

MYSQL

2/16/20

Creating/Checking Connection

```
$conn = mysqli_connect ($servername, $username, $password);  
// can wrap in try-catch
```

```
if (!$conn) {  
    die ("conn failed: " . mysqli_connect_error());  
}
```

DB

```
if ($conn->query("CREATE DATABASE name") === TRUE)
```

creates DB (returns true if successful)

table goes here

```
$sql = "CREATE TABLE name (
```

```
• id INT(6) UNSIGNED AUTO-INCREMENT  
  name      type      +1 for every new entry
```

PRIMARY KEY,
identifies rows in table

```
• email VARCHAR(50) NOT NULL
```

must contain value

)

inserting data

• \$sql = "INSERT INTO table_name (column1, 2, ...) VALUES (value1, value2, ...)";

Get last ID

\$conn -> insert_id

WHERE

• can use * to select all
\$sql = "SELECT name, name2 FROM table WHERE name = 'hi'";

\$result = \$conn -> query(\$sql);

Prepared Statements

• executes similar / same commands quickly

Process

- ① Prepare - save statement of ?? commands
- ② database parses info (doesn't execute)
- ③ executes statements

```

SS  JAVASCRIPT  SQL  PYTHON  PHP  BOOTSTRAP  HOW TO  MORE  REFERENCES  EXER
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// prepare and bind
$stmt = $conn->prepare("INSERT INTO MyGuests (firstname, lastname, email) VALUES (?, ?, ?)");
$stmt->bind_param("sss", $firstname, $lastname, $email);

// set parameters and execute
$firstname = "John";
$lastname = "Doe";
$email = "john@example.com";
$stmt->execute();

$firstname = "Mary";
$lastname = "Moe";
$email = "mary@example.com";
$stmt->execute();

$firstname = "Julie";
$lastname = "Dooley";
$email = "julie@example.com";
$stmt->execute();

echo "New records created successfully";

$stmt->close();
$conn->close();
?>

```

Handwritten annotations on the image:

- (1) next to the first `$stmt->execute();`
- (2) ← at DB next to the second `$stmt->execute();`
- (3) next to the third `$stmt->execute();`
- A bracket groups the three `$stmt->execute();` lines.

order by

`$sql = "SELECT ... ORDER BY var"`
↑
order output

Deleting

`$sql = "DELETE FROM table WHERE var = val"`
↑
thing you want to delete

Updating

`$sql = "UPDATE table.name SET column 1 = val
 WHERE some-column = some-val"`

Limit

`$sql = "SELECT * FROM table LIMIT 30"`
└──┘
gets records 1-30

to get ^{records} 16-25

" \$sql = "SELECT * FROM table LIMIT 10
OFFSET 15";

or LIMIT 15, 10
 ↑ ↑
 offset limit
(starts at 16)