1. Introduction to Databases

1.1 Database basics

Data

Data is numeric, textual, visual, or audio information that describes real-world systems.

analog

Historically, data was mostly analog, encoded as continuous variations on various physical media.

digital

Today, data is mostly digital, encoded as zeros and ones on electronic and magnetic media.

database

A database is a collection of data in a structured format. In principle, databases can be stored on paper or even clay tablets. In practice, however, modern databases are invariably stored on computers.

database system / database management system / DBMS

A database system, also known as a database management system or DBMS, is software that reads and writes data in a database. Database systems ensure data is secure, internally consistent, and available at all times. These functions are challenging for large databases with many users, so database systems are complex.

query language

A query language is a specialized programming language, designed specifically for database systems.

database application

A database application is software that helps business users interact with database systems.

database administrator

A database administrator is responsible for securing the database system against unauthorized users. A database administrator enforces procedures for user access and database system availability.

database designer

A database designer determines the format of each data element and the overall database structure. Database designers must balance several priorities, including storage, response time, and support for rules that govern the data. Since these priorities often conflict, database design is technically challenging.

database programmer

A database programmer develops computer programs that utilize a database.

database user

A database user is a consumer of data in a database. Database users request, update, or use stored data to generate reports or information. Database users usually access the database via applications but can also submit queries directly to the database system.

1.2 Database systems

transaction

A transaction is a group of queries that must be either completed or rejected as a whole. Execution of some, but not all, queries results in inconsistent or incorrect data.

architecture

The architecture of a database system describes the internal components and the relationships between components.

query processor

The query processor interprets queries, creates a plan to modify the database or retrieve data, and returns query results to the application.

query optimization

The query processor performs query optimization to ensure the most efficient instructions are executed on the data.

storage manager

The storage manager translates the query processor instructions into low-level file-system commands that modify or retrieve data.

indexes

The storage manager uses indexes to quickly locate data.

transaction manager

The transaction manager ensures transactions are properly executed.

log

The log is a file containing a complete record of all inserts, updates, and deletes processed by the database.

catalog / data dictionary

The catalog, also known as a data dictionary, is a directory of tables, columns, indexes, and other database objects.

relational database

A relational database stores data in tables, columns, and rows, similar to a spreadsheet.

SQL

SQL stands for Structured Query Language and includes statements that read and write data, create and delete tables, and administer the database system.

big data

The growth of the internet in the 1990s generated massive volumes of online data, called big data, often with poorly structured or missing information.

NoSQL

The newer non-relational systems are called NoSQL, for 'not only SQL', and are optimized for big data.

Open source

Open source software is software that anyone can inspect, copy, and modify with no licensing fee.

1.3 Query languages

query

A query is a command for a database that typically inserts new data, retrieves data, updates data, or deletes data from a database.

query language

A query language is a computer programming language for writing database queries.

CRUD

The four common queries are sometimes referred to as CRUD operations, an acronym for Create, Read, Update, and Delete data.

Structured Query Language / SQL

Structured Query Language, or SQL, is the standard query language of relational database systems.

statement

An SQL statement is a database command, such as a query that inserts, selects, updates, or deletes data: .

INSERT

INSERT inserts rows into a table.

SELECT

SELECT retrieves data from a table.

UPDATE

UPDATE modifies data in a table.

DELETE

DELETE deletes rows from a table.

CREATE TABLE

The SQL CREATE TABLE statement creates a new table by specifying the table and column names.

data type

Each column is assigned a data type that indicates the format of column values. Data types can be numeric, textual, or complex.

1.4 Database design and programming

database design

A database design is a specification of database objects such as tables, columns, data types, and indexes. Database design also refers to the process used to develop the specification.

analysis

The analysis phase specifies database requirements without regard to a specific database system.

ER diagrams

Entities, relationships, and attributes are depicted in ER diagrams: .

logical design

The logical design phase implements database requirements in a specific database system.

key

A key is a column used to identify individual rows of a table.

table diagram

The logical design is depicted in a table diagram.

schema

The logical design, as specified in SQL and depicted in a table diagram, is called a database schema.

physical design

The physical design phase adds indexes and specifies how tables are organized on storage media.

data independence

The principle that physical design never affects query results is called data independence.

application programming interface / API

An application programming interface, or API, is a library of procedures or classes that links a host programming language to a database.

1.5 MySQL

MySQL

MySQL is a leading relational database system sponsored by Oracle.

MySQL Community / MySQL Server

MySQL Community, commonly called MySQL Server, is a free edition.

MySQL Enterprise

MySQL Enterprise is a paid edition for managing commercial databases. MySQL Enterprise includes MySQL Server and additional administrative applications.

root account

The root account, the administrative account that has full control of MySQL.

MySQL Command-Line Client

The MySQL Command-Line Client is a text interface included in the MySQL Server download. The Command-Line Client allows developers to connect to the database server, perform administrative functions, and execute SQL statements.

error code

MySQL Server returns an error code and description when an SQL statement is syntactically incorrect or the database cannot execute the statement.

MySQL Workbench

MySQL Workbench is installed with MySQL Server and allows developers to execute SQL commands using an editor.

2. Relational Databases

2.1 Relational model

database model

A database model is a conceptual framework for database systems, with three parts:

Data structures that prescribe how data is organized.

Operations that manipulate data structures.

Rules that govern valid data.

