April 11, 2025

Keith Jhaeron Cayatoc

B.I.T Student

Nelson Marlborough Institute of Technology

DAT601 – Assessment 1

Tutor: Todd Cochrane

Table of Contents

[Assignment 1 Part 1: Getting to know TSQL. SQL Queries 2](#_Toc195144776)

[Assignment 1 Part 2: Conceptual Database Design 3](#_Toc195144777)

[0. Introduction to data modelling 3](#_Toc195144778)

[1. Description of Extended Chen ERD Notation 4](#_Toc195144779)

[2. Conceptual ERD of ParkWorks 5](#_Toc195144780)

[3. Conceptual Data Dictionary 7](#_Toc195144781)

[4. Business Rules and Assumptions 1](#_Toc195144782)

[References 3](#_Toc195144783)

# Assignment 1 Part 1: Getting to know TSQL. SQL Queries

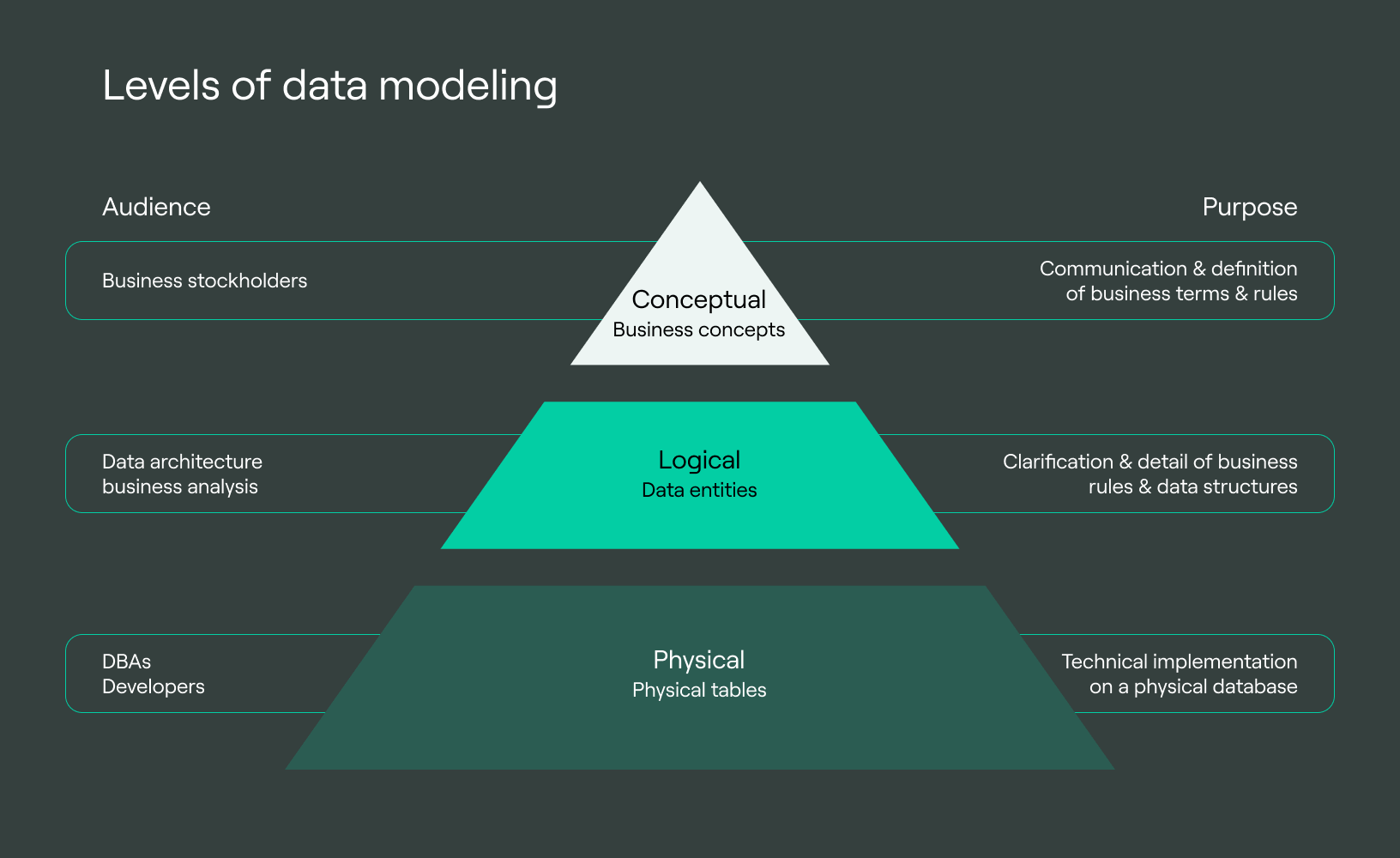
Link to the GitHub Repository for the SQL scripts:

<https://github.com/Yugenzariah/DAT601-Assessment-One.git>

# Assignment 1 Part 2: Conceptual Database Design

## Introduction to data modelling

Data modelling is the process of visually representing a system’s data to show how it is structured, connected and managed. It plays a critical role in designing or improving databases by making sure that data is organized in a way that supports accuracy, consistency, and relevance to business needs. There are three key types of data models, the conceptual model which provides a high-level overview of data entities and relationships, the logical model which introduces more detailed structures and attributes and the physical model, which defines how data is stored and accessed within a specific database system. These models not only support effective data management but also allow teams to identify potential design issues early, whether they are building a new system or documenting existing ones (Gillis, Stedman, & Vaughan, n.d.).



(DoubleCloud, 2023)

## Description of Extended Chen ERD Notation

The extended Chen entity-relationship diagram (ERD) notation is a visual modelling technique used to represent data and its relationships in a structured and more detailed way. It builds on the original Chen notation by introducing advanced components like weak entities, multivalued attributes, and generalisations/specialisations making it ideal for enterprise-level systems such as ParkWorks.

##### Main Components of Extended Chen ERD

**Entities** are represented as rectangles and define real-world objects like FRED, Subscriber, or Contract. Each entity typically includes a primary key which uniquely identifies each record.

**Attributes** are shown as ovals and describe properties of entities.

* Simple attributes hold one value for example just the Name attribute in a person entity.
* Composite attributes are made up of sub-parts for example a Full Name attribute has First name and Last Name.
* Multivalued attributes can hold multiple values for example a FRED may be able to perform multiple services sown with double ovals.

**Relationships** appear as diamonds and connect two or more entities.

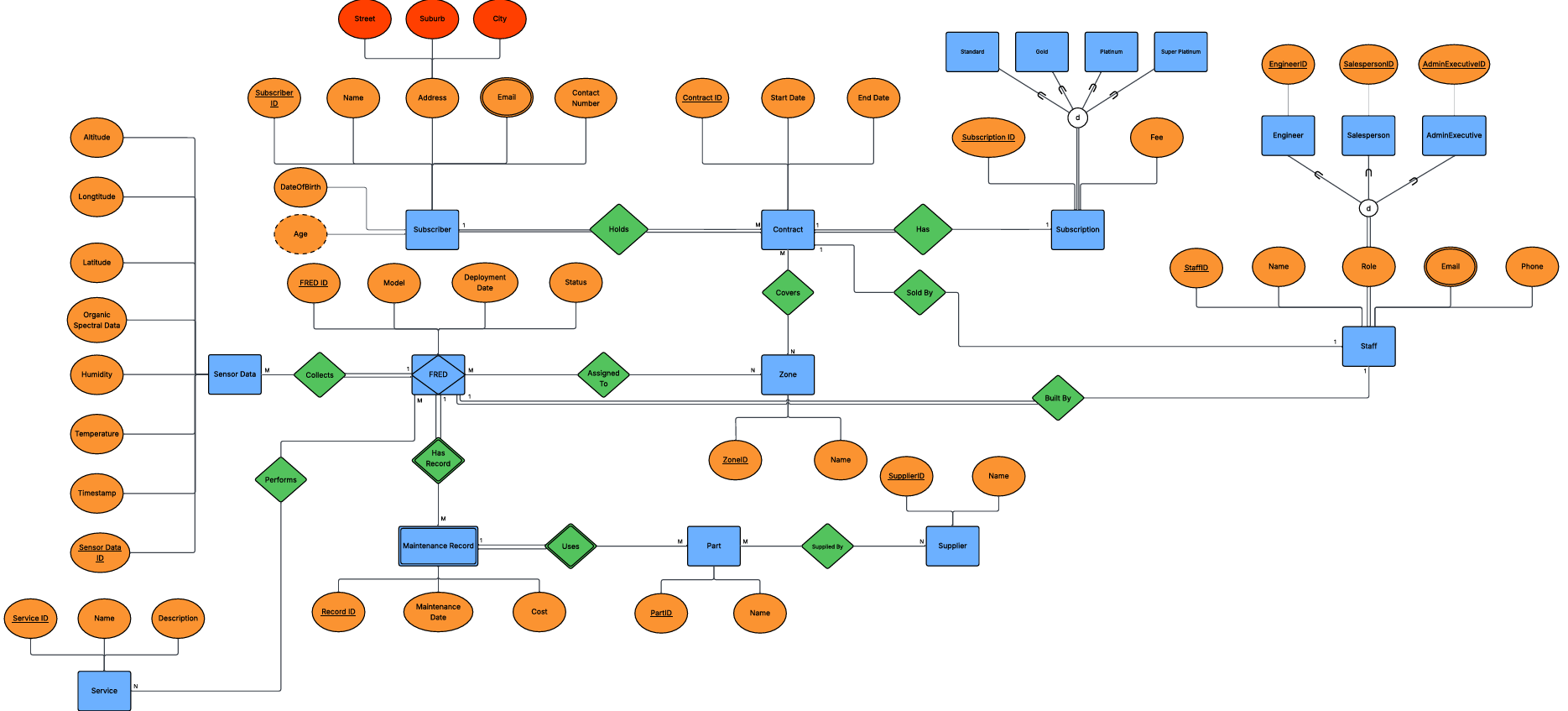
* Cardinality represented by “1”, “N” or “M” and participation that can either be a total or partial participation are used to specify how entities interact with each other.
* Weak entities depend on other entities for identification. For example, in the context of the project brief maintenance record is a weak entity dependent on both FRED and Part, since it cannot exist without referencing both entities.
* Generalisations/Specialisation also known as superclass and subclass are shown using a triangle to represent inheritance. In the ParkWorks project brief, the subscription entity can be specialised into Standard, Gold, Platinum, and Super Platinum each with its own constraints.

