



Basics of Databases

Tessema Mengistu (Ph.D.)

Department of Computer Science

Virginia Tech

Mengistu@vt.edu



Outline

- Introduction
- Relational Databases
- SQL
 - Data Definition Language
 - Data Manipulation Language



Introduction

- A typical web-based application has three layers:
 - Presentation - handles user interactions on the client application
 - Application – handles the business logic
 - Data - manages data storage and management

Introduction

- Data source
 - Is where an application pulls data from
 - Can be
 - Database
 - Unstructured data (flat files)
 - Sensors
 - Web services / APIs
- Data storage
 - Cloud
 - DBaaS
 - On-premise
 - Local
 - Hybrid



Database

- The most common kind of data source
- Is a collection of related data
- Defining a database involves specifying the data types, structures, and constraints of the data to be stored in the database



Database

- A database management system (DBMS) is a general-purpose software system that facilitates the processes of defining, constructing, manipulating, and sharing databases among various users and applications
 - Example: MySQL, MongoDB
- **Query** typically causes some data to be retrieved
 - **CRUD** operations
- Transaction may cause some data to be read and some data to be written into the database
 - **ACID** property



Database

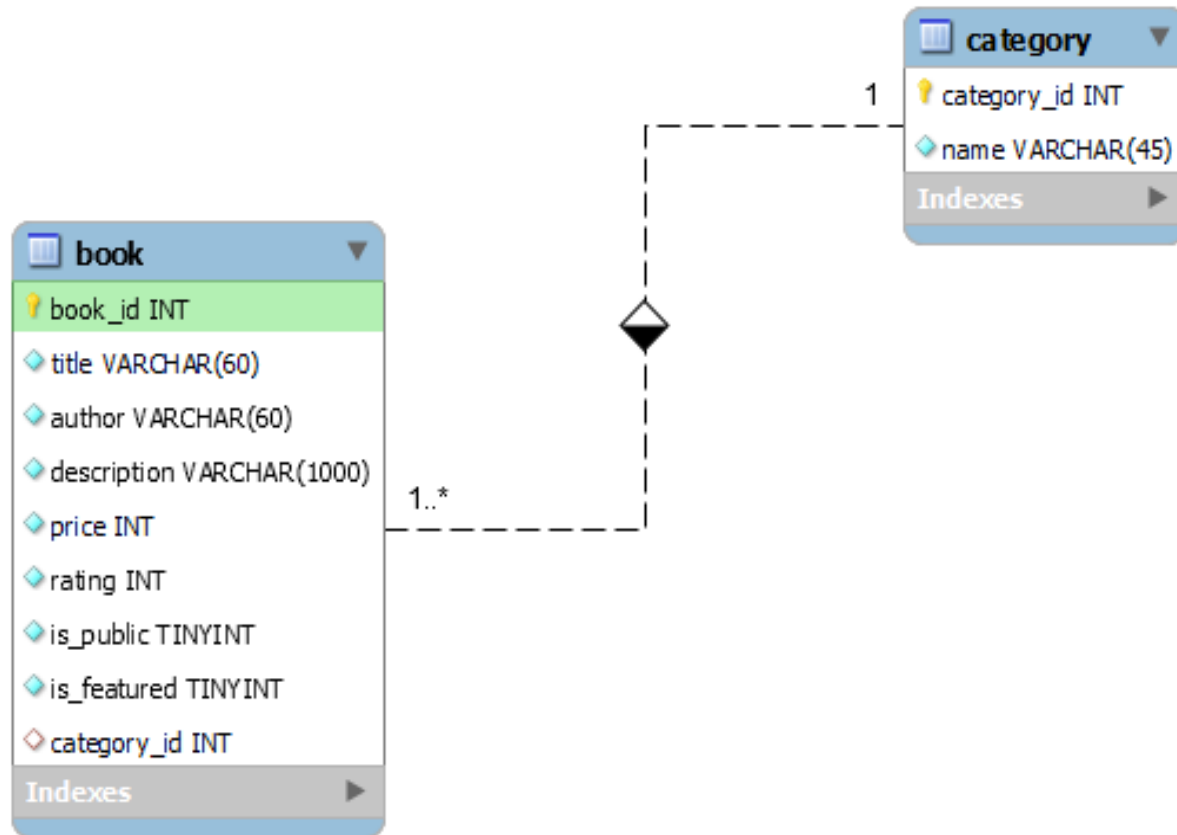
- Relational Databases
 - Organized data in terms of **tables** (relations)

Book_id	Title	Author	Price	Category_id
1001	The Iliad	Homer	6.99	1001
1002	The Brothers Karamazov	Fyodor Dostoyevski	10.55	1001
1003	Little Women	Louisa May Alcott	7.65	1005
1004	Little Dorrit	Charles Dickens	8.00	1006
...



