ICT 700
Introduction To
Business Information Systems

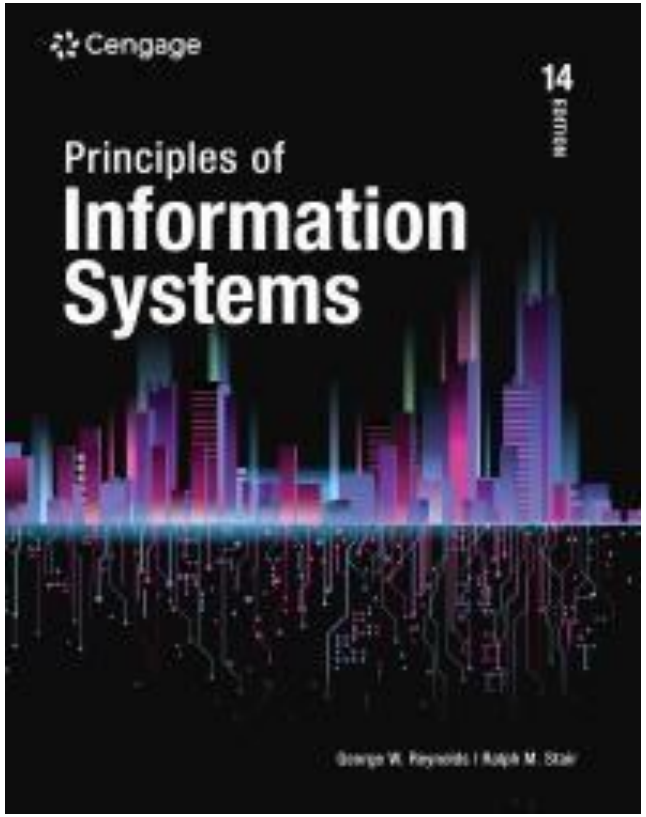
LECTURE 10

Use Of Database systems and Data Management To Develop and Acquire Information systems

Unit Coordinator:
Sajad Ghatrehsamani

Reading Chapter:

Chapter 5 – Stair & Reynolds



Learning Objectives

1. Distinguishing data from information and knowledge
2. Defining the term database management system
3. Understanding Database Structure and Its Design
4. Understanding Entity Relationship Diagram
5. Identifying Database Normalization Form
6. Identifying Relational Database
7. Knowing SQL Commands
8. Introducing Popular Relational Database Management Systems

Why Learn about Database Systems and Data Management

Databases capture data about changes

- for analysis and decision making
- To recognize new challenges and opportunities
- To track progress toward meeting key goals
- To identify when a change in tactics or strategy is needed
- Need to understand database systems and data management

King's Own Institute

Data, Information, and Knowledge

1. **Data** refers to raw facts.
2. **Information**
   - Collection of organized and processed data.
   - Has additional value beyond the value of the individual facts.
3. **Knowledge**
   - Provides awareness and understanding of a set of information.
   - Shows how information can support a specific task or be used to reach a decision.