.

relational model

The relational model is a database model based on a tabular data structure. The model was published in 1970 by E. F. Codd of IBM and released in commercial products around 1980. The data structure, operations, and rules are standardized in SQL, the universal query language of relational databases.

big data

The rise of the internet in the 1990s generated big data, characterized by unprecedented data volumes and rapidly changing data structures.

set

A set is an unordered collection of elements enclosed in braces.

tuple

A tuple is an ordered collection of elements enclosed in parentheses.

table

A table has a name, a fixed tuple of columns, and a varying set of rows.

column

A column has a name and a data type.

row

A row is an unnamed tuple of values. Each value corresponds to a column and belongs to the column's data type.

data type

A data type is a named set of values, from which column values are drawn.

relational algebra

These operations are collectively called relational algebra and are the theoretical foundation of the SQL language.

Relational rules

Relational rules are part of the relational model and govern data in every relational database.

Business rules

Business rules are based on business policy and specific to a particular database.

constraints

Relational rules are implemented as SQL constraints and enforced by the database system.

2.2 Structured Query Language

Structured Query Language / SQL

Structured Query Language (SQL) is a high-level computer language for storing, manipulating, and retrieving data.

statement

An SQL statement is a complete command composed of one or more clauses.

clause

A clause groups SQL keywords like SELECT, FROM, and WHERE with table names like City, column names like Name, and conditions like Population > 100000.

Data Definition Language

Data Definition Language (DDL) defines the structure of the database.

Data Query Language

Data Query Language (DQL) retrieves data from the database.

Data Manipulation Language

Data Manipulation Language (DML) manipulates data stored in a database.

Data Control Language

Data Control Language (DCL) controls database user access.

Data Transaction Language

Data Transaction Language (DTL) manages database transactions.

2.3 Managing databases

database system instance

A database system instance is a single executing copy of a database system. Personal computers usually run just one instance of a database system. Shared computers, such as computers used for cloud services, usually run multiple instances of a database system.

CREATE DATABASE DatabaseName

CREATE DATABASE DatabaseName creates a new database.

DROP DATABASE DatabaseName

DROP DATABASE DatabaseName deletes a database, including all tables in the database.

USE DatabaseName

USE DatabaseName selects a default database for use in subsequent SQL statements.

SHOW DATABASES

SHOW DATABASES lists all databases in the database system instance.

SHOW TABLES

SHOW TABLES lists all tables in the default database.

SHOW COLUMNS FROM TableName

SHOW COLUMNS FROM TableName lists all columns in the TableName table of the default database.

SHOW CREATE TABLE TableName

SHOW CREATE TABLE TableName shows the CREATE TABLE statement for the TableName table of the default database.

2.4 Tables

table

A table has a name, a fixed sequence of columns, and a varying set of rows.

column

A column has a name and a data type.

row

A row is an unnamed sequence of values. Each value corresponds to a column and belongs to the column's data type.

cell

A cell is a single column of a single row.

empty table

A table without rows is called an empty table.

data independence

Rule 4 is called data independence.

CREATE TABLE

The CREATE TABLE statement creates a new table by specifying the table name, column names, and column data types.

DROP TABLE

The DROP TABLE statement deletes a table, along with all the table's rows, from a database.

ALTER TABLE

The ALTER TABLE statement adds, deletes, or modifies columns on an existing table.

2.5 Data types

data type

A data type is a named set of values from which column values are drawn.

Integer

Integer data types represent positive and negative integers.

Decimal

Decimal data types represent numbers with fractional values.

Character

Character data types represent textual characters.

Date and time

Date and time data types represent date, time, or both. Some date and time data types include a time zone or specify a time interval.

Binary

Binary data types store data exactly as the data appears in memory or computer files, bit for bit.

Spatial

Spatial data types store geometric information, such as lines, polygons, and map coordinates.

Document

Document data types contain textual data in a structured format such as XML or JSON.

signed

A signed number may be negative.

unsigned

An unsigned number cannot be negative.

2.6 Selecting rows

operator / operands

An operator is a symbol that computes a value from one or more other values, called operands:

Arithmetic operators compute numeric values from numeric operands.

Comparison operators compute logical values TRUE or FALSE. Operands may be numeric, character, and other data types.

Logical operators compute logical values from logical operands.

.

unary

A unary operator has one operand.

binary

A binary operator has two operands.

expression

An expression is a string of operators, operands, and parentheses that evaluates to a single value. Operands may be column names or fixed values. The value of an expression may be any data type.

operator precedence

Operators in an expression are evaluated in the order of operator precedence, shown in the table below. Operators of the same precedence are evaluated from left to right. Regardless of operator precedence, expressions enclosed in parentheses are evaluated before any operators outside the parentheses are applied.

SELECT / FROM

The SELECT statement selects rows from a table. The statement has a SELECT clause and a FROM clause. The FROM clause specifies the table from which rows are selected. The SELECT clause specifies one or more expressions, separated by commas, that determine what values are returned for each row.

result table

The SELECT statement returns a set of rows, called the result table.

LIMIT

MySQL has a LIMIT clause that limits the number of rows returned by a SELECT statement.

condition

A condition is an expression that evaluates to a logical value.

WHERE

A SELECT statement has an optional WHERE clause that specifies a condition for selecting rows. A row is selected when the condition is TRUE for the row values. A row is omitted when the condition is either FALSE or NULL.

2.7 Null values

NULL

NULL is a special value that represents either unknown or inapplicable data.

NOT NULL

The NOT NULL constraint prevents a column from having a NULL value. Statements that insert NULL, or update a value to NULL, are automatically rejected. NOT NULL follows the column name and data type in a CREATE TABLE statement.

IS NULL / IS NOT NULL

Instead, the IS NULL and IS NOT NULL operators must be used to select NULL values. Value IS NULL returns TRUE when the value is NULL. Value IS NOT NULL returns TRUE when the value is not NULL.

truth tables

The value of logical expressions containing NULL operands is defined in truth tables.

