

## Week 12: Homework 2: Project: Online Jukebox + DOS attack using Selenium JUnit/WebDriver + Python Web Server

- Continue the previous [Online Jukebox + DOS attack using Selenium IDE + Python Web Server](#) task
- Replacing "DOS attack using **Selenium IDE**" with "DOS attack using **Selenium JUnit/WebDriver**".
- The DOS attack must be done on a command line using [Cron Job](#)

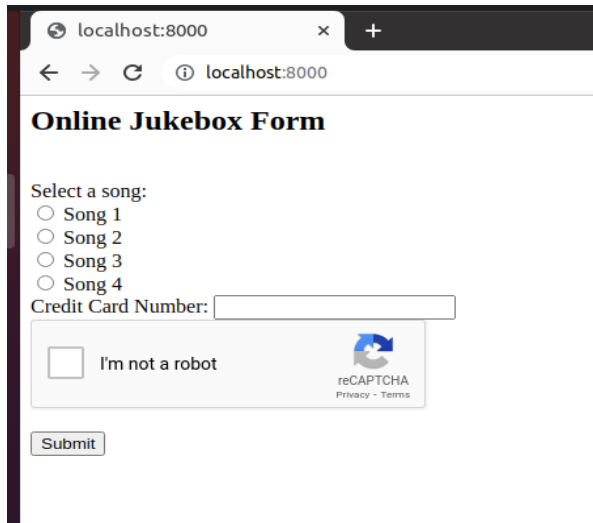
**Jukebox code will be available in my GitHub account**

<https://github.com/santhinagalla/Software-Quality-Assurance-and-Test-Automation/tree/main/Jukebox>

Steps to run Code -

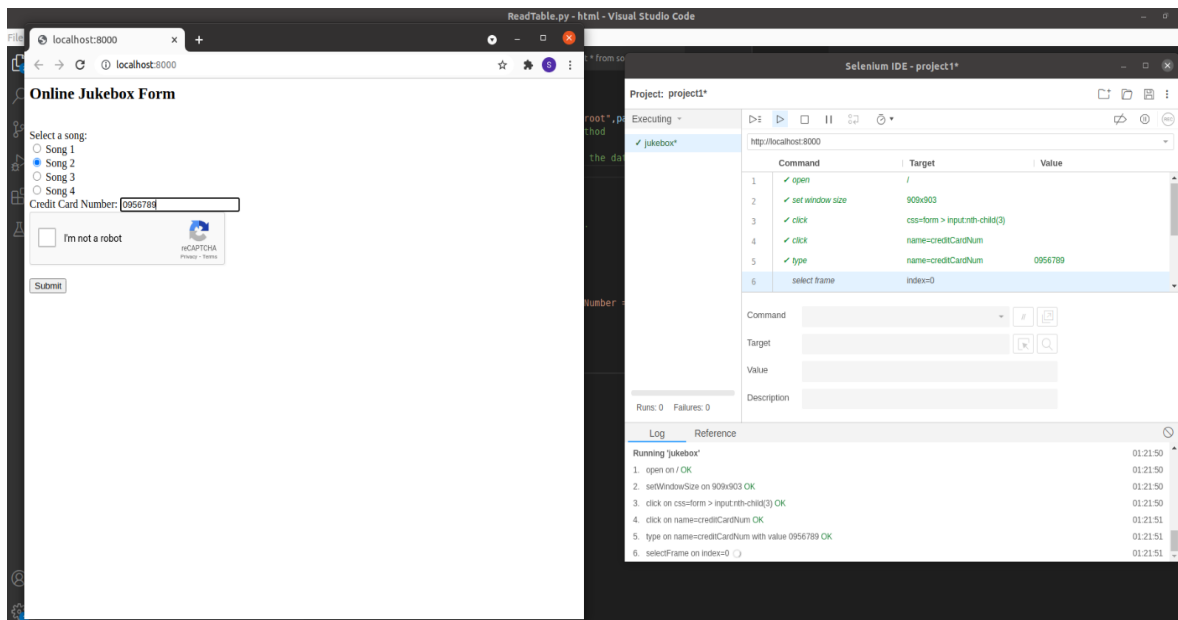
- Make sure the mysql server up and running
- And start python3 server using below command.  
**python3 -m http.server --bind 127.0.0.1 --cgi 8000**
- Save and change the mode of the Files using command Ex - `chmod 755 dos.sh`
- Go to <https://www.google.com/recaptcha/admin/create> to create a reCAPTCHA to prevent the attack.
- Copy the client-side key and use in index.html.
- Copy the Server-side key and use in songlist.py.
- Launch the browser and Open `http://localhost:8000/` to see Online Jukebox form.

