# Module 1.3 Assignment

## In the context of relational databases, what are relationships? Describe at least two, and provide an example of their use.

A relationship describes how data in two tables relate to each other. Let’s say we have a table that stores employee information. This employee would have the following information in the table (keeping this very simple):

Employee ID – PRIMARY KEY

Employee Name

Job Title

Let’s look at the employee’s job title and refer to them as a Shift Leader. This is fine, but if we have 10 shift leaders and, at some point, we decide to change the name of that job title (because the company wants to call it a Guest Service Leader instead), you will now have to change that Title in 10 locations, and you might miss one. But, if we store a Job Code and link it as a FOREIGN KEY to a different database, where that Job Code is connected to the Job Title, all I have to do is change the Job Title in that database, and it changes for all employees with that Job Code. So we have two “relational databases,” one that stores employee information, and one that stores the job titles. The employee database stores a job code, and when the data is retrieved, the program can look up the corresponding job title using the job code stored in the database.

Use Cases:

One use case is as listed above, where we keep a database of Job Titles, Departments, or any other information that MIGHT be changed in the future, and substitute a code or ID in for the title so we can keep that information separate, and use a database to store what the job code references.

Another use case would be linking a customer database to the orders they place. A customer database may contain extensive information about a customer, including their orders placed. Another database that tracks orders sold would reference the customer ID, linking the order to that customer’s account. In this example, a customer might have MANY orders, and each order is linked to that customer using their customer ID in the order. So, if the customer wants to view past orders, they can query the database for any order associated with their customer ID and retrieve the pertinent information.

## What are the advantages of relational databases? What are the advantages of NoSQL databases?

#### Advantages of Relational Databases:

* **Data Integrity and Accuracy:** As mentioned above, if a Job Title changes, you update the Job Title in the database and in every occurrence of reference to the Job code, ensuring all information is consistent and accurate.
* **Data Normalization:** Using relationship databases can minimize redundancy in data (you don’t need to store order information in the customer database or customer information in the orders database when they can reference each other where necessary).

#### Advantages of NoSQL:

* **Document-based:** Data tends to be more flexible, semi-structured formats like JSON.
* **Flexible**: NoSQL Databases are highly adaptable, allowing developers to store and retrieve data dynamically and support multiple data types and changing data structure.
* **Performance:** For large datasets and real-time analytics, performance is superior to that of relational databases.

## What are the disadvantages of relational databases? What are the disadvantages of NoSQL databases?

#### Disadvantages to relational databases:

* **More Complex:** As the title says, relational databases tend to be far more complex than non-relational ones.
* **Performance and Scalability Bottlenecks:** Because the databases reference each other, sometimes, as the amount of data increases, it can create bottlenecks when there are large amounts of data or many users trying to access the data simultaneously, making non-relational databases faster.
* **Cost**: Relational databases are more expensive because their design is more complex and typically involves multiple databases (as separate databases are required on the server for items such as Job Codes, Department Codes, etc.).

#### Disadvantages of NoSQL Databases:

* **Narrow Focus:** Great for data storage, but lacks functionalities as transaction management.
* **No Complex Query Support:** They are not a good fit for applications that require complex data analysis or reporting.
* **Management complexity:** For large datasets,maintenancecan be difficult.

## Identify at least two features of MySQL and two features of MongoDB, and describe what they are and how they are used.

#### MySQL:

1. **ACID Compliance:** MySQL supports **ACID (Atomicity, Consistency, Isolation, Durability)** properties, ensuring reliable transactions and data integrity. This means that operations such as inventory updates or payment processing occur completely or not at all, which is ideal for systems where accuracy is critical.
2. **Structured Query Language (SQL):** MySQL uses **SQL**, a standardized language for defining, manipulating, and querying structured data. SQL enables complex joins, filtering, sorting, and aggregations, making MySQL highly effective for analytical reporting and applications that require well-defined relationships between tables.

#### MongoDB

1. **Document-Oriented Storage:** MongoDB stores data in **BSON (Binary JSON)** documents rather than rows and tables. This allows flexible, hierarchical structures, such as embedding arrays or objects, making it ideal for modern applications that need to handle semi-structured or rapidly evolving data.
2. **Horizontal Scalability with Sharding:** MongoDB supports **sharding**, which distributes large datasets across multiple servers, enabling horizontal scalability. This enables high availability and performance by balancing the load and allowing the database to scale horizontally as the data volume grows, a feature commonly used in large-scale web and analytics applications.

## References:

Gaddis, T. (2024). Chapter 14: Database programming. In Starting out with Python (6th ed.). Pearson.

GeeksforGeeks. (2023, October 16). Introduction to NoSQL. GeeksforGeeks. <https://www.geeksforgeeks.org/dbms/introduction-to-nosql/>

GeeksforGeeks. (2023, October 17). RDBMS benefits and limitations. GeeksforGeeks. [https://www.geeksforgeeks.org/dbms/rdbms-benefits-and-limitations/](https://www.geeksforgeeks.org/dbms/rdbms-benefits-and-limitations/?utm_source=chatgpt.com)

MongoDB, Inc. (2024). Resources. MongoDB. <https://www.mongodb.com/resources>

Oracle. (2024). MySQL documentation and resources. MySQL. <https://dev.mysql.com/>