2.8 Inserting, updating, and deleting rows

INSERT

The INSERT statement adds rows to a table.

INSERT INTO

The INSERT INTO clause names the table and columns where data is to be added. The keyword INTO is optional.

VALUES

The VALUES clause specifies the column values to be added.

DEFAULT

The optional DEFAULT keyword and default value follow the column name and data type in a CREATE TABLE statement. The column is assigned the default value, rather than NULL, when omitted from an INSERT statement.

UPDATE

The UPDATE statement modifies existing rows in a table.

SET

The UPDATE statement uses the SET clause to specify the new column values.

DELETE

The DELETE statement deletes existing rows in a table.

FROM

The FROM keyword is followed by the table name whose rows are to be deleted.

TRUNCATE

The TRUNCATE statement deletes all rows from a table.

2.9 Primary keys

primary key

A primary key is a column, or group of columns, used to identify a row.

simple primary key

A simple primary key consists of a single column.

composite primary key

A composite primary key consists of multiple columns.

Minimal

Minimal. All primary key columns are necessary for uniqueness. When any column is removed, the resulting simple or composite column is no longer unique.

PRIMARY KEY

The PRIMARY KEY constraint in a CREATE TABLE statement names the table's primary key. The PRIMARY KEY constraint ensures that a column or group of columns is always unique and non-null.

auto-increment column

An auto-increment column is a numeric column that is assigned an automatically incrementing value when a new row is inserted.

AUTO\_INCREMENT

The AUTO\_INCREMENT keyword defines an auto-increment column. AUTO\_INCREMENT follows the column's data type in a CREATE TABLE statement.

2.10 Foreign keys

foreign key

A foreign key is a column, or group of columns, that refer to a primary key.

Referential integrity

Referential integrity requires foreign key values must either be NULL or match some value of the referenced primary key.

FOREIGN KEY / REFERENCES

A foreign key constraint is added to a CREATE TABLE statement with the FOREIGN KEY and REFERENCES keywords.

2.11 Referential integrity

fully NULL

A fully NULL foreign key is a simple or composite foreign key in which all columns are NULL.

Referential integrity

Referential integrity is a relational rule that requires foreign key values are either fully NULL or match some primary key value.

RESTRICT

RESTRICT rejects an insert, update, or delete that violates referential integrity.

SET NULL

SET NULL sets invalid foreign keys to NULL.

SET DEFAULT

SET DEFAULT sets invalid foreign keys to the foreign key default value.

CASCADE

CASCADE propagates primary key changes to foreign keys.

ON UPDATE / ON DELETE

Actions are specified in the optional ON UPDATE and ON DELETE clauses of the FOREIGN KEY constraint. ON UPDATE and ON DELETE are followed by either RESTRICT, SET NULL, SET DEFAULT, or CASCADE.

2.12 Constraints

constraint

A constraint is a rule that governs allowable values in a database. Constraints are based on relational and business rules, and implemented with special keywords in a CREATE TABLE statement. The database automatically rejects insert, update, and delete statements that violate a constraint.

column constraint

A column constraint appears after the column name and data type in a CREATE TABLE statement. Column constraints govern values in a single column.

table constraint

A table constraint appears in a separate clause of a CREATE TABLE statement and governs values in one or more columns.

UNIQUE

The UNIQUE constraint ensures that values in a column, or group of columns, are unique.

CHECK

The CHECK constraint specifies an expression on one or more columns of a table. The constraint is violated when the expression is FALSE and satisfied when the expression is either TRUE or NULL.

CONSTRAINT

Table constraints may be named using the optional CONSTRAINT keyword, followed by the constraint name and declaration.

3. Complex Queries

3.1 Special operators and clauses

IN

The IN operator is used in a WHERE clause to determine if a value matches one of several values.

BETWEEN

The BETWEEN operator provides an alternative way to determine if a value is between two other values. The operator is written value BETWEEN minValue AND maxValue and is equivalent to value >= minValue AND value <= maxValue.

LIKE

The LIKE operator, when used in a WHERE clause, matches text against a pattern using the two wildcard characters % and \_.

BINARY

The LIKE operator performs case-insensitive pattern matching by default or case-sensitive pattern matching if followed by the BINARY keyword.

DISTINCT

The DISTINCT clause is used with a SELECT statement to return only unique or 'distinct' values.

ORDER BY

The ORDER BY clause orders selected rows by one or more columns in ascending (alphabetic or increasing) order.

DESC

The DESC keyword with the ORDER BY clause orders rows in descending order.

3.2 Simple functions

function / argument

A function operates on an expression enclosed in parentheses, called an argument, and returns a value. Usually, the argument is a simple expression, such as a column name or fixed value. Some functions have several arguments, separated by commas, and a few have no arguments at all.

3.3 Aggregate functions

aggregate function

An aggregate function processes values from a set of rows and returns a summary value.

COUNT()

COUNT() counts the number of rows in the set.

MIN()

MIN() finds the minimum value in the set.

MAX()

MAX() finds the maximum value in the set.

SUM()

SUM() sums all the values in the set.

AVG()

AVG() computes the arithmetic mean of all the values in the set.

GROUP BY

The GROUP BY clause consists of the GROUP BY keyword and one or more columns. Each simple or composite value of the column(s) becomes a group. The query computes the aggregate function separately, and returns one row, for each group.

HAVING

The HAVING clause is used with the GROUP BY clause to filter group results.

3.4 Join queries

join / left table / right table

A join is a SELECT statement that combines data from two tables, known as the left table and right table, into a single result.

AS

To simplify queries or result tables, a column name can be replaced with an alias. The alias follows the column name, separated by an optional AS keyword.

join clause

A join clause determines how a join query handles unmatched rows. Two common join clauses are: .

INNER JOIN

INNER JOIN selects only matching left and right table rows.

FULL JOIN

FULL JOIN selects all left and right table rows, regardless of match.

ON

The ON clause specifies the join columns.

LEFT JOIN

LEFT JOIN selects all left table rows, but only matching right table rows.

RIGHT JOIN

RIGHT JOIN selects all right table rows, but only matching left table rows.

outer join

An outer join is any join that selects unmatched rows, including left, right, and full joins.

UNION

The UNION keyword combines the two results into one table.

3.5 Equijoins, self-joins, and cross-joins

equijoin

An equijoin compares columns of two tables with the = operator.

non-equijoin

A non-equijoin compares columns with an operator other than =, such as < and >.

self-join

A self-join joins a table to itself.

cross-join

A cross-join combines two tables without comparing columns.

CROSS JOIN

A cross-join uses a CROSS JOIN clause without an ON clause.

3.6 Subqueries

subquery / nested query / inner query

A subquery, sometimes called a nested query or inner query, is a query within another SQL query.

correlated

A subquery is correlated when the subquery's WHERE clause references a column from the outer query.

alias

An alias is a temporary name assigned to a column or table.

AS

The AS keyword follows a column or table name to create an alias.

EXISTS

Correlated subqueries commonly use the EXISTS operator, which returns TRUE if a subquery selects at least one row and FALSE if no rows are selected.

NOT EXISTS

The NOT EXISTS operator returns TRUE if a subquery selects no rows and FALSE if at least one row is selected.

flattening

Replacing a subquery with an equivalent join is called flattening a query.

3.8 View tables

view table / view query

A view table is a table name associated with a SELECT statement, called the view query.

CREATE VIEW

The CREATE VIEW statement creates a view table and specifies the view name, query, and, optionally, column names. If column names are not specified, column names are the same as in the view query result table.

base table

A table specified in the view query's FROM clause is called a base table.

materialized view

A materialized view is a view for which data is stored at all times.

WITH CHECK OPTION

When WITH CHECK OPTION is specified, the database rejects inserts and updates that do not satisfy the view query WHERE clause.

3.9 Relational algebra

relational algebra

In his original paper on the relational model, E. F. Codd introduced formal operations for manipulating tables. Codd's operations, called relational algebra, have since been refined and are the theoretical foundation of SQL.

select operation

The select operation selects table rows based on a logical expression. The select operation is written as

and is equivalent to SELECT \* FROM Table WHERE expression.

project operation

The project operation selects table columns. The project operation is written as

and is equivalent to SELECT Column1, Column2, ... FROM Table.

product operation

The product operation combines two tables into one result. The result includes all columns and all combinations of rows from both tables. The product operation is written as

and is equivalent to SELECT \* FROM Table1 CROSS JOIN Table2.

join

The join operation, denoted with a "bowtie" symbol, is written as

and is identical to a product followed by a select:

The join operation is equivalent to SELECT \* FROM Table1 INNER JOIN Table2 ON expression.

theta join

Because of theta notation, the join operation is sometimes called a theta join.

Compatible tables

Compatible tables have the same number of columns with the same data types. Column names may be different.

set operations

Union, intersect, and difference operate on compatible tables and, collectively, are called set operations.

union

The union operation combines all rows of two compatible tables into a single table. Duplicate rows are excluded from the result table. The union operation is written as

and is equivalent to SELECT \* FROM Table1 UNION SELECT \* FROM Table2.

Intersect

Intersect operates on two compatible tables and returns only rows that appear in both tables. The intersect operation is written as

and is equivalent to SELECT \* FROM Table1 INTERSECT SELECT \* FROM Table2.

difference

The difference operation removes from a table all rows that appear in a second compatible table. The difference operation is written as

and is equivalent to SELECT \* FROM Table1 MINUS SELECT \* FROM Table2.

rename operation

The rename operation specifies new table and column names. The rename operation is written as

.

aggregate operation

The aggregate operation applies aggregate functions like SUM(), AVG(), MIN(), and MAX(). The aggregate operation is written as

and is equivalent to SELECT GroupColumn, Function(Column) FROM Table GROUP BY GroupColumn.

equivalent

Relational algebra expressions are equivalent if the expressions operate on the same tables and generate the same result.

query optimizer / query execution plan

A query optimizer converts an SQL query into a sequence of low-level database actions, called the query execution plan. The query execution plan specifies precisely how to process an SQL statement.

cost

The cost of an operation is a numeric estimate of processing time. The cost estimate usually combines both storage media access and computation time in a single measure.

4. Database Design

4.1 Entities, relationships, and attributes

entity-relationship model

An entity-relationship model is a high-level representation of data requirements, ignoring implementation details.

entity

An entity is a person, place, product, concept, or activity.

relationship

A relationship is a statement about two entities.

attribute

An attribute is a descript­ive property of an entity.

reflexive relationship

A reflexive relationship relates an entity to itself.

entity-relationship diagram / ER diagram

An entity-relationship diagram, commonly called an ER diagram, is a schematic picture of entities, relationships, and attributes.

glossary / data dictionary / repository

A glossary, also known as a data dictionary or repository, documents additional detail in text format.

entity type

An entity type is a set of things. Ex: All employees in a company.

relationship type

A relationship type is a set of related things. Ex: Employee-Manages-Department is a set of (employee, department) pairs, where the employee manages the department.

attribute type

An attribute type is a set of values. Ex: All employee salaries.

entity instance

An entity instance is an individual thing. Ex: The employee Sam Snead.

relationship instance

A relationship instance is a statement about entity instances. Ex: "Maria Rodriguez manages Sales." .

attribute instance

An attribute instance is an individual value. Ex: The salary $35,000.

Analysis

Analysis develops an entity-relationship model, capturing data requirements while ignoring implementation details.

Logical design

Logical design converts the entity-relationship model into tables, columns, and keys for a particular database system.

Physical design

Physical design adds indexes and specifies how tables are organized on storage media.

4.3 Cardinality

cardinality

In entity-relationship modeling, cardinality refers to maxima and minima of relationships and attributes.

Relationship maximum

Relationship maximum is the greatest number of instances of one entity that can relate to a single instance of another entity.

singular / plural

A related entity is singular when the maximum is one and plural when the maximum is many.

Relationship minimum

Relationship minimum is the least number of instances of one entity that can relate to a single instance of another entity.

optional / required

A related entity is optional when the minimum is zero and required when the minimum is one.

Attribute maximum

Attribute maximum is the greatest number of attribute values that can describe each entity instance. Attribute maximum is usually specified as one (singular) or many (plural).

Attribute minimum

Attribute minimum is the least number of attribute values that can describe each entity instance. Attribute minimum is usually specified as zero (optional) or one (required).

unique attribute

Each value of a unique attribute describes at most one entity instance.

4.4 Strong and weak entities

identifying attribute

An identifying attribute is unique, singular, and required.

identify

Identifying attribute values correspond one-to-one to, or identify, entity instances.

strong entity

A strong entity has one or more identifying attributes.

weak entity / identifying relationship / identifying entity

A weak entity does not have an identifying attribute. Instead, a weak entity usually has a relationship, called an identifying relationship, to another entity, called an identifying entity. Cardinality of the identifying entity is 1(1).

4.5 Supertype and subtype entities

subtype entity / supertype entity

A subtype entity is a subset of another entity type, called the supertype entity.

IsA relationship

A supertype entity identifies its subtype entities. The identifying relationship is called an IsA relationship.

Similar entities

Similar entities are entities that have many common attributes and relationships.

partition

A partition of a supertype entity is a group of mutually exclusive subtype entities.

partition attribute

Each partition corresponds to an optional partition attribute of the supertype entity.

4.6 Alternative modeling conventions

crow's foot notation

Variations in cardinality conventions are common. One popular convention, called crow's foot notation, depicts cardinality as a circle (zero), a short line (one), or three short lines (many). The three short lines look like a bird's foot, hence the name "crow's foot notation".

subject area

Decompose a complex model into a group of related entities, called a subject area.

independent / dependent

Refer to strong entities as independent and weak entities as dependent.

Unified Modeling Language / UML

Unified Modeling Language, or UML, is commonly used for software development. Software data structures are similar to database structures, so UML includes ER conventions.

IDEF1X

IDEF1X stands for Information DEFinition version 1X. IDEF1X became popular, in part, due to early adoption by the United States Department of Defense.

Chen notation

Chen notation appeared in an early ER modeling paper by Peter Chen. Chen notation is not standardized but often appears in literature and tools.

4.7 Implementing entities

strong table

A strong entity becomes a strong table. The primary key must be unique and non-NULL, and should be stable, simple, and meaningless. Single-column primary keys are best, but if no such column exists, a composite primary key may have the required properties.

artificial key

An artificial key is a single-column primary key created by the database designer when no suitable single-column or composite primary key exists.

subtype table

A subtype entity becomes a subtype table .

weak table

A weak entity becomes a weak table.

4.10 First, second, and third normal form

depends on

Column A depends on column B means each B value is related to at most one A value.

functional dependence

Dependence of one column on another is called functional dependence.

Multivalued dependence / join dependence

Multivalued dependence and join dependence entail dependencies between three or more columns.

Redundancy

Redundancy is the repetition of related values in a table.

Normal forms

Normal forms are rules for designing tables with less redundancy.

first normal form

Every cell of a table contains exactly one value. A table is in first normal form when, in addition, the table has a primary key.

second normal form

A table is in second normal form when all non-key columns depend on the whole primary key.

third normal form

Informally, a table is in third normal form when all non-key columns depend on the key, the whole key, and nothing but the key. A formal definition appears elsewhere in this material.

4.11 Boyce-Codd normal form

candidate key / Minimal

A candidate key is a simple or composite column that is unique and minimal. Minimal means all columns are necessary for uniqueness.

non-key

A non-key column is a column that is not contained in a candidate key.

third normal form

A table is in third normal form if, whenever a non-key column A depends on column B, then B is unique. Columns A and B may be simple or composite.

Boyce-Codd normal form

A table is in Boyce-Codd normal form if, whenever column A depends on column B, then B is unique. Columns A and B may be simple or composite.

4.12 Applying normal form

Normalization

Normalization eliminates redundancy by decomposing a table into two or more tables in higher normal form.

depends on

Column A depends on column B when each B value is related to at most one A value. A and B may be simple or composite columns.

Boyce-Codd normal form

In a Boyce-Codd normal form table, if column A depends on column B, then B must be unique.

Denormalization

Denormalization means intentionally introducing redundancy by merging tables.

5. Data Storage

5.1 Storage media

Access time

Access time is the time required to access the first byte in a read or write operation.

Transfer rate

Transfer rate is the speed at which data is read or written, following initial access.

Volatile memory

Volatile memory is memory that is lost when disconnected from power.

Non-volatile memory

Non-volatile memory is retained without power.

Main memory / random-access memory (RAM)

Main memory, also called random-access memory (RAM), is the primary memory used when computer programs execute.

Flash memory / solid-state drive (SSD)

Flash memory, also called solid-state drive (SSD), is less expensive and higher capacity than main memory.