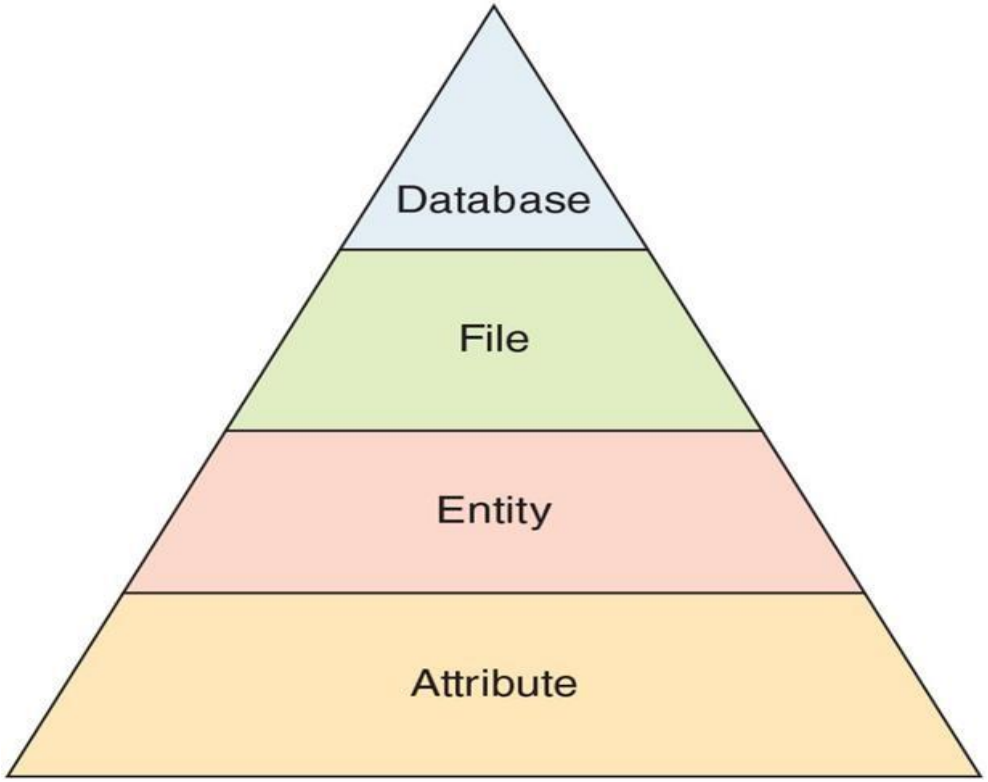
KOI
King's Own Institute

Data Types

Data	Represented By
Alphanumeric data	Numbers, letters, and other characters
Audio data	Sounds, noises, or tones
Image data	Graphic images and pictures
Video data	Moving images or pictures

TABLE 5.1 Types of data

The Data Hierarchy



KOI
King's Own Institute

Database File Structure

1. **Entity**
   Person, place, or thing (object) for which data is collected, stored, and maintained
2. **File**
   Collection of entities
3. **Attribute**
   Characteristic of an entity
4. **Domain**
   Range of allowable values for a data attribute

Database Structure

1. **Data item**
   Specific value of a data attribute
2. **Record**
   Collection of attributes about a specific entity
3. **Primary key**
   Attribute or set of attributes that uniquely identifies the record
4. **Foreign key**
   Attribute in one table that refers to the primary key in another table

Database Management System

- Database management system (DBMS)
- Group of programs provided by the DBMS supplier
- Programs used to access and manage a database
- Provides an interface between the database and its users and other application programs

Database Activities

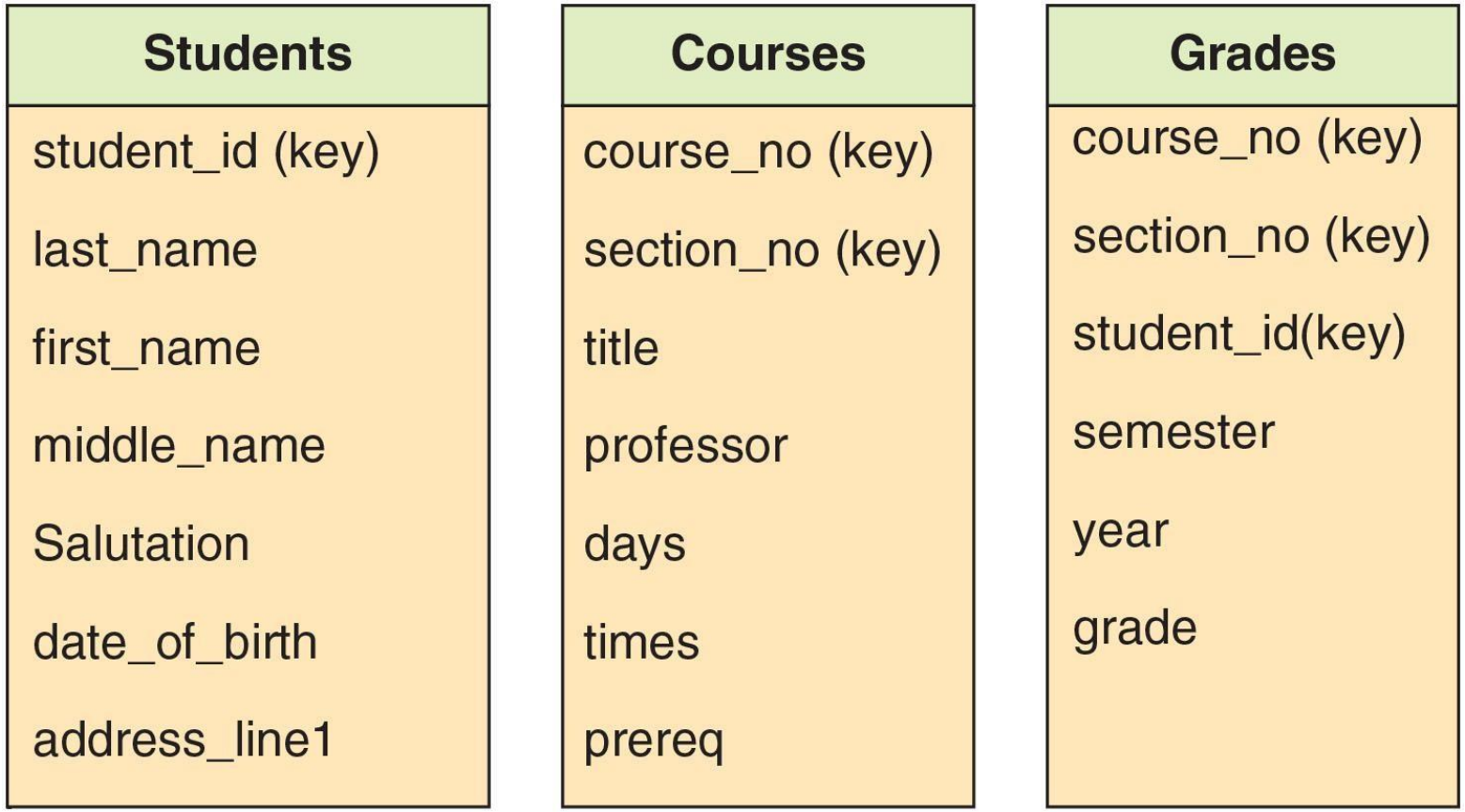


FIGURE 5.4 Database schema represented in a visual diagram

Creating and modifying the database

Data definition language (DDL)

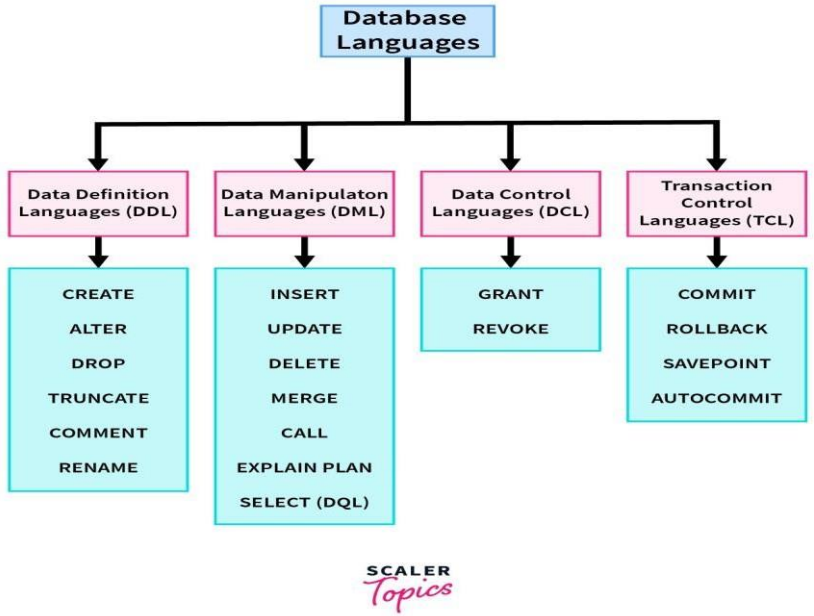
- Collection of instructions and commands
- Defines and describes data and relationshi ps in a specific database

Data dictionary

- Detailed description of data stored in the database
- Adherence to data dictionary standards
- Makes it easy to share data among organizations

Data manipulation language (DML)

- A specific language provided with a DBMS
- Allows users to access and modify the data, make queries, and generate reports



Source: https://www.scaler.com/topics/database-languages-in-dbms/

King's Own Institute

Database Activities

- **Storing and retrieving data**
  - DBMS function

    Interface between application program and database
  - To obtain database data

    Request it through the DBMS
  - **Concurrency control**

    Addresses situation where two or more users or applications access the same record at the same time

Database Design

- **Necessary to keep data well organized**
- **Database design**
  - Store all relevant data
  - Provide quick access and easy modification
  - Reflect organization’s business processes
- **Considerations**
  - Content and access
  - Logical structure and physical organization
  - Response time, archiving, security
- **Data modelling**

  Tool used to design a database

  Occurs at organizational level

  Called enterprise data modelling

  Occurs at specific business application level

Database Design

Data modelling

- Tool used to design a database Occurs at organizational level
  - Called enterprise data modelling Occurs at specific business application level

