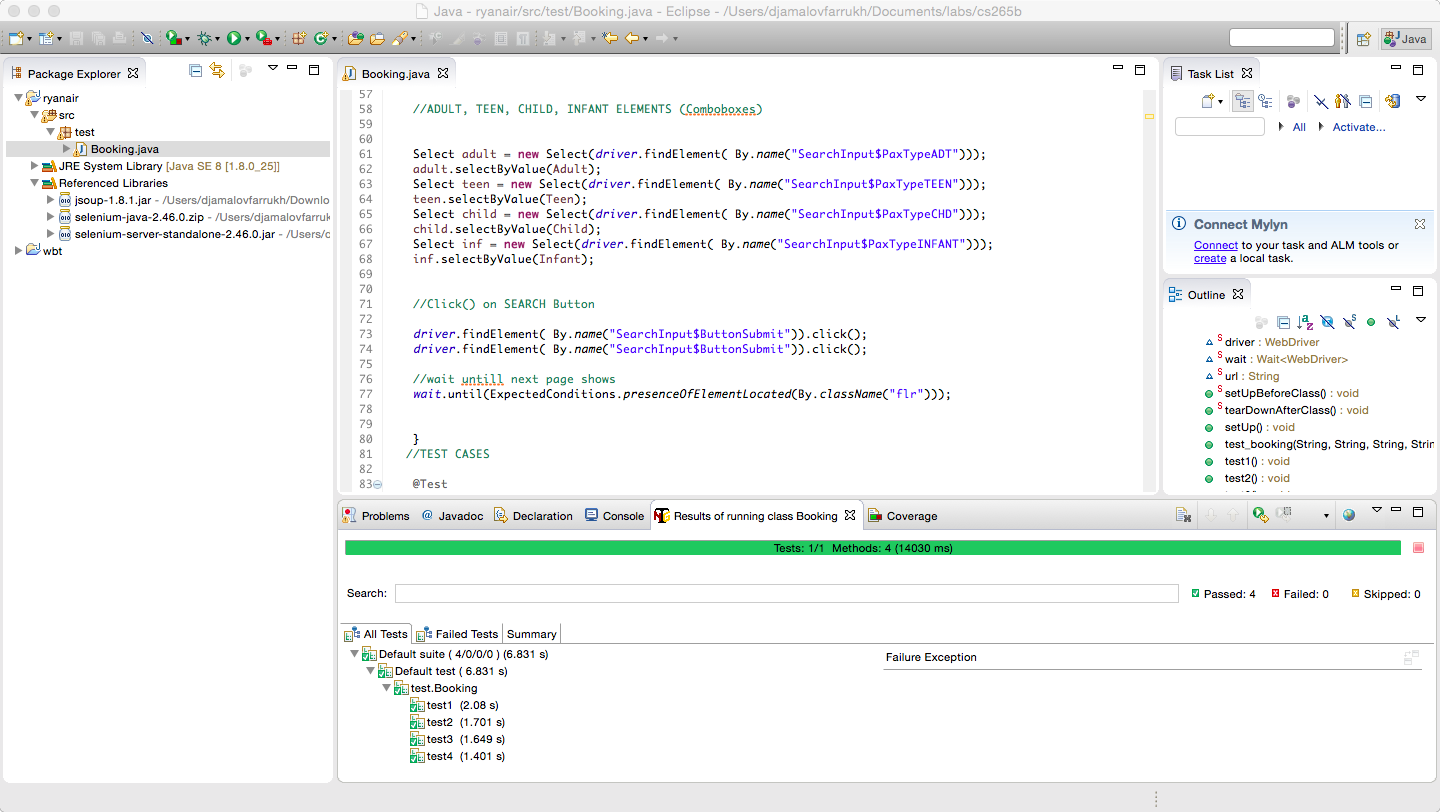
*}*



webdriver opens firefox and runs the TEST DATA, see below.



After cycling through the Test Cases Eclipse (JUnit/TestNG) shows that all tests passed.see below.



In this Task we can reproduce as many TEST DATA/CASES as we want.

ONLY thing is that RYANAIR SERVER Responds some times with errors. However it depends on tickets availabilities and etc.

Below on the page I am providing a full source code for this task.

public class Booking {

static WebDriver driver;

static Wait<WebDriver> wait;

static String url="https://www.bookryanair.com/SkySales/Booking.aspx?#Search";//RYANAIR URL

@BeforeClass

public static void setUpBeforeClass() throws Exception {

// setup selenium test

driver = new FirefoxDriver();

wait = new WebDriverWait(driver, 30);

}

@AfterClass

public static void tearDownAfterClass() throws Exception {

driver.close();

driver.quit();

}

@BeforeMethod

public void setUp() throws Exception {

driver.get(url);

// wait until displayed

wait.until(ExpectedConditions.presenceOfElementLocated(By.id("SearchInput\_ButtonSubmit")));

}

public void test\_booking(String Orig, String Dest, String DeptDate, String RetDate,

String Adult, String Teen, String Child, String Infant){

//Looking for Elements By their Name & Passing Values them.

driver.findElement( By.name("SearchInput$Orig")).sendKeys(Orig);

driver.findElement( By.name("SearchInput$Dest")).sendKeys(Dest);

//DEPARTURE and RETURN ELEMENTS

driver.findElement( By.name("SearchInput$DeptDate")).clear();

driver.findElement( By.name("SearchInput$DeptDate")).sendKeys(DeptDate);

driver.findElement( By.name("SearchInput$RetDate")).clear();

driver.findElement( By.name("SearchInput$RetDate")).sendKeys(RetDate);

//ADULT, TEEN, CHILD, INFANT ELEMENTS (Comboboxes)

Select adult = new Select(driver.findElement( By.name("SearchInput$PaxTypeADT")));

adult.selectByValue(Adult);

Select teen = new Select(driver.findElement( By.name("SearchInput$PaxTypeTEEN")));

teen.selectByValue(Teen);

Select child = new Select(driver.findElement( By.name("SearchInput$PaxTypeCHD")));

child.selectByValue(Child);

Select inf = new Select(driver.findElement( By.name("SearchInput$PaxTypeINFANT")));

inf.selectByValue(Infant);

//Click() on SEARCH Button

driver.findElement( By.name("SearchInput$ButtonSubmit")).click();

driver.findElement( By.name("SearchInput$ButtonSubmit")).click();

//wait untill next page shows

wait.until(ExpectedConditions.presenceOfElementLocated(By.className("flr")));

}

//TEST CASES

@Test

public void test1() {

test\_booking( "DUB", "BRE", "11/06/2015", "16/06/2015", "1","2","3","1");

}

@Test

public void test2() {

test\_booking( "DUB", "BRE", "11/06/2015", "16/06/2015", "1","2","3","1");

}

@Test

public void test3() {

test\_booking( "DUB", "BRE", "11/06/2015", "16/06/2015", "1","2","3","1");

}

@Test

public void test4() {

test\_booking( "DUB", "BRE", "11/06/2015", "16/06/2015", "1","2","3","1");

}

}