The browser looks like below -

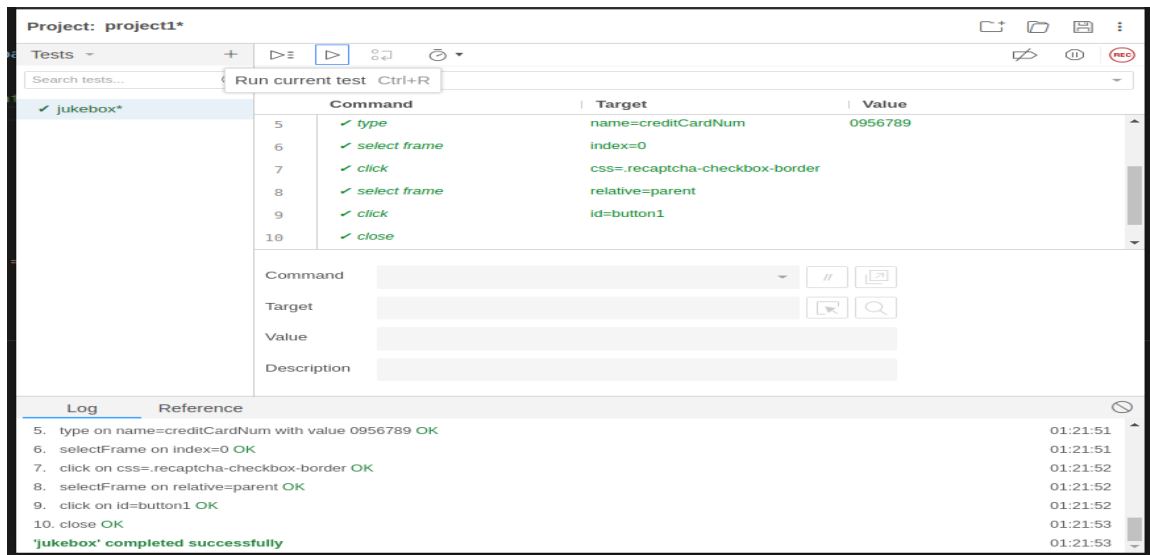


## DOS attack using Selenium IDE

1. Open IDE and record the steps by using the Base URL - **http://localhost:8000/** to test Online Jukebox form and select song and enter credit card number then select the simple captcha checkbox then click on submit button.
2. Once you are done the recording, stop the recording and playback and check whether the test case running successfully.



4. Check all the steps run successfully.



5. See whether the data is saving in the database by using command “Python3 ReadTable.py”

```
sans@sans-vb:~/Desktop/html$ python3 ReadTable.py
songName = song1, Credit Card Number = 123456789
songName = song2, Credit Card Number = 987654321
songName = song2, Credit Card Number = 0956789
songName = song2, Credit Card Number = 0956789
songName = song2, Credit Card Number = 0956789
```

## DOS attack using Selenium Web driver

1. Write testcases using “Selenium JUnit/WebDriver” to access the Jukebox form or export the code in “Java Junit” using Selenium IDE.
2. Run the Selenium Web driver code in Command line.

