SQL Basics Part 2

* this PDF about Data Definition Language [DDL]

/\*

=======================================

== 08 Execution Order VS Coding Order =

=======================================

------------------

-- Coding Order --

------------------

- syntax

SELECT DISTINCT TOP <num>

column1, column2, ...

FROM table\_name

WHERE condition

GROUP BY <column name>

ORDER BY column\_name ASC|DESC

---------------------

-- Execution Order --

---------------------

1- FROM Clause

2- WHERE Clause

3- GROUP BY Statement

4- HAVING Clause

5- SELECT DISTINCT Statement

6- ORDER BY Keyword

7- TOP Statement

\*/

/\*

===============================

== 09 Write Multiple Queries ==

===============================

- [1] at the end of each query write semicolomn (;)

- [2] static (fixed) values, that means output data from us not in the table

\*/

SELECT \*

FROM Customers;

SELECT \*

FROM Orders;

-- static (fixed) values, you can output any value using SELECT

SELECT 'HELLO WORLD' AS static\_string;

SELECT 123 AS static\_value;

-- add column from us

SELECT

id,

first\_name,

'New Customers' AS customer\_type

FROM customers;

/\*

=====================

== 10 CREATE TABLE ==

=====================

- CREATE TABLE is a statement used to create a new table in SQL

- NOT NULL: is a conatrain used to make the field unempty

- in each table must have a primary key

- primary key: is a unique identifier for each record (row) in a database,

It ensures that every row can be uniquely identified and cannot contain duplicate or NULL value

- syntax

CREATE TABLE table\_name (

column1 datatype CONSTRAINT,

column2 datatype CONSTRAINT,

column3 datatype CONSTRAINT,

....

);

\*/

-- create a new table calles persons, with columns: id, person\_name, birth\_date, and phone

CREATE TABLE persons (

id INT NOT NULL,

person\_name VARCHAR(50) NOT NULL,

birth\_date DATE,

phone VARCHAR(15) NOT NULL,

/\*

- create the primary key for the table

- syntax

CONSTRAIN constraint\_name PRIMARY KEY (field\_name)

\*/

CONSTRAINT pk\_persons PRIMARY KEY (id)

);

/\*

========================

== 11 ALTER Statement ==

========================

- ALTER Statement: used to modify an existing table without deleting it,

it allows you to add, delete, or change columns, rename tables, or add and remove constraints

- syntax

ALTER TABLE table\_name

ADD | DROP | MODIFY | RENAME | ...;

-- example to add new column

ALTER TABLE Students

ADD email VARCHAR(100);

-- example to modify an existing column

ALTER TABLE Students

MODIFY name VARCHAR(150);

\*/

-- add a new column called email to the persons table

ALTER TABLE persons

ADD email VARCHAR(100);

-- remove the column phone from the persons table

ALTER TABLE persons

DROP COLUMN phone;

SELECT \*

FROM persons;

/\*

=======================

== 12 DROP Statement ==

=======================

- DROP Statemen is used to permanently delete database objects,

such as tables, databases, views, indexes, or constraints

- when use DROP, all the data and structre are deleted, it can't be rolled back

- we use "IF EXISTS" to avoid errors

- the DROP statement is very simple, but risky to use

\*/

-- delete the table persons from the database

DROP TABLE persons;

-- delete database

DROP DATABASE IF EXISTS people;

-- delete a column from the database

ALTER TABLE perons

DROP COLUMN email;

-- delete a primary key

ALTER TABLE persons

DROP PRIMARY KEY pk\_persons;