SQL INJECTION 101

By Mohammed Alharbi

AGENDA

- CREATE A USERS TABLE
- SELECT QUERY
- LOGICAL OPERATORS
- BACK TO SELECT QUERY
- SQL INJECTION
- SQL INJECTION EXAMPLE
- ANOTHER EXAMPLE
- PREPARED STATEMENT

CREATE A USERS TABLE

Users table schema

ID	USERNAME	PASSWORD
1	admin	123456
2	Mohammed	1122334455
3	Abdulaziz	12121212

CREATE TABLE

```
create table users (
   id int(10),
   username varchar(50),
   password varchar(50)
);
```

INSERT DATA

```
INSERT INTO users
VALUES (1, 'admin', '123456');
...
```

SELECT QUERY

SELECT SPECIFIC COLUMN

```
SELECT id FROM users
```

SELECT SPECIFIC COLUMNS

```
SELECT id , username FROM users
```

SELECT ALL COLUMNS

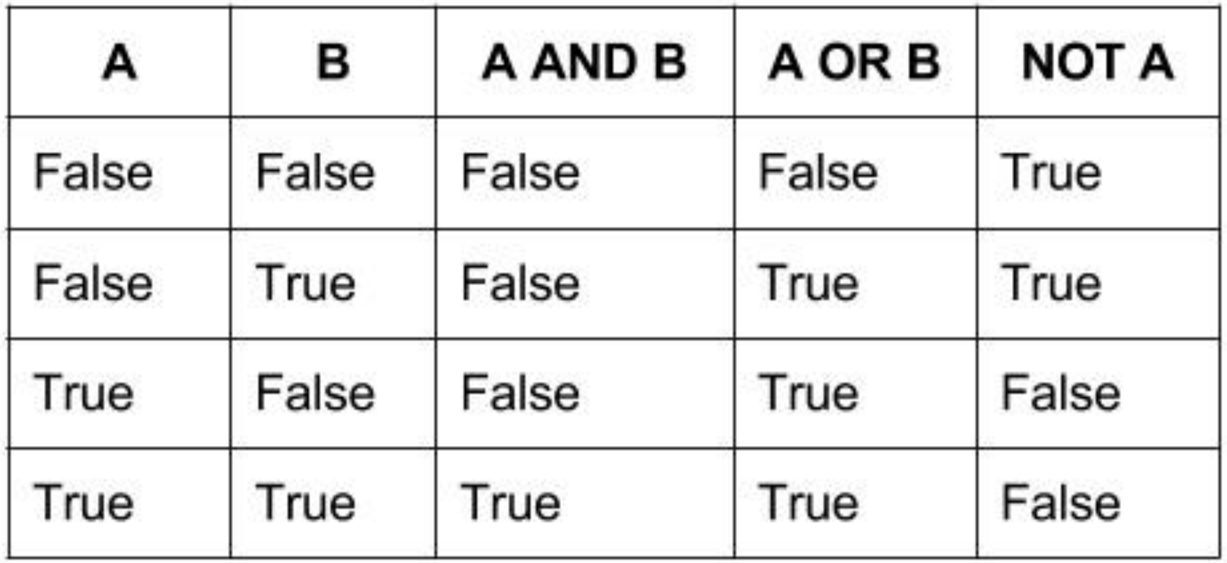
```
SELECT * FROM users
```

SELECT WITH CONDITION

```
SELECT * FROM users WHERE id = 1
```

LOGICAL OPERATORS

Logical operators



A	B	A AND B	A OR B	NOT A
False	False	False	False	True
False	True	False	True	True
True	False	False	True	False
True	True	True	True	False

BACK TO SELECT QUERY

SELECT WITH CONDITIONS

```sql
SELECT * FROM users WHERE username
= 'admin' AND password = '123456'
```

SELECT WITH CONDITIONS

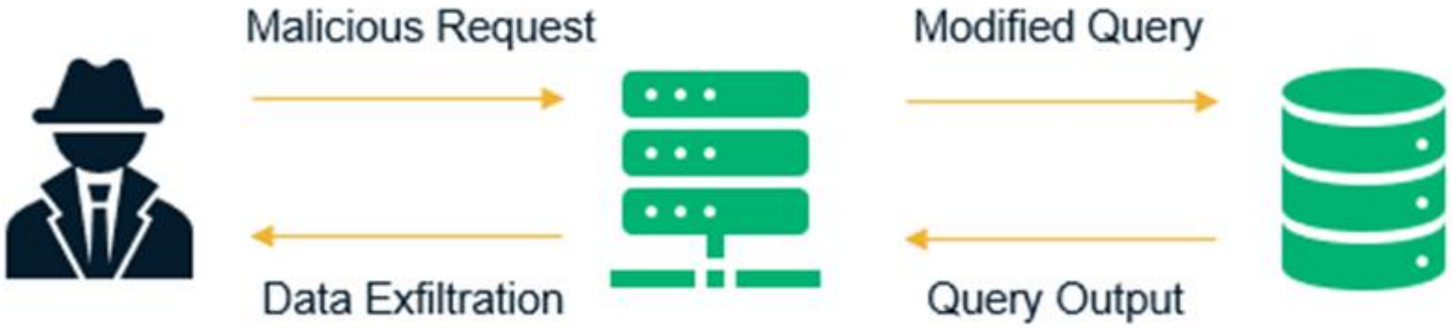
```
SELECT * FROM users WHERE username
= 'admin' OR username = 'mohammed'
```

SQL INJECTION

SQL INJECTION

It’s a **common web vulnerability** allow the attackers to **inject malicious SQL query** to access information that was not intended to be displayed, and these queries can help them to **manipulate the data**

SQL INJECTION



SQL INJECTION EXAMPLE

SELECT WITH INPUTS

```
SELECT * FROM users WHERE id = INPUT
```

SELECT WITH INJECTION

```
SELECT * FROM users WHERE id = 1337 OR 1 = 1
```

SELECT WITH INJECTION

```
SELECT * FROM users WHERE id = 1337 OR 1 = 1
```

ANOTHER EXAMPLE

SELECT WITH INPUTS

```
SELECT * FROM users WHERE username
= 'INPUT' AND password = 'INPUT'
```

SELECT WITH INJECTION

```
SELECT * FROM users WHERE username = 'ADMIN' -- - ' AND password = ' INPUT '
```

SELECT WITH INJECTION

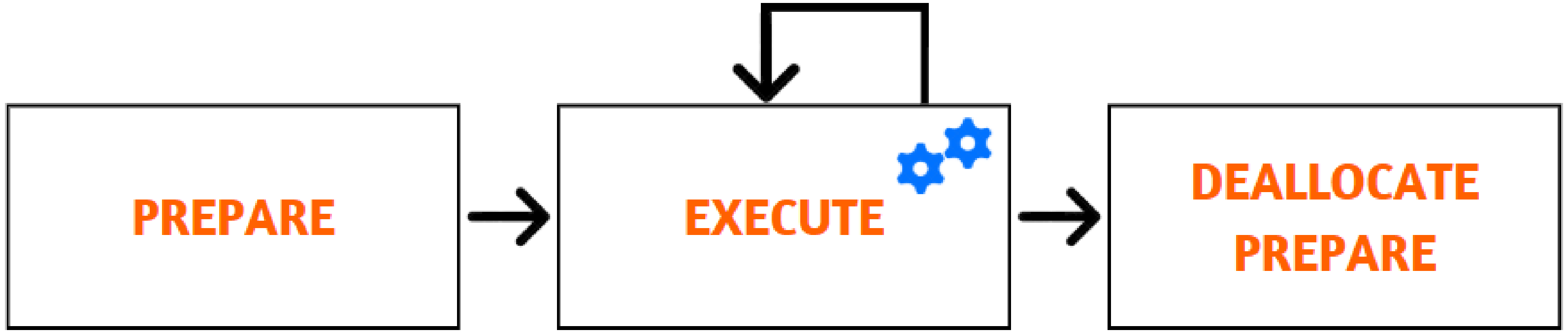
```
SELECT * FROM users WHERE username = 'ADMIN' -- - ' AND password = ' INPUT '
```

PREPARED STATEMENT

Prepared statement

prepared statement or a **parameterized statement** is used to execute the same statement repeatedly with high efficiency and protect against SQL injections.

Prepared statement



PREPARE

EXECUTE

DEALLOCATE PREPARE

Prepared statement

```
SELECT * FROM users WHERE
username = ? AND password = ?
```

Prepared statement

https://dev.mysql.com/doc/apis-php/en/apis-php-mysqli.quickstart.prepared-statements.html

THANK YOU