### Selenium web driver code

```
import org.junit.Test;
import org.junit.Before;
import org.junit.After;
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.chrome.ChromeDriverService;
```

```

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 Jukeboxselenium {
    private WebDriver driver;
    private Map<String, Object> vars;
    JavascriptExecutor js;
    @Before
    public void setUp() {

System.setProperty(ChromeDriverService.CHROME_DRIVER_EXE_PROPERTY, "/home/sans/Downloads/jar/chromedriver");

        driver = new ChromeDriver();
        js = (JavascriptExecutor) driver;
        vars = new HashMap<String, Object>();
    }
    @After

```

```
public void tearDown() {
    driver.quit();
}

@Test
public void test1() {
    try{
        driver.get("http://localhost:8000/");
        driver.manage().window().setSize(new Dimension(909, 903));
        driver.findElement(By.name("song")).click();
        driver.findElement(By.name("creditCardNum")).click();
        driver.findElement(By.name("creditCardNum")).sendKeys("111111123");
        driver.switchTo().frame(0);
        driver.findElement(By.cssSelector(".recaptcha-checkbox-border")).click();
        driver.switchTo().defaultContent();
        driver.findElement(By.id("button1")).click();

        try{
            driver.switchTo().alert().dismiss();
        } catch (org.openqa.selenium.NoAlertPresentException e) {
        }

        System.out.println(driver.findElement(By.tagName("h2")).getText());
        driver.close();
    } catch (Exception e){
    }
    finally{
        driver.quit();
    }
}
```

```
}  
  
}  
  
}
```

2.To run the above code in command prompt need below Jar files.

3.Download the Jar files before execute the code in command line.

- **Selenium Standalone server** - <https://selenium-release.storage.googleapis.com/index.html?path=3.141/selenium-server-standalone-3.141.59.jar>
- **Chrome Driver** - <https://chromedriver.chromium.org/downloads>
- **junit-4.11** - <http://www.java2s.com/Code/Jar/j/Downloadjunit411jar.htm>
- **hamcrest-core-1.3** - <http://www.java2s.com/Code/Jar/h/Downloadhamcrestcore13jar.htm>

4.Compile the jukebox code using below syntax with jar files.

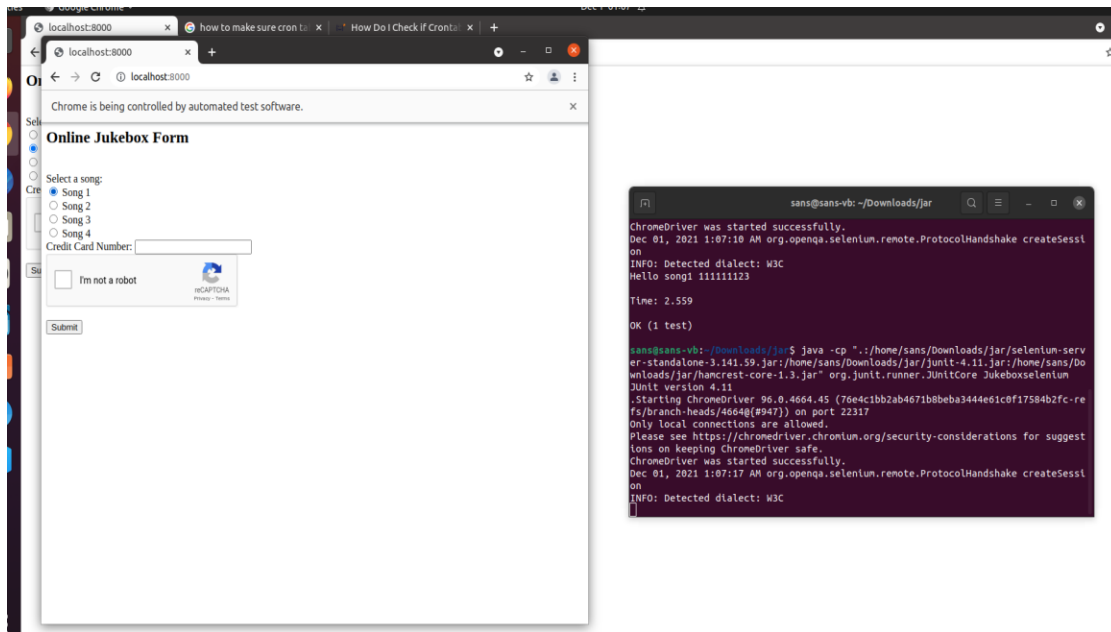
.class will be generated.

```
javac -cp ".:./home/sans/Downloads/jar/selenium-server-standalone-3.141.59.jar:./home/sans/Downloads/jar/junit-4.11.jar:./home/sans/Downloads/jar/hamcrest-core-1.3.jar" -d . Jukeboxselenium.java
```