ParkWorks Context:

The FRED entity has attributes like FRED id, deployment date, and status. The sensor data entity has multivalued attributes like temperature, humidity, and organic spectral data collected every second. The subscription entity which is considered as the “Superclass” is specialised into types like standard subscription and platinum subscription. The maintenance record is a weak entity because it is connected to both FRED and Part entities.

Extended Chen ERD notation supports enterprise data management by promoting clarity, consistency, and flexibility in system design. By visually structuring entities, relationships, and advanced modelling features such as weak entities and specialisation, it helps enforce data integrity through primary and foreign keys. This ensures that the data remains valid across complex interactions. The use of specialisation and generalisation also allows organisations to implement role-based control, supporting secure access to data based on user responsibilities. Furthermore, clearly separating entities and applying normalisation principles reduces redundancy and improves accuracy. The modular structure of Extended Chen ERDs makes systems easier to scale, allowing enterprises to adapt to new requirements such as adding new services, user roles, or integrations without needing to redesign the entire database (Gillis, Stedman, & Vaughan, n.d.).

## Conceptual ERD of ParkWorks



I chose to use Extended Chen ERD notation because it allowed me to model the complexity of the ParkWorks system more accurately. Features like weak entities, multivalued and composite attributes and specialisation helped me represent things like sensor data collected by FREDs, maintenance records linked to both FREDs and parts, as well as staff roles broken down into Engineer, Salesperson and Admin Executive.

My design choices were based closely on the project brief. For example, I used generalisation to show different subscription types like Standard, Gold, Platinum, Super Platinum and made sure each staff member had a role using disjoint total participation. Relationships were built with correct cardinality and participation, and I used mandatory/optional lines to show which connections were required.

Using EERD helped me apply data management principles like data integrity with primary and foreign keys, accuracy by reducing redundancy, and scalability so that new services or roles can be added easily. I also made sure the model avoided fan and chasm traps by keeping relationships clear and direct.

This task helped me better understand how to turn business rules into a clear data structure. My EERD may not be perfect, but I learned a lot about using ERD features properly and feel more confident designing databases in the future with the help of my tutors feedback and advise how to improve my diagrams.

## Conceptual Data Dictionary

**Table 1: Document Entities**

|  |  |  |  |
| --- | --- | --- | --- |
| **Entity Name** | **Description** | **Aliases** | **Occurrence** |
| Subscriber | Organisation or individual holding contracts |  | One per subscribing organisation |
| Contract | Defines the subscription agreement between ParkWorks and a subscriber |  | Many per subscriber |
| Subscription | Defines the type of subscription |  | One per contract |
| Zone | A defined geographic area serviced by FREDs |  | Many per contract or FRED |
| FRED | Autonomous park maintenance robot |  | Many per zone |
| Sensor Data | Environmental data collected by FREDs |  | Many per FRED |
| Maintenance Record | Maintenance logs for FRED parts |  | Many per FRED-Part combination |
| Part | Component used by FREDs |  | Many used across FREDs |
| Supplier | Supplies parts to ParkWorks |  | Many per part |
| Service | Types of work performed by FREDs |  | Many per FRED |
| Staff | Employee involved in building, selling, or managing services |  | Many per organisation |

**Table 2: Document Relationships/Specializations- Generalizations**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Entity Name** | **Cardinality** | **Participation** | **Relationship** | **Participation** | **Cardinality** | **Entity Name** |
| Subscriber | 1 | Mandatory | Holds | Mandatory | N | Contract |
| Contract | 1 | Mandatory | Has | Mandatory | 1 | Subscription |
| FRED | M | Optional | Assigned To | Optional | M | Zone |
| Contract | M | Optional | Covers | Optional | M | Zone |
| FRED | 1 | Mandatory | Collects | Mandatory | N | Sensor Data |
| FRED | 1 | Mandatory | Maintained With | Mandatory | N | Maintenance Record |
| Maintenance Record | N | Mandatory | Uses | Mandatory | 1 | Part |
| Part | M | Optional | Supplied By | Optional | M | Supplier |
| FRED | M | Optional | Performs | Optional | M | Service |
| Contract | 1 | Optional | Sold By | Optional | 1 | Staff |
| FRED | 1 | Mandatory | Built By | Mandatory | 1 | Staff |
| Staff | 1 | Mandatory | Is A | Specialisation | N | Engineer/Sales/Admin Executive |
| Subscription | 1 | Mandatory | Is A | Specialisation | N | Standard/Gold/Platinum/Super |

**Table 3: Document Attributes**

| **Entity Name** | **Attributes** | **Description** | **Domain** | **Aliases** | **Composite** | **Derived** | **Nulls** | **Primary Key?** | **Default Value** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Subscriber | SubscriberID | Unique identifier | Integer |  | No | No | No | Yes |  |
| Subscriber | Name | Subscriber name | Varchar |  | No | No | No | No |  |
| Subscriber | Email | Contact email | Varchar |  | No | No | No | No |  |
| Subscriber | Address | Physical address | Text |  | Yes | No | No | No |  |
| Subscriber | DateOfBirth | Subscribers birth date | Date Time |  | No | No | No | No |  |
| Subscriber | Age | Age of subscriber | Int |  | No | Yes | No | No |  |
| Contract | Contract ID | Unique contract ID | Integer |  | No | No | No | Yes |  |
| Contract | StartDate | Start of service | Date |  | No | No | No | No |  |
| Contract | End Date | End of service | Date |  | No | No | Yes | No |  |
| FRED | FRED ID | Robot ID | Integer |  | No | No | No | Yes |  |
| FRED | Model | FRED model | Varchar |  | No | No | No | No |  |
| SensorData | Sensor Data ID | Unique data ID | Integer |  | No | No | No | Yes |  |
| SensorData | Timestamp | Time collected | Datetime |  | No | No | No | No |  |
| SensorData | Temperature | Temperature value | Decimal |  | No | No | Yes | No |  |
| SensorData | Humidity | Humidity level | Decimal |  | No | No | Yes | No |  |
| SensorData | Ambient Light | Light level | Decimal |  | No | No | Yes | No |  |
| SensorData | Organic Spectral Data | Spectral data | Text |  | No | No | Yes | No |  |
| SensorData | Latitude | Geo location | Decimal |  | No | No | No | No |  |
| SensorData | Longitude | Geo location | Decimal |  | No | No | No | No |  |
| SensorData | Altitude | Altitude reading | Decimal |  | No | No | No | No |  |
| MaintenanceRecord | Record ID | Maintenance record ID | Integer |  | No | No | No | Yes |  |
| MaintenanceRecord | Maintenance Date | Date of maintenance | Date |  | No | No | No | No |  |
| MaintenanceRecord | Cost | Maintenance cost | Decimal |  | No | No | Yes | No |  |
| Part | Part ID | Unique part ID | Integer |  | No | No | No | Yes |  |
| Part | Name | Part name | Varchar |  | No | No | No | No |  |
| Supplier | Supplier ID | Supplier ID | Integer |  | No | No | No | Yes |  |
| Supplier | Name | Supplier name | Varchar |  | No | No | No | No |  |
| Service | Service ID | Unique service ID | Integer |  | No | No | No | Yes |  |
| Service | Service Name | Name of service | Varchar |  | No | No | No | No |  |
| Service | Description | What the service does | Text |  | No | No | Yes | No |  |
| Staff | Staff ID | Employee ID | Integer |  | No | No | No | Yes |  |
| Staff | Name | Employee name | Varchar |  | No | No | No | No |  |
| Staff | Role | Employee role | Varchar |  | No | No | No | No |  |
| Staff | Email | Work email | Varchar |  | No | No | No | No |  |
| Staff | Phone | Work contact number | Varchar |  | No | No | Yes | No |  |
| Zone | Zone ID | Unique zone ID | Integer |  | No | No | No | Yes |  |
| Zone | Name | Name of the zone | Varchar |  | No | No | No | No |  |
| Subscription | Subscription ID | Unique subscription ID | Integer |  | No | No | No | Yes |  |
| Subscription | Type | Type of subscription | Varchar |  | No | No | No | No |  |
| Subscription | Fee | Monthly subscription fee | Decimal |  | No | No | No | No |  |

Creating the data dictionary helped me apply important data management principles by presenting accurate and consistent data. By clearly defining each attributes type whether it’s a primary key and whether it allows nulls I was able to make sure the data structure supports valid and meaningful data. For example, marking primary keys and preventing nulls in important fields help maintain data reliability. Listing all entities and relationships also helped me avoid redundancy and made the model easier to understand. I also found that including things like composite attributes example: Address helped show how data can be organised clearly. Overall, the data dictionary made my design more structured, reusable, and ready for implementation in a real system.

## Business Rules and Assumptions

**Business Rules**

1. Each Subscriber must hold at least one Contract.

* This is modelled with a mandatory 1:N relationship from Subscriber to Contract to reflect business rules and ensure data integrity (Gillis, Stedman, & Vaughan, n.d.).

1. Each Contract must include one Subscription type.

* A 1:1 mandatory relationship between Contract and Subscription ensures a single, valid subscription is linked to each contract (Coronel & Morris, 2018).

1. A Contract may cover multiple Zones, and each Zone can be covered by many Contracts.

* This M:N optional relationship allows for flexible coverage mapping of FREDs across park areas.

1. Each FRED is assigned to one or more Zones, and Zones may have multiple FREDs.

* This is also modelled as an M:N optional relationship to support enterprise system scalability (Gillis et al., n.d.)

1. Every FRED collects Sensor Data regularly.

* A 1:N mandatory relationship between FRED and Sensor Data ensures completeness of system monitoring and supports accurate environmental tracking (Elmasri & Navathe, 2017)

1. FREDs have maintenance records that depend on both the FRED and the Part used.

* This is captured using a weak entity structure with identifying relationships, which improves data consistency and models real-world dependency (Elmasri & Navathe, 2017)

1. Parts are supplied by one or more Suppliers.

* An M:N optional relationship allows flexibility in sourcing and supplier management.

1. FREDs perform various services depending on configuration.

* This is represented using an M:N optional relationship to allow for future system expansion and modular functionality.

1. Only Salespeople can sell Contracts, and only Engineers can build FREDs.

* Disjoint total specialisation is used to enforce this role-based control by restricting relationships to certain staff types (Elmasri & Navathe, 2017).

1. Each Subscription belongs to one of four levels: Standard, Gold, Platinum, or Super Platinum.

* This is modelled using a generalisation hierarchy to reduce redundancy and provide clear type enforcement (Coronel & Morris, 2018).

**Assumptions**

1. A Subscriber is linked to a Subscription only through their Contract.
2. Each Contract and FRED is handled by one staff member, determined by their role.
3. Zones are created independently of whether they are assigned to a Contract or a FRED.
4. Sensor Data is grouped by Sensor Data ID and contains multivalued attributes such as temperature, humidity, and spectral data.
5. Maintenance Records must have a valid Part and FRED, reflecting dependency and referential integrity.
6. Subscription fees are fixed by Subscription Type, not by Contract.
7. Staff roles are implemented using the Role attribute in the Staff entity, with specialisation handled conceptually through a disjoint total hierarchy (Gillis et al., n.d.).
8. The list of services a FRED can perform is not stored as attributes, but as instance-level data for flexibility.

# References

Gillis, A. S., Stedman, C., & Vaughan, J. (n.d.). *Data modeling*. TechTarget. Retrieved March 30, 2025, from <https://www.techtarget.com/searchdatamanagement/definition/data-modeling>

DoubleCloud. (2023, August). *Levels of data modeling* [Image]. DoubleCloud Blog. <https://double.cloud/blog/posts/2023/08/what-is-data-modeling/>

Coronel, C., & Morris, S. (2018). *Database systems: Design, implementation, and management* (13th ed.). Cengage Learning.

Elmasri, R., & Navathe, S. B. (2017). *Fundamentals of database systems* (7th ed.). Pearson.