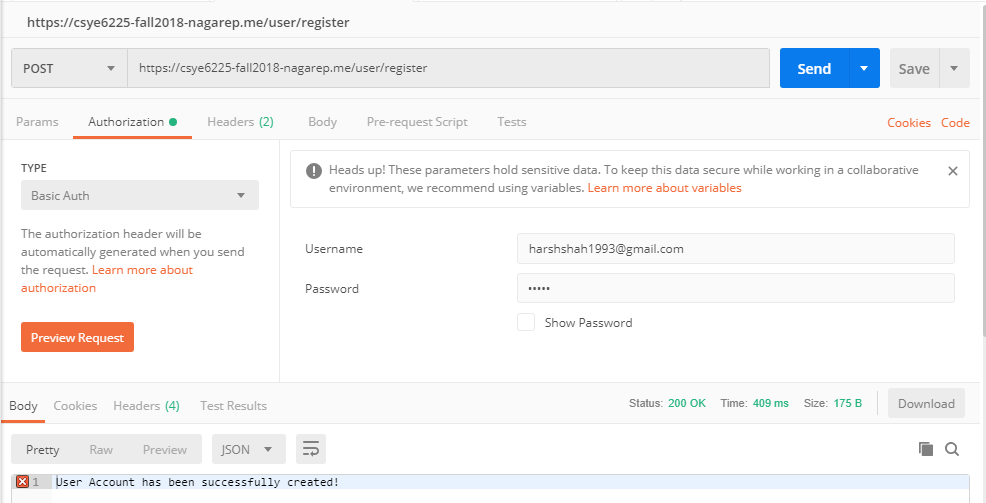
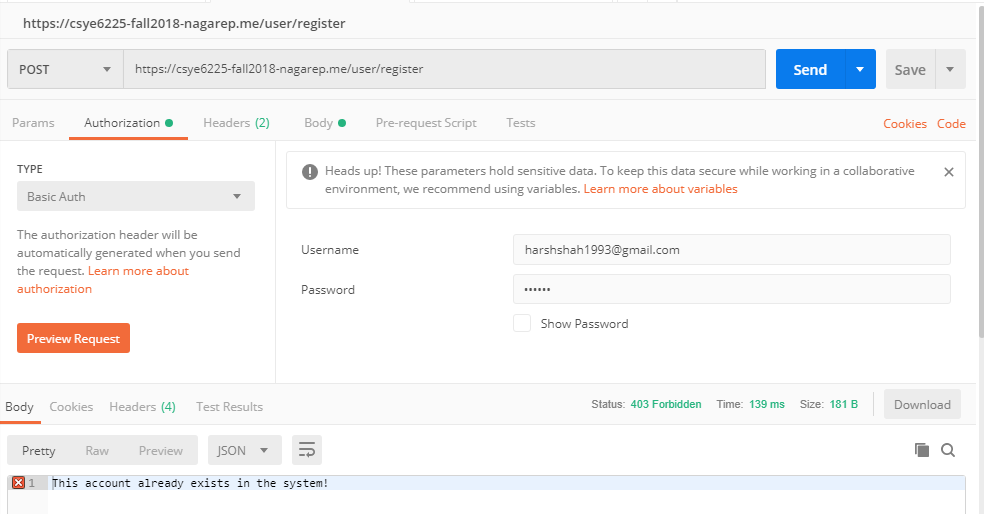
Accessing Webapplication using domain name:

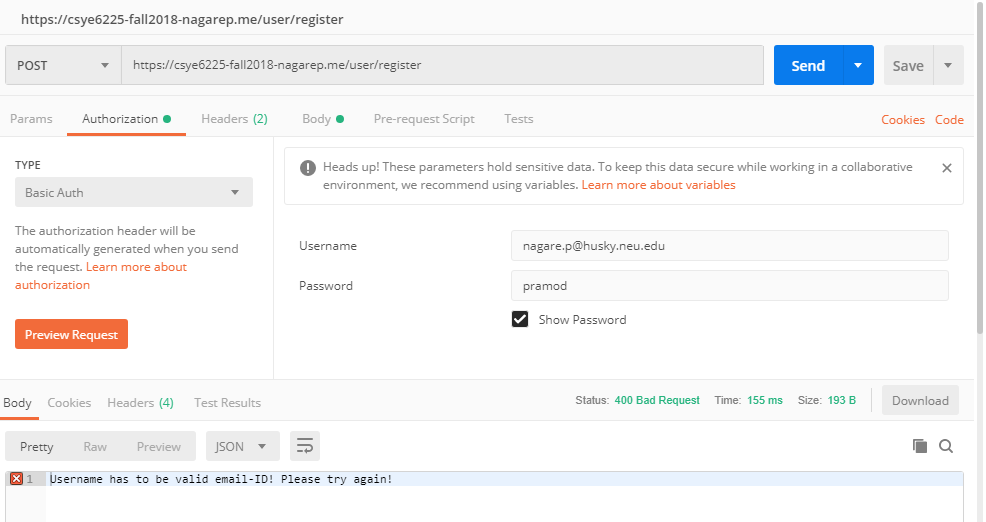
Ex: <https://csye6225-fall2018-nagarep.me/user/register>



**Trying to register the same user again:**

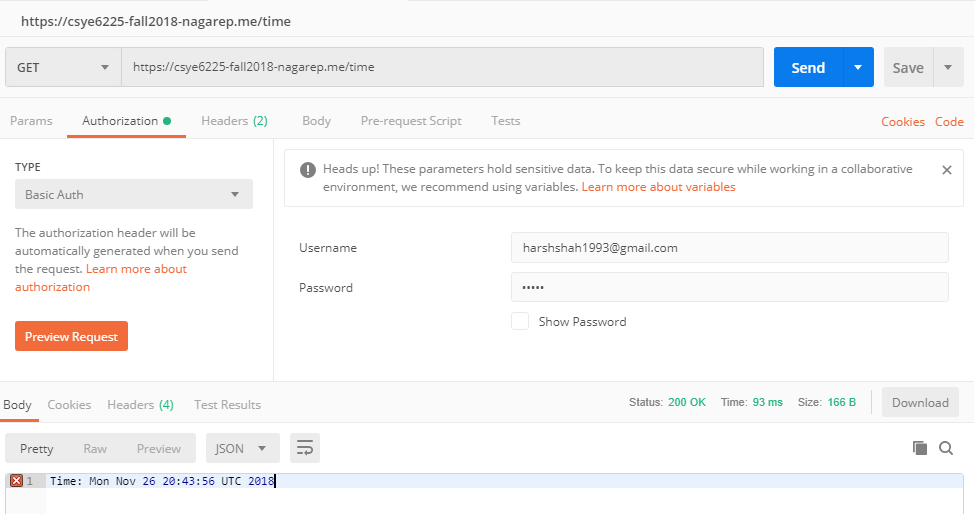


Trying to create user with .neu.edu domain ID.



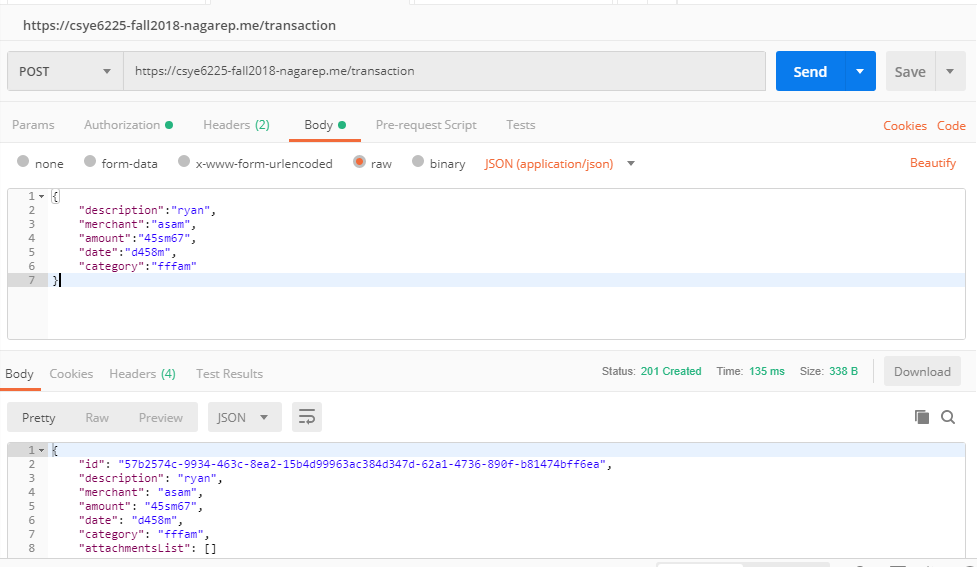
Getting current time:

<https://csye6225-fall2018-nagarep.me/time>

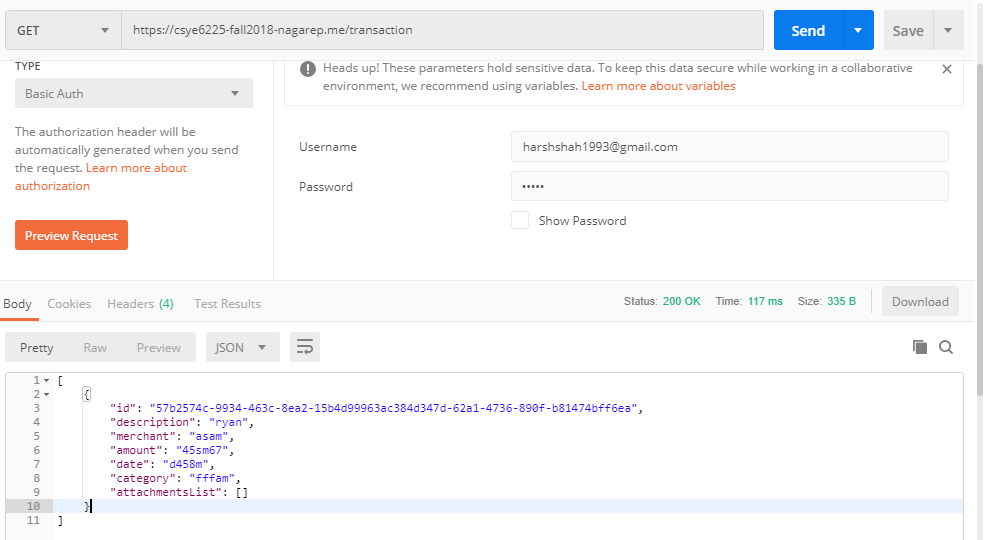


Creating Transaction:

<https://csye6225-fall2018-nagarep.me/transaction>

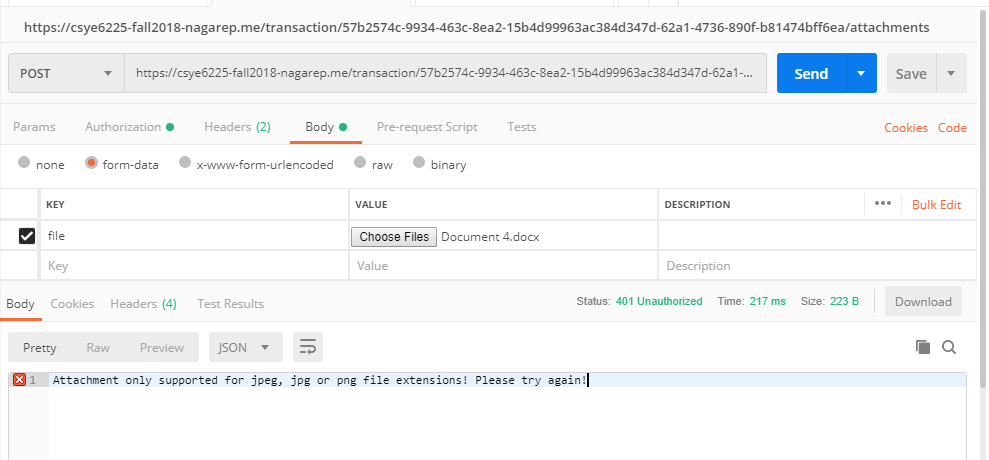


Getting all transactions:

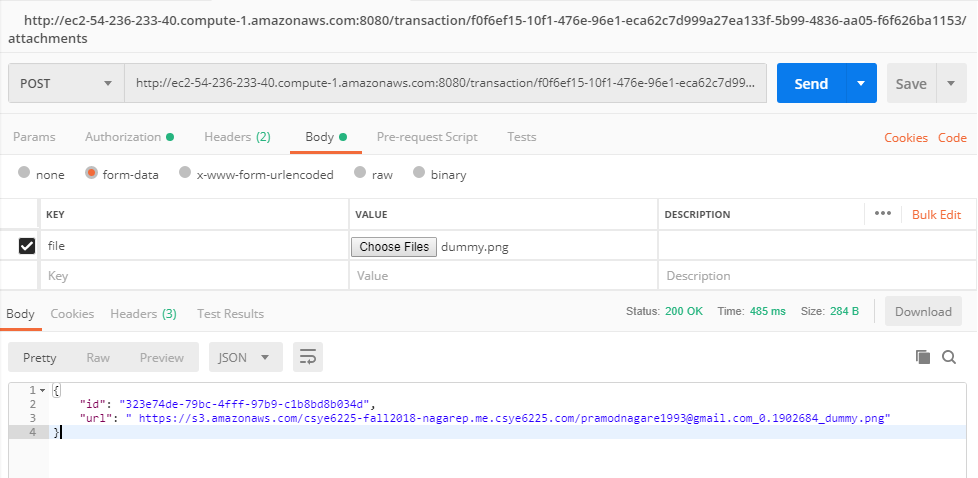


Attachment for the transaction:

1. Other than png or jpeg files:

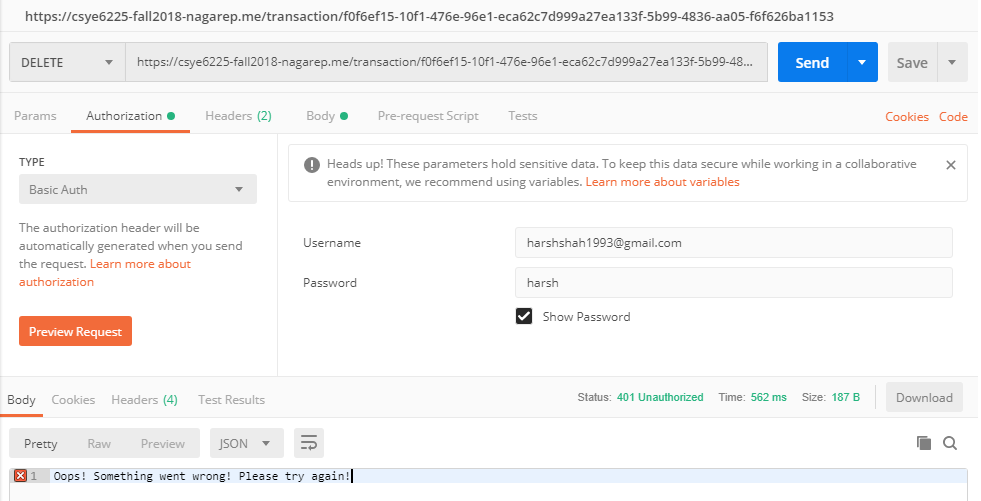


1. With the supported file formats.

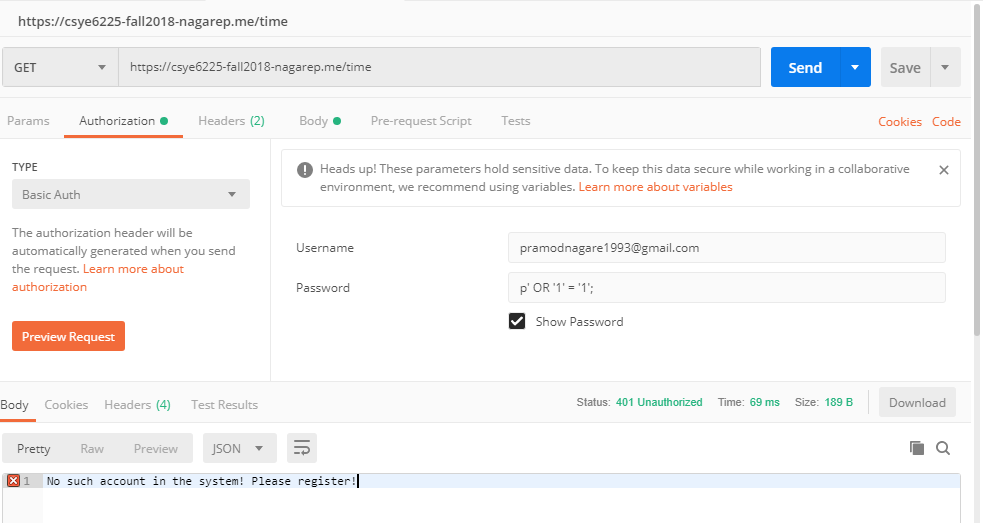


1. Cross user interface:

User [harshshah1993@gmail.com](mailto:harshshah1993@gmail.com) trying to delete transaction for user pramodnagare1993@gmail.com

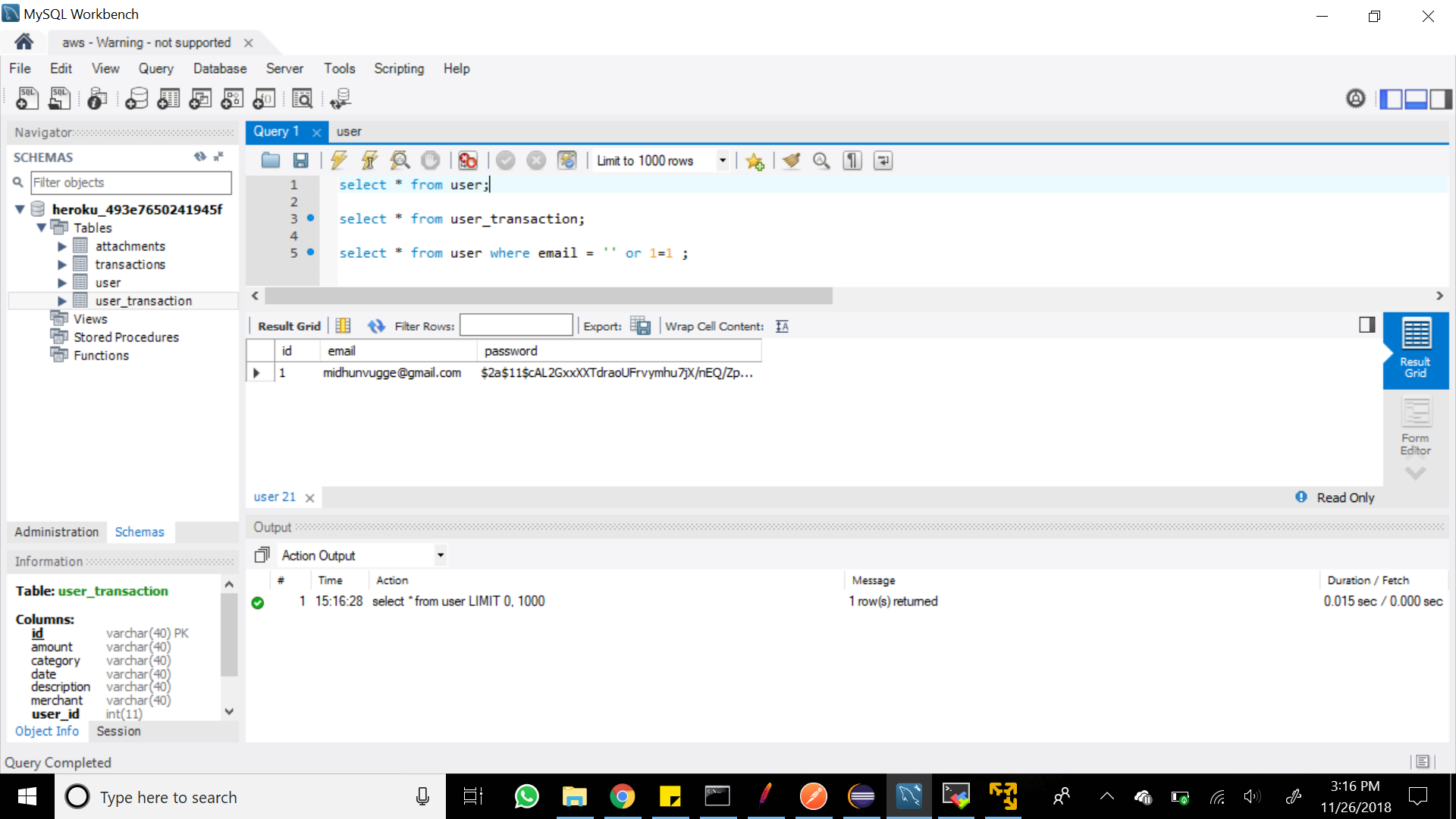


1. SQL Injection:

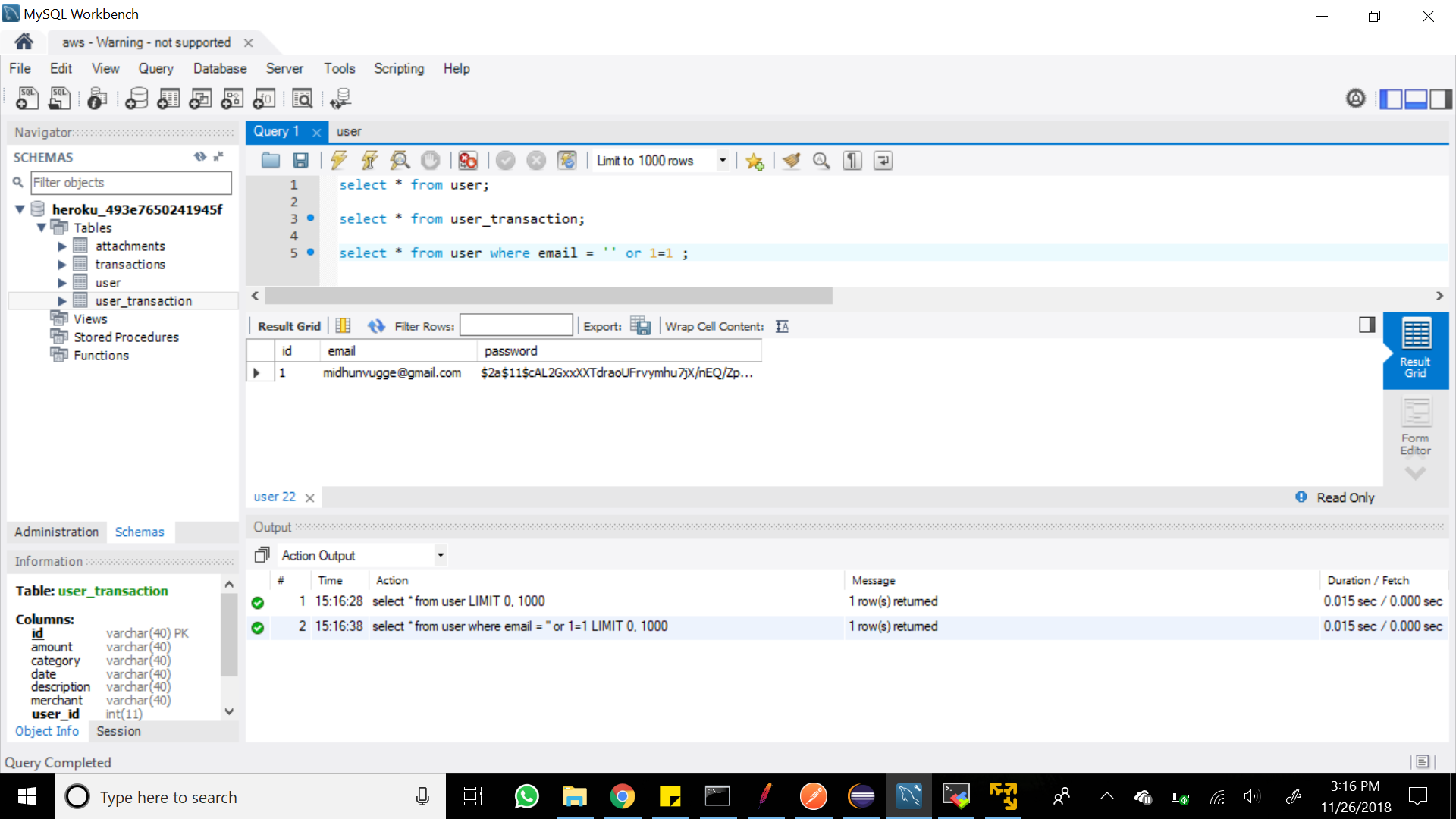


**Attack Vector: SQL Injection**

It is a code injection technique for data driven applications in which we test the username and password authentication while in place of username or password when given 1=1(which is always true) the code which is vulnerable to attacks displays the result as when authenticated username and password is passed.



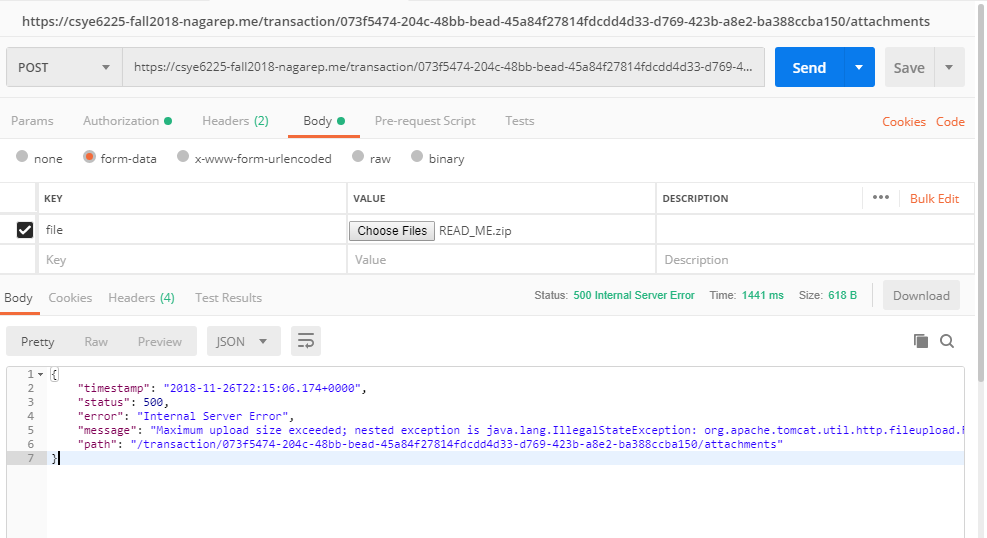
**Test with ‘OR 1=1’ field:**



SQL Injection helps test the vulnerability of the code. Attackers can access the data base even without knowledge of our entries. Since our web application has excessive usage of sql data base with user and user\_transaction tables, it is logical and useful to test the code for SQL Injection.

1. Attachment size:

Attachment is failing, because the max limit is 1048576 Bytes or 1MB.



**Attack Vector: Attachment maximum size**

Attachment maximum size vulnerability test is done to verify set restrictions on the files being uploaded to our S3 bucket. In this test we pass heavy files and in a well written code, these files should not be posted as it induces considerable latency on our instance.

We are testing ‘attachment post’ in our code. An attacker can pass a large file to the S3 bucket deliberately and can induce considerable delays to our instance and can also crash the instance as well. To prevent this, we can restrict the maximum size of the files being posted into our S3 bucket.

XSS Penetration: Cross site scripting:

