

Лекція 06: Мова запитів SQL

Lecture 06: SQL query language

SQL

SQL – це структурована мова запитів для зберігання, обробки та отримання даних, що зберігаються в реляційній базі даних. SQL є стандартною мовою ANSI (American National Standards Institute), але існує безліч різних версій мови SQL.

SQL is Structured Query Language, used for storing, manipulating and retrieving data stored in a relational database. SQL is an ANSI (American National Standards Institute) standard language, but there are many different versions of the SQL language.

Історія SQL

1970 – Доктор Кодд з IBM описав реляційну модель для баз даних.

1974 – З'явилася мова SQL.

1978 – IBM працював над розробкою ідей Кодда і випустив продукт під назвою System/R.

1986 – IBM розробив перший прототип реляційної бази даних, стандартизований ANSI. Перша реляційна база даних була випущена компанією Relational Software, яка згодом стала відома як Oracle.

History of SQL

- 1970** – Dr. Edgar Codd of IBM described a relational model for databases.
- 1974** – Structured Query Language appeared.
- 1978** – IBM worked to develop Codd's ideas and released a product named System/R.
- 1986** – IBM developed the first prototype of relational database and standardized by ANSI. The first relational database was released by Relational Software which later came to be known as Oracle.

Використання SQL / SQL usage

Всі реляційні СУБД, такі як MySQL, MS Access, Oracle, Sybase, Informix, PostgreSQL та SQL Server використовують SQL як стандартну мову бази даних.

All the Relational Database Management Systems (RDMS) like MySQL, MS Access, Oracle, Sybase, Informix, PostgreSQL and SQL Server use SQL as their standard database language.

| Vendor | Relational DBMS | SQL Dialects |
|-------------|-----------------|--|
| Microsoft | SQL Server | T-SQL (Transact-SQL) |
| | Access | JET SQL (Native format) |
| Oracle | Oracle Database | PL/SQL (Procedural Language/SQL) |
| SAP | Sybase IQ | ANSI SQL (Native format) T-SQL |
| Open-Source | PostgreSQL | ANSI SQL PL/pgSQL (similar to PL/SQL) SQL:2011 |
| Oracle | MySQL | SQL:2003 |

Переваги SQL

- Дозволяє користувачам отримувати доступ до даних в реляційних СУБД.
- Дозволяє користувачам описувати дані.
- Дозволяє користувачам визначати дані в базі даних та маніпулювати ними.
- Дозволяє вбудовувати інші мови за допомогою модулів SQL, бібліотек та пре-компіляторів.
- Дозволяє користувачам створювати та видаляти бази даних та таблиці.
- Дозволяє користувачам створювати представлення, збережені процедури, функції в базі даних.
- Дозволяє користувачам встановлювати права на таблиці, процедури та представлення.

