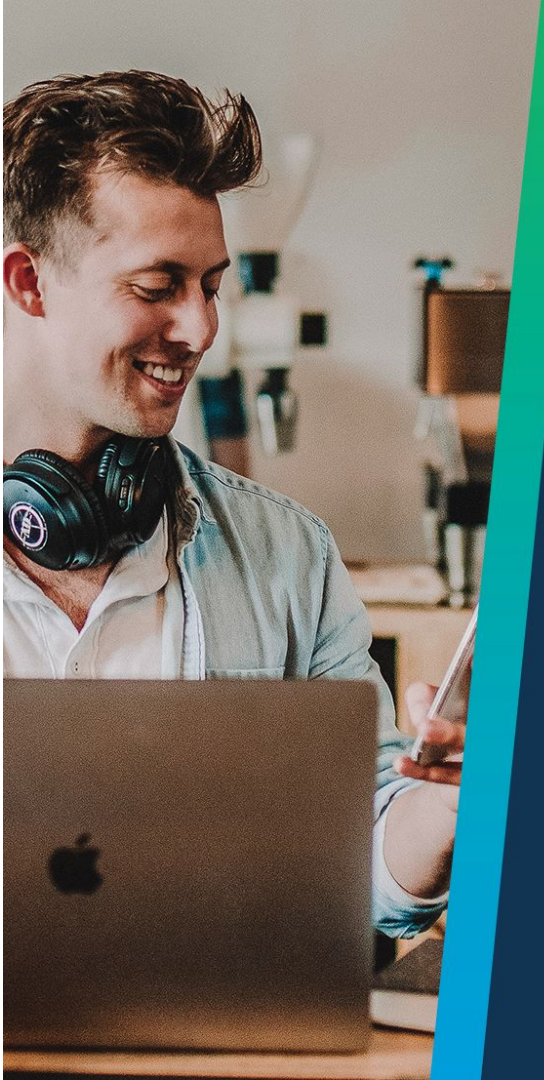


Cyber Security Bootcamp

Hyperiondev

Databases: SQL + SQL injections

Lecture - Housekeeping

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly.
- No question is daft or silly - ask them!
- There are Q/A sessions at the end of the session, should you wish to ask any follow-up questions.
- For all non-academic questions, please submit a query: www.hyperiondev.com/support
- Report a safeguarding incident: http://hyperiondev.com/safeguardreporting

Objectives

1. Define the purpose of SQL in relational databases.
2. Identify and Describe the two main categories of SQL commands: Data Definition Language(DDL) & Data Manipulation Language(DML)
3. Execute basic SQL commands to create and query tables.
4. Identify common SQL injection techniques and attack vectors by analyzing sample SQL injection payloads.
5. Apply mitigation strategies to prevent SQL injection vulnerabilities in SQL queries by implementing best practices such as prepared statements and input validation.

Previously:

Defined what MITM & XSS attacks and vulnerabilities are.

We explored the different types and effectively explored how to mitigate the above.

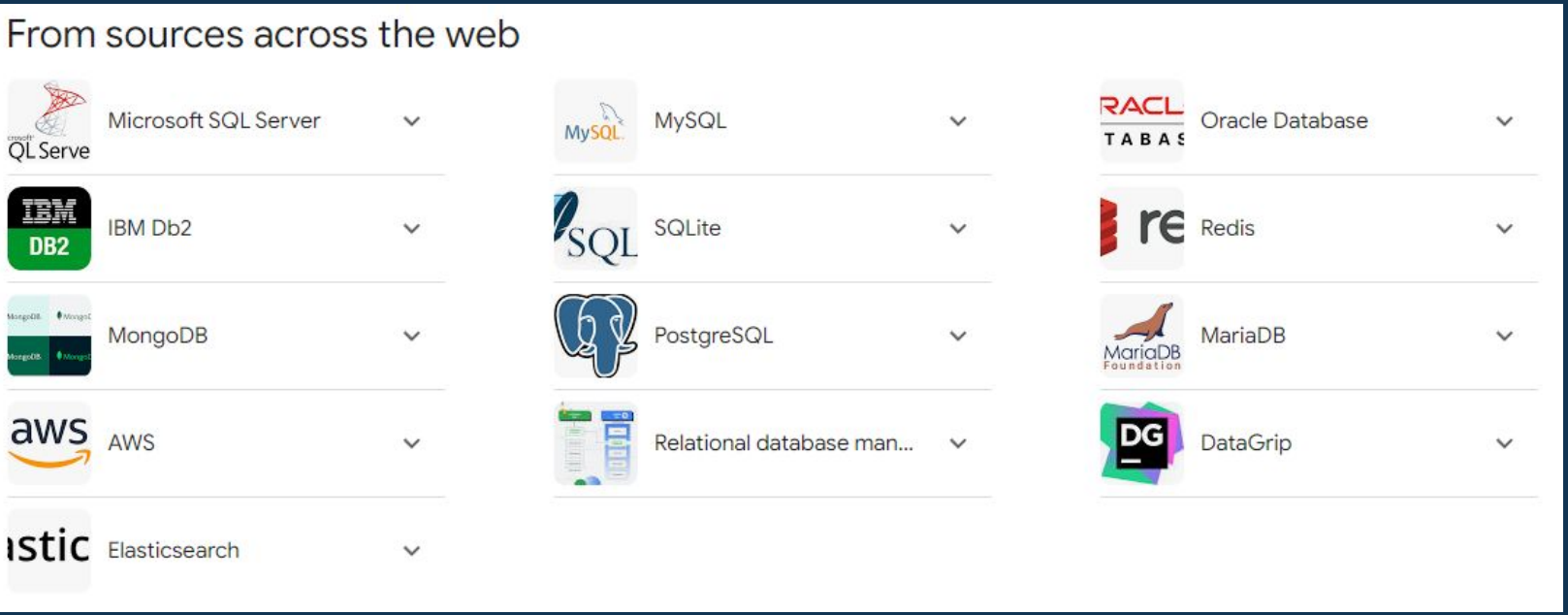
SQL

Introduction to SQL

- **Definition:** SQL (Structured Query Language) is a standardized language used for managing and manipulating relational databases.
- **Purpose:** Create and manage database structures, insert and manipulate data, and query information.
- **Portability:** SQL commands work similarly across different DBMSs, making it a versatile tool for database management.

SQL Platforms

From sources across the web

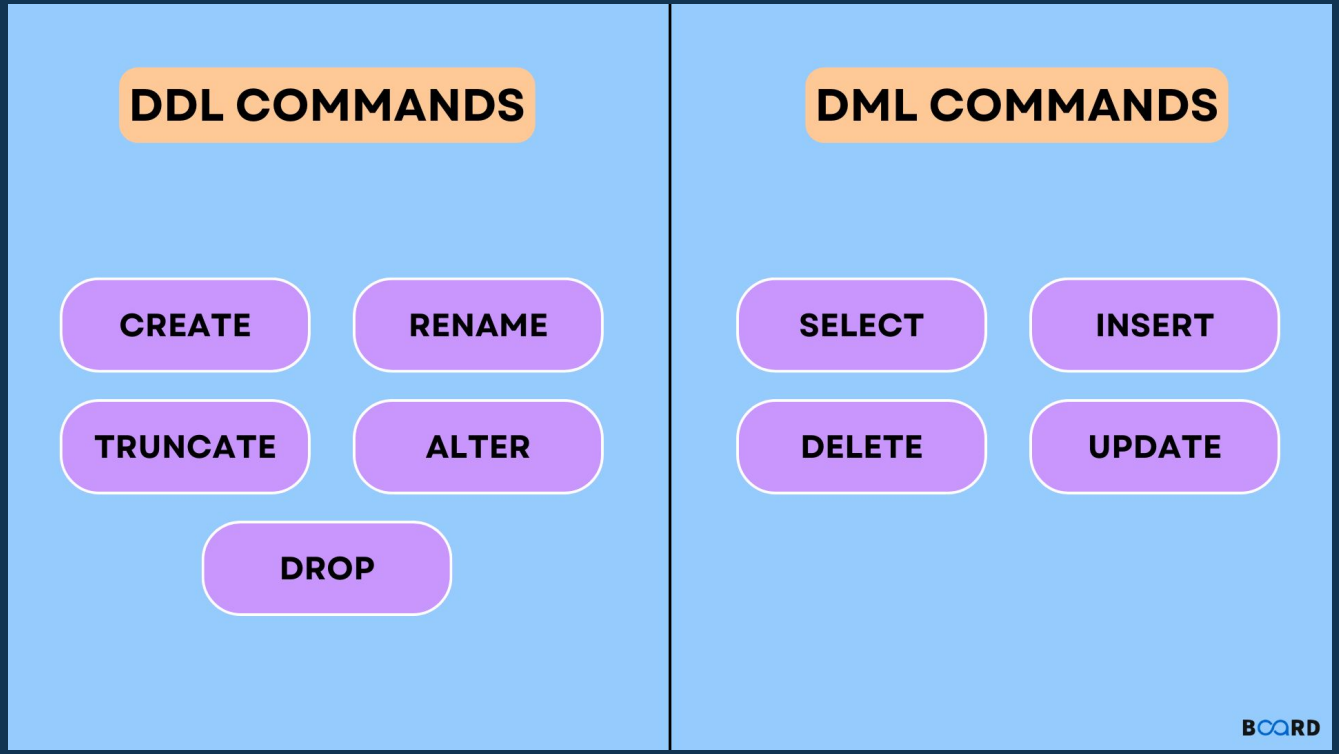


SQL Command Categories

DDL	DML
Data Definition Language	Data Manipulation Language
It affects the entire table	It affects only specific rows of a table
Helps in defining fields/columns in a table	Helps in defining rows or records of a table
Changes are permanent	We can roll back our changes

https://www.boardinfinity.com/blog/content/images/2023/03/DDL-AND-DML-COMMANDS.png

SQL Command Categories



Hyperiondev

https://www.boardinfinity.com/blog/content/images/2023/03/DDL-AND-DML-COMMANDS.png

SQL Syntax

Tables

Employee

EmployeeID (int)	LastName	FirstName	Address	Number	Email
736	Naidoo	Liano	8th Cyber Ave	098 123 7822	liano@test.com
...	...	...	...	...	...
...	...	...	...	...	...

CREATE statement

```sql
CREATE TABLE Employee (
    EmployeeID int NOT NULL,
    LastName varchar(255) NOT NULL,
    FirstName varchar(255),
    Address varchar(255),
    PhoneNumber varchar(255),
    PRIMARY KEY (EmployeeID)
);
```

Alter Table

```
10 ALTER TABLE Employee
11 ADD Email varchar(255);
```

INSERT INTO

```
13 INSERT INTO Employee (EmployeeID, LastName, FirstName, Address, PhoneNumber, Email)
14 VALUES
15       (1234, 'Smith', 'John', '25 Oak Rd', '0837856767', 'john@smith.com'),
16       (4321, 'Man', 'Jelly', '33 Sweety Ave', '0913405097', 'Jelly@man.com');
```

Conditionally SELECT

```
18 SELECT * FROM Employee
19 WHERE FirstName = 'John';
```

Conditionally SELECT

```
18 SELECT * FROM Employee
19 WHERE FirstName = 'John';
```

Conditionally SELECT and sort

```
21 SELECT * FROM Employee
22 ORDER BY LastName DESC;
```

Conditionally UPDATE a value

```
24 UPDATE Employee
25 SET Address = '78 Oak St'
26 WHERE EmployeeID = 1234;
```

Conditionally UPDATE a value

```
24 UPDATE Employee
25 SET Address = '78 Oak St'
26 WHERE EmployeeID = 1234;
```

Conditionally UPDATE a value

```
24 UPDATE Employee
25 SET Address = '78 Oak St'
26 WHERE EmployeeID = 1234;
```

Conditionally UPDATE a value

```
31 DROP TABLE Employee;
```

Infotek Solutions

SQL Injections

SQL Injection

SQL injection (SQLi) is a web security vulnerability that allows an attacker to interfere with the queries that an application makes to its database.

In some situations, an attacker can escalate a SQL injection attack to compromise the underlying server or other back-end infrastructure. It can also enable them to perform denial-of-service attacks.

Session Recap

SQL Basics:

- **We reviewed fundamental SQL concepts, including simple queries using SELECT statements, and the distinction between Data Definition Language (DDL) and Data Manipulation Language (DML).**

Highlight Vulnerabilities:

- **We explored how SQL injection can exploit vulnerabilities in SQL queries, particularly focusing on the risks associated with not using prepared statements.**

Practical Exercise:

- **You examined a deliberately vulnerable Python script linked to a SQLite3 database, demonstrating how SQL injection can be executed through input manipulation using an external text file.**

Mitigation Strategies:

- **We discussed and applied strategies to prevent SQL injection, including the use of prepared statements and input validation, to secure SQL queries against common attack vectors.**

Key Takeaways:

- **Understanding SQL injection techniques and implementing robust security practices are crucial for protecting databases and applications from potential attacks. Ensure to apply these best practices in your coding to enhance security and safeguard data integrity.**

Hyperiondev

Q & A Section

Please use this time to ask any questions relating to the topic explained, should you have any

Hyperiondev

Thank you for joining us

Take regular breaks.
Stay hydrated.
Avoid prolonged screen time.
Remember to have fun :)