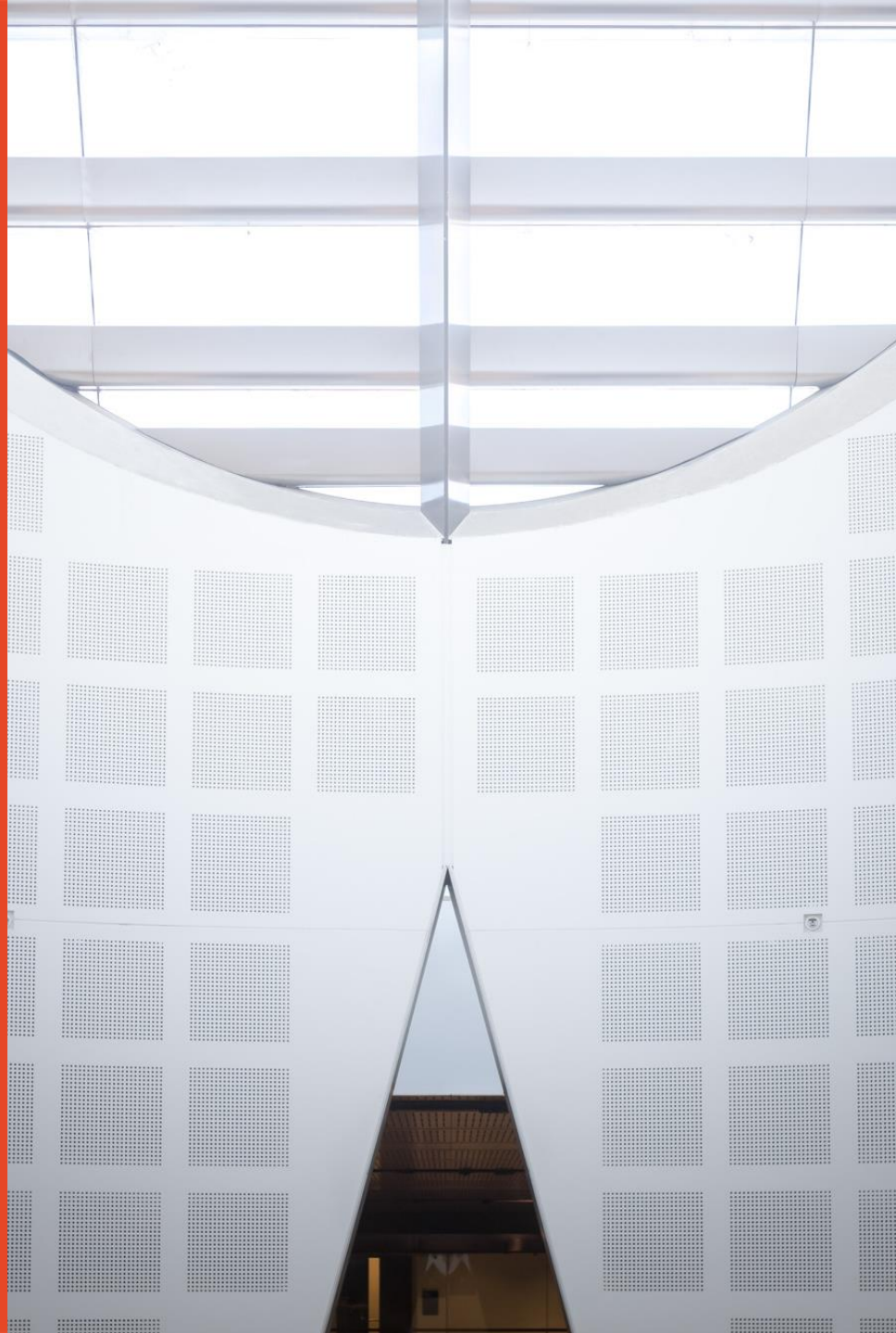
ISYS2110
Analysis and Design of Web Information Systems

Lecture 9
Data Design PART-A

Semester 1, 2018
Dr Rabiul Hasan

THE UNIVERSITY OF SYDNEY



Recapture From Lecture 8

What we have covered on the topic: User Interface Design

- Concept of user interface design
- Human-computer interaction
- Habits of Successful Interface Designers
- Guidelines for User Interface Design
- Challenges of designing for different devices

What Will We Do Today ?

- Lecture
  - Data Design Concepts
  - Database Environment
  - Data Design Terms
  - Entity Relationship Diagram

- Class activities
  - Critical Thinking / No Problem Solving Today
  - https://padlet.com
  - https://answergarden.ch

- Tutorial: ?
- Assessment ?
- Announcement (if any): We will have a guest speaker with us for about 30 minutes in week10 lecture (next week)

Learning Objectives

- Discuss file-oriented systems and how they differ from database management systems
- Explain data design terminology— entities, fields, records, files, tables, and key fields
- Describe data relationships and Entity Relationship Diagram

Data Design Concepts

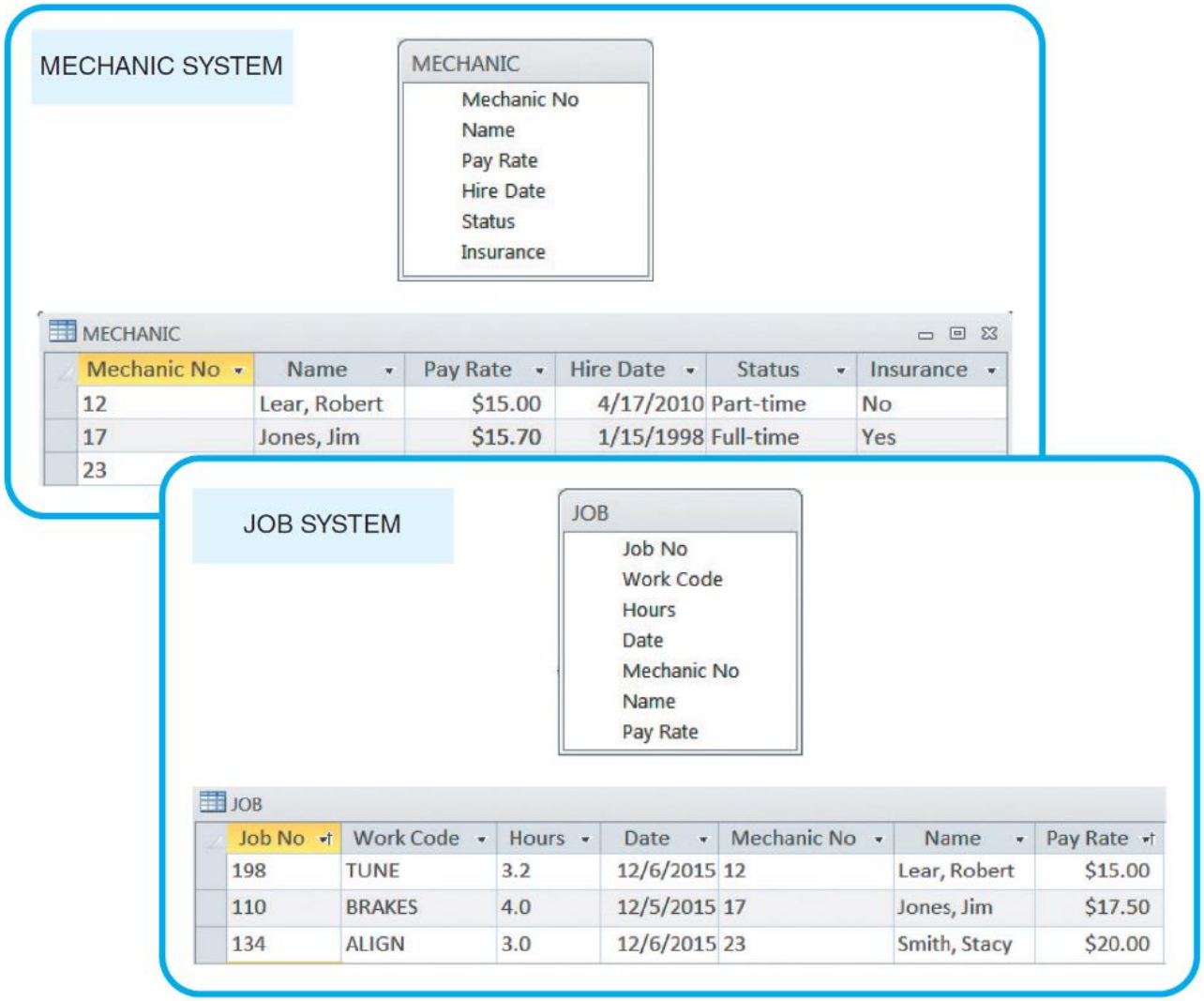
- **Data Structures**
  - Framework for organizing, storing, and managing data
  - Comprises of files or tables that interact in various ways
    - Each **file** or **table** contains data about people, places, things, or events

Data Design Concepts (Cont. 1)

- **Mario and Danica - A Data Design Example**
  - Mario's auto shop uses **file-oriented systems**
    - MECHANIC SYSTEM uses the MECHANIC file to store data about shop employees
    - JOB SYSTEM uses the JOB file to store data about work performed at the shop
  - Danica's auto shop uses a relational model
    - SHOP OPERATIONS SYSTEM - Tables are linked by a common field named Mechanic No field

Data Design Concepts (Cont. 2)

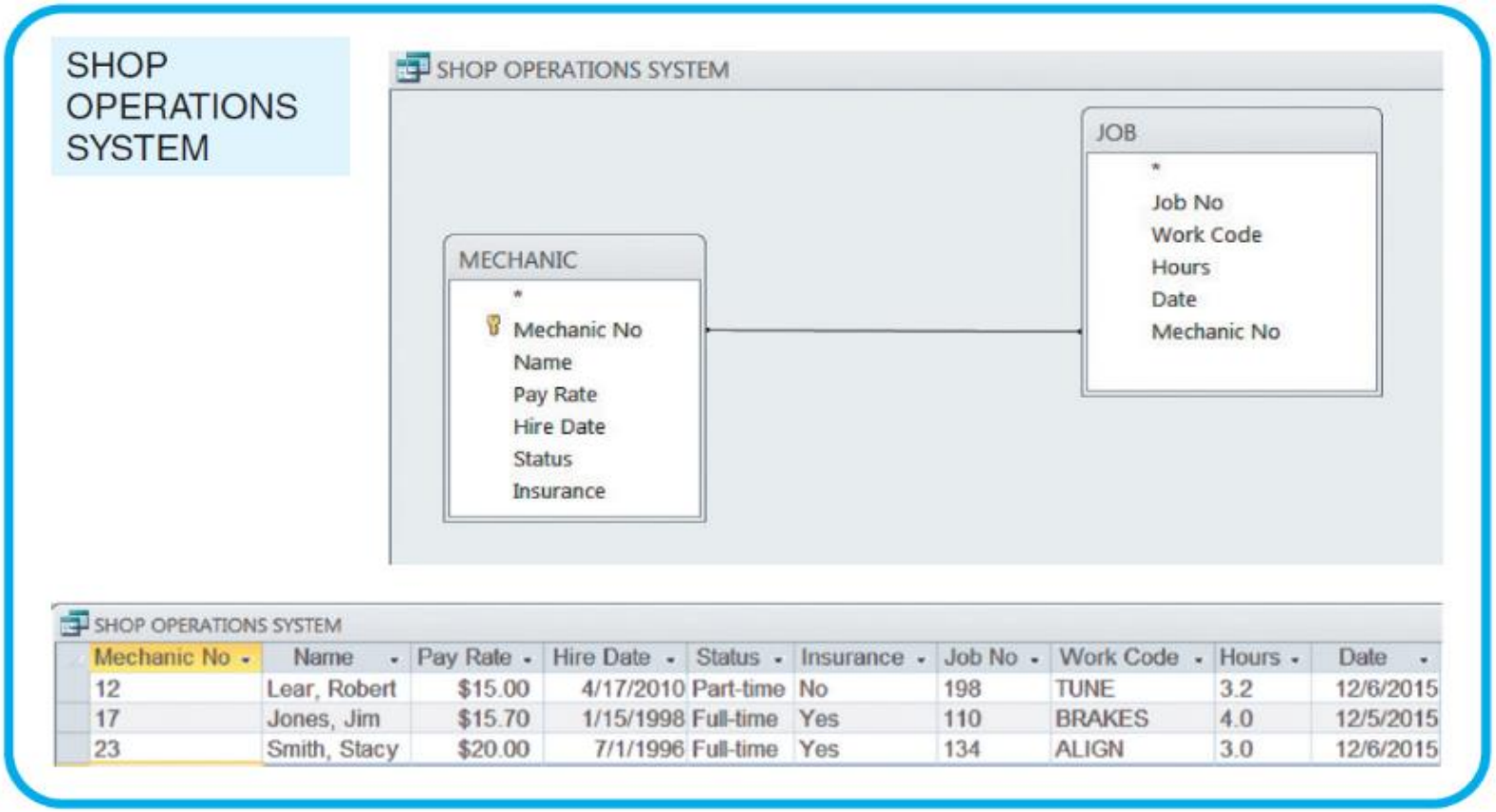
Mario's Auto Shop



Mario's shop uses two separate systems, so certain data must be entered twice. This redundancy is inefficient and can produce data errors.

Data Design Concepts (Cont. 2)

Danica's Auto Shop



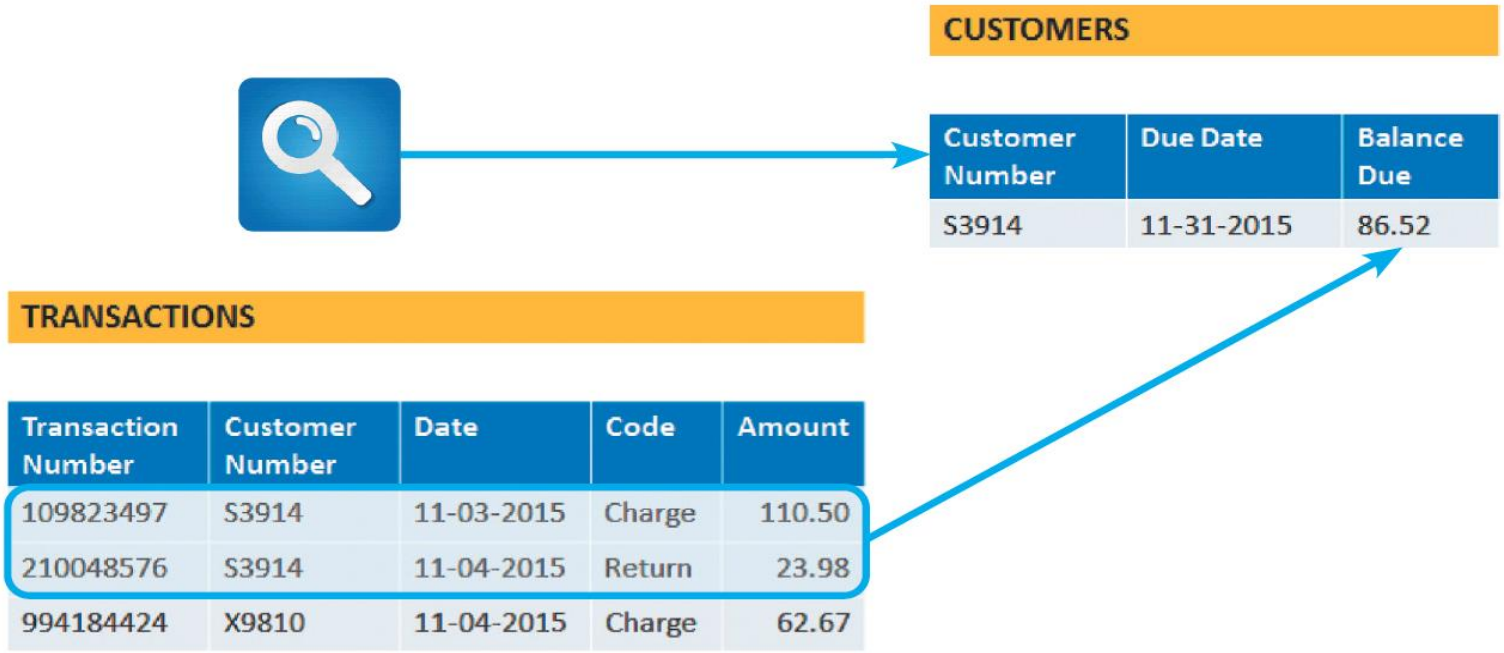
SHOP OPERATIONS SYSTEM

Mechanic No	Name	Pay Rate	Hire Date	Status	Insurance	Job No	Work Code	Hours	Date
12	Lear, Robert	$15.00	4/17/2010	Part-time	No	198	TUNE	3.2	12/6/2015
17	Jones, Jim	$15.70	1/15/1998	Full-time	Yes	110	BRAKES	4.0	12/5/2015
23	Smith, Stacy	$20.00	7/1/1996	Full-time	Yes	134	ALIGN	3.0	12/6/2015

Danica's SHOP OPERATIONS SYSTEM uses a database design, which avoids duplication. The data can be viewed as if it were one large table, regardless of where the data is stored physically.

Data Design Concepts (Cont. 3)

- **Is File Processing Still Important?**
  - Used by some companies to handle large volumes of structured data on a regular basis
    - Cost-effective in certain situations



A credit card company that posts thousands of daily transactions might consider a file processing option.

The Database Environment

- **Database management system (DBMS):** Collection of tools, features, and interfaces that enables users to add, update, manage, access, and analyze data

In this example, a sales database can support four separate business systems.

The Database Environment

- DBMS advantages

✓ **Scalability** - System can be expanded, modified, or downsized

✓ **Economy of scale**

- Database design allows better utilization of hardware

The Database Environment

- DBMS advantages (cont….)

✓ **Enterprise-wide application**

- A database administrator (DBA) assesses overall requirements and maintains the database

✓ **Stronger standards**

- Standards for data names, formats, and documentation are followed uniformly throughout the organization

✓ **Better security**

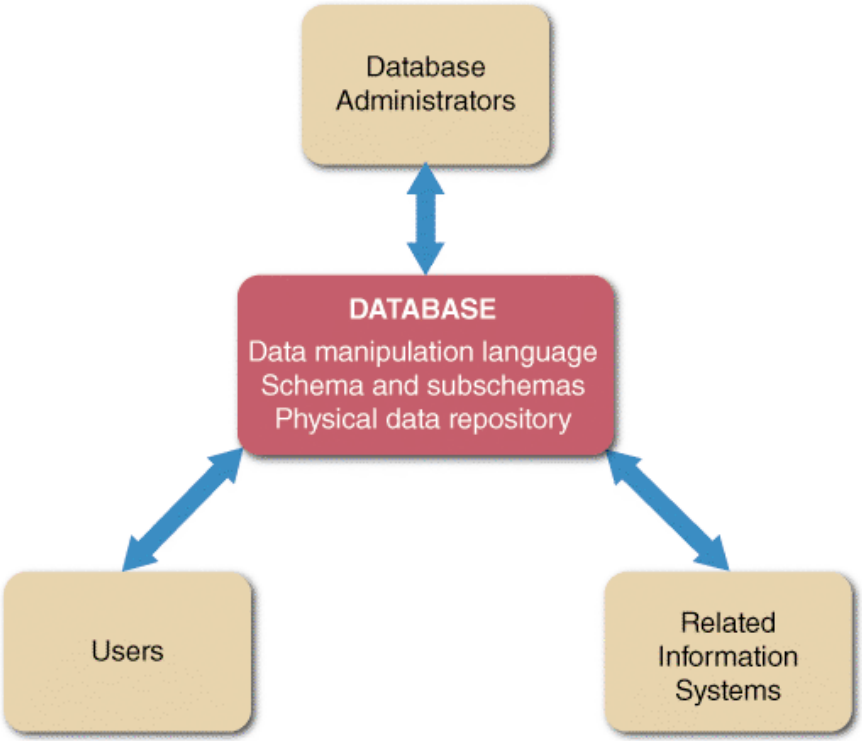
  - Only legitimate users can access the database
  - Different users have different levels of access

✓ **Data independence**

  - Systems that interact with a DBMS are relatively independent of how physical data is maintained

DBMS Components

- **Interfaces for Users, Database Administrators, and Related Systems**
  - Users
    - Work with predefined queries and switchboard commands
    - Use query languages to access stored data
  - Database administrators
    - Responsible for DBMS management and support
  - Related information systems
    - DBMS provides support to related information systems



In addition to interfaces or users, database administrators, and related information systems, a DBMS also has a data manipulation language, a schema and subschemas, and a physical data repository.

DBMS Components (Cont.)

- **Data Manipulation Language (DML)**
  - Controls database operations
- **Schema**
  - Descriptions of all fields, tables, and relationships
    - Subschema: Portions of the database that a particular system or user needs or is allowed to access
- **Physical Data Repository**
  - Contains the schema and subschemas
  - Can be centralized or distributed at several locations
  - Uses open database connectivity (ODBC)-compliant software that enables communication among the systems and DBMSs

Web-Based Data Design

- **Connecting to the Web**
  - Databases are created and managed by using languages and commands that have nothing to do with HTML
    - Objective - To connect the database to the Web and enable data to be viewed and updated
    - Middleware is used to integrate different applications and allow them to exchange data

- **Data Security**
  - Web-based data must be secure, yet easily accessible to authorized users

Web-Based Data Design (Cont. 1)

CHARACTERISTIC	EXPLANATION
Global access	The Internet enables worldwide access, using existing infrastructure and standard telecommunications protocols.
Ease of use	Web browsers provide a familiar interface that is user-friendly and easily learned.
Multiple platforms	Web-based design is not dependent on a specific combination of hardware or software. All that is required is a browser and an Internet connection.
Cost effectiveness	Initial investment is relatively low because the Internet serves as the communication network. Users require only a browser, and Web-based systems do not require powerful workstations. Flexibility is high because numerous outsourcing options exist for development, hosting, maintenance, and system support.
Security issues	Security is a universal issue, but Internet connectivity raises special concerns. These can be addressed with a combination of good design, software that can protect the system and detect intrusion, stringent rules for passwords and user identification, and vigilant users and managers.
Adaptability issues	The Internet offers many advantages in terms of access, connectivity, and flexibility. Migrating a traditional database design to the Web, however, can require design modification, additional software, and some added expense.

Table: A Web-based design characteristics. In a Web-based design, the Internet serves as the front end, or interface, for the database management system. Access to the database requires only a Web browser and an Internet connection.

Data Design Terms

- **Definitions**
  - **Entity** - Person, place, thing, or event for which data is collected and maintained
  - **Table** or **file**: Contains a set of related records that store data about a specific entity
  - **Field** (attribute) - Single characteristic or fact about an entity
    - Common field: Attribute that appears in more than one entity
  - **Tuple** (record): Set of related fields that describes one instance, or occurrence, of an entity

Data Design Terms (Cont. 1)

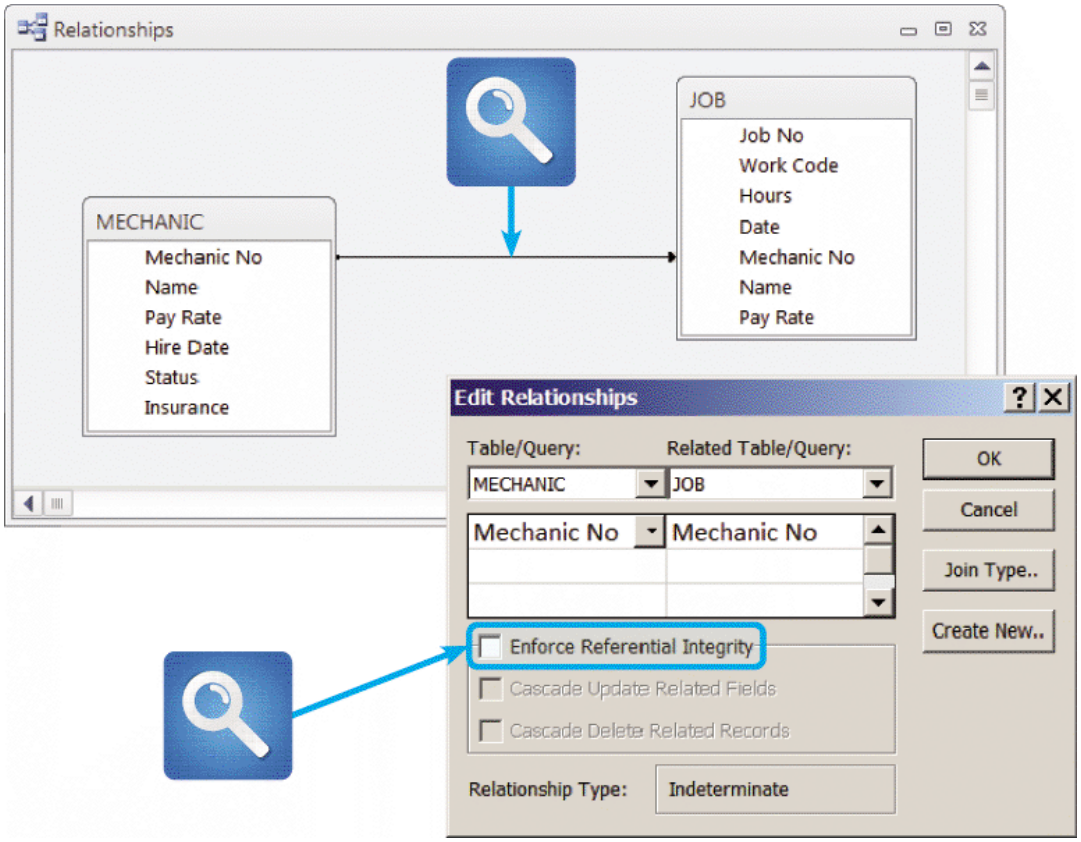
- **Key Fields**
  - **Primary key:** Field or combination of fields that uniquely and minimally identifies a particular member of an entity
    - Called a **combination key**
  - **Candidate key**: Any field that could serve as a primary key
  - **Foreign key:** Field in one table that must match a primary key value in another table for a relationship between the two tables to exist

CustomerID	FisrtName	LastName	DoB

Customer table with it's attributes

Data Design Terms (Cont. 2)

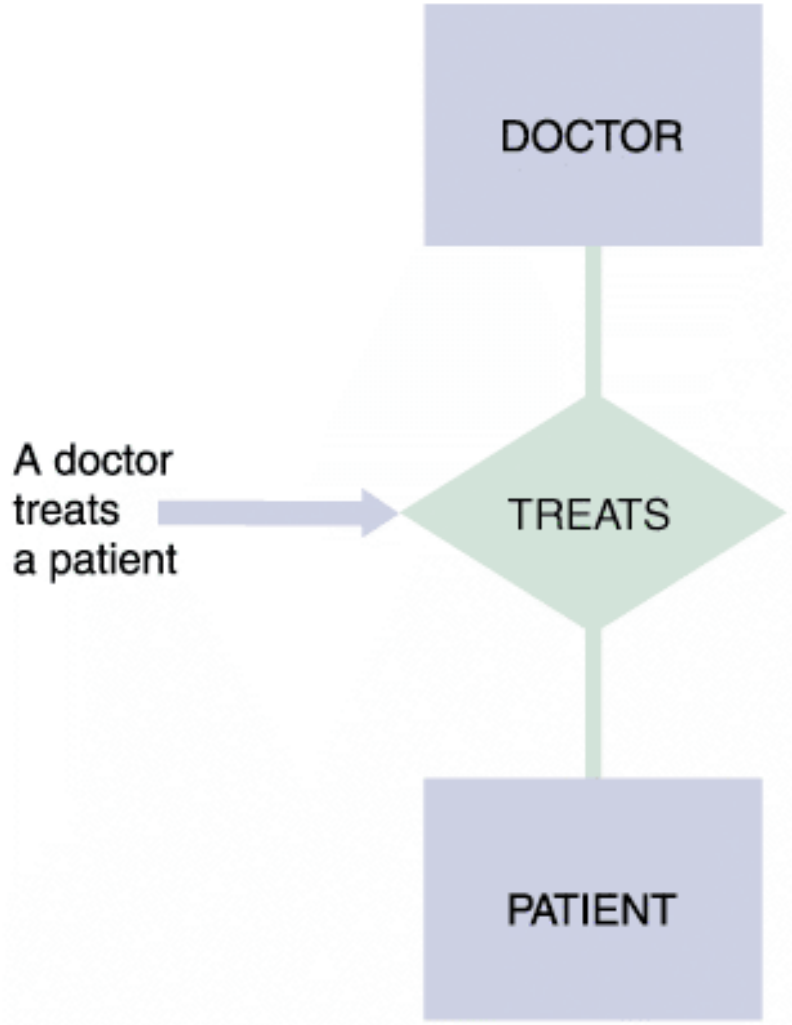
- **Referential Integrity**
  - Set of rules that avoids data inconsistency and quality problems
  - Refers to the accuracy and consistency of data within a relationship. In relationships, data is linked between two or more tables
  - Referential integrity requires that, whenever a foreign key value is used it must reference a valid, existing primary key in the parent table.



Microsoft Access allows a user to specify that referential integrity rules will be enforced in a relational database design.

Entity-Relationship Diagrams

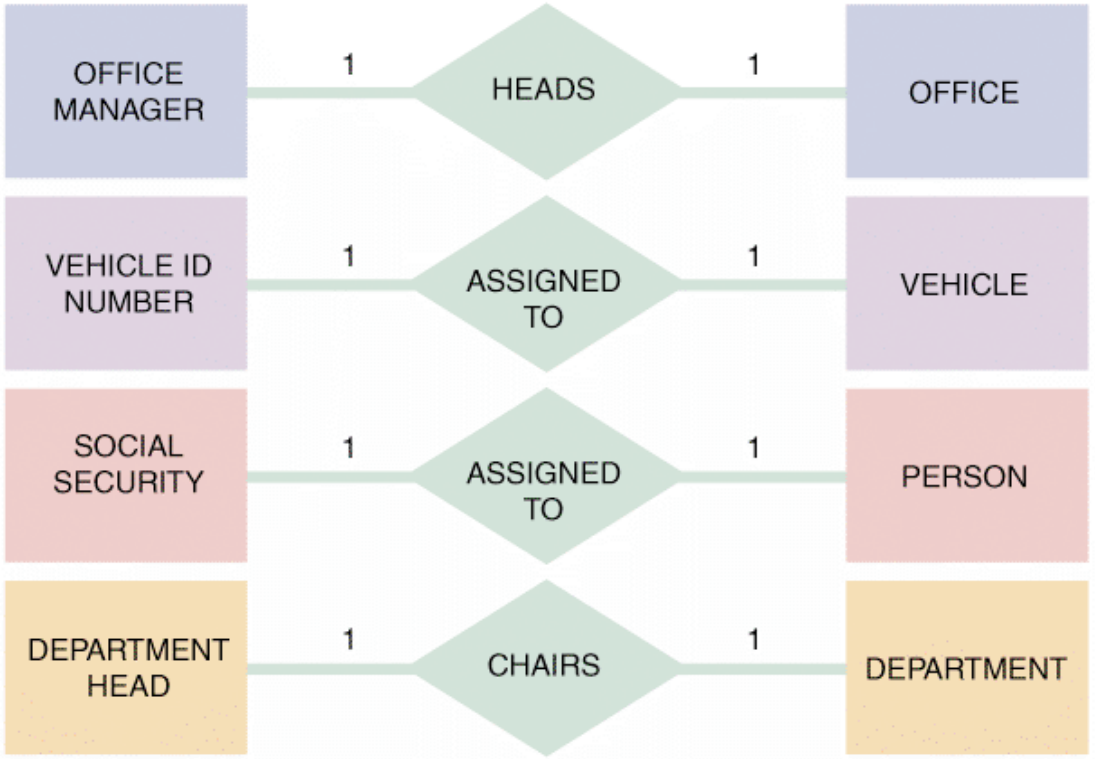
- **Drawing an ERD**
  - List the entities that were identified during the systems analysis phase
    - Consider the nature of the relationships that link them



In an entity-relationship diagram, entities are labeled with singular nouns and relationships are labeled with verbs. The relationship is interpreted as a simple English sentence.

Entity-Relationship Diagrams (Cont. 1)

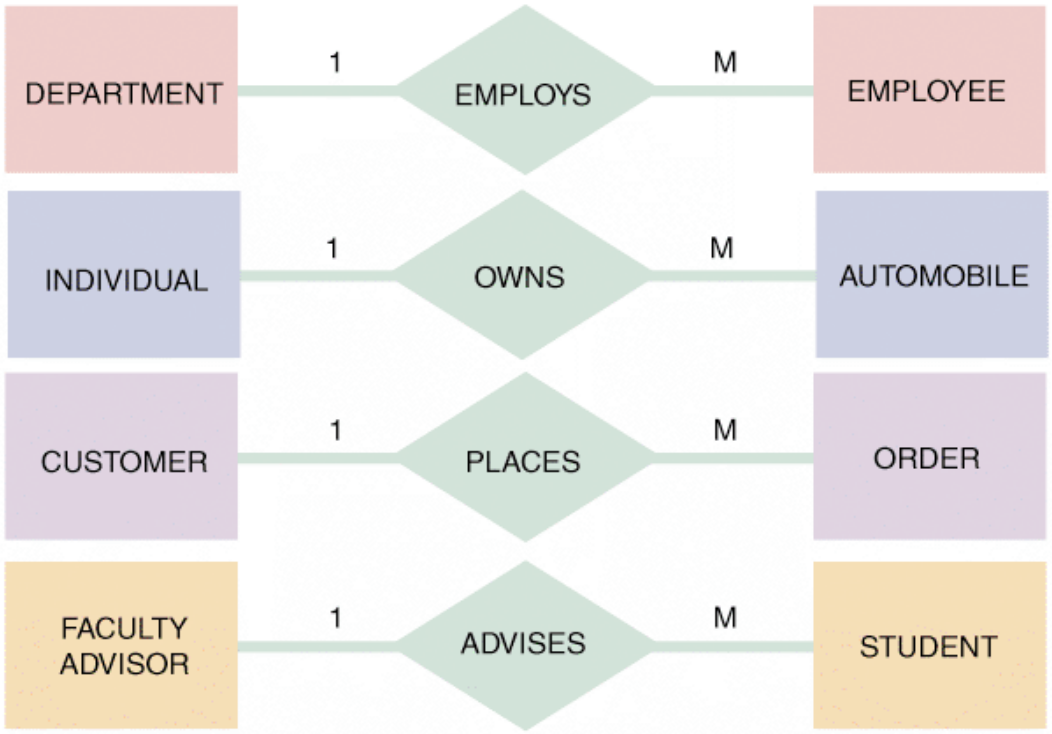
- **Types of Relationships**
  - **One-to-one relationship:** Exists when exactly one of the second entity occurs for each instance of the first entity
    - Abbreviated **1:1**



Examples of one-to-one (1:1) relationships.

Entity-Relationship Diagrams (Cont. 2)

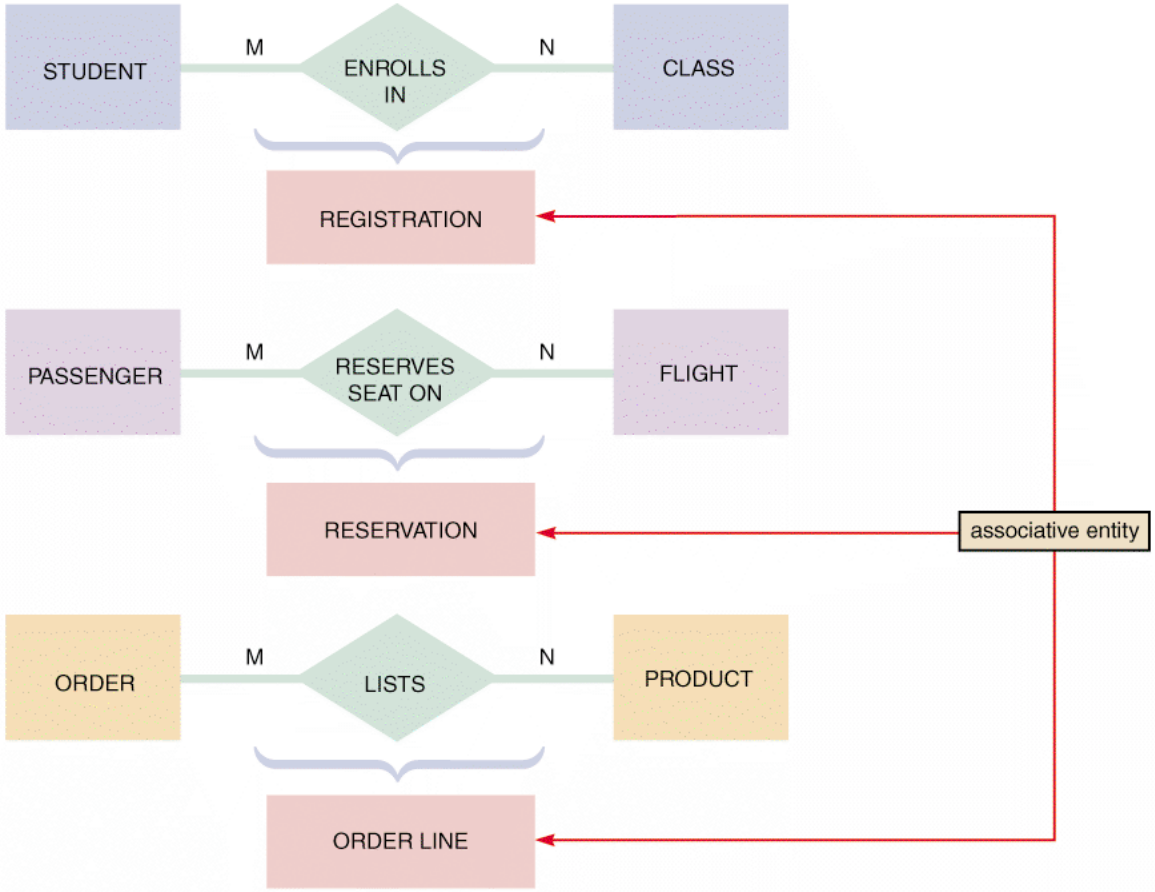
- **One-to-many relationship:** Exists when one occurrence of the first entity can relate to many instances of the second entity
  - Each instance of the second entity can associate with only one instance of the first entity
  - Abbreviated **1:M**



Examples of one-to-many (1:M) relationships.

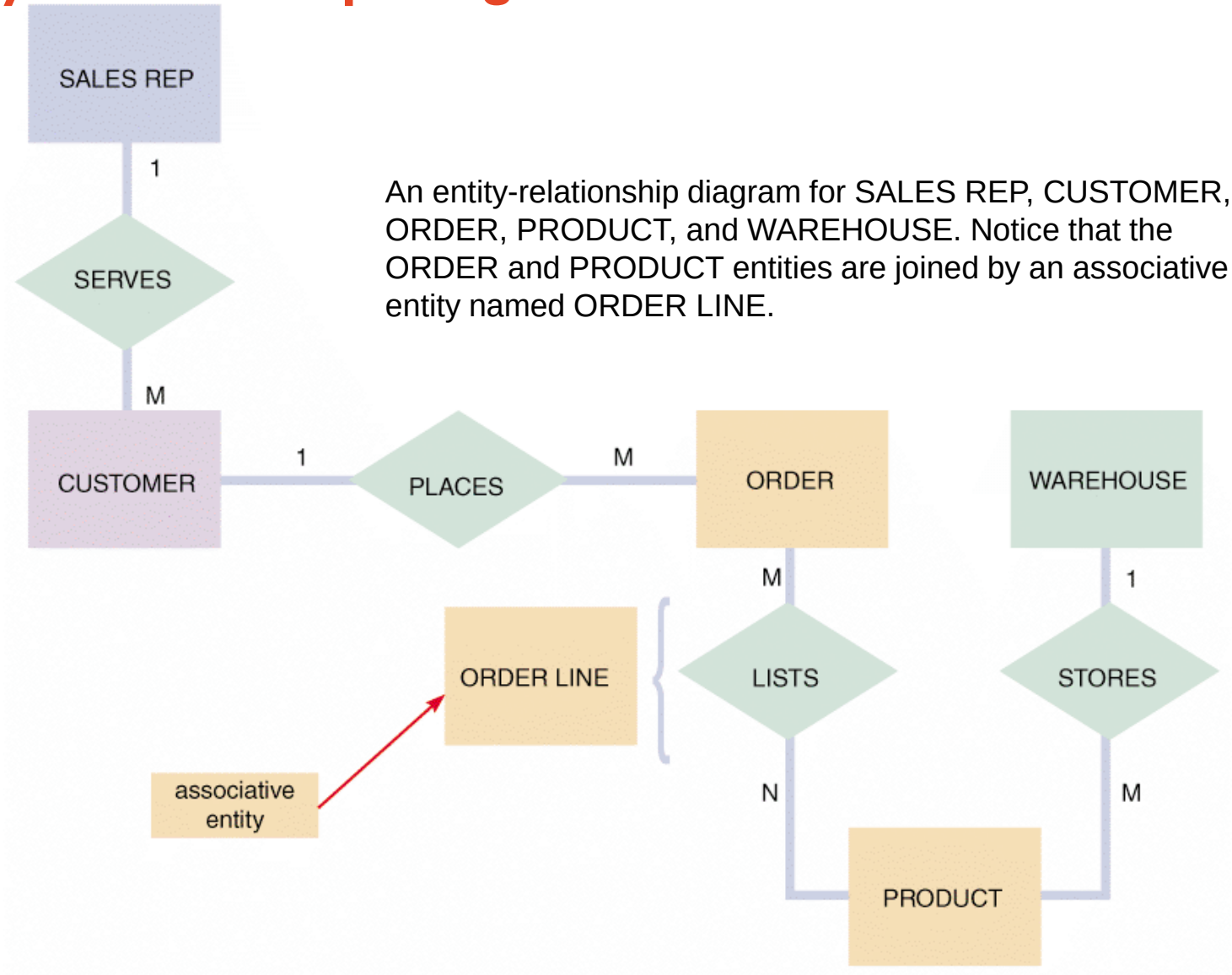
Entity-Relationship Diagrams (Cont. 3)

- **Many-to-many relationship**
  - Exists when one instance of the first entity can relate to many instances of the second entity, and vice versa
  - Abbreviated **M:N**



Examples of many-to-many (M:N) relationships. Notice that the event or transaction that links the two entities is an associative entity with its own set of attributes and characteristics

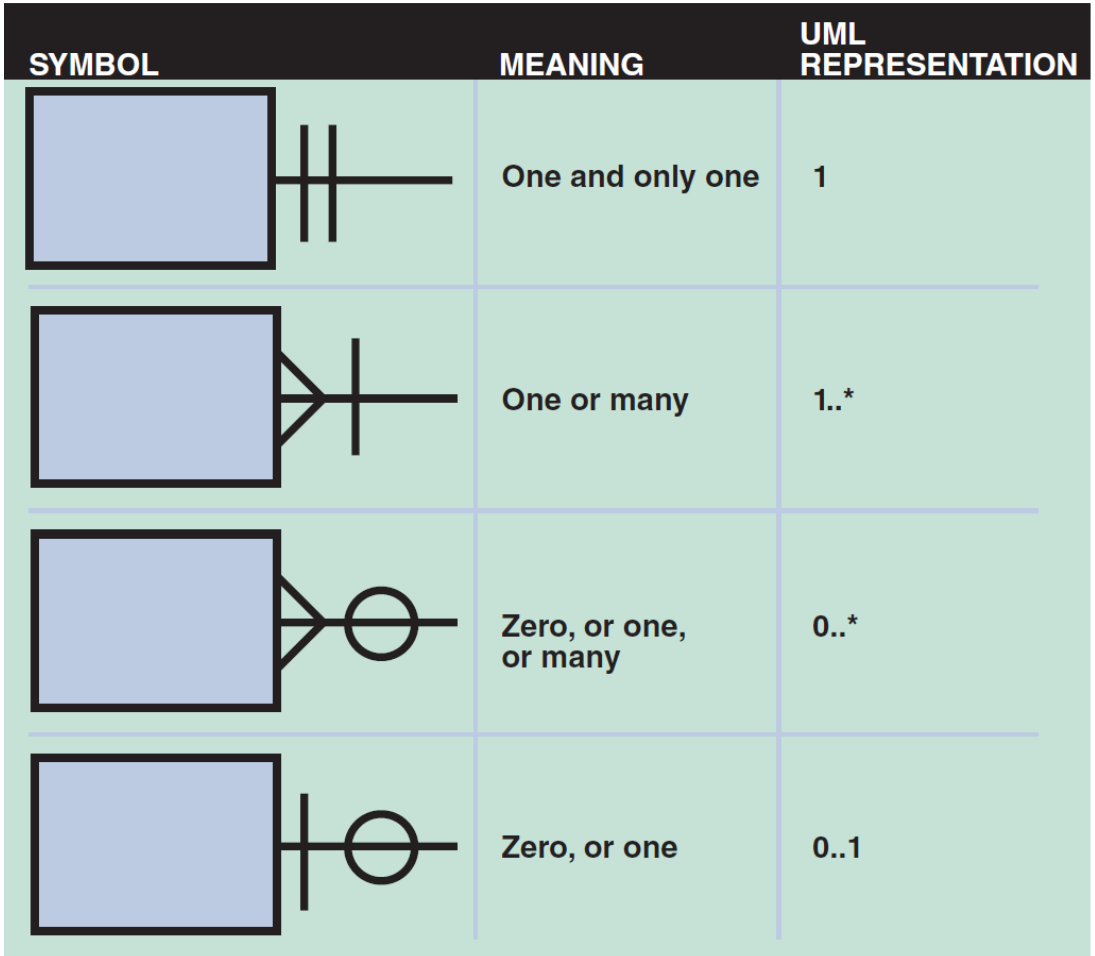
Entity-Relationship Diagrams (Cont. 4)



An entity-relationship diagram for SALES REP, CUSTOMER, ORDER, PRODUCT, and WAREHOUSE. Notice that the ORDER and PRODUCT entities are joined by an associative entity named ORDER LINE.

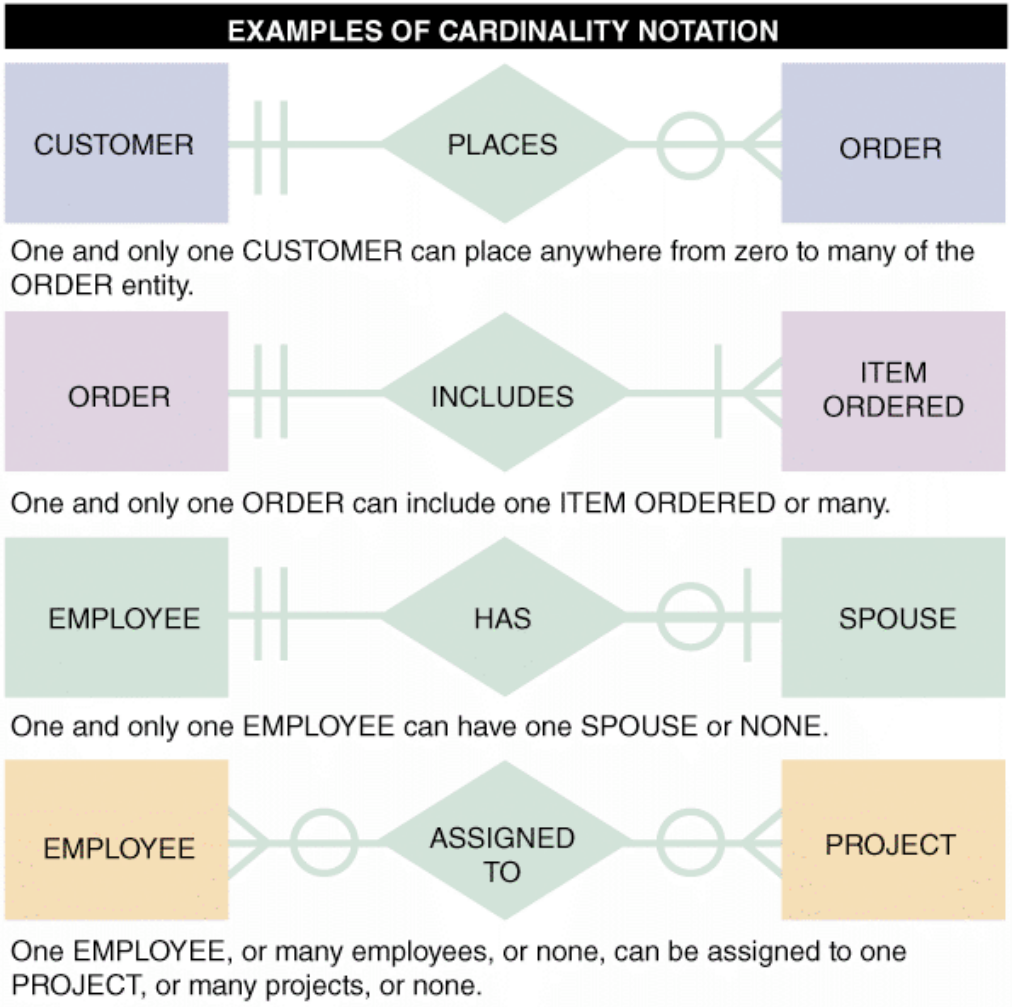
Entity-Relationship Diagrams (Cont. 5)

- **Cardinality**
  - Describes the numeric relationship between two entities
  - Shows how instances of one entity relate to instances of another entity



Crow's foot notation is a common method of indicating cardinality. The four examples show how you can use various symbols to describe the relationships between entities.

Entity-Relationship Diagrams (Cont. 6)



In the first example of cardinality notation, one and only one CUSTOMER can place anywhere from zero to many of the ORDER entity.

In the second example, one and only one ORDER can include one ITEM ORDERED or many.

In the third example, one and only one EMPLOYEE can have one SPOUSE or none.

In the fourth example, one EMPLOYEE, or many employees, or none, can be assigned to one PROJECT, or many projects, or none.

Class Exercise 1

- Q1: What could be the primary key in a student table?
- Q2: How about candidate key in a student table? Give an example.

Class Exercise 2

"Many authors write more than one book, and more than one author writes some books".

- Q1: Identify the possible entities based on the scenario above
- Q2: Draw an ERD

Additional Exercise

- "There are many students enrolled in a course and a specific lecture teaches that course".
- Use any assumptions needed and draw an ERD

Use of Codes

- **Overview of Codes**
  - Codes are shorter than the data they represent
    - Save storage space and costs
    - Decrease data entry time and transmission time
  - Codes can:
    - Reveal or conceal information
    - Reduce data input errors
  - Coded data is easier to remember

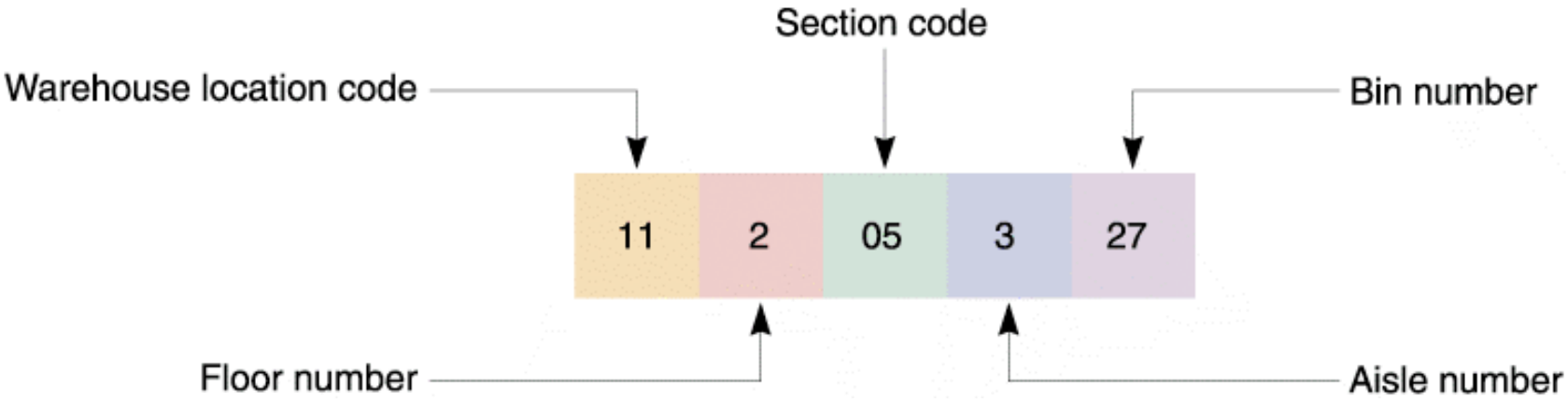
Use of Codes (Cont. 1)

- **Types of Codes**
  - **Sequence codes:** Numbers or letters assigned in a specific order
    - Contain no additional information other than an indication of order of entry into the system
  - **Block sequence codes:** Use blocks of numbers for different classifications
    - Sequence of numbers in a particular block can have additional meaning
  - **Significant digit codes:** Distinguish items by using a series of subgroups of digits

Use of Codes (Cont. 2)

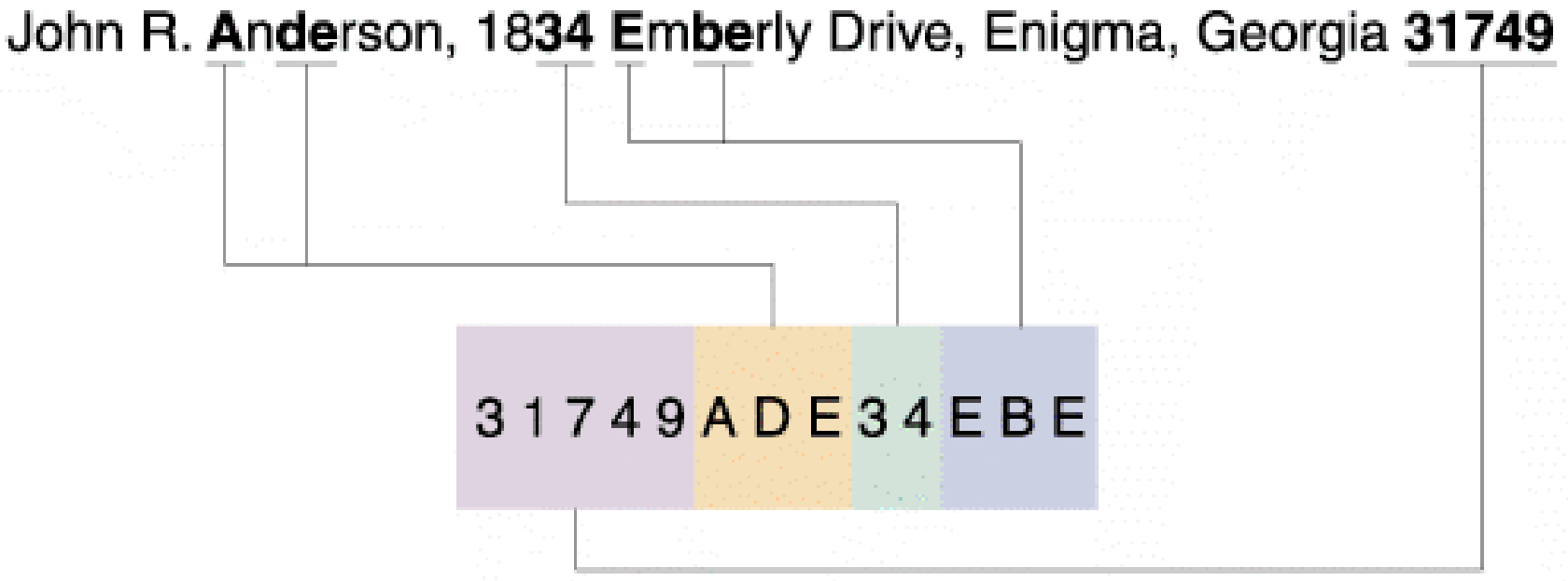
- **Alphabetic codes:** Use letters to distinguish one item from another
  - **Category codes:** Identify a group of related items
  - **Abbreviation codes:** Alphabetic abbreviations
  - **Mnemonic codes:** Use specific combination of letters that are easy to remember
- **Derivation codes:** Combine data from different item attributes, or characteristics
- **Cipher codes:** Use a keyword to encode a number
- **Action codes:** Indicate what action is to be taken with an associated item

Use of Codes (Cont. 4)



Sample of a code that uses significant digits to pinpoint the location of an inventory item.

Use of Codes (Cont. 5)



A magazine subscriber code is derived from various parts of the name and address.

Use of Codes (Cont. 6)

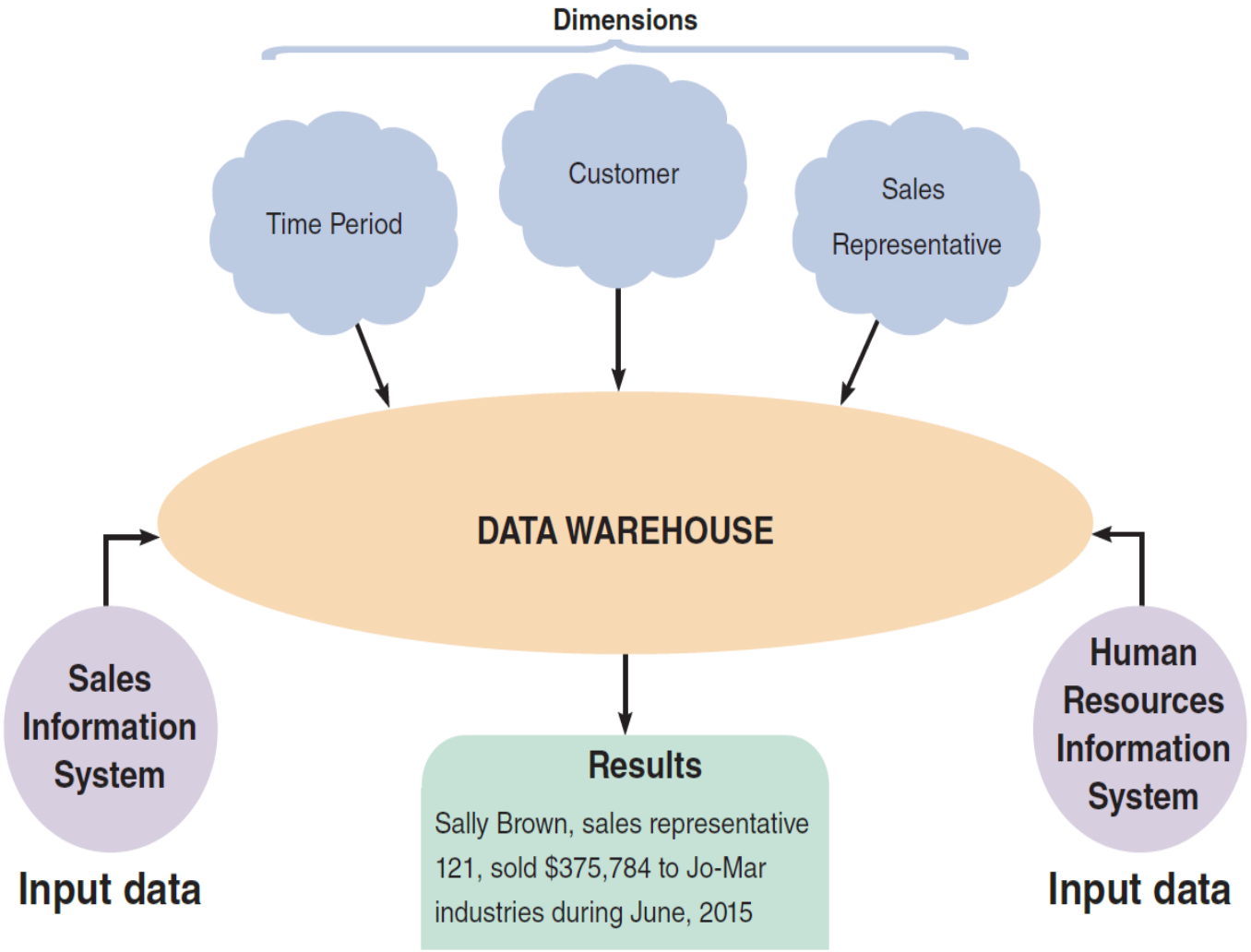
- **Best Practices in Designing Codes**
  - Keep codes concise
  - Allow for expansion
  - Keep codes stable
  - Make codes unique
  - Use sortable codes

  - Use a simple structure
  - Avoid confusion
  - Make codes meaningful
  - Use a code for a single purpose
  - Keep codes consistent

Data Storage and Access

- **Tools and Techniques**
  - **Data warehouse**
    - An integrated collection of data that can include seemingly unrelated information, no matter where it is stored in the company
  - **Data mart**
    - Designed to serve the needs of a specific department

Data Storage and Access



A data warehouse stores data from several systems. By selecting data dimensions, a user can retrieve specific information without having to know how or where the data is stored.

Data Storage and Access (Cont. 1)

- **Data Mining (clickstream storage)**
  - Looks for meaningful data patterns and relationships
  - Suggested goals for data mining
    - Increase the number of pages viewed per session and referred customers
    - Reduce **clicks to close**
    - Increase checkouts per visit and average profit per checkout
  - Can be used to build a profile of new customers

Data Storage and Access (Cont. 2)

- **Logical versus Physical Storage**
  - **Logical storage:** Data that a user can view, understand, and access, regardless of how or where that information actually is organized or stored
  - **Physical storage:** Strictly hardware-related
    - Involves the process of reading and writing binary data to physical media

Data Storage and Access (Cont. 3)

- **Data Coding**
  - **EBCDIC** (Extended Binary Coded Decimal Interchange Code)
    - Used on mainframe computers and high-capacity servers
  - **ASCII** (American Standard Code for Information Interchange)
    - Used on most personal computers
  - **Binary storage format**
    - Represents numbers as actual binary values
  - **Unicode:** Uses two bytes per character

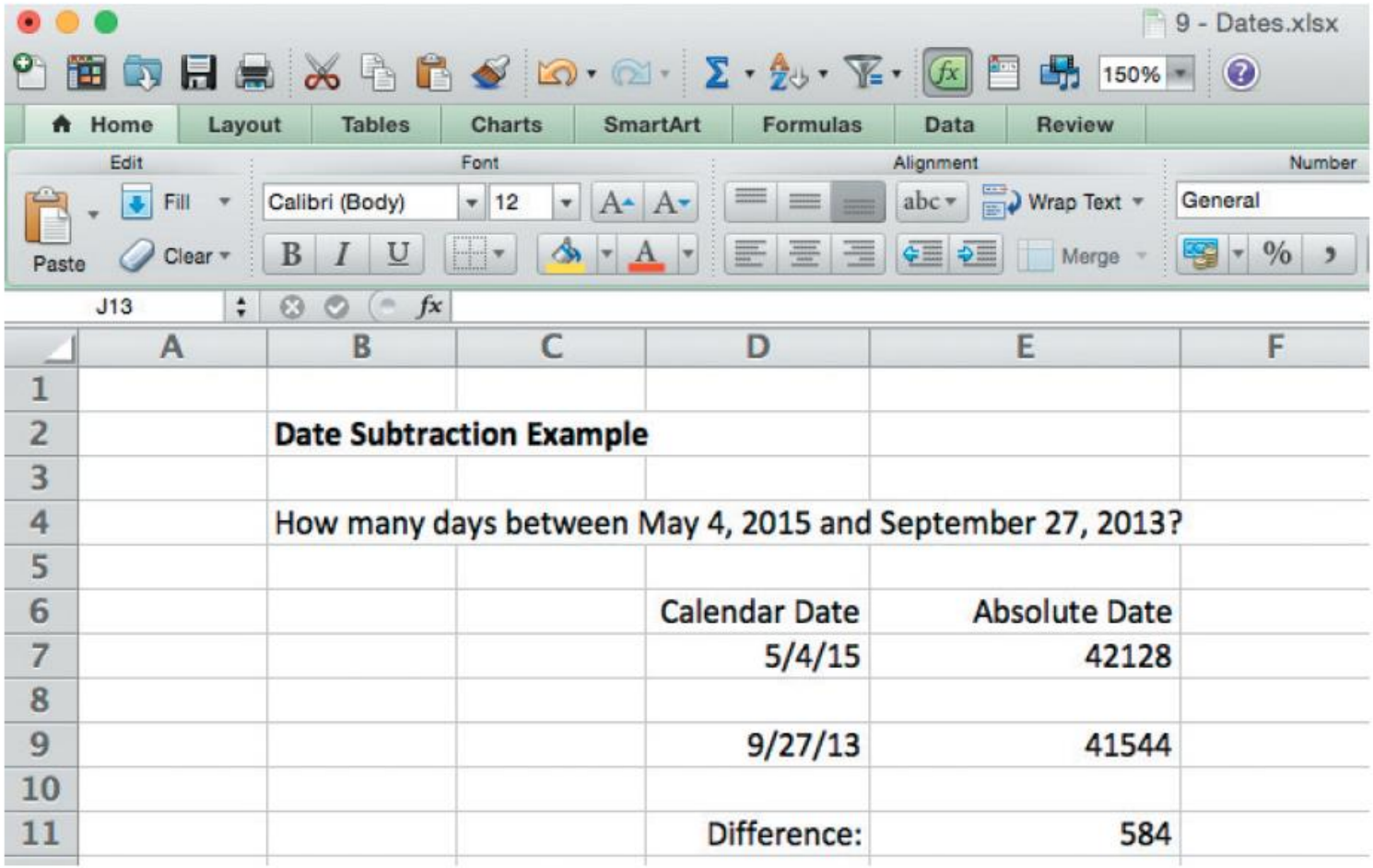
Data Storage and Access (Cont. 5)

- Storing dates
  - **International Organization for Standardization (ISO)** requires a format of four digits for the year, two for the month, and two for the day (YYYYMMDD)

  - **Absolute date:**

  Total number of days from some specific base date

Data Storage and Access (Cont. 5)



	A	B	C	D	E	F
1						
2		**Date Subtraction Example**				
3						
4		How many days between May 4, 2015 and September 27, 2013?				
5						
6				Calendar Date	Absolute Date	
7				5/4/15	42128	
8						
9				9/27/13	41544	
10						
11				Difference:	584	

Microsoft Excel uses **absolute dates** in calculations. In this example, May 4, 2015, is displayed as 42128, and September 27, 2013, is displayed as 41544. The difference between the dates is 584 days.

Lecture Summary

- A database consists of linked tables that form an overall data structure
  - DBMS enable users to add, update, manage, access, and analyze data in a database

- DBMS designs are more powerful and flexible than traditional file-oriented systems
  - Components include interfaces for users, database administrators, and related systems

- In an information system, an entity is a person, place, thing, or event for which data is collected and maintained

Lecture Summary (Cont. 1)

- Key fields include primary keys, candidate keys, foreign keys, and secondary keys

- An entity-relationship diagram (ERD) is a graphic representation of all system entities and the relationships among them
  - Relationship between two entities is referred to as cardinality

Announcement (if any)

Q &A?

Thanks everyone !