

# Module\_2

## Entity Relationship Diagrams for Data Analysis

---

CS 3410 Introduction to Database



# Introduction

Businesses are built on understanding and using the data they collect.

However, as size increases for business, the complexity of the data does as well

- **Entity Relationship Diagrams** are used to represent the complex relations between any facets or items in a business. These commonly take the form of a wire frame representation between tables, leading to this having many uses outside of big business.
- **Entities, Attributes, and Relations** make up the bulk of a diagram, allowing both users and creators to effectively understand complex relations between parts of an environment.

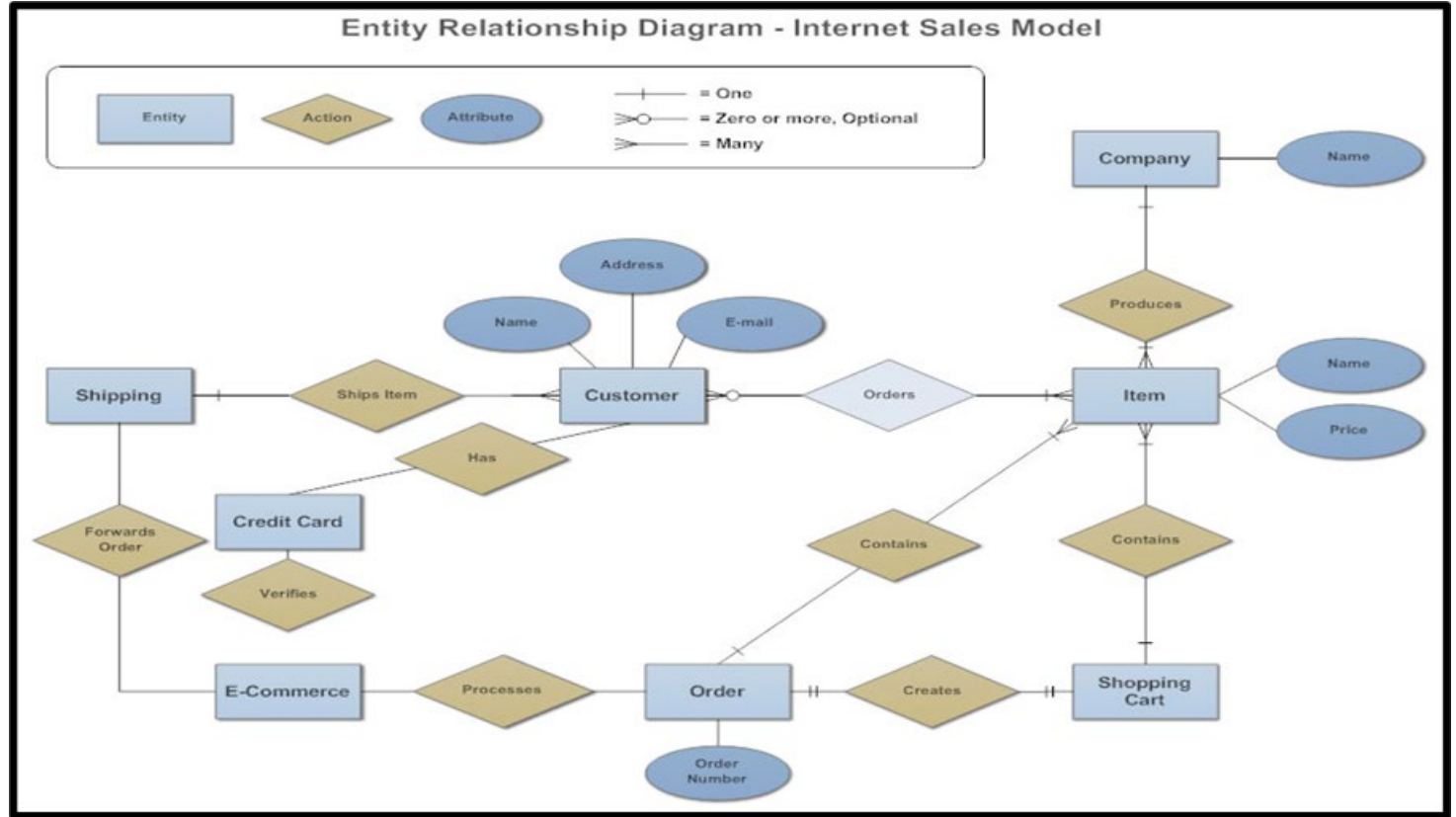
# History of the ERD

- Suggested in the 1960-70s due to an increase in business complexity
- Unified system to aid in visually representing both company data, and its relations through departments
- This led to an easier understanding and management of the interconnected systems

# Original Design of the ERD

- Style focused on shapes determining the type of data
- Lines for entities used actions for clarification of the relation
- Other variations of the lines focused on explaining the quantity of the relation, like one or many.

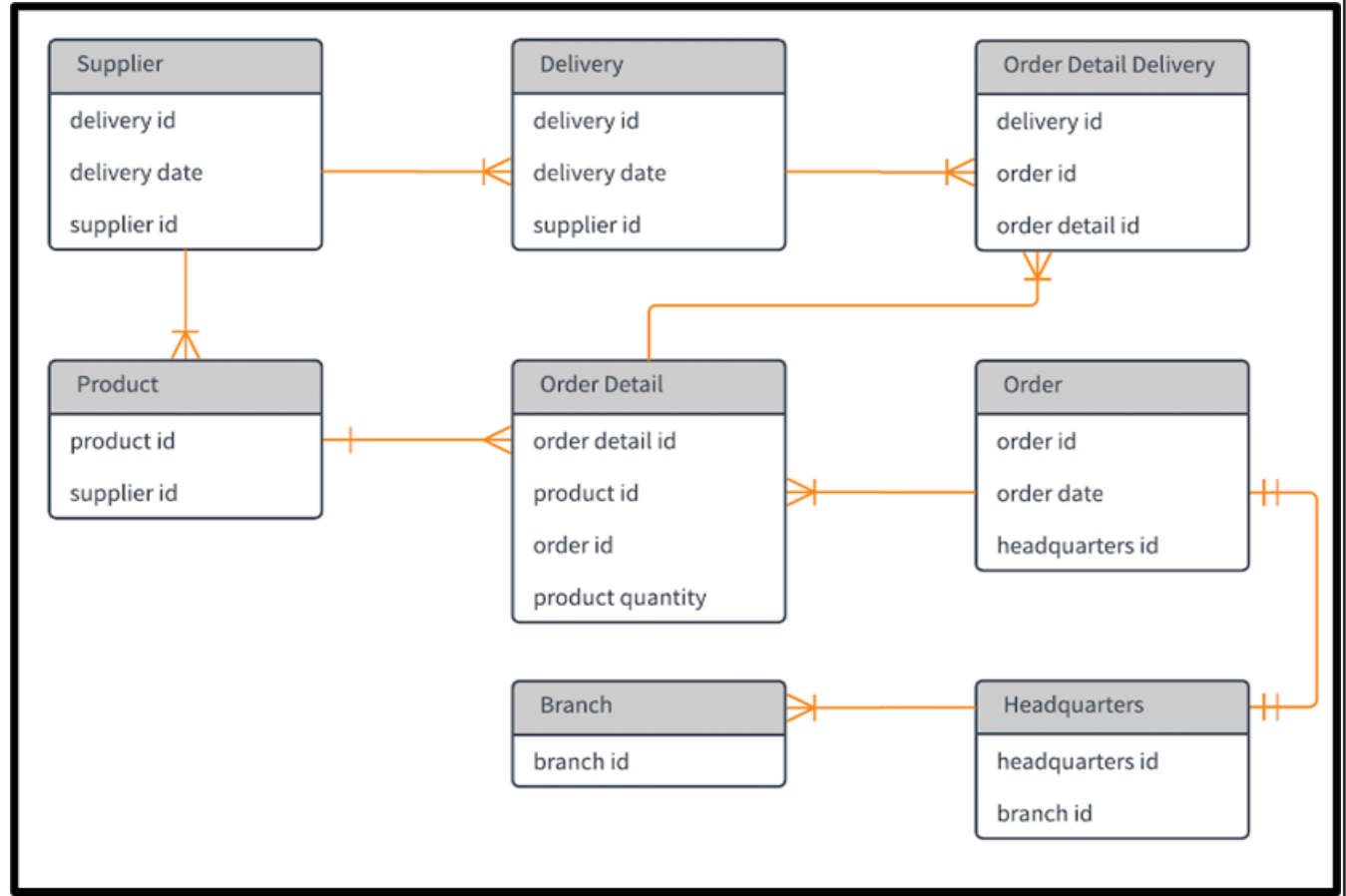
**Figure 1** – Original style ERD (SmartDraw) Depicting an e-commerce company



# Newer variant of ERD

- Focuses on representing data in a less cluttered manner
- Uses tables to represent entities and their attributes
- Relation lines lack action description in diamonds

**Figure 2** - New Style ERD (Lucidchart) depicting a similar e-commerce company



# Why use an ERD?

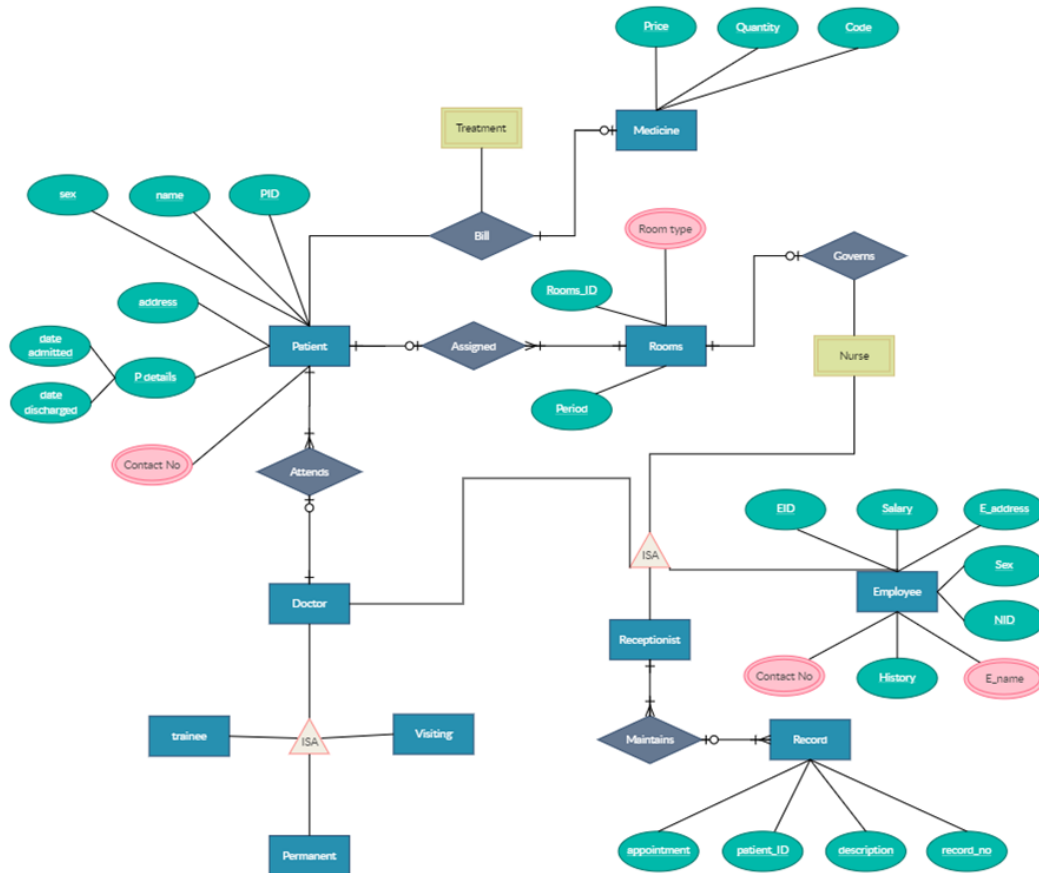
- An ERD is useful for overall business management to maintain a high level of efficiency.
- Models allow users to effectively:
  1. **Understand** integral and complex concepts to the company
  2. **Represent** data in a simple manner
  3. **Model** certain systems ahead of creation
  4. **Identify** and solve problems with future or current systems



# Using an ERD to model a Hospital

- When using this system, it is easy to see the benefits of an interconnected system such as the examples below:
  1. Upon patient arrival, the Doctor can access all medical records regarding the patient, even if they went to a different hospital.
  2. If the patient is given a prescription, the details are logged on that record.
  3. Costs are easily communicated to the Patient or their insurance, which are then logged for easy viewing.
  4. All these systems collaborate for open information for the patient and doctor to access.

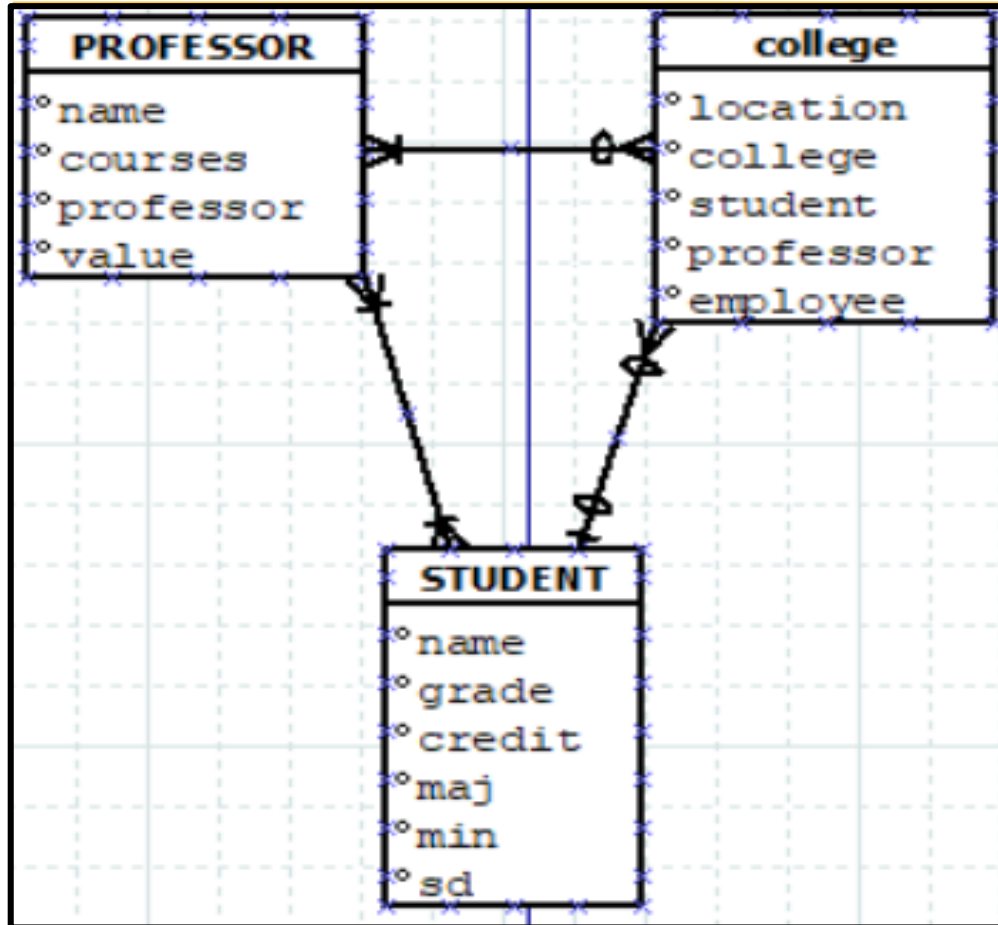




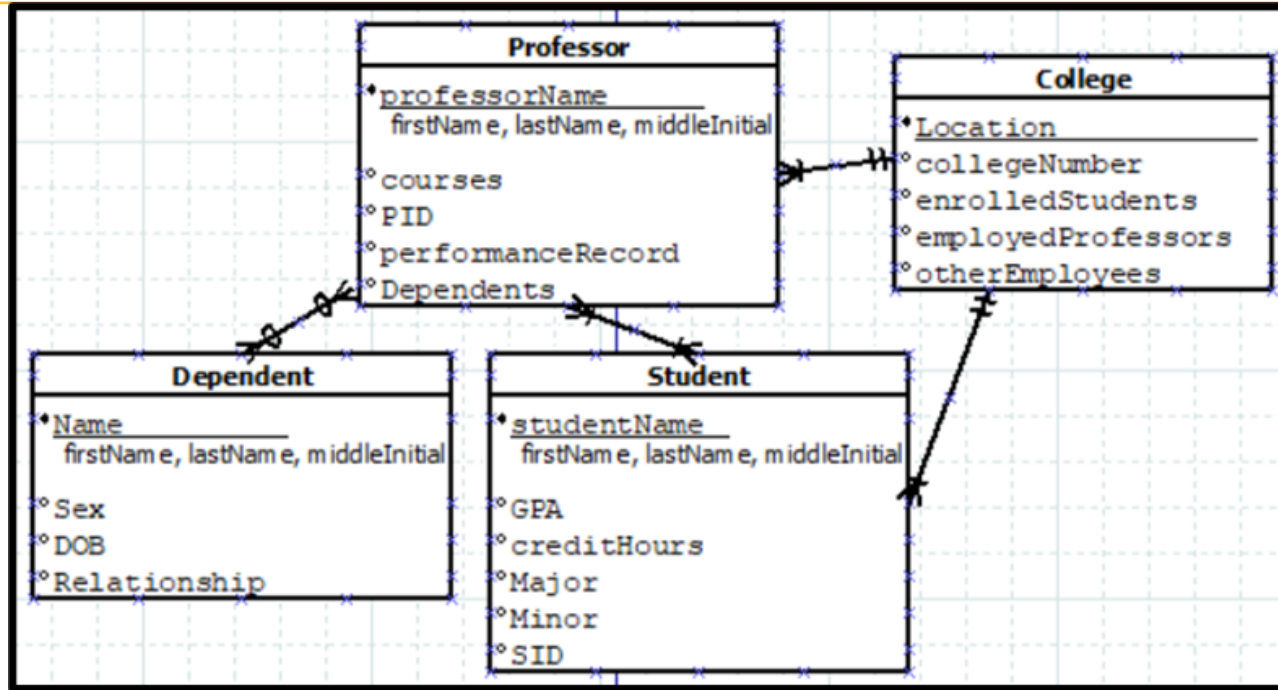
**Figure 3 - Old Style ERD (Createrly, 2008) depicting basic hospital management**

# Introduction to the Implementation of ERDs

- Representing data, especially considering the simplification that goes into creating an effective ERD, is difficult
- **Variable Design:**
  - When creating an ERD, it is important to follow the below conventions:
    1. Entities and Attributes should have self explanatory names.(EX student and professor should be named accordingly)
    2. These names should not be too long, but also not too short. Generally, two to three-word names should be the most you should see regarding an attribute or Entity. Acronyms also work for shortening names.
    3. These names should follow conventions like camelCase and Capitalization so as to be coherently read by a layman.



**Figure 4** - New Style ERD (Bartz, 2020) depicting a College/Professor/Student relation with several discernable issues.



**Figure 5** - New Style ERD (Bartz, 2020) depicting an improved College/Professor/Student relation using several different conventions outlined in slide 11.

# An Iterative approach to ERDs

- **First, create an ERD**
  - This does not have to be perfect; it only has to outline the next step.
- **Optimize the Diagram**
  - Remove repeat or useless entities and their attributes. An ER diagrams' strength is in its' simplicity.
- **Analyze the Relations**
  - Relations should make sense and should only be included if they contribute to the displayed information.
- **Final step**
  - Make sure to spread out the Diagram. Having a cluttered model is sure to confuse people, and it's better not to take that risk.

# Business Intelligence Systems and Data Warehouse

- **Business Intelligence (BI) Systems** are any database system that has data, programs, and personnel and are specialized for the preparation of data for BI processing.
- A **Data Warehouse** is a system used for reporting and data analysis, and is considered a core component of business intelligence.
- This includes day-to-day transactions and other internal data or external data from sources by the **Extract, Transform, and load (ETL) System**.

# Business Intelligence Systems and Data Warehouse Cont...

- **Database SKUs** where the **primary key** consists of one or more column will uniquely identify each row or record in the table.
- Some tables may have one or more **foreign keys** that are used to create **relationship** between tables logically link the tables together.
- **Database SKUs** (A **stock-keeping unit (SKU)** is a scannable bar code, most often seen printed on product labels in a retail store) where the **primary key** consists of one or more column will uniquely identify each row or record in the table.
- Some tables may have one or more **foreign keys** that are used to create **relationship** between tables logically link the tables together.



# Introduction to Structure Query Language (SQL)

- SQL is the universal query language of relational database management systems (DBMS) that is almost always behind user-friendly GUIs.
- In this section, we will briefly talk about SQL queries. We will visit SQL in more detail in later chapters.
- SQL statements can also include a **SQL comment**, which is a block of text used to document the SQL statement but not executed as part of the statement.
- SQL comments are enclosed in the symbols **/\* and \*/**, and any text between these symbols is ignored when the SQL statement is executed. Here is an example:
  - `/* SQL-Query-Ch02 */`



# SQL Cont...

- The **fundamental statement of SQL** query can apply to Microsoft Access, SQL Server, Oracle Database, and MySQL. The basic form of SQL queries uses **SQL SLECT – FROM – WHERE framework**. Here are some basic specifications:
- The **SQL SELECT** statement specifies which **columns** are to be listed in the query results.
- The **SQL FROM** statement specifies which **tables** are to be listed in the query results.
- The **SQL WHERE** statement specifies which **rows** are to be listed in the query results.

**Reading Specified Columns from Single Table** – to obtain values from the SKU\_Data table, we write a **SQL SELECT** statement that contains all of the column names in the table:

```
/* SQL-Query-Ch_02 */
```

	SKU	SKU_Description	Department	Buyer
1	200201	Women's T-Shirts	Clothing	Tyra Perry
2	200202	Mans's T-Shirts	Clothing	Bradly Cooper
3	200203	Children T-Shirt	Clothing	Cora Mathis

```
SELECT      SKU, SKU_Description, Department, Buyer  
  
FROM        SKU_DATA;
```

Below is a SQL queries from a single table, which obtains just the value of the Department and Buyer columns of the SKU\_Data table:

Department	Buyer
Clothing	Tyra Perry
Clothing	Bradly Cooper
Clothing	Cora Mathis

```
SELECT      Department, Buyer
FROM        SKU_DATA;
```

The Catalog\_SKU\_2019 table shows DateOnWebSite. To see these items, we use the follow query:

```
/* SQL-Query-Ch_02 */
```

	SKU	SKU_Description	Department	Buyer	Date on WebSite
1	200201	Women's T-Shirts	Clothing	Tyra Perry	2020-10-10
2	200202	Mans's T-Shirts	Clothing	Bradly Cooper	2019-11-11
3	200203	Children T-Shirt	Clothing	Cora Mathis	2020-20-20

```
SELECT      *  
  
FROM        CATALOG_SKU_2020  
  
WHERE       DateOnWebSite = '10-Oct-2020';
```



# Submitting SQL Statement to the DBMS

Using **SQL in Microsoft Access 2016**, we can execute SQL statements and run Query windows in **Design view**.

- Click the **Create** command tab to display the Create command groups, as shown in Figure 2-13.
- Click the **Query Design** button.
- The Query1 tabbed document window is displayed in Design view, along with the Show Table dialog box, as shown in Figure 2-10.
- Click the **Close** button on the Show Table dialog box. The Query1 document window now looks as shown in Figure 2-11, with the Query Tools contextual command tab and the Design command tab displayed. This window is used for creating and editing Microsoft Access queries in Design view and is used with Microsoft Access QBE (Query By Design).

File Home Create External Data Database Tools Help Fields Table Tell me what you want to do



Application  
Parts ▾

Templates



Table

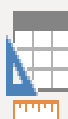


Table  
Design

Tables

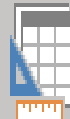


SharePoint  
Lists ▾



Query  
Wizard

Queries



Query  
Design



Form



Form  
Design



Blank  
Form

Forms



Form Wizard



Navigation ▾



More Forms ▾



Report



Report  
Design



Blank  
Report

Reports



Report Wizard



Labels

All Access Objects

Search...



Tables



Table1



Table1



ID



Click to Add



(New)

# In Conclusion

## ERDs:

1. Model Real world businesses
2. Manage Databases
3. Prototype ideas or concepts
4. Allow Informative and Cohesive management efforts



# Summary

- An Entity Relationship Diagram, or ER diagram, is a visual representation of a database and the interactions between its entities and attributes. The need for ER diagrams arose from the need to structure complex databases in the 1970s. Aspects of ER diagrams that are appealing include their not only being easy to understand but also easy to create and teach.

# Summary Cont...

- One of the earliest models was developed by Peter Chen, which included entities (rectangles), attributes (ovals) and their relationships (diamonds).
- Newer **ER diagrams** (using Unifying Modeling Language) simplify this by attaching attributes to their entities in a rectangle and by having relationships designated by lines with specific end-connection symbols that represent the following:
  - One-to-one
  - One-to-many
  - Many-to-one
  - Many-to-many

# Resources Used

1. SmartDraw. Entity Relationship Diagram. (n.d.). Retrieved from <https://www.smartdraw.com/entity-relationship-diagram/#whatIsERD>
2. Lucidchart. Template: Entity Relationship Diagram. (n.d.). Retrieved from <https://lucidchart.zendesk.com/hc/en-us/articles/360000478183-Template-Entity-Relationship-Diagram>
3. Creately 2008. R Diagram for Hospital Management System ( Entity Relationship Diagram). (n.d.). Retrieved March 22, 2020, from [https://creately.com/diagram/example/hwj0b7oy1/E-R Diagram for Hospital Management System](https://creately.com/diagram/example/hwj0b7oy1/E-R%20Diagram%20for%20Hospital%20Management%20System)