Introduction to SQL (Part I)

Fundamentals

- A **query** is a specific request for data manipulation issued by the end-user or the application to the DBMS.
- SQL:
  - Stands for Structured Query Language
  - Pronounced as S-Q-L or "sequel"
  - Consists of commands that:
    - Create database and table structures
    - Perform various types of data manipulation and data administration
    - Query the database to extract useful information
- Popular Database Management Tools
  - Microsoft SQL Server
  - MySQL
  - Oracle RDBMS
  - Microsoft Access
- **SQL Data Types:**

Category	Common Data Types
Exact numeric	bigint, bit, decimal, int, money, numeric
Approximate numeric	float, real
Date and time	date, datetime, time
Character strings	char, text, varchar
Unicode character strings	nchar, ntext, nvarchar
Binary strings	binary, image, varbinary
Other data types	cursor, sql_variant, table, uniqueidentifier, xml

- **SQL Operators:**

Category	Operators
Arithmetic	+, -, *, /, %
Comparison	=, >, <, >=, <=, <>
Compound	+=, -=, *=, /=, %=
Logical	AND, OR, NOT, LIKE, IN, BETWEEN, EXISTS, ANY, ALL

SQL Data Definition Commands

- **CREATE DATABASE** – creates a new database
  - *Syntax*: **CREATE DATABASE** *database_name*;
  - *Example*: CREATE DATABASE myDB;
- **DROP DATABASE** – deletes an existing database
  - *Syntax*: **DROP DATABASE** *database_name*;
  - *Example*: DROP DATABASE myDB;
- **CREATE TABLE** – creates a new table in a database
  - *Syntax*: **CREATE TABLE** *table_name* (*column1 datatype, …*);
  - *Example*: CREATE TABLE Students (StudentID varchar(11), LastName varchar(99), FirstName varchar(99), Section varchar(5));
- **DROP TABLE** – deletes an existing table in a database
  - *Syntax*: **DROP TABLE** *table_name*;
  - *Example*: DROP TABLE Students;
  - To delete only the table's data:
    - *Syntax*: TRUNCATE TABLE *table_name*;
    - *Example*: TRUNCATE TABLE Students;
- **ALTER TABLE** – Adds, deletes, or modifies columns in an existing table
  - *Syntax to add*: **ALTER TABLE** *table_name* **ADD** *column datatype*;
  - *Example*: ALTER TABLE Students ADD MiddleName varchar(99);
  - *Syntax to delete*: **ALTER TABLE** *table_name* **DROP** *column*;
  - *Example*: ALTER TABLE Students DROP COLUMN Section;
  - *Syntax to modify*: **ALTER TABLE** *table_name* **ALTER COLUMN** *column datatype*;
  - *Example*: ALTER TABLE Students ALTER COLUMN MiddleName nvarchar(99);

SQL Constraints

- **NOT NULL** on CREATE TABLE – ensures that a column cannot have a NULL value upon creating a table
  - *Example*: CREATE TABLE Students (StudentID varchar(11) NOT NULL, LastName varchar(99) NOT NULL, FirstName varchar(99) NOT NULL, Section varchar(5));

- **NOT NULL** on ALTER TABLE – ensures that a column in an existing table cannot have a NULL value
    - *Example*: ALTER TABLE Students ALTER COLUMN Section varchar(5) NOT NULL;
- **UNIQUE** on CREATE TABLE – ensures that all values in a column are different upon creating a table
    - *Example*: CREATE TABLE Students (StudentID varchar(11) NOT NULL UNIQUE, LastName varchar(99) NOT NULL, FirstName varchar(99) NOT NULL, Section varchar(5));
- **UNIQUE** on ALTER TABLE – creates a UNIQUE constraint on a column of an existing table
    - *Syntax*: **ALTER TABLE** *table_name* **ADD UNIQUE** *(column);*
    - *Example*: ALTER TABLE Students ADD UNIQUE (StudentID);
- **PRIMARY KEY** on CREATE TABLE – uniquely identifies each row in a table
    - *Example*: CREATE TABLE Students (StudentID varchar(11) NOT NULL PRIMARY KEY, LastName varchar(99) NOT NULL, FirstName varchar(99) NOT NULL, Section varchar(5));
- **PRIMARY KEY** on ALTER TABLE – creates a PRIMARY KEY constraint on a column of an existing table
    - *Syntax*: **ALTER TABLE** *table_name* **ADD PRIMARY KEY** *(column);*
    - *Example*: ALTER TABLE Students ADD PRIMARY KEY (StudentID);
- **FOREIGN KEY** on CREATE TABLE – uniquely identifies a row in another table
    - *Example*: CREATE TABLE Orders (OrderID int NOT NULL PRIMARY KEY, TableNumber int NOT NULL, CustomerID int FOREIGN KEY REFERENCES Customers (CustomerID));
- **FOREIGN KEY** on ALTER TABLE – creates a FOREIGN KEY constraint on a column of an existing table
    - *Syntax*: **ALTER TABLE** *table1_name* **ADD FOREIGN KEY** *(table1_column)* **REFERENCES** *table2_name (table2_column);*
    - *Example*: ALTER TABLE Orders ADD FOREIGN KEY (CustomerID) REFERENCES Customers (CustomerID);
- **CHECK** on CREATE TABLE – ensures that all values in a column satisfy a specific condition upon creating a table
    - *Example*: CREATE TABLE Students (StudentID varchar(11) NOT NULL, LastName varchar(99) NOT NULL, FirstName varchar(99) NOT NULL, Age int CHECK (Age>=15));
- **CHECK** on ALTER TABLE – ensures that all values in a column of an existing table satisfy a specific condition
    - *Syntax*: **ALTER TABLE** *table_name* **ADD CHECK** *(condition);*
    - *Example*: ALTER TABLE Students ADD CHECK (Age>=15);
- **DEFAULT** on CREATE TABLE – sets a default value for a column when there is no value specified
    - *Example*: CREATE TABLE Students (StudentID varchar(11) NOT NULL, LastName varchar(99) NOT NULL, FirstName varchar(99) NOT NULL, Section varchar(5) DEFAULT 'Not yet enrolled');
- **DEFAULT** on ALTER TABLE – sets a default value for a column of an existing table when there is no value specified
    - Syntax: **ALTER TABLE** *table_name* **ADD CONSTRAINT** *constraint_name* **DEFAULT** *'value'* **FOR** *column*;
    - Example: ALTER TABLE Students ADD CONSTRAINT df_section DEFAULT 'Not yet enrolled' FOR Section;

References:

Coronel, C. and Morris, S. (2017). *Database systems: design, implementation, and management* (12th ed.). USA: Cengage Learning.

Elmasri, R. and Navathe, S. (2016). *Fundamentals of database systems* (7th ed.). USA: Pearson Higher Education.

Kroenke, D. and Auer, D. (2016). *Database processing: fundamentals, design, and implementation*. England: Pearson Education Limited.