Data Modelling and Database Applications - IDATA2303  
Course notes

# Data models

Data models can be classified as *Conceptual models, Logical and physical models* at an abstract level.  
A conceptual model serves users by portraying data entities, relationships, and attributes. This model does not concern itself with theoretical implementation but rather focuses on capturing user needs and business rules during the early stages of database design.  
  
Requirement analysis – Process of gathering, documenting, and understanding the needs an expectation for a system or project.

Conceptual model – Abstract representation of essential system elements and their relationships, focusing on concepts and ideas rather than specifics.

Logical model – Detailed representation of the system’s structure and functionality, emphasizing how data is organized and processed. Logical models include hierarchical models, network models, relational models, object-oriented models, object relational models.

Physical model – Concrete implementation of the logical model into a specific technology or platform, considering hardware, software, and infrastructure. Physical model is the lowest level abstract of data. It is to describe the storage and access methods of data inside the hardware systems.  
  
Evaluation – Assessing the system to ensure it meets requirements and performs as intended, often involving testing, feedback, and analysis.  
  
Maintenance – Ongoing tasks to support, update, and enhance the system after its development, including fixing issues and adapting to changing needs or environments.

Databases are abstract descriptions of data features from an objective world. It should describe the world in a relatively real way, easy to understand, easy to be implemented in computers.

MAC and DAC are two fundamental models used in computer security to regulate access to resources within a system. They differ in their approach and level of control.

MAC – Mandatory Access Control  
Is an approach which is strictly governed by predefined labels or classification set by the system of administrators. It is commonly used in high-security environments like government or military sectors, ensuring stringent control over sensitive information based on security clearances.  
DAC – Discretionary Access Control  
DAC is decentralized and offers flexibility, allowing resource owners or users to determine access permissions. This model, prevalent in personal computing, grants users’ autonomy over their resources, but relies on responsible management to avoid potential security risks stemming from users granting excessive permissions or overlooking security measures.

# Database

A database system is a structured collection of data organized to allow easy access, management, and updating. It comprises software applications, database management system, data models, and the actual data. It enables users to store, retrieve and manipulate data.

The basic concepts we use in the relational data model:

* **Entities** are things that exist in objective world and can be identified. Can be people, objects, events, or abstract concept.
* An **entity type** is a template for entity instances, including entity name and attribute sets.
* An **entity set** is a collection of entities with the same entity type.
* **Relation**, which is a table representing entities with rows and columns (tuples, attributes)
* **Tuples** are single rows in a relation, containing specific data related to an entity.
* **Attributes** are columns in a relation, representing a specific characteristic or property of an entity. One entity can be described by several attributes.
* **Et bilde som inneholder tekst, skjermbilde, diagram, Font

  Automatisk generert beskrivelseKeys** are attributes or sets of attributes that uniquely identify each tuple in a relation.
* **Domains** are collections of the values with the same data type. The range of a certain attribute usually comes from a domain.
* An **element** is an attribute value in a tuple.
* **Normalization** is the process of organizing data attributes into multiple related tables to minimize redundancy and dependency.

The relational model is a way of organizing data based on simple ideas and math principles. It is like fitting information into tables where everything – like people, things, and connections between them – can be shown as relationships. When you ask for data, the answer you get also follow this table-like structure. This model keeps data safe and makes it independent, which is good for security. It also makes things easier for programmers and those building databases. But, compared to other ways of organizing data, it might take longer to find specific information. To make it work faster, you need to be smart about how you ask for things, which can be a bit tricky. And because of this, building and improving databases using this model can be tough and complex. So, while it is simple and safe, making it fast and building databases with it can be a bit of a challenge.

# SQL

SQL, meaning Structured Query Language, is a programming language designed for managing, manipulating, and querying data stored in relational databases. SQL enables users to perform various operations such as retrieving data, inserting new records, updating existing data, and deleting information from databases.

Some crucial SQL functions:

Query

* SELECT: Retrieves data from one or multiple tables based on specified criteria using queries.
* CONSTRAINT: Helps enforce rules or restrictions on data within tables. Common constraints include primary key, foreign key, unique, not null and check constraints.
* JOIN: Combines data from multiple tables based on related columns, allowing for the retrieval of data from multiple sources simultaneously.
* AGGREGATE FUNCTIONS: Perform operations on sets of values and return a single result. Some functions
* GROUP BY: Groups rows that have the same values in specified columns into summary rows and allows applying aggregate functions to these groups.
* ORDER BY A clause used to sort the result set based on specified columns in either ascending or descending order.

Define

* CREATE: Used to create database objects like tables, views, indexes, or databases themselves.
* DROP: Used to remove objects like tables, views, indexes, or databases themselves.
* ALTER: Enables modification of existing database objects, allowing changes such as adding, modifying, or dropping columns in a table, altering constraints, or renaming objects.

Manipulation

* INSERT: Adds new records or rows in a table, allowing users to insert data into specific columns within a table.
* UPDATE: Modifies existing records in a table by changing values in specified columns based on defined conditions.
* DELETE: Removes records from a table based on specific conditions, deleting entire rows of data.

Control

* GRANT: Provides specific privileges or permissions to database users, allowing them to perform actions like SELECT, INSERT, UPDATE, DELETE, etc, on specific database objects.
* REVOKE: Revokes previously granted privileges or permissions from users, restricting their access to certain database objects or actions.

Three-layer structure commonly supported in relational databases:

Base tables are tables which store data independently. Each relation or table in a database typically corresponds to one base table. These tables hold the actual data within the database and can have various indices for optimization purposes.

Views are virtual tables generated from one or multiple base tables. They store the definitions or queries based on the data from the base tables but do not contain the actual data themselves. Users and create views, allowing for layered or hierarchical structure of querying and organizing data without duplicating the underlying information.

Storage file is a physical storage aspect specifically for the DBMS. It represents the file or files on the storage medium where the database’s tables and associated data are stored. It is essentially the physical representation of the tables and views in the database.

Regarding SQL commands, be aware of the following details:

* Case insensitive: It does not matter whether letters for the SQL commands are capitalized or not.
* Values of strings are case sensitive, meaning ‘Seattle’ is not the same as ‘seattle’
* Use single quotes when typing a string (like this ‘abc’, not “abc”)

## Data definition

A schema in a database context refers to a logical container that holds various database objects like tables, views, procedures.  
  
*CREATE SCHEMA Sprockets AUTHORIZATION Annik  
CREATE TABLE NineProngs (source int, cost int, partnumber int)  
GRANT SELECT ON SCHEMA::Sprockets TO Mandar  
DENY SELECT ON SCHEMA::Sprockets TO Prasanna;*

In this example , a schema named ‘Sprockets’ is created, and it is authorized or owned by the user ‘Annik’. Inside this schema, a table called ‘NineProngs’ is defined with columns ‘source’, ‘cost’, and ‘partnumber’. Furthermore, permissions are granted and denied within this schema:

* User ‘Mandar’ is given permission to execute SELECT queries on objects within the ‘Sprockets schema
* User ‘Prasanna’ is denied SELECT access, restricting their ability to retrieve data from these schema objects.

To delete a schema, you can type the following command:

DROP SCHEMA Schema\_Name CASCADE | RESTRICT;

CASCADE: Delete schema means to delete all database objects inside.  
RESTRICT: If there are definitions in the schema, then deny the execution of this command. Execute only if there are no objects defined in it.

A base table in SQL will look something like this:

* + CREATE TABLE *table\_name*(  
    *attribute\_name datatype Attribute\_constraint*,  
    *attribute\_name datatype Attribute\_constraint*,  
    *…*  
       *Table\_constraint*  
    );

If integrity constraints, such as those ensuring data accuracy or relationships between columns, apply to multiple attributes, they are included within the table-level constrains, ensuring that the defined rules cover the entire table rather than individual attributes. This method helps maintain consistency and data integrity across the entire table structure in the database.  
  
Assertions define complex constraints among multiple tables and aggregate functions. When an assertion is established, any operation in related tables will activate DBMS to check this constraint. Any operation which can return FALSE for assertion will be rejected to execute.  
An assertion which will define a constraint as course “Database” with only 60 students or less:

CREATE ASSERTION check\_less\_than\_60\_students  
CHECK (60 >= (SELECT COUNT(\*)  
 FROM Course, SC  
 WHERE SC.Cno = Course.Cno AND Course.Cname = ‘Database’   
);

Assertions must not be too complex as to not be costly in both time and computation resources.

Trigger is a series of special event-driven processes to the tables that are defined by users. Triggers are usually triggered by events like: INSERT, UPDATE or DELETE, where every time one of these are called, a predefined action is called. A trigger in SQL can look something like this:

CREATE TRIGGER trigger\_name  
BEFORE UPDATE ON table\_name  
FOR EACH ROW  
BEGIN  
 SET NEW.column\_name = (NEW.column\_name \* 2);  
END;

Database design is to build optimized logical and physical structure for a specific application environment. The aim is to effectively store and manage data, including information and data operations for different users. Database design includes several stages:  
  
Requirement analysis – Data dictionary, system description for data items, structure, flow, and stage.

Conceptual model – Usually visualized through an E/R diagram. Data dictionary. Modeling in digital world. Should be easy to understand, modify and transfer to relational, graph or hierarchical data models.

Logical Structure design – Either relational or non-relational model. Both entity sets and relationships become relations (tables) in DBMS.

Physical Structure design – Storage organization, Access method selection (DAC, MAC), Access path set up.

Database implementation – Create DB schema, load data, DB deployment.

Operation and Maintenance – Performing monitoring, Migration and recovery, Reorganization and refactoring.