Magnetic disk / hard-disk drive (HDD)

Magnetic disk, also called hard-disk drive (HDD), is used to store large amounts of data.

sectors

Magnetic disk groups data in sectors, traditionally 512 bytes per sector but 4 kilobytes with newer disk formats.

pages

Flash memory groups data in pages, usually between 2 kilobytes and 16 kilobytes per page.

block

Databases and file systems use a uniform size, called a block, when transferring data between main memory and storage media.

row-oriented storage

To minimize block transfers, relational databases usually store an entire row within one block, which is called row-oriented storage.

column-oriented / columnar storage

In column-oriented storage, also called columnar storage, each block stores values for a single column only.

5.2 Table structures

table structure

A table structure is a scheme for organizing rows in blocks on storage media.

heap table

In a heap table, no order is imposed on rows.

sorted table / sort column

In a sorted table, the database designer identifies a sort column that determines physical row order.

hash table

In a hash table, rows are assigned to buckets.

bucket

A bucket is a block or group of blocks containing rows.

hash key

The hash key is a column or group of columns, usually the primary key.

hash function

The hash function computes the bucket containing the row from the hash key.

modulo function

The modulo function is a simple hash function with four steps.

dynamic hash function

A dynamic hash function automatically allocates more blocks to the table, creates additional buckets, and distributes rows across all buckets. With more buckets, fewer rows are assigned to each bucket and, on average, buckets contain fewer linked blocks.

Table clusters / multi-tables

Table clusters, also called multi-tables, interleave rows of two or more tables in the same storage area.

cluster key

Table clusters have a cluster key, a column that is available in all interleaved tables.

5.3 Single-level indexes

single-level index

A single-level index is a file containing column values, along with pointers to rows containing the column value.

multi-column index

In a multi-column index, each index entry is a composite of values from all indexed columns. In all other respects, multi-column indexes behave exactly like indexes on a single column.

table scan

A table scan is a database operation that reads table blocks directly, without accessing an index.

index scan

An index scan is a database operation that reads index blocks sequentially, in order to locate the needed table blocks.

Hit ratio / filter factor / selectivity

Hit ratio, also called filter factor or selectivity, is the percentage of table rows selected by a query.

binary search

In a binary search, the database repeatedly splits the index in two until it finds the entry containing the search value: .

primary index / clustering index

A primary index, also called a clustering index, is an index on a sort column.

secondary index / nonclustering index

A secondary index, also called a nonclustering index, is an index that is not on the sort column.

dense index

A dense index contains an entry for every table row.

sparse index

A sparse index contains an entry for every table block.

5.4 Multi-level indexes

multi-level index

A multi-level index stores column values and row pointers in a hierarchy.

fan-out

The number of index entries per block is called the fan-out of a multi-level index.

branch

Each path from the top-level block to a bottom-level block is called a branch.

balanced / imbalanced

Multi-level indexes are called balanced when all branches are the same length and imbalanced when branches are different lengths.

B+tree

B+tree. All indexed values appear in the bottom level. Pointers to table blocks appear only in the bottom level. Since some indexed values also appear in higher levels, values are occasionally repeated in the index.

B-tree

B-tree. If an indexed value appears in a higher level, the value is not repeated at lower levels. Instead, a pointer to the corresponding table block appears in the higher level along with the value.

5.5 Other indexes

hash index

In a hash index, index entries are assigned to buckets.

bucket

A bucket is a block or group of blocks containing index entries.

hash function

The bucket containing each index entry is determined by a hash function, which computes a bucket number from the value of the indexed column.

bitmap index

A bitmap index is a grid of bits: .

physical index

A single- or multi-level index normally contains pointers to table blocks and is called a physical index.

logical index

A logical index is a single- or multi-level index in which pointers to table blocks are replaced with primary key values.

function index

In a function index, the database designer specifies a function on the column value. Index entries contain the result of the function applied to column values, rather than the column values.

5.6 Tablespaces and partitions

tablespace

A tablespace is a database object that maps one or more tables to a single file.

fragmented

As files are updated, blocks become scattered, or fragmented, across many tracks.

partition

A partition is a subset of table data. One table has many partitions that do not overlap and, together, contain all table data.

horizontal partition

A horizontal partition is a subset of table rows.

vertical partition

A vertical partition is a subset of table columns.

shard

Like a partition, a shard is a subset of table data, usually a subset of rows rather than columns. Unlike partitions, which are stored on different storage devices of a single computer, shards are stored on different computers of a distributed database.

partition expression / partition columns

To partition a table, the database administrator specifies a partition expression based on one or more partition columns. The partition expression may be simple, such as the value of a single partition column, or a complex expression based on several partition columns. Rows are assigned to partitions in one of the following ways: .

range partition

A range partition associates each partition with a range of partition expression values. The VALUES LESS THAN keywords specify the upper bound of each range. The MAXVALUE keyword represents the highest column value, and VALUES LESS THAN MAXVALUE specifies the highest range. Each partition is explicitly named by the database administrator.

list partition

A list partition associates each partition with an explicit list of partition expression values using the VALUES IN keywords. Like a range partition, each partition is explicitly named.

hash partition

A hash partition requires a partition expression with positive integer values. The database administrator specifies the number of partitions, N, and partitions are automatically named p0 through p(N-1). The partition number for each row is computed as: (partition expression value) modulo N.

key partition

A key partition is similar to a hash partition, except the partition expression is determined automatically by the database.

5.7 Physical design

Logical design

Logical design specifies tables, columns, and keys. The logical design process is described elsewhere in this material.

Physical design

Physical design specifies indexes, table structures, and partitions. Physical design affects query performance but never affects query results.

storage engine / storage manager

A storage engine or storage manager translates instructions generated by a query processor into low-level commands that access data on storage media. Storage engines support different index and table structures, so physical design is dependent on a specific storage engine.

CREATE INDEX

The CREATE INDEX statement creates an index by specifying the index name and table columns that compose the index.

DROP INDEX

The DROP INDEX statement deletes a table's index.

SHOW INDEX

The SHOW INDEX statement displays a table's index.

EXPLAIN

The EXPLAIN statement generates a result table that describes how a statement is executed by the storage engine.

slow query log

The MySQL slow query log is a file that records all long-running queries submitted to the database.

6. Database Architecture

6.1 MySQL architecture

Architecture

Architecture describes the components of a computer system and the relationships between components.

Tools

Tools interact directly with database users and administrators, and send queries to the query processor.

query processor

The query processor manages connections from multiple users and compiles queries into low-level instructions for the storage engine.

storage engine / storage manager

The storage engine, also called a storage manager, executes instructions, manages indexes, and interacts with the file system.

file system

The file system accesses data on storage media. The file system contains both system and user data, such as log files, tables, and indexes.

MySQL Server / MySQL Enterprise Edition

MySQL is available in a free version, called MySQL Server, and a paid version, called MySQL Enterprise Edition.

Monitor

Monitor collects and displays information on CPU, memory, and index utilization, as well as queries and results. Database administrators use Enterprise Monitor to manage and tune large databases with many users.

Audit

Audit keeps track of all database changes. For each change, Audit tracks the time of change and who made the change. Audit supports government and business audit requirements for sensitive databases such as financial, medical, and defense.

Utility programs

Utility programs include approximately 30 tools, grouped in five categories: installation, client, administrative, developer, and miscellaneous tools.

connection

A connection is a link between tools and the query processor. Each connection specifies a database name, server address, logon name, and password.

execution plan

An execution plan is a detailed, low-level sequence of steps that specify exactly how to process a query.

query parser

The query parser checks each query for syntax errors and converts valid queries to an internal representation.

query optimizer

The query optimizer reads the internal representation, generates alternative execution plans, estimates execution times, and selects the fastest plan. Estimates are based on heuristics and statistics about data, like the number of rows in each table and the number of values in each column. These statistics are maintained in the data dictionary, described below.

cache manager

For optimal performance, the query processor layer has a cache manager that stores reusable information in main memory.

buffer manager

To reduce data access time, the buffer manager retains data blocks from the file system for possible reuse.

buffer

The data blocks are retained in an area of main memory called the buffer.

least recently used / LRU

The InnoDB buffer manager uses a least recently used or LRU algorithm. The LRU algorithm tracks the time each block was last accessed and, when space is needed, discards 'stale' blocks.

catalog / data dictionary

A catalog, also known as a data dictionary, is a directory of tables, columns, keys, indexes, and other objects in a relational database.

6.2 Cloud databases

tiers

Multiple computers linked by a network are often grouped in layers, called tiers, and arranged in a hierarchy.

single-tier architecture

Prior to 1990, most software ran in a single-tier architecture, consisting of a personal or corporate computer connected directly to monitors.

multi-tier architecture

Since 1990, complex corporate and government applications have increasingly been implemented in a multi-tier architecture:

The top tier consists of computers interacting directly with end-users.

The bottom tier consists of servers managing resources like databases and email.

One or more middle tiers execute a variety of functions, such as user authorization, business logic, and communication with other computers.

.

Web architecture

Web architecture is a multi-tier architecture consisting of web browsers and web servers communicating over the internet:

Web browsers, on the top tier, manage user interaction.

Web servers, on a middle tier, generate web pages for display on web browsers and transmit user requests to services running on lower tiers.

Application servers run application software, process user requests, and communicate with databases and other services.

Services, such as database and authentication, comprise the bottom tier.

.

on-premise

Prior to 2000, most commercial software was on-premise, or installed and run on customer computers.

cloud services

With cloud services, a vendor such as Amazon, Microsoft, or Google implements computer services on lower tiers of a web architecture. For a fee, cloud services are made available over the internet to customers.

Infrastructure-as-a-service / IaaS

Infrastructure-as-a-service, or IaaS, provides computer processing, memory, and storage media, as if the customer were renting a computer.

Platform-as-a-service / PaaS

Platform-as-a-service, or PaaS, provides tools and services, such as databases, application development tools, and messaging services.

Software-as-a-service / SaaS

Software-as-a-service, or SaaS, provides complete applications, usually through web browsers on customer machines.

virtual machine / VM

A virtual machine, or VM, is a software layer that emulates a complete, independent computing environment.

cloud database

A cloud database is a database offered as a PaaS cloud service.

6.3 Distributed databases

parallel computer

A parallel computer consists of multiple processors managed by a single operating system instance.

shared memory

In a shared memory computer, processors share the same memory and storage media.

shared storage

In a shared storage computer, processors share storage media only.

shared nothing

In a shared nothing computer, processors share neither memory nor storage media.

local area network

A local area network consists of cables extending over a small area, typically within one facility. Local area networks usually use the Ethernet communication protocol.

wide area network

A wide area network spans multiple facilities in different geographic locations, separated by many miles. Wide area networks may communicate via cables, satellite, or telephone lines, often using internet communication protocols.

node

A node is one of a group of computers connected by either a local or wide area network.

cluster

A cluster is a group of nodes connected by a local area network, managed by separate operating system instances, and coordinated by specialized cluster management software.

parallel database

A parallel database runs on a parallel computer or cluster.

distributed database

A distributed database runs on multiple computers connected by a wide area network.

distributed transaction

A distributed transaction updates data on multiple nodes of a distributed database. In a distributed transaction, either all nodes or no nodes must be successfully updated.

two-phase commit

Databases commonly implement distributed transactions with a technique called two-phase commit. The two-phase commit has four steps:

In phase 1, a central transaction coordinator notifies all participating nodes of the required updates.

Participating nodes receive the notification, store the update in a local log, and send a confirmation message to the transaction coordinator. Participating nodes do not yet commit the update to the database.

Phase 2 begins when the transaction coordinator receives confirmation from all participating nodes. The transaction coordinator now instructs all nodes to commit.

Participating nodes receive the commit message, commit the update to the database, and notify the transaction coordinator of success.

.

local transaction

A local transaction updates data on a single node of a distributed database.

synchronous

Updates in a distributed transaction are synchronous, since the updates occur at the same time from the perspective of the database user.

asynchronous

Updates in separate local transactions are asynchronous.

eventually consistent

Databases that use local rather than distributed transactions are called eventually consistent.

consistent

A consistent database conforms to all rules at all times.

available

In an available database, 'live' nodes must respond to queries at all times.

network partition

A network partition forms when a network error prevents nodes from communicating.

partition-tolerant

A partition-tolerant database continues to function when a network partition occurs.

CAP theorem

The CAP theorem states that a distributed database cannot simultaneously be Consistent, Available, and Partition-tolerant. A distributed database can guarantee any two, but not all three, of these properties.

6.4 Replicated databases

replica

A replica is a copy of an entire database, a table, or a subset of table data.

replicated database

A replicated database maintains two or more replicas on separate storage devices.

storage arrays

Some storage devices, called storage arrays, manage replicas internally, without database intervention.

primary/secondary

The primary/secondary technique designates one node as primary. All updates are first applied to the primary node in local transactions. Secondary nodes are updated after the primary node commits, with independent local transactions.

group replication

The group replication technique applies updates to any node in a group. Prior to committing, a node broadcasts transaction information to other nodes, which look for conflicts with concurrent transactions. If any node detects a conflict, an algorithm determines which transaction commits and which rolls back.

central catalog

In a central catalog, the entire catalog resides on a single node.

replicated catalog

In a replicated catalog, a copy of the catalog resides on each node.

6.5 Data warehouses

operational data

Organizations use operational data to conduct daily business functions.

analytic data

Organizations use analytic data to understand, manage, and plan the business.

reporting data / decision support data

Analytic data is sometimes called reporting data or decision support data.

data warehouse

A data warehouse is a separate database optimized for analytics rather than operations.

data mart

A data mart is a data warehouse designed for a specific business area, such as sales, human resources, or product development.

extract-transform-load / ETL

The five-step process is commonly referred to as the extract-transform-load, or ETL, process.

ETL tools

Since the ETL process is time-consuming and difficult to automate, many organizations use special software products, called ETL tools, to minimize programming.

6.6 Data warehouse design

dimensional design / star schema

A dimensional design, also called a star schema, consists of fact and dimension tables: .

fact table

A fact table contains numeric data used to measure business performance, such as sales revenue or number of employees. Each row in a fact table consists of numeric fact columns and foreign keys that reference dimension tables.

dimension table

A dimension table contains textual data that describes the fact data, such as product line, organizational unit, and geographical region.

dimension hierarchy

A dimension hierarchy is a sequence of columns in which each column has a one-many relationship to the next column.

date dimension

Each row of the date dimension table corresponds to a day. If an organization tracks data for 100 years, the date dimension contains 36,500 rows (100 years × 365 days per year).

time dimension

Each row of the time dimension table corresponds to a minute of the day. The time dimension contains 1,440 rows (24 hours × 60 minutes per hour).

type 2 design for slowly changing dimensions

Adding start and end foreign keys to the fact table is called type 2 design for slowly changing dimensions.

6.8 Other database architectures

in-memory database

An in-memory database is a database that stores data in main memory, instead of or in addition to storage media.

embedded database / in-process database

An embedded database, sometimes called an in-process database, is a database that is packaged with a programming language. An embedded database and application program execute together in a single software process.

SQLite

SQLite is the dominant embedded relational database.

SQL Server Compact

SQL Server Compact is an embedded database from Microsoft. The last major release of SQL Server Compact was in 2011, and Microsoft will discontinue support after 2021.

libmysqld

The MySQL software library libmysqld configures MySQL as an embedded database but was discontinued as of MySQL release 8.0.

federated database

A federated database is a collection of two or more participating databases underneath a coordinating software layer. The participating databases are autonomous and heterogeneous: .

autonomous database

An autonomous database operates independently of other participating databases. An autonomous database is administered and can be queried as if the database were not part of a federated database.

Heterogeneous databases

Heterogeneous databases either run under different database systems or have incompatible schema.

middleware

The coordinating software layer is called middleware, since the software lies between application programs and database software.

global catalog

A global catalog is a directory of participating database objects, such as tables, columns, and indexes.

database wrapper

A database wrapper converts the decomposed queries to the appropriate syntax for each participating database.

SQL/Management of External Data / SQL/MED

Some products support SQL/Management of External Data, or SQL/MED, an extension of the SQL standard for federated databases.

nickname

A nickname is a federated database name for a participating database object, such as tables and columns.

user mapping

A user mapping associates a federated database user with a participating database user.

data lake

A data lake is an analytic database of raw, unprocessed data copied from multiple data sources. Data lakes share some characteristics of data warehouses and some characteristics of federated databases: .

8. Additional Material

8.1 MySQL Workbench: Import and export

backtick

The backtick (`) delimits literals that represent identifiers, which allows spaces and reserved words to be used as identifiers.

9. Data Modeling

9.1 Binary relationships

9.2 Unary relationships

9.3 Ternary relationships