SQL CHEAT SHEET

MANAGING TABLES

```
CREATE TABLE tbl (
   column1 INT PRIMARY KEY,
   column2 VARCHAR NOT NULL,
   column3 INT DEFAULT 0
);
```
Create a new table with three columns

```
DROP TABLE tbl;
```
Delete the table

```
ALTER TABLE tbl ADD clm4;
```
Add a new column to the table

```
ALTER TABLE tbl DROP COLUMN clm4;
```
Drop column4 from the table

```
ALTER TABLE tbl ADD constraint;
```
Add constraint

```
ALTER TABLE tbl DROP constraint;
```
Drop a constraint

```
ALTER TABLE tbl1 RENAME TO tbl2;
```
Rename a table from tbl1 to tbl2

```
ALTER TABLE tbl1 RENAME clm1 TO clm2;
```
Rename column clm1 to clm2

```
TRUNCATE TABLE tbl;
```
Remove all data in a table

USING SQL CONSTRAINTS

```
CREATE TABLE tbl (
   clm1 INT, clm2 INT, clm3
   VARCHAR, PRIMARY KEY (clm1,
   clm2)
);
```
Set clm1 and clm2 as a primary key

```
CREATE TABLE tbl1 (
   clm1 INT PRIMARY KEY,
   clm2 INT,
   FOREIGN KEY (clm2) REFERENCES
tbl2(clm2)
);
```
Set clm2 column as a foreign key

```
CREATE TABLE tbl (
   clm1 INT, clm2 INT,
   UNIQUE (clm2,
   clm3)
);
```
Make the values in clm1 and clm2 unique

```
CREATE TABLE tbl (
   clm1 INT, clm2 INT,
   CHECK (clm1> 0 AND clm1 >= clm2)
);
```
Ensure clm1> 0 and values in clm2 >= clm2

```
CREATE TABLE tbl (
   clm1 INT PRIMARY KEY,
   clm2 VARCHAR NOT
   NULL
```

MODIFYING DATA

```
INSERT INTO tbl (column_list)
VALUES (value_list);
```
Insert one row into a table

```
INSERT INTO tbl(column_list)
VALUES (value_list), (value_list), ...;
```
Insert multiple rows into a table

```
INSERT INTO tbl1(column_list)
SELECT column_list
FROM tbl2;
```
Insert rows from tbl2 into tbl1

```
UPDATE tbl
SET clm1 = new_value;
```
Update new value in column clm1 for all rows

```
UPDATE tbl
SET clm1 = new_value, clm2= new_value
WHERE condition;
```
Update values in the column clm1, clm2 that match the condition

```
DELETE FROM tbl;
```
Delete all data in a table

```
DELETE FROM tbl
WHERE condition;
```
Delete subset of rows in a table

QUERYING DATA FROM A TABLE

```
SELECT clm1, clm2 FROM tbl;
```
Query data in columns clm1, clm2 from a table

```
SELECT * FROM tbl;
```
Query all rows and columns from a table

```
SELECT clm1, clm2 FROM tbl
WHERE condition;
```
Query data and filter rows with a condition

```
SELECT DISTINCT clm1 FROM tbl
WHERE condition;
```
Query distinct rows from a table

```
SELECT clm1, clm2 FROM tbl
ORDER BY clm ASC [DESC];
```
Sort the result set in ascending or descending order

```
SELECT clm1, clm2 FROM tbl
ORDER BY clm1
LIMIT n OFFSET offset;
```
Skip offset of rows and return the next n rows

```
SELECT clm1, aggregate(clm2)
FROM tbl
GROUP BY clm1;
```
Group rows using an aggregate function

```
SELECT clm1, aggregate(clm2)
FROM tbl
GROUP BY clm1
HAVING condition;
```
Filter groups using HAVING clause

QUERYING FROM MULTIPLE TABLES

```
SELECT clm1, clm2
FROM tbl
INNER JOIN tbl2 ON condition;
```
Inner join tbl1 and tbl2

```
SELECT clm1, clm2
FROM tbl1
LEFT JOIN tbl2 ON condition;
```
Left join tbl1 and tbl2

```
SELECT clm1, clm2
FROM tbl1
RIGHT JOIN tbl2 ON condition;
```
Right join tbl1 and tbl2