



# SQL INJECTION

## Lesson Description:

Students will learn about SQL injection and how it fits into the cybersecurity concepts of confidentiality, integrity, defense in depth, and thinking like an adversary. Before modeling SQL injection with the students, we will review truth tables, and/or statements, and inputs.

## Prerequisite Knowledge:

Students should know a little bit about databases, and truth tables.

## Length of Completion:

This lesson is for a 40 minute period.

## Level of Instruction:

This lesson is aimed for junior high/high schoolers. It can be a part of an introductory or intermediate course.

## Applicable Concepts:

### GenCyber Cybersecurity Concepts

Availability	X Defense in Depth
X Confidentiality	X Think Like an Adversary
X Integrity	Keep it Simple

## NYS Standards:

9-12.CY.1

Determine the types of personal and organizational information and digital resources that an individual may have access to that need to be protected.

9-12.CY.2



Describe physical, digital, and behavioral safeguards that can be employed to protect the confidentiality, integrity, and accessibility of information.

9-12.CY.5

Recommend multiple actions to take prior and in response to various types of digital security breaches.

### **Resources that are Needed:**

Students will need access to a web browser. This lesson can be used in tandem with curriculum programs such as codeHS. And an article of the first well known SQL injection hack.

### **Accommodations Needed:**

The included videos have closed captioning available.

## **LEARNING OUTCOMES**

### **LESSON LEARNING OUTCOMES**

- Recall and/or statements, truth tables, and user inputs.
- Discuss how user inputs may affect code and explain how SQL injection works.
- Identify the security issues and possible ways of protecting the database.
- Able to use a simple SQL injection on a weak website.

## **LESSON DETAILS**

### **Interconnection:**

This lesson could follow lessons on databases and SQL. This fits nicely in Unit 8 of the Cybersecurity course on codeHS. The lesson can be followed with teaching prevention of SQLi.



## Assessment:

A review of truth tables will be done together in class. This will ensure that students understand why SQL injections work. Students will also be given a chance to use SQL injection on a website. The teacher will walk around and have class discussions to make sure students are understanding.

## Extension Activities:

- Show students other prevention techniques as well as types of SQL injections. Have students do some research on real life SQL injections.
- Show students a video (from codeHS Lesson 8.9) on Security Engineers at Google and discuss a possible career branch for students:
  - <https://www.youtube.com/watch?v=-6ZbrfSRWKc>

## Differentiated Learning Opportunities:

For more of a challenge, there is a list of different types of SQLi as well as different prevention tips.

For struggling learners, focus more on what data the hackers have access to as opposed to how they hack. This could involve more of an ethics based discussion than a why it works discussion.

# LESSON

## Lesson 1 Details:

### Warm Up:

- Hand out or read together as a class the article: "Teenage hacker facing court case for data theft".  
(<https://web.archive.org/web/20090612081311/http://www.taipeitimes.com/News/front/archives/2006/01/22/2003290158>)
- Ask for students' thoughts and lead into a discussion of ethics as well as introducing the topic for today, SQL injection.
  - Was it right for Hung to steal customer information?
  - Should he have received the praise he did?
  - What should the punishment be for hackers?



## Lesson:

- Firstly, review the truth tables with students. One way to do this is make a game of it, have students play tic tac toe where they can only make a move if they answer a question on truth statements correctly. And instead of X's and O's, play with 1's and 0's!
- Show students that you can crack into a database using SQL injection. This can be done via codeHS lesson 8.10 (the example is the last part of the lesson) and/or using the following website.
  - Click on Hacking Activity, it will show you, the teacher, what to do.
    - <https://www.guru99.com/learn-sql-injection-with-practical-example.html#3>
  - Or go directly to the website to hack into using the email xxx@xx.xxx and password: 123 or 1=1
    - <http://www.techpanda.org/index.php>
  - Let students try and then tell them what you did. Ask them to edit existing information or add a new contact.
  - Ask students why this is bad and what information the "hacker" had access to.
- Now it's time to see why it works.
  - Have students go to <https://www.hacksplaining.com/exercises/sql-injection> and work through it on their own.
  - Have a discussion with the class on how it worked.
    - Asking for an input allows us to "inject" new code into the existing code. This is a possible weakness in website design.
    - If the website is weak in security, it is pretty easy to use this loophole.
    - By using a statement that is always true into a code that puts user input directly into a return statement, we are able to "hack" in and have a return of all the information in the database.
- Wrap Up: Prevention.
  - For the example students did in class, there is an easy fix.
    - In the code, don't use the user input directly into the return code.
    - Make it so the user is not allowed to use spaces or quotation marks.



- Other resources:
  - CodeHS Fundamentals of Cybersecurity course, Unit 8 Software Security (there is a free version). The first part of this unit talks about databases so could be used as a prior lesson. The second half introduces SQL injection:  
<https://codehs.com/curriculum/catalog?q=cybersecurity>
  - Explains SQL injection:  
[https://www.w3schools.com/sql/sql\\_injection.asp](https://www.w3schools.com/sql/sql_injection.asp)
  - OWASP prevention: [https://owasp.org/www-community/attacks/SQL\\_Injection](https://owasp.org/www-community/attacks/SQL_Injection)
  - Overview video -  
<https://www.youtube.com/watch?v=Yqu93GXx0vI&t=119s>
  - SQL injection explained:  
<https://www.youtube.com/watch?v=FwIUkAwKzG8>
  - For those that can't use a mock example, here is a youtube video: <https://www.youtube.com/watch?v=Cm0BihCxAjE>
  - Types of Protection:  
<https://www.youtube.com/watch?v=9FeZ7AYMB40>
  - Protection Overview:  
<https://www.mssqltips.com/sqlservertip/3637/protecting-yourself-from-sql-injection-in-sql-server-part-1/>

