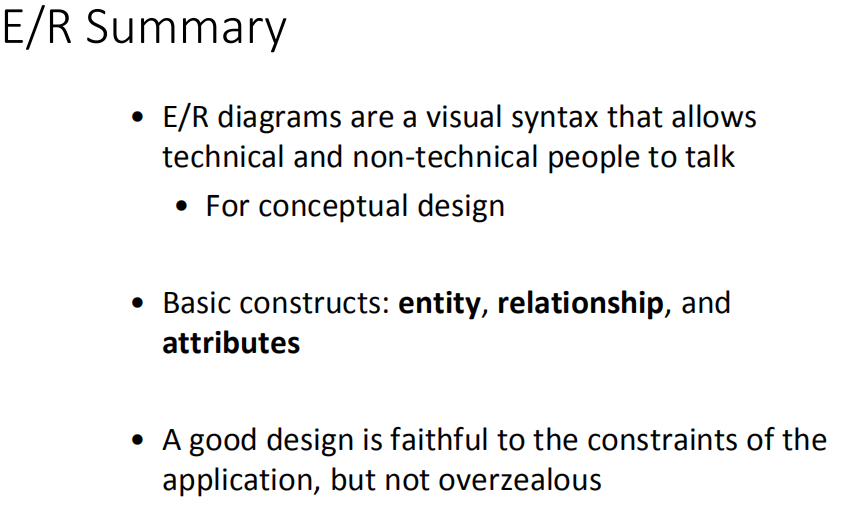
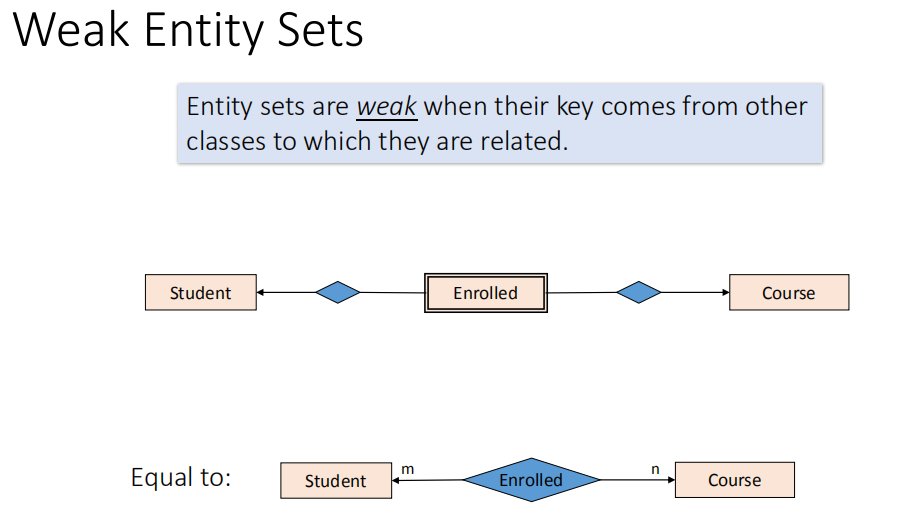
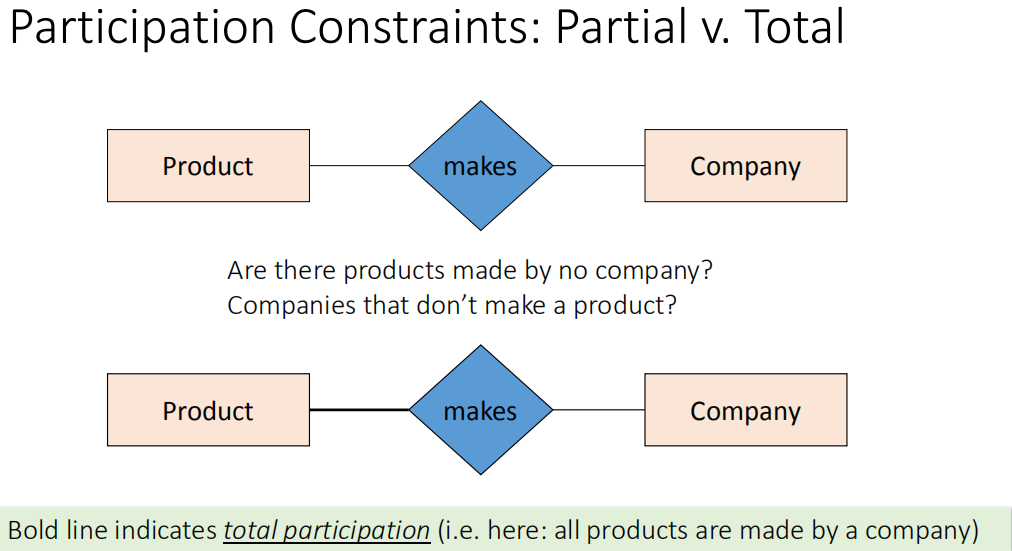
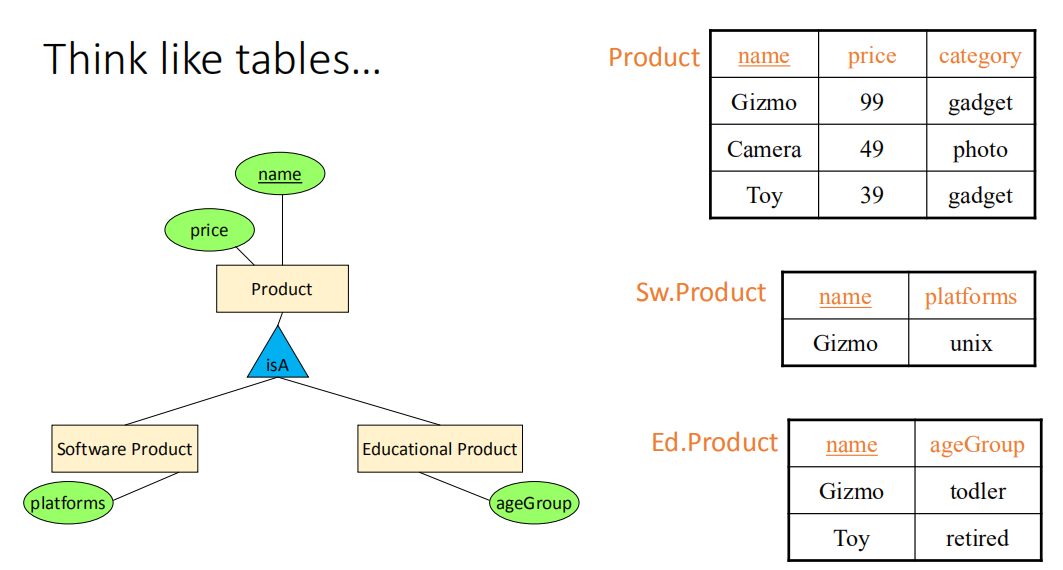