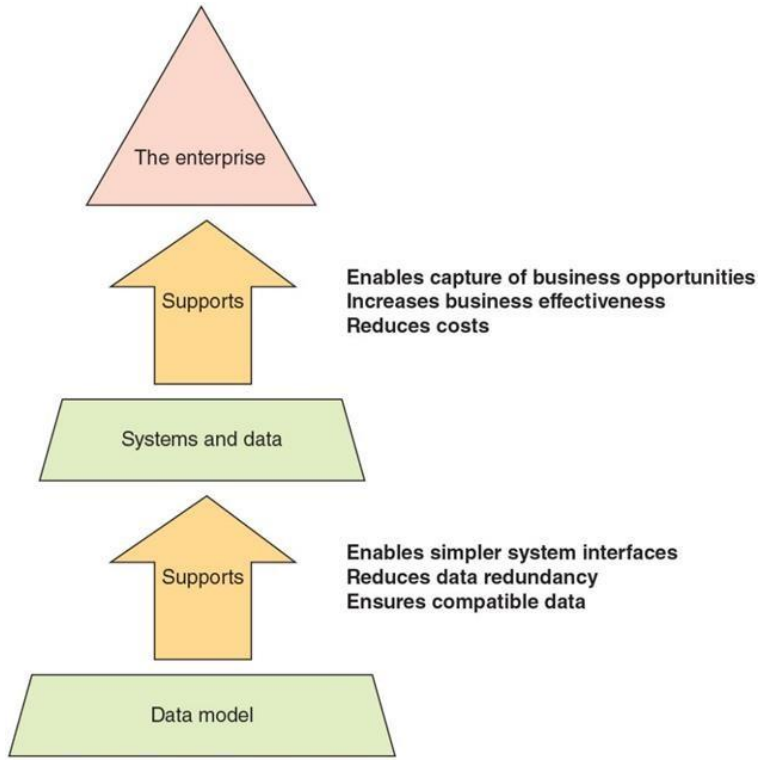
Enterprise data model

- Identifies data entities and data attributes of greatest interest to the organization
- Identifies their associated standard data definitions, data length and format, domain of valid values, and any business rules for their use

Database Design Continued ..

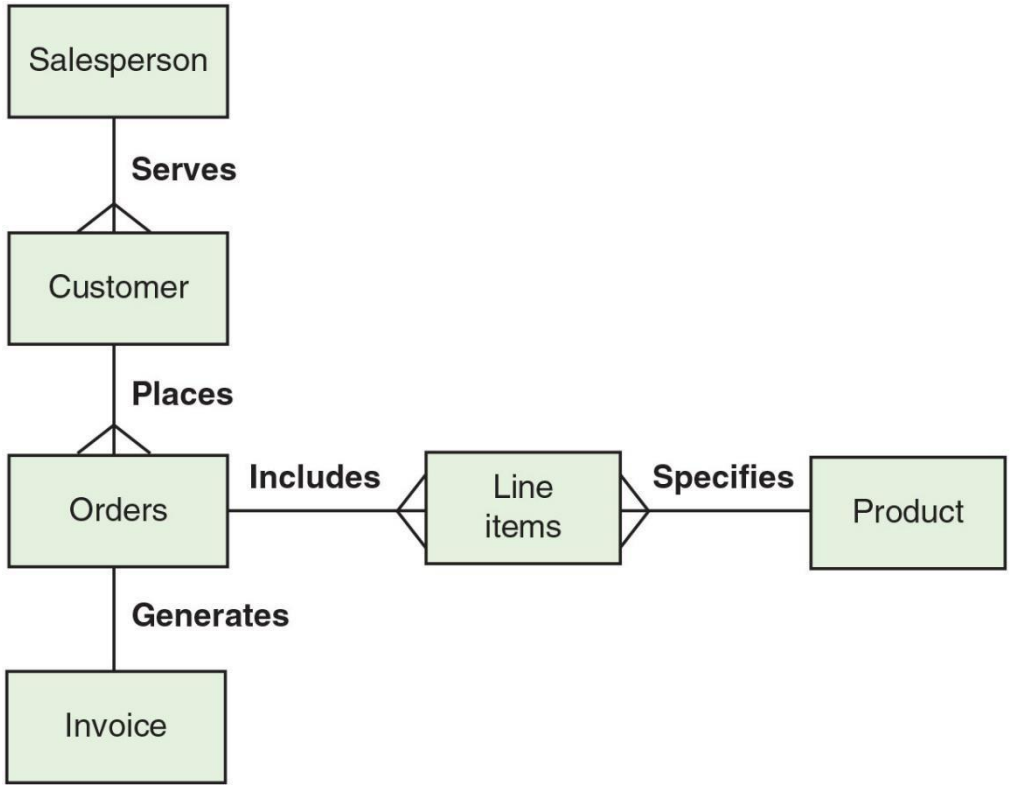
Enterprise data model

- Identifies data entities and data attributes of greatest interest to the organization
- Identifies their associated standard data definitions, data length and format, domain of valid values, and any business rules for their use



King's Own Institute

Database Design Continued

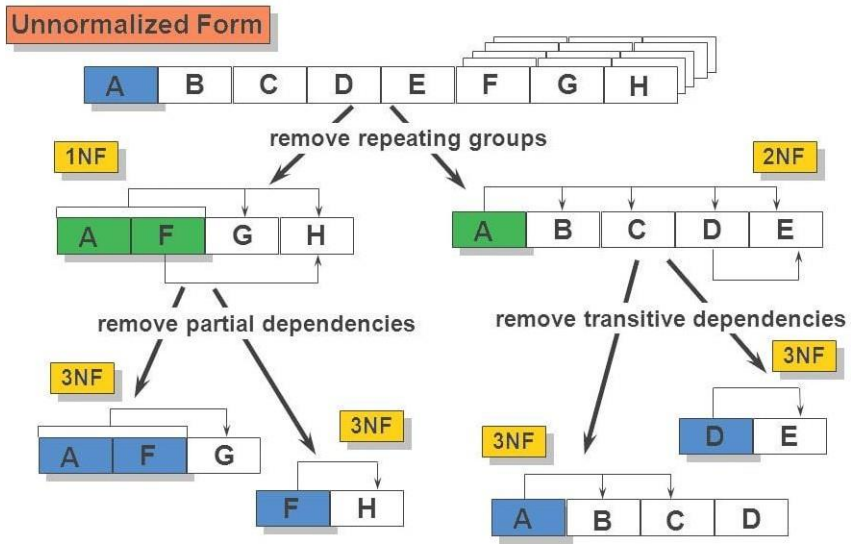


King's Own Institute

Normalization

- Data model used to analyze and communicate data needs
- Works at the individual project or application level
- Uses graphical symbols
  - Identify data entities and their associated data attributes
  - Identify the relationships among the entities of interest
- Many notation styles exist for drawing an ER diagram

- 1NF, 2NF, 3NF, 4NF



Source: https://galaktika-soft.com/blog/data-mining-normalization.html

King's Own Institute

Data Normalization Example

Full Names	Physical Address	Movies Rented	Salutation
Janet Jones	First Street Plot No 4	Pirates of the Caribbean, Clash of the Titans	Ms.
Robert Phil	3rd Street 34	Forgetting Sarah Marshal, Daddy's Little Girls	Mr.
Robert Phil	5th Avenue	Clash of the Titans	Mr.

NF (First Normal Form) Rules
- Each table cell should contain a single value.
- Each record needs to be unique.

Full Names	Physical Address	Movies Rented	Salutation
Janet Jones	First Street Plot No 4	Pirates of the Caribbean	Ms.
Janet Jones	First Street Plot No 4	Clash of the Titans	Ms.
Robert Phil	3rd Street 34	Forgetting Sarah Marshal	Mr.
Robert Phil	3rd Street 34	Daddy's Little Girls	Mr.
Robert Phil	5th Avenue	Clash of the Titans	Mr.

2NF (Second Normal Form) Rules
- Rule 1- Be in 1NF
- Rule 2- Single Column Primary Key that does not functionally dependent on any subset of candidate key relation

Membership ID	Full Names	Physical Address	Salutation
1	Janet Jones	First Street Plot No 4	Ms.
2	Robert Phil	3rd Street 34	Mr.
3	Robert Phil	5th Avenue	Mr.

Membership ID	Movies Rented
1	Pirates of the Caribbean
1	Clash of the Titans
2	Forgetting Sarah Marshal
2	Daddy's Little Girls
3	Clash of the Titans

3NF (Third Normal Form) Rules
- Rule 1- Be in 2NF
- Rule 2- Has no transitive functional dependencies

To move our 2NF table into 3NF, we again need to again divide our table.

Membership ID	Full Names	Physical Address	Salutation ID
1	Janet Jones	First Street Plot No 4	2
2	Robert Phil	3rd Street 34	1
3	Robert Phil	5th Avenue	1

Membership ID	Movies Rented
1	Pirates of the Caribbean
1	Clash of the Titans
2	Forgetting Sarah Marshal
2	Daddy's Little Girls
3	Clash of the Titans

Salutation ID	Salutation
1	Mr.
2	Ms.
3	Mrs.
4	Dr.

KOI
King's Own Institute

Relational Databases (1 of 2)

Data Table 1: Project Table

Project	Description	Dept. number
155	Payroll	257
498	Widgets	632
226	Sales manual	598

Data Table 2: Department Table

Dept. number	Dept. name	Manager SSN
257	Accounting	005-10-6321
632	Manufacturing	549-77-1001
598	Marketing	098-40-1370

Data Table 3: Manager Table

SSN	Last name	First name	Hire date	Dept. number
005-10-6321	Johns	Francine	10-07-2013	257
549-77-1001	Buckley	Bill	02-17-1995	632
098-40-1370	Fiske	Steven	01-05-2001	598

FIGURE 5.9 Relational database model
In the relational model, data is placed in two-dimensional tables, or relations. As long as they share at least one common attribute, these relations can be linked to provide output useful information. In this example, all three tables

include the dept. number attribute.

Relational Database

- Fundamental characteristics
- Data is organized into relations
- Rows represent entities and columns represent attributes
- Rows uniquely identified by a primary key
- Column table data
- Integer number, decimal number, date, text, etc.
- Constrained to be certain type, length, or to have a value between two limits
- Primary and foreign keys enable table relationships
- User queries perform operations on the database

Manipulating Data in a Relational Database

1. **Selecting**
   - Eliminating rows according to certain criteria
2. **Projecting**
   - Eliminating columns in a table
3. **Joining**
   - Combining two or more tables through common data attributes to create a new table
4. **Data normalization**
   - Eliminates data redundancy

King's Own Institute

Manipulating Data in a Relational Database

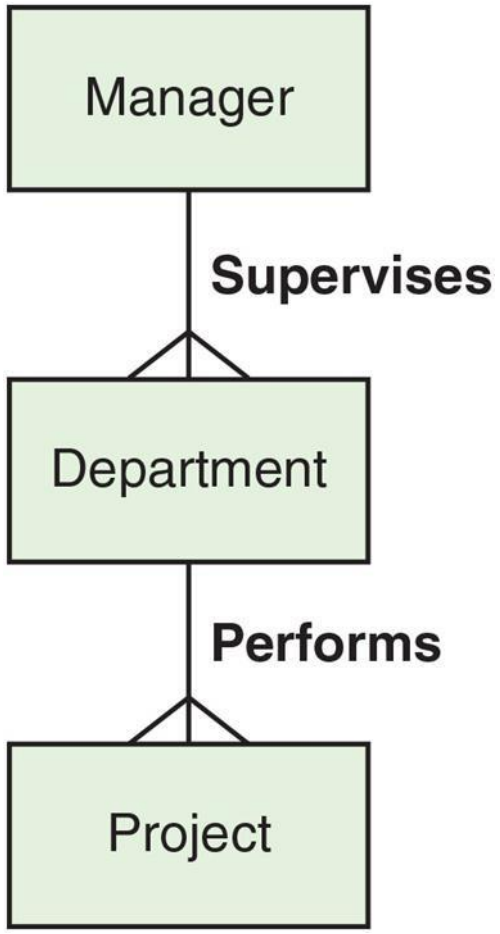
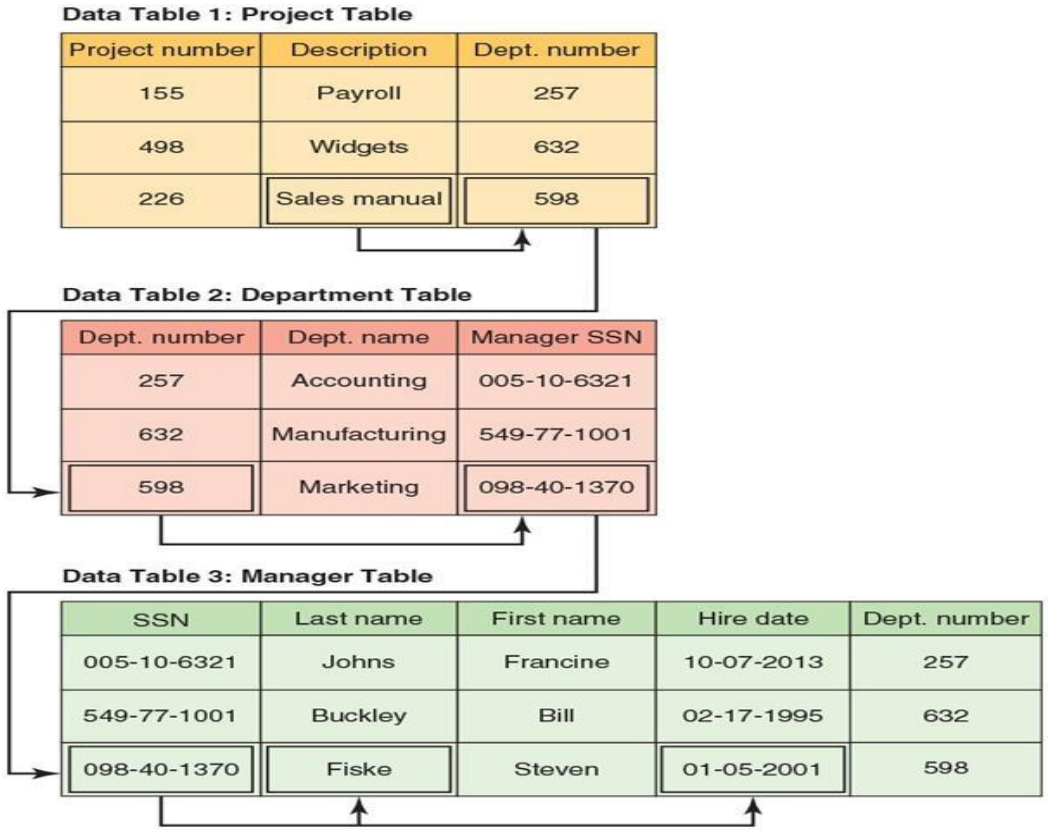


FIGURE 5.10 ER diagram
This diagram shows the relationship among the manager, department, and project tables.

Manipulating Data in a Relational Database



Data Table 1: Project Table

Project number	Description	Dept. number
155	Payroll	257
498	Widgets	632
226	Sales manual	598

Data Table 2: Department Table

Dept. number	Dept. name	Manager SSN
257	Accounting	005-10-6321
632	Manufacturing	549-77-1001
598	Marketing	098-40-1370

Data Table 3: Manager Table

SSN	Last name	First name	Hire date	Dept. number
005-10-6321	Johns	Francine	10-07-2013	257
549-77-1001	Buckley	Bill	02-17-1995	632
098-40-1370	Fiske	Steven	01-05-2001	598

FIGURE 5.11 Linking data tables to answer an inquiry
To find the name and hire date of the manager working on the sales manual project, the president needs three tables: project, department, and manager. The project description (sales manual) leads to the department number (598) in the project table, which leads to the manager's social security number (098-40-1370) in the department table, which leads to the manager's last name (fiske) and hire date (01-05-2001) in the manager table

SQL Databases

SQL

- Special-purpose programming language
- Used for accessing and manipulating relational database data
- SQL databases conform to **ACID properties**
  - Atomicity, consistency, isolation, durability
  - Guarantees the database transactions are processed reliably
  - Ensures the integrity of data in the database

SQL COMMAND

SQL Command	Description
SELECT ClientName, Debt FROM Client WHERE Debt > 1000	This query displays clients (ClientName) and the amount they owe the company (Debt) from a database table called Client; the query would only display clients who owe the company more than $1,000 (WHERE Debt > 1000).

TABLE 5.6 Examples of SQL commands

SQL COMMAND CONTINUED

SQL Command	Description
SELECT ClientName, ClientNum, OrderNum FROM Client, Order WHERE Client. ClientNum=Order.ClientNum	This command is an example of a join command that combines data from two tables: the Client table and the Order table (FROM Client, Order). The command creates a new table with the client name, client number, and order number (SELECT ClientName, ClientNum, OrderNum). Both tables include the client number, which allows them to be joined. This ability is indicated in the WHERE clause, which states that the client number in the Client table is the same as (equal to) the client number in the Order table (WHERE Client.ClientNum=Order.ClientNum).

TABLE 5.7 Examples of SQL commands

SQL COMMAND CONTINUED

SQL Command	Description
GRANT INSERT ON Client to Guthrie	This command is an example of a security command. It allows Bob Guthrie to insert new values or rows into the Client table.

TABLE 5.9 Examples of SQL commands

SQL COMMAND CONTINUED

SQL Command	Description
SELECT ClientName, Debt FROM Client WHERE Debt > 1000	This query displays clients (ClientName) and the amount they owe the company (Debt) from a database table called Client; the query would only display clients who owe the company more than $1,000 (WHERE Debt > 1000).

TABLE 5.10 Examples of SQL commands

SQL COMMAND CONTINUED

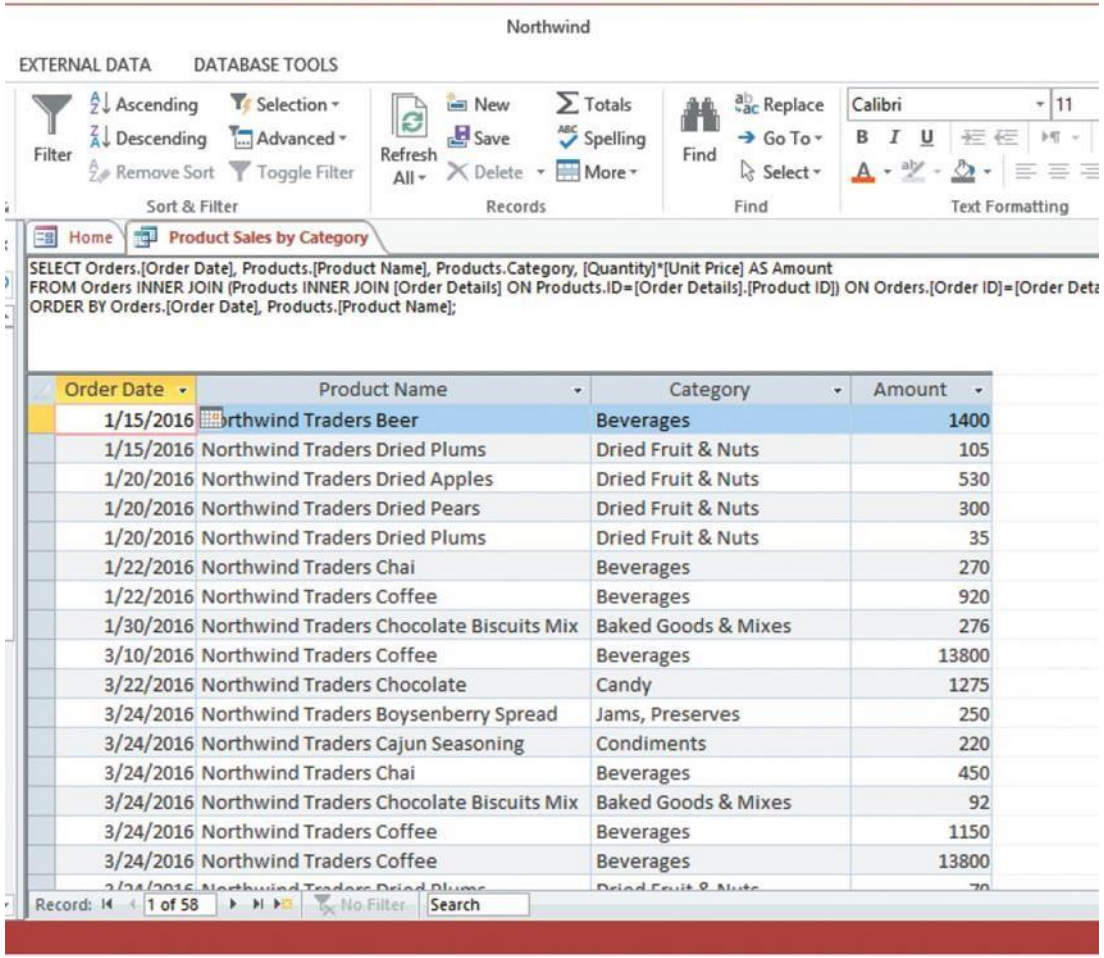


TABLE 5.11 Examples of SQL commands

King's Own Institute

SQL COMMAND CONTINUED

SQL Command	Description
SELECT ClientName, Debt FROM Client WHERE Debt > 1000	This query displays clients (ClientName) and the amount they owe the company (Debt) from a database table called Client; the query would only display clients who owe the company more than $1,000 (WHERE Debt > 1000).

TABLE 5.12 Examples of SQL commands

Popular Relational Database Management Systems

Open-Source Relational DBMS	Relational DBMS for Individuals and Workgroups	Relational DBMS for Workgroups and Enterprise
MySQL Summary	Microsoft Access	Oracle
PostgreSQL	IBM Lotus Approach	IBM DB2
MariaDB	Google Base	Sybase Adaptive Server
SQL Lite	OpenOffice Base	Teradata
CouchDB	Airtable	Microsoft SQL Server
MongoDB	Knack	Progress OpenEdge

Summary

- Well-designed and well-managed databases help with decision making
  - High-quality data benefits
    - Improve decision making, increase customer satisfaction, increase sales, improve innovation, raise productivity, and ensure compliance
- Strong data management program needed to ensure high-quality data
  - Data governance is the core component of data management

Any Questions?

KOI
King's Own Institute