

# HOW TO RUN THE CODE:

1. Download and extract “**Iridium.zip**” from e-learning.
2. Download and install Java to run the code in Java.
3. Download and install VS Code to edit and run the Java program.
4. Open VS Code. Open folder Iridium in VS Code.
5. In the “**IRIDIUMSQL/src**” folder, there is a file called DavisBase.java. Run the file from VS Code.

## IRIDIUM SQL PROMPT:

6. A prompt will be visible where we can type our SQL queries. By default, a “catalog” database is created.
7. The following queries are supported by Iridium SQL (all queries should be written in small):
  - a. **Create Database:** CREATE DATABASE <db\_name>;
  - b. **Drop Database:** DROP DATABASE <db\_name>;
  - c. **Use Database:** USE <db\_name>;
  - d. **Show Databases:** SHOW DATABASES;
  - e. **Create Table:** CREATE TABLE <table\_name> (<column1\_name> <column1\_dtype> [ARGUMENTS], <column2\_name> <column2\_dtype> [ARGUMENTS]);

Example: **create table test (id int, name text not null);**

Note: The first column is the primary key by default.

- f. **Drop Table:** DROP TABLE <table\_name>;
- g. **Show Tables:** SHOW TABLES;
- h. **Create Index:** CREATE INDEX <table\_name> ON <column\_name>;

- i. **Select Table:** SELECT [column\_names] FROM <table\_name> WHERE <column\_name> <condition> <value>;

Example: **select name from test where id > 1;**

```
Iridiumsql> show databases;
Case: SHOW

Databases
-----
catalog

Query Successful
1 row affected

Iridiumsql> create database newdb;
Case: CREATE
newdb
isCreated : true PATH: data/newdb
Query Successful
1 row affected

Iridiumsql> create table test(id int, name text not null);
Case: CREATE
Stub: parseCreateTable method
Command: create table test ( id int , name text not null )
Table test created successfully.
Query Successful
1 row affected
```

- j. **Insert Table:** INSERT INTO <table\_name>([column\_definitions])  
VALUES([column\_values]);

Example: insert into test(id, name) values(1, 'Davis');

```
Iridiumsql> create table test(id int, name text not null);
Case: CREATE
Stub: parseCreateTable method
Command: create table test ( id int , name text not null )
Table test created successfully.
Query Successful
1 row affected

Iridiumsql> insert into test(id, name) values(1, 'Davis');
Case: INSERT
Command: insert into test ( id , name ) values ( 1 , 'davis' )
Stub: This is the insertRecord method
Query Successful
1 row affected

Iridiumsql> insert into test(id, name) values(2, 'Davis_2');
Case: INSERT
Command: insert into test ( id , name ) values ( 2 , 'davis_2' )
Stub: This is the insertRecord method
Query Successful
1 row affected

Iridiumsql> insert into test(id, name) values(3, 'Davis_3');
Case: INSERT
Command: insert into test ( id , name ) values ( 3 , 'davis_3' )
Stub: This is the insertRecord method
Query Successful
1 row affected
```

- k. **Update Table:** UPDATE <table\_name> SET  
<column\_name>=<new\_values> WHERE <column\_name>  
<condition> <value>;

Example: **update test set name='newname' where id=1;**

```
Iridiumsql> update test set name='newname' where id=1;
Case: UPDATE
Command: update test set name='newname' where id=1
Stub: This is the parseUpdate method
Table test updated successfully.

Query Successful
1 row affected

Iridiumsql> select * from test;
Case: SELECT
Command: select * from test
Stub: This is the parseQuery method
INSERTED
select * from test
-----
-----
id  |name      |
-----
1   |'newname' |
2   |'davis_2' |
3   |'davis_3' |

Query Successful
1 row affected
```

- I. **Delete in Table:** DELETE FROM <table\_name> WHERE  
<column\_name> <condition> <value>;

Example: **delete from test where id=3;**

```
Iridiumsql> delete from test where id=3;
Case: DELETE
Command: delete from test where id=3
Stub: This is the deleteRecord method

Query Successful
1 row affected

Iridiumsql> select * from test;
Case: SELECT
Command: select * from test
Stub: This is the parseQuery method
INSERTED
select * from test
-----
-----
id      | name      |
-----
1       | 'newname' |
2       | 'davis_2' |

Query Successful
1 row affected
```