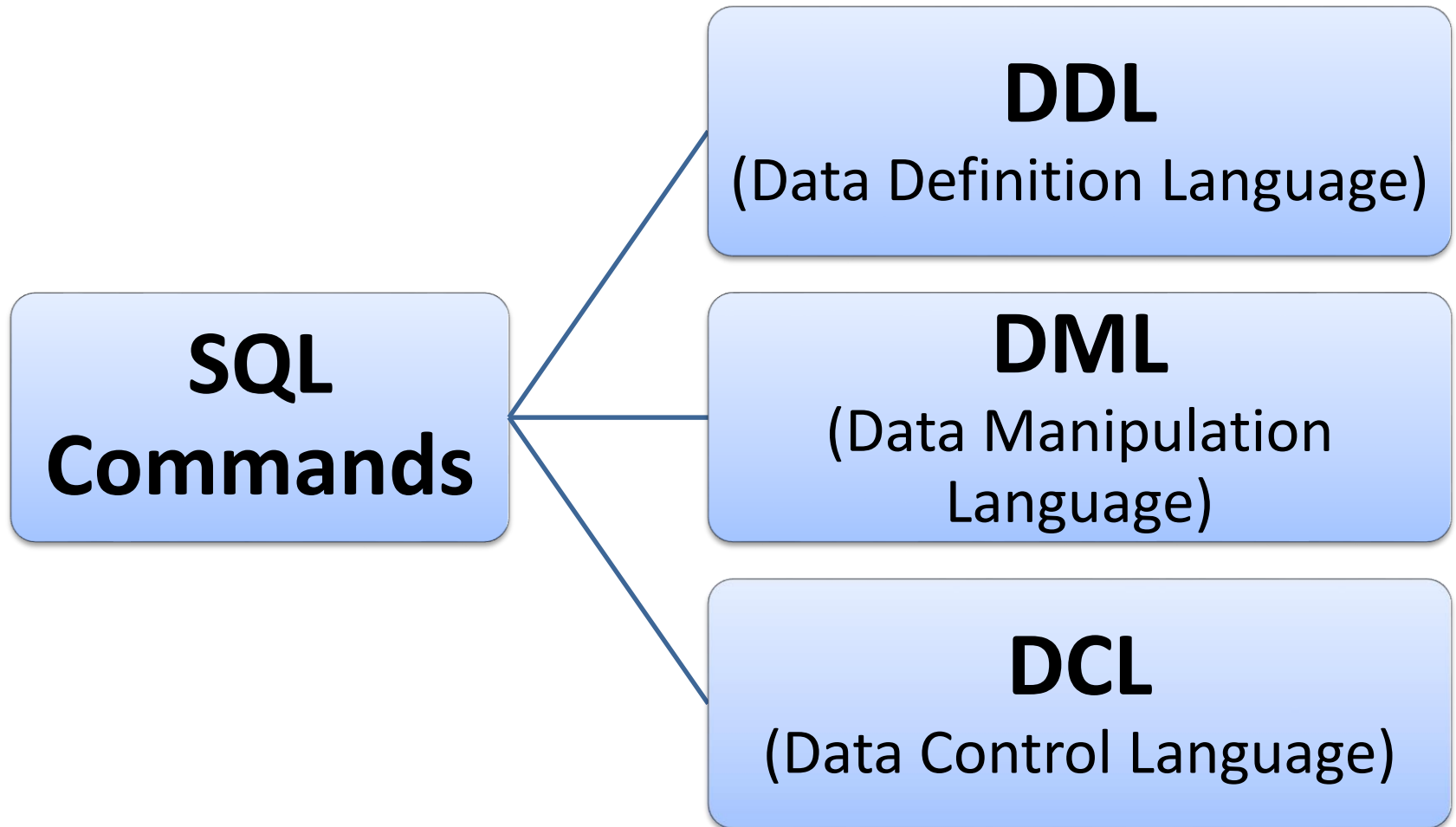
Advantages of SQL

- Allows users to access data in the relational database management systems.
- Allows users to describe the data.
- Allows users to define the data in a database and manipulate that data.
- Allows to embed within other languages using SQL modules, libraries & pre-compilers.
- Allows users to create and drop databases and tables.
- Allows users to create view, stored procedure, functions in a database.
- Allows users to set permissions on tables, procedures and views.

Команди SQL / SQL commands

Стандартні команди SQL для взаємодії з реляційними базами даних – CREATE, SELECT, INSERT, UPDATE, DELETE і DROP. Ці команди можна розділити на наступні групи залежно від їх характеру.

The standard SQL commands to interact with relational databases are CREATE, SELECT, INSERT, UPDATE, DELETE and DROP. These commands can be classified into the following groups based on their nature.



Data Definition Language (DDL)

| Номер | Команда та опис |
|-------|---|
| 1 | CREATE Створює нову таблицю, представлення таблиці або інший об'єкт у базі даних Creates a new table, a view of a table, or other object in the database |
| 2 | ALTER Модифікує існуючий об'єкт бази даних, такий як таблиця Modifies an existing database object, such as a table |
| 3 | DROP Видаляє цілу таблицю, представлення таблиці або інші об'єкти в базі даних Deletes an entire table, a view of a table or other objects in the database |

Data Manipulation Language (DML)

| Номер | Команда та опис |
|-------|---|
| 1 | SELECT Отримує певні записи з однієї або декількох таблиць Retrieves certain records from one or more tables |
| 2 | INSERT Створює запис Creates a record |
| 3 | UPDATE Модифікує записи Modifies records |
| 4 | DELETE Видаляє записи Deletes records |

Data Control Language (DCL)

| Номер | Команда та опис |
|-------|---|
| 1 | GRANT Дає привілей користувачу Gives a privilege to user |
| 2 | REVOKE Відміняє привілеї, надані користувачу Takes back privileges granted from user |

Процес виконання команди SQL / SQL command execution process

Коли ви виконуєте команду SQL для будь-якої реляційної СУБД, система визначає найкращий спосіб виконання вашого запиту, і движок SQL обчислює, як інтерпретувати завдання.

When you are executing an SQL command for any RDBMS, the system determines the best way to carry out your request and SQL engine figures out how to interpret the task.

У цьому процесі приймають участь різні
компоненти.

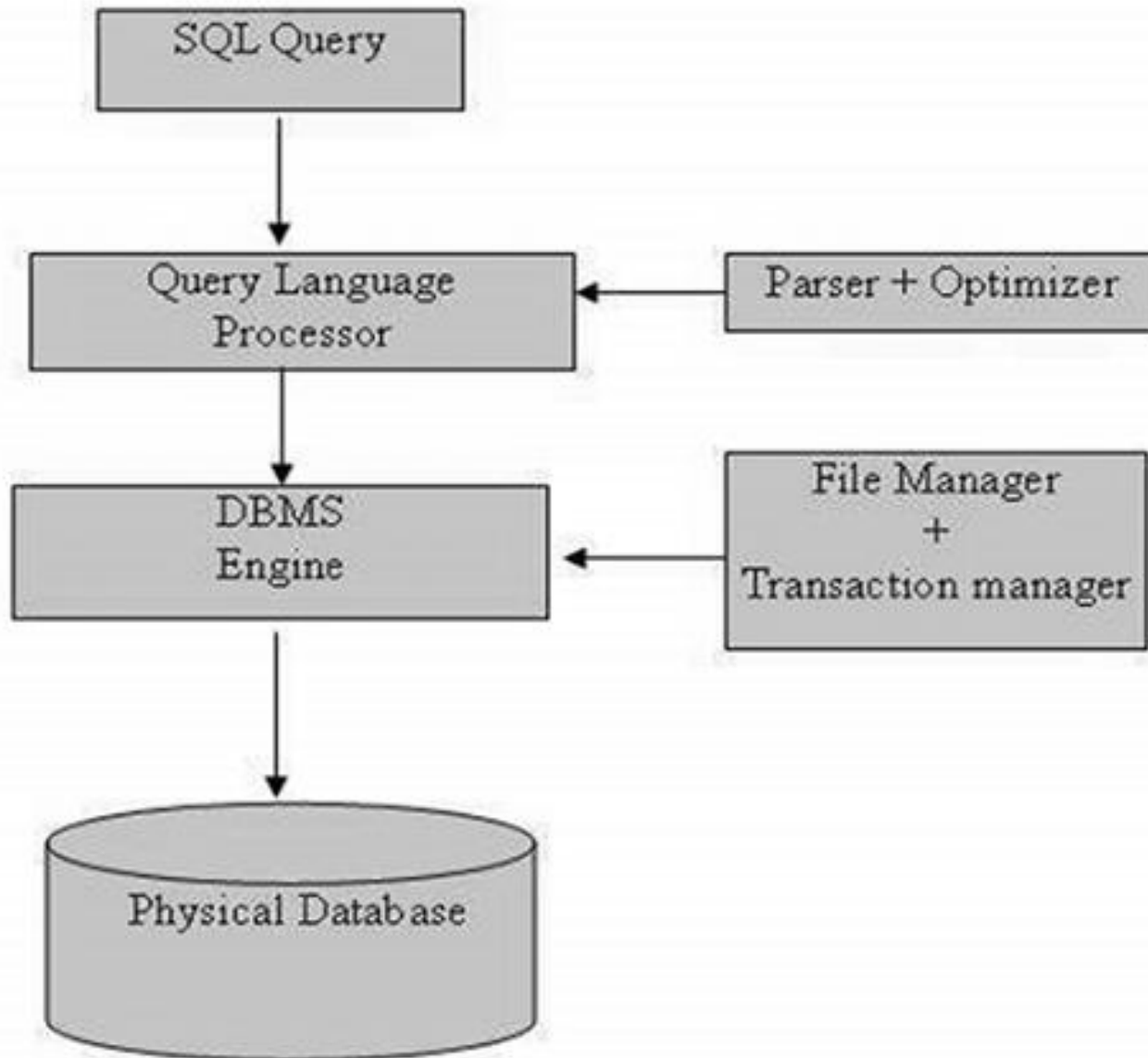
There are various components included in this
process.