5. Now run the .class file using below syntax.

```
java -cp ".:./home/sans/Downloads/jar/selenium-server-standalone-3.141.59.jar:./home/sans/Downloads/jar/junit-4.11.jar:./home/sans/Downloads/jar/hamcrest-core-1.3.jar" org.junit.runner.JUnitCore Jukeboxselenium
```

6. After running the above commands browser will launch and execute the steps.



7. Output will be printed in the terminal and test passed.

```
sans@sans-vb: ~/Downloads/jar

Time: 2.559

OK (1 test)

sans@sans-vb:~/Downloads/jar$ java -cp ".:~/home/sans/Downloads/jar/selenium-server-standalone-3.141.59.jar:/home/sans/Downloads/jar/junit-4.11.jar:/home/sans/Downloads/jar/hamcrest-core-1.3.jar" org.junit.runner.JUnitCore Jukeboxselenium
JUnit version 4.11
Starting ChromeDriver 96.0.4664.45 (76e4c1bb2ab4671b8beba3444e61c0f17584b2fc-re
fs/branch-heads/4664@#947)) on port 22317
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggest
ions on keeping ChromeDriver safe.
ChromeDriver was started successfully.
Dec 01, 2021 1:07:17 AM org.openqa.selenium.remote.ProtocolHandshake createSessi
on
INFO: Detected dialect: W3C
Hello song1 111111123

Time: 2.745

OK (1 test)

sans@sans-vb:~/Downloads/jar$
```

8. write a shell script to invoke your java program with the necessary arguments.  
Example: - Juke.sh

9.Copy the code in Juke.sh file.

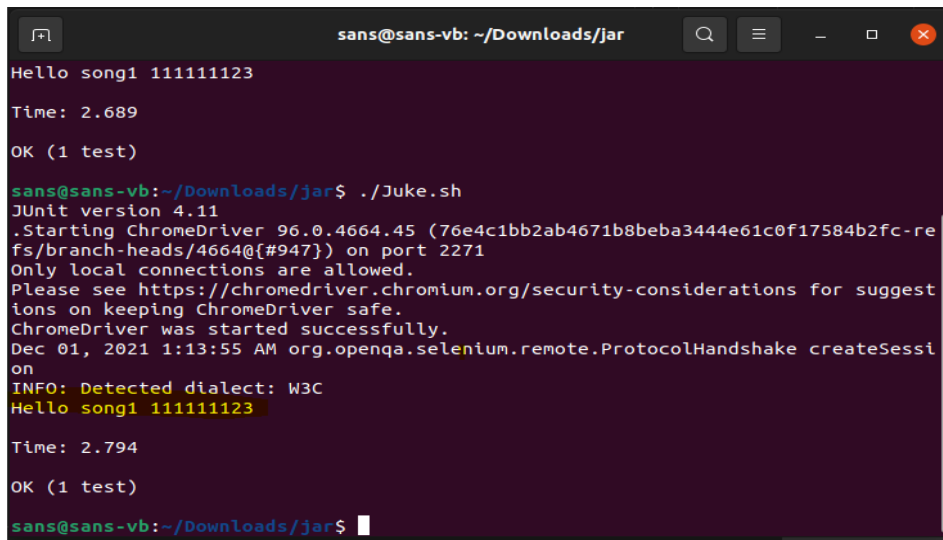
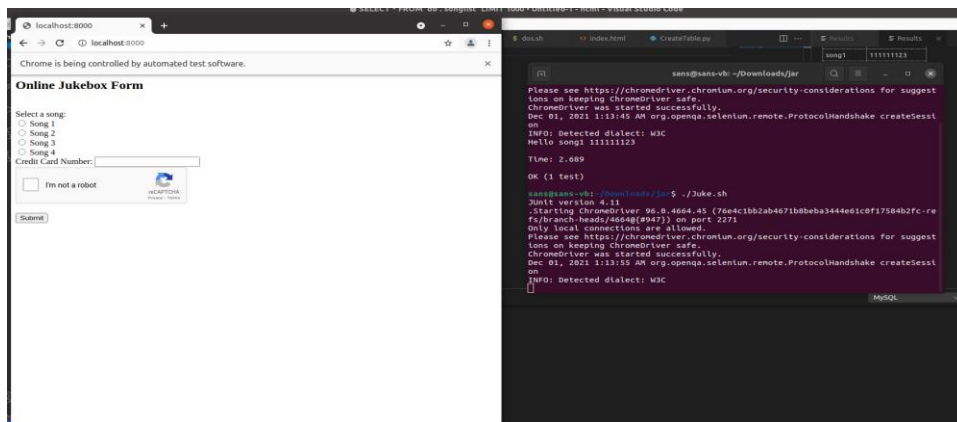
**#!/bin/bash**

