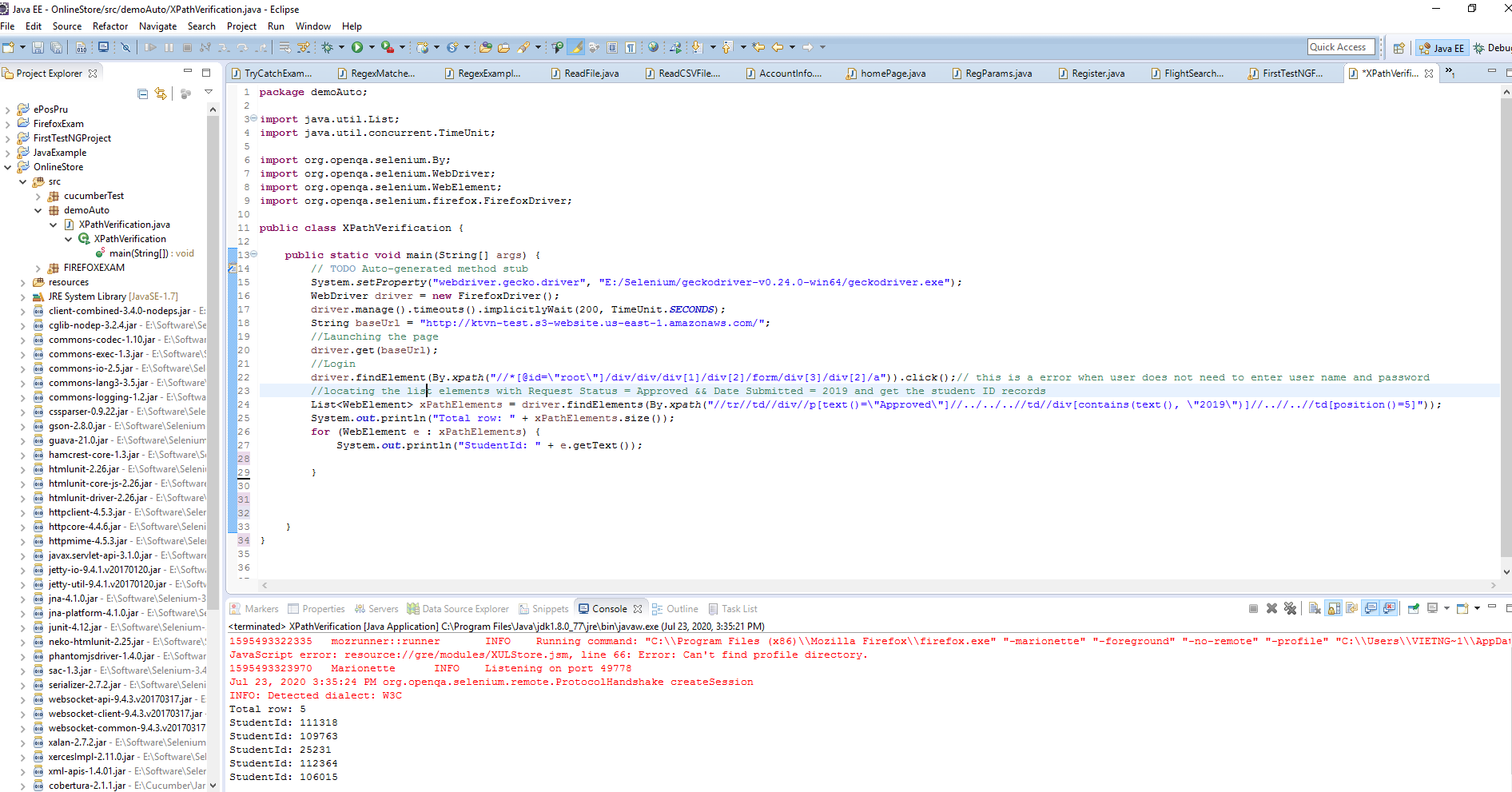
# - Provide the ONE XPath to return the requests which are submitted and approved in 2019 (Request Status = Approved && Date Submitted = 2019)

***//tr//td//div//p[text()=\"Approved\"]//../../..//td//div[contains(text(), \"2019\")]***

Simple code to test the one xPath:



**package** demoAuto;

**import** java.util.List;

**import** java.util.concurrent.TimeUnit;

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.WebElement;

**import** org.openqa.selenium.firefox.FirefoxDriver;

**public** **class** XPathVerification {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

System.*setProperty*("webdriver.gecko.driver", "E:/Selenium/geckodriver-v0.24.0-win64/geckodriver.exe");

WebDriver driver = **new** FirefoxDriver();

driver.manage().timeouts().implicitlyWait(200, TimeUnit.***SECONDS***);

String baseUrl = "http://ktvn-test.s3-website.us-east-1.amazonaws.com/";

//Launching the page

driver.get(baseUrl);

//Login

driver.findElement(By.*xpath*("//\*[@id=\"root\"]/div/div/div[1]/div[2]/form/div[3]/div[2]/a")).click();// this is a error when user does not need to enter user name and password

//locating the list elements with Request Status = Approved && Date Submitted = 2019 and get the student ID records

List<WebElement> xPathElements = driver.findElements(By.*xpath*("//tr//td//div//p[text()=\"Approved\"]//../../..//td//div[contains(text(), \"2019\")]//..//..//td[position()=5]"));

System.***out***.println("Total row: " + xPathElements.size());

**for** (WebElement e : xPathElements) {

System.***out***.println("StudentId: " + e.getText());

}

}

}

# Write the automation script to verify for

+TC1: Verify filter Student Access Request with INACTIVE

+ TC2: Verify sorting of First Name column

+ TC3: The API endpoint - GET https://my-json-server.typicode.com/typicode/demo/posts/1

For TC1 and TC2, please refer to Github link:

For TC3, not sure how to write automation script to test it but the idea is to verify the status code and the body response. Below is simple test using Postman:

