

# 40 Lecture - CS403

## Important Subjective

1. **What is a view in a database, and how is it different from a table?**

Answer: A view is a virtual table in a database that displays a subset of data from one or more tables. It differs from a table in that it does not actually store any data, but rather is created by defining a SELECT statement as a table.

**How can a view be used to simplify complex queries in a database?**

Answer: Views can be used to simplify complex queries by allowing users to define a SELECT statement that retrieves only the data they need, rather than having to write a more complex query that retrieves data from multiple tables.

**What is the purpose of creating a view in a database, and how can it improve security?**

Answer: The purpose of creating a view in a database is to display a subset of data from one or more tables. Views can improve security by allowing users to access only the data they are authorized to see.

**Can a view be used to enforce data constraints in a database?**

Answer: No, views cannot be used to enforce data constraints in a database.

**How can a view be modified in a database?**

Answer: A view can be modified in a database using the ALTER VIEW statement.

**Can a view be used in a JOIN operation in a database?**

Answer: Yes, views can be used in JOIN operations in a database.

**What is the difference between a simple view and a complex view in a database?**

Answer: A simple view is a view that is created from a single table, while a complex view is created from multiple tables.

**How can a view improve performance in a database?**

Answer: Views can improve performance by reducing the amount of data that needs to be retrieved from the database, and by simplifying complex queries.

**Can a view be dropped in a database, and if so, how?**

Answer: Yes, a view can be dropped in a database using the DROP VIEW statement.

**How does the use of views affect database design and development?**

Answer: Views can impact database design and development by allowing for a more modular approach, where complex queries are encapsulated in views that can be easily reused and modified.