**cd /home/sans/Downloads/jar**

```
#javac -cp ".:~/home/sans/Downloads/jar/selenium-server-standalone-3.141.59.jar:~/home/sans/Downloads/jar/junit-4.11.jar:~/home/sans/Downloads/jar/hamcrest-core-1.3.jar" -d . Jukeboxselenium.java
```

```
java -cp ".:~/home/sans/Downloads/jar/selenium-server-standalone-3.141.59.jar:~/home/sans/Downloads/jar/junit-4.11.jar:~/home/sans/Downloads/jar/hamcrest-core-1.3.jar" org.junit.runner.JUnitCore Jukeboxselenium
```

10.Run the command on terminal - **./Juke.sh**



## Cron job

- Schedule the script to be invoked by setting up a cron job.
- Run from a terminal: **crontab -e**
- This will open your crontab editor. You can add a job in this way:



```
MAILTO="sans.nagalla@gmail.com"
```

```
*/5 * * * * bash /home/sans/Downloads/jar/Juke.sh
```

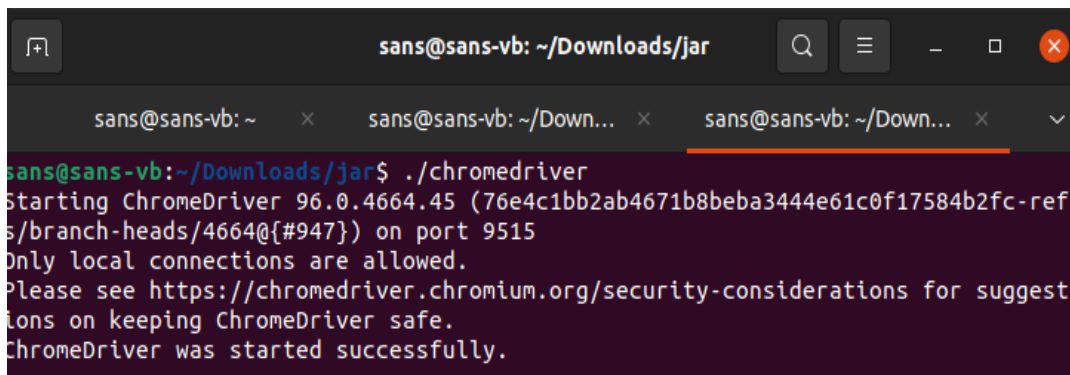
- This job is set to run every 5 minutes.

**I have made few changes in code for cron job to launch browser using RemoteWebdriver.**

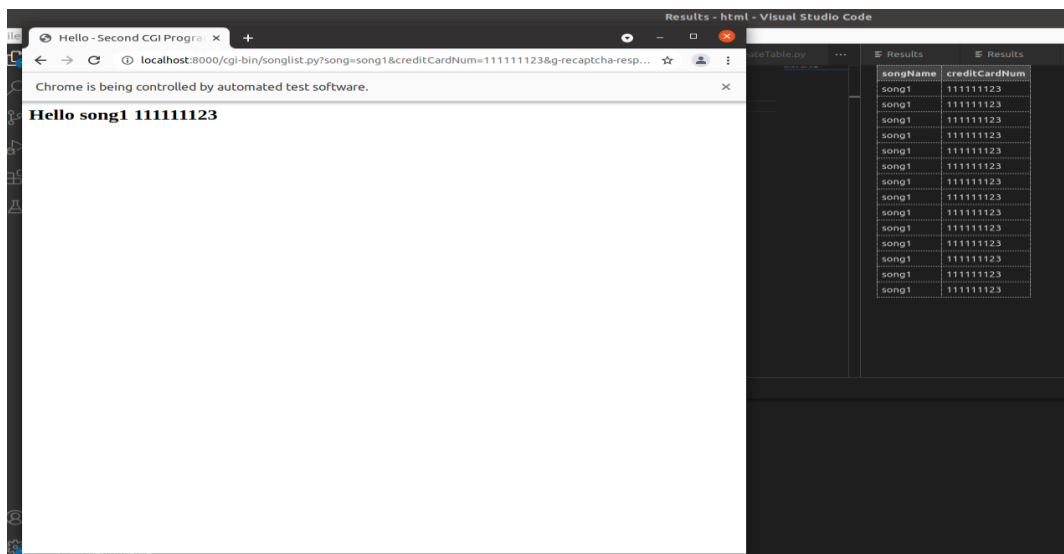
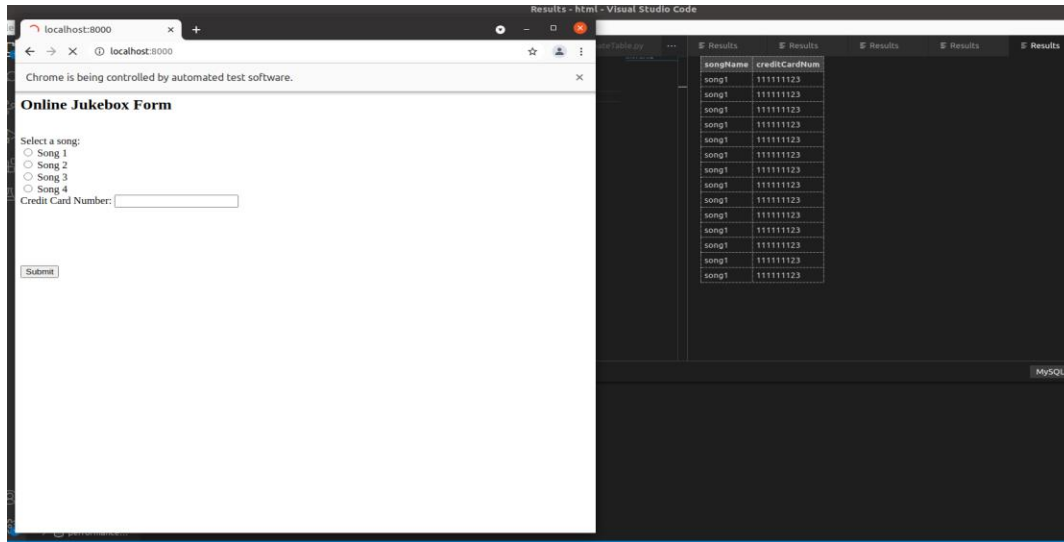
@Before

```
public void setUp() throws MalformedURLException{  
    //  
    System.setProperty(ChromeDriverService.CHROME_DRIVER_EXE_PROPERTY,  
        "/home/sans/Downloads/jar/chromedriver");  
  
    ChromeOptions options = new ChromeOptions();  
  
    driver = new RemoteWebDriver(new URL("http://localhost:9515"), options);  
  
    js = (JavascriptExecutor) driver;  
  
    vars = new HashMap<String, Object>();  
  
}
```

**Run chrome driver in the terminal.**

A screenshot of a terminal window with a dark background. The title bar shows 'sans@sans-vb: ~/Downloads/jar'. The terminal content shows the command './chromedriver' being executed. The output text is: 'Starting ChromeDriver 96.0.4664.45 (76e4c1bb2ab4671b8beba3444e61c0f17584b2fc-ref/s/branch-heads/4664@{#947}) on port 9515', 'Only local connections are allowed.', 'Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.', and 'ChromeDriver was started successfully.'

And let's wait for cron job to launch the browser and check the data is storing in the database.



Check the mail received after each cron job.

Remove cron job once you are done using Command **crontab -r**.