Query Dispatcher (диспетчер запитів)

Optimization Engines (інструмент оптимізації)

Classic Query Language (движок запитів)

SQL Query Engine (движок запитів)



SQL data types

Тип даних SQL – це атрибут, який визначає тип даних будь-якого об'єкта. Кожен стовпець, змінна та вираз мають відповідний тип даних в SQL. Ці типи даних використовуються при створенні таблиць. Тип даних для стовпця таблиці обирається відповідно до вимог.

SQL Data Type is an attribute that specifies the type of data of any object. Each column, variable and expression has a related data type in SQL. You can use these data types while creating your tables. You can choose a data type for a table column based on your requirement.

Numeric data types

| DATA TYPE | FROM | TO |
|------------|----------------------------|---------------------------|
| bigint | -9,223,372,036,854,775,808 | 9,223,372,036,854,775,807 |
| int | -2,147,483,648 | 2,147,483,647 |
| smallint | -32,768 | 32,767 |
| tinyint | 0 | 255 |
| bit | 0 | 1 |
| decimal | $-10^{38} + 1$ | $10^{38} - 1$ |
| numeric | $-10^{38} + 1$ | $10^{38} - 1$ |
| money | -922,337,203,685,477.5808 | +922,337,203,685,477.5807 |
| smallmoney | -214,748.3648 | +214,748.3647 |
| float | $-1.79E + 308$ | $1.79E + 308$ |
| real | $-3.40E + 38$ | $3.40E + 38$ |

Date and time data types

| DATA TYPE | FROM | TO |
|---------------|--------------------------------------|--------------|
| datetime | Jan 1, 1753 | Dec 31, 9999 |
| smalldatetime | Jan 1, 1900 | Jun 6, 2079 |
| date | Stores a date like June 30, 1991 | |
| time | Stores a time of day like 12:30 P.M. | |

Character strings data types

| No. | DATA TYPE & Description |
|-----|---|
| 1 | char Maximum length of 8,000 characters. (Fixed length non-Unicode characters) |
| 2 | varchar Maximum of 8,000 characters. (Variable-length non-Unicode data). |
| 3 | varchar(max) Maximum length of 2^{E+31} characters. (Variable-length non-Unicode data (SQL Server 2005 only)). |
| 4 | text Variable-length non-Unicode data with a maximum length of 2,147,483,647 characters. |

Unicode character strings data type

| No. | DATA TYPE & Description |
|-----|---|
| 1 | nchar Maximum length of 4,000 characters. (Fixed length Unicode) |
| 2 | nvarchar Maximum length of 4,000 characters. (Variable length Unicode) |
| 3 | nvarchar(max) Maximum length of 2E + 31 characters (SQL Server 2005 only). (Variable length Unicode) |
| 4 | ntext Maximum length of 1,073,741,823 characters. (Variable length Unicode) |

Binary data types

| No. | DATA TYPE & Description |
|-----|---|
| 1 | binary Maximum length of 8,000 bytes. (Fixed-length binary data) |
| 2 | varbinary Maximum length of 8,000 bytes. (Variable length binary data) |
| 3 | varbinary(max) Maximum length of 2E + 31 bytes (SQL Server 2005 only). (Variable length Binary data) |
| 4 | image Maximum length of 2,147,483,647 bytes. (Variable length Binary Data) |