Relational Database

- Relationships exist among tables





Structured Query Language (SQL)

- SQL is a standard language for accessing and manipulating relational databases
- SQL is a declarative language
 - Describes the type of information needed, not how the information is retrieved
- Can be divided into two parts:
 - The Data Definition Language (DDL)
 - The Data Manipulation Language (DML)



SQL - DDL

- Allows database tables to be created or deleted, specifies links between tables, and imposes constraints between tables, etc.
- Some important DDL statements in SQL:
 - *CREATE DATABASE* - creates a new database
 - *ALTER DATABASE* - modifies a database
 - *CREATE TABLE* - creates a new table
 - *ALTER TABLE* - modifies a table
 - *DROP TABLE* - deletes a table
 - *CREATE INDEX* - creates an index (search key)
 - *DROP INDEX* - deletes an index

SQL - DDL

- The CREATE DATABASE Statement
- SYNTAX:
 - *CREATE DATABASE database_name*

SQL - DDL

- The CREATE TABLE Statement
- Syntax

```
CREATE TABLE table_name  
(field1 datatype [constraints ],  
field2 datatype [constraints],  
field3 datatype [constraints]...)
```

- Constraints include:
 - PRIMARY KEY
 - FOREIGN KEY
 - NOT NULL
 - UNIQUE
 - CHECK
 - ...



SQL - DDL

- SQL data type
 - CHAR, VARCHAR, LONGVARCHAR
 - NUMERIC, DECIMAL
 - INTEGER
 - TINYINT, SMALLINT, BIGINT
 - BIT
 - REAL
 - FLOAT, DOUBLE
 - BINARY, VARBINARY, LONGVARBINARY
 - DATE, TIME, TIMESTAMP
 - BOOLEAN
 - BLOB
 - ...

SQL - DDL

- Implementing Constraints
 - Primary Key Constraints
 - Adding a primary key when creating a table

```
CREATE TABLE category  
(  
    category_id int PRIMARY KEY,  
    category_name varchar(50) NOT NULL  
)
```

SQL - DDL

- Implementing Constraints
 - Referential integrity Constraints
 - Adding a primary key when creating a table

```
CREATE TABLE book
(
    book_id int PRIMARY KEY,
    title varchar(50) NOT NULL,
    author varchar(50) NOT NULL,
    . . .
    FOREIGN KEY(category_id) REFERENCES
        (category)
)
```



SQL - DML

- The query and update commands form the DML part of SQL:
 - *INSERT* – inserts a new data to the database
 - *SELECT* - extracts data from a database
 - *UPDATE* - updates data in a database
 - *DELETE* - deletes data from a database



SQL - DML

- Inserting Records (INSERT SQL Command)
- Syntax:

```
INSERT INTO    tablename  
[(first_column,...,last_column)]  
VALUES (first_value,...,last_value)
```



SQL - DML

- Updating Records (UPDATE SQL Command)
- Syntax:

UPDATE tablename

SET columnname = "newvalue"

[, "nextcolumn" = "newvalue2"...]

WHERE columnname OPERATOR value

[AND|OR columnname OPERATOR value]



SQL - DML

- Deleting Records (DELETE Command)
- Syntax:

```
DELETE [FROM] table_name  
[WHERE search_condition]
```

SQL - DML

- Selecting Records from a table
- Syntax:

```
SELECT [ALL | DISTINCT] column1 [,  
column2]  
FROM table1 [, table2]  
[WHERE conditions]  
[GROUP BY column-list]  
[HAVING conditions]  
[ORDER BY column-list [ASC | DESC] ]
```



SQL - DML

- Comparison Operators used in the *WHERE* clause are the following:
 - = Equal
 - > Greater than
 - < Less than
 - >= Greater than or equal
 - <= Less than or equal
 - <> or != Not equal to
 - ...



SQL - DML

- Aggregate functions
 - Are used to summarize the results of a particular column
 - These are:
 - *MIN ()*
 - *MAX ()*
 - *SUM ()*
 - *AVG ()*
 - *COUNT ()*
 - . . .



SQL - DML

- *GROUP BY* **clause**

- Gather all of the rows together that contain data in the specified column(s) and will allow aggregate functions to be performed on the one or more columns

SQL - DML

- *HAVING* clause
 - Similar to where condition except that HAVING is used with group of records instead of single records
 - Example:
 - Show all the average salaries of departments that are greater than 1200

SQL - DML

- *ORDER BY* clause

- Syntax:

```
SELECT column1, column2  
FROM list-of-tables  
ORDER BY column-list [ASC | DESC];
```



SQL - DML

Table Joins

```
SELECT list-of-columns  
FROM table1, table2  
WHERE search-condition(s)
```



References

- Fundamentals of database systems. Elmasri, Ramez, ; Navathe, Sham. 7th Edition, 2016