

## README file

The Project was developed using Eclipse IDE for Java.

### Instructions for running the application:

From eclipse:

- 1) Extract the compressed folder downloaded from eLearning.
- 2) Import the folder TeamEDavisBase into the Eclipse IDE.
- 3) Run the DavisBase.java to start the application.

(or)

From local terminal:

- 4) Extract the compressed folder TeamEDavisBase
- 5) Open terminal or command prompt
- 6) Navigate to the path where the folder has been extracted.
- 7) Change directory to the TeamEDavisBase\src\ in terminal and compile all files using the below command:

**javac \*.java**

- 8) Change directory to parent directory (TeamEDavisBase\src)
- 9) Run DavisBase.class file using the below command:

**java database.DavisBase**

### The following commands are supported:

#### **(a) Displays a list of all tables in DavisBase**

Syntax: *SHOW TABLES;*

```
TeamEsq!> show tables;
STUB: Calling the method to process the command
Parsing the string:"show tables"
-----
table_name |
-----
davisbase_tables |
davisbase_columns |
test |
test1 |
```

# README file

## (b) Create a new table

Syntax: CREATE TABLE <table\_name> (<column name> <constraint>, ..) ;

- The first column is primary key, so it has to be of data type Integer.
- There is an inbuilt AUTO INCREMENT function implemented for the primary key.
- The columns followed by primary key can be UNIQUE/NOT NULL.

```
*****
Welcome to DavisBase
DavisBase Version V1.0
```

Type "help;" to display supported commands.

```
*****
```

```
TeamEsq!> CREATE TABLE testcase (rowid INT, name TEXT UNIQUE, desc TEXT NOT NULL);
```

STUB: Calling your method to process the command

Parsing the string:"create table testcase (rowid int, name text unique, desc text not null)"

```
TeamEsq!> SELECT * FROM davisbase_columns WHERE table_name=testcase;
```

STUB: Calling the method to process the command

Parsing the string:"select \* from davisbase\_columns where table\_name=testcase"

rowid	table_name	column_name	data_type	ordinal_position	is_nullable	is_unique
16	testcase	desc	TEXT	3	NO	NO
14	testcase	rowid	INT	1	YES	NO
15	testcase	name	TEXT	2	NO	YES

```
TeamEsq!>
```

## (c) Drop table from the database

Syntax: DROP TABLE <table\_name>;

```
TeamEsq!> drop table testcase;
```

STUB: Calling the method to process the command

Parsing the string:"drop table testcase"

```
TeamEsq!> show tables;
```

STUB: Calling the method to process the command

Parsing the string:"show tables"

table_name
davisbase_tables
davisbase_columns
test
test1

```
TeamEsq!>
```

## README file

### (d) Insert record into table

Syntax: INSERT INTO table\_name <column\_list> VALUES <value\_list>;

```
TeamEsq!> insert into testcase values (1,t1,testcase1);
STUB: Calling the method to process the command
Parsing the string:"insert into testcase values (1,t1,testcase1)"
Checking for unique constraint violation
```

```
TeamEsq!> select * from testcase;
STUB: Calling the method to process the command
Parsing the string:"select * from testcase"
```

rowid	name	desc
1	t1	testcase1

With the auto increment feature on primary, the value for the column can be skipped for it to be generated dynamic.

```
TeamEsq!> insert into testcase values (1,t1,testcase1);
STUB: Calling the method to process the command
Parsing the string:"insert into testcase values (1,t1,testcase1)"
Checking for unique constraint violation
```

```
TeamEsq!> select * from testcase;
STUB: Calling the method to process the command
Parsing the string:"select * from testcase"
```

rowid	name	desc
1	t1	testcase1

When null is inserted into any not null column, then the database engine throws error.

```
TeamEsq!> insert into testcase values (,t3,null);
STUB: Calling the method to process the command
Parsing the string:"insert into testcase values (,t3,null)"
NULL-value constraint violation
```

When duplicate value is passed for column with unique constraint, then the database engine throws error.

```
TeamEsq!> insert into testcase values (,t2,test);
STUB: Calling the method to process the command
Parsing the string:"insert into testcase values (,t2,test)"
Checking for unique constraint violation
```

Duplicate key found for name

```
TeamEsq!>
```

## README file

### (e) Update one or more records in table

Syntax: UPDATE table\_name SET column\_name = value WHERE <condition>;

```
TeamEsq!> update testcase set desc=test where rowid=2;
STUB: Calling the method to process the command
Parsing the string:"update testcase set desc=test where rowid=2"
TeamEsq!> select * from testcase;
STUB: Calling the method to process the command
Parsing the string:"select * from testcase"
-----
rowid  |name  |desc      |
-----
1      |t1    |testcase1 |
2      |t2    |test      |
TeamEsq!>
```

---

### (f) Display records from a table

Syntax: SELECT \* FROM <table\_name> [WHERE <condition>];

```
TeamEsq!> select * from testcase;
STUB: Calling the method to process the command
Parsing the string:"select * from testcase"
-----
rowid  |name  |desc      |
-----
1      |t1    |testcase1 |
2      |t2    |test      |
TeamEsq!> select * from testcase where rowid>=2;
STUB: Calling the method to process the command
Parsing the string:"select * from testcase where rowid>=2"
-----
rowid  |name  |desc      |
-----
2      |t2    |test      |
TeamEsq!> select * from testcase where desc=test;
STUB: Calling the method to process the command
Parsing the string:"select * from testcase where desc=test"
-----
rowid  |name  |desc      |
-----
2      |t2    |test      |
TeamEsq!>
```

When NULL is used in where condition, the database engine equate null value in data with the where condition.

```
TeamEsq!> select * from test;
STUB: Calling the method to process the command
Parsing the string:"select * from test"
-----
id  |name  |
-----
1  |test  |
2  |null  |
TeamEsq!> select * from test where name=null;
STUB: Calling the method to process the command
Parsing the string:"select * from test where name=null"
Empty Set
TeamEsq!>
```

## **README file**

**(g) To get help on the command and syntax**

Syntax: help;

**(h) Exit from the database**

Syntax: exit;