SQL SL Review LIJIA XU(1/)

Primary keys | Structured Query Language

unique value for each row
can not contain NULL values
* one primary key each must be different for each row

MySQL is a program that understands SQL
SQL is ANSI standard

SHOW DATABASES; lists the databases managed by server

SHOW TABLES; display all tables in the current database

SHOW COLUMNS FROM customers; Field Type key Default Extra

SELECT Firstname FROM customers; // stored in a result table ⇒ result-set

* end each SQL statement with a semicolon;

SQL is case insensitive | while spaces and multi lines are ignored

SELECT Fn, Ln, City FROM customers;

only in SQL | Unique SELECT DISTINCT column1, column2 FROM tablename;

SELECT * FROM customers; // select all from customers

LIMIT 3,4; // show 4 results, start from 3rd location ** The count starts from 0

SELECT ID, City FROM customers LIMIT 5; // only show 5 results

SELECT customers.City From customers; ↓ qualified name

SELECT City From customers; | fully

SELECT * FROM customers | Default is ascending order
ORDER BY FirstName;
ORDER BY LastName, Age;

WHERE City='NY';

= != > < >= <=

SELECT * FROM customers WHERE ID BETWEEN 3 AND 7

must be single quotation 'Can't'

AND OR IN NOT

SELECT * FROM customers WHERE City NOT IN ('NY','LA','CA');

BETWEEN

SELECT CONCAT(FirstName,',',City) FROM customers;

use the AS to assign name

SELECT CONCAT(FirstName,',',City) AS new_column FROM customers;

+ - * /

SELECT ID, FirstName, Salary+500 AS Salary FROM employees;

UPPER LOWER convert to uppercase or to lowercase

SELECT FirstName, UPPER(LastName) AS LastName FROM employees;

SQRT AVG SUM MIN

#A subquery is a query within another query

SELECT FirstName, Salary FROM employees WHERE Salary > (SELECT AVG(Salary) FROM employees) ORDER BY Salary DESC

'_' single char '%' any number of char

SELECT * FROM table WHERE name LIKE

SELECT customers.ID, customers.Name, orders.Name, orders.Amount FROM customers AS ct, orders AS ord WHERE customers.ID = orders.Customer_ID ORDER BY customers.ID; // AS is optional to use

INNER JOIN | ON ⇒ specifying inner join condition

LEFT RIGHT | OR it NO match is found for a row, NULL is returned

SELECT column_name(s) FROM table1 INNER JOIN table2 ON table1.column_name = table2.column_name;

UNION ⇒ removes duplicates
UNION ALL ⇒ not remove duplicates

SELECT table1.column1, table2.column2 FROM table1 LEFT OUTER JOIN table2 ON table1.column_name = table2.column_name

In order to use UNION, must have same number of columns

INSERT INTO provide values for every column

SELECT ID, FirstName FROM table1 UNION SELECT ID, FirstName From table2

does not have default or does not support NULL

SELECT ID, NULL (or other) FROM table2

INSERT INTO Employees Values (8,'Anthony','Young',35);

or INSERT INTO employees(ID, FirstName, LastName, Age) VALUES (8,'A','B',24);

UPDATE Employees SET Salary=5000, FirstName='A' WHERE ID=1

DELETE FROM Employees WHERE ID=1

* forget WHERE will delete everything

CREATE TABLE Users (
UserID int,
City varchar(100),
PRIMARY(UserID)
);

Auto increment starts from '1' by default

SQL constrains

INT FLOAT DOUBLE DATE DATETIME TIMESTAMP TIME CHAR VARCHAR BLOB // Binary Large Objects TEXT

NOT NULL UNIQUE PRIMARY KEY CHECK DEFAULT

ALTER TABLE * add, delete or modify columns in table

CRETE TABLE Users(
id int NOT NULL AUTO INCREMENT,
username varchar(40) NOT NULL,
password varchar(10) NOT NULL,
PRIMARY KEY(id)
);

ALTER TABLE People ADD DateOfBirth Date;

** All new rows will have the default values

delete a column: | delete the whole table:

ALTER TABLE People DROP COLUMN DateOfBirth; | DROP TABLE People;

ALTER TABLE People CHANGE FirstName name varchar(100); | modify the columns

Rename the table: | a VIEW is a virtual table

RENAME TABLE People TO Users;

CREATE VIEW List AS SELECT FirstName, Salary FROM Employees; | ** a view is always up to date

CREATE OR REPLACE VIEW List AS | * remove ⇒ DROP VIEW List;

SELECT FirstName, LastName, Salary FROM Employees; | * sometimes is easier to drop and create new