# Lab 2 - SQL Injection (Part I)

Ryan Foster

CIS 483-01

## 1. Subverting application logic

* Take a screenshot of the outcome like below. (Do your own work!)
* Graphical user interface, application, website

  Description automatically generated
* Graphical user interface, application, website

  Description automatically generated

## 2. Retrieving data from other database tables

### 2.1 Lab: SQL injection UNION attack, determining the number of columns returned by the query

* Take a screenshot of the outcome like below. (Do your own work!) There are three columns using three NULL statements
* Graphical user interface, text, application, website

  Description automatically generated

### 2.2 Lab: SQL injection UNION attack, finding a column containing text

* Take a screenshot of the outcome.

Graphical user interface, text, application, website

Description automatically generated

### 2.3 Lab: SQL injection UNION attack, retrieving data from other tables

* Take a screenshot of the outcome like below.
* Graphical user interface, text, application

  Description automatically generatedGraphical user interface, website

  Description automatically generated

## 3. Examining the database in SQL injection attacks

### 3.1 Querying the database type and version

* Take a screenshot of the outcome.
* **So I tried this one but it seemed not to work for me. I tried to use the ‘—’ at the end of the url to comment out and it didn’t work. I also used the “#” to try to comment out the rest of the query. I’m not sure what else to try for this one. My final url was to go to the accessories tab and input web-security-academy.net' UNION select @@version, NULL#. Returned a internal error so im not sure what to do after. I input my screenshots to show good faith effort.**
* Graphical user interface, text

  Description automatically generated

Graphical user interface, text

Description automatically generated