**Purpose:**

Review the importance of conceptual database design in the overall database development process.

**Tasks:**

1. What are the pros and cons of conceptual database design using Entity Relationship Diagrams? Please give one pro and con.
2. Please, respond to the initial post and respond to at least two of your classmates. The initial discussion question (DQ) response message is due by 11:59 PM on Thursday, and at least two (2) responses to other peers are due by 11:59 PM EST on Sunday.

Pro: Flexible Design

Conceptual database design using Entity Relationship Diagrams is advantageous because of its flexible and agile approach when designing the schema. By breaking down the business problem into the different entities, relations, attributes, and constraints (as mentioned in the module), you represent the data at a higher level. This means that as the requirements change and evolve during the lifespan of the project design, changes can be made to the data model without having to worry about specific implementation details. The module gives a great example of this with the addition of the requirement that all managers work in a building. We were able to represent this change as a modification to the diagram without having to worry about how to actually implement the change. This means that changes do not require significant overall changes to the schema design.

Con: Potential for Ambiguous Interpretation

A con of using Entity Relationship Diagrams for conceptual database design is they may be interpreted differently by different people. For example, two designers may have been taught using different notations. If they are designing a system together, but have not come to an agreement on notation, it is possible that the diagram is inconsistent when noting things such as constraints or cardinality. And speaking of cardinality - even if the notation is correct and consistent - if there is not proper or extensive documentation on the diagram itself, a non-technical person trying to understand data relationships may misunderstand if a relationship should be one-to-many or many-to-one. Or for constraints, it is possible that a person confuses a required entity in a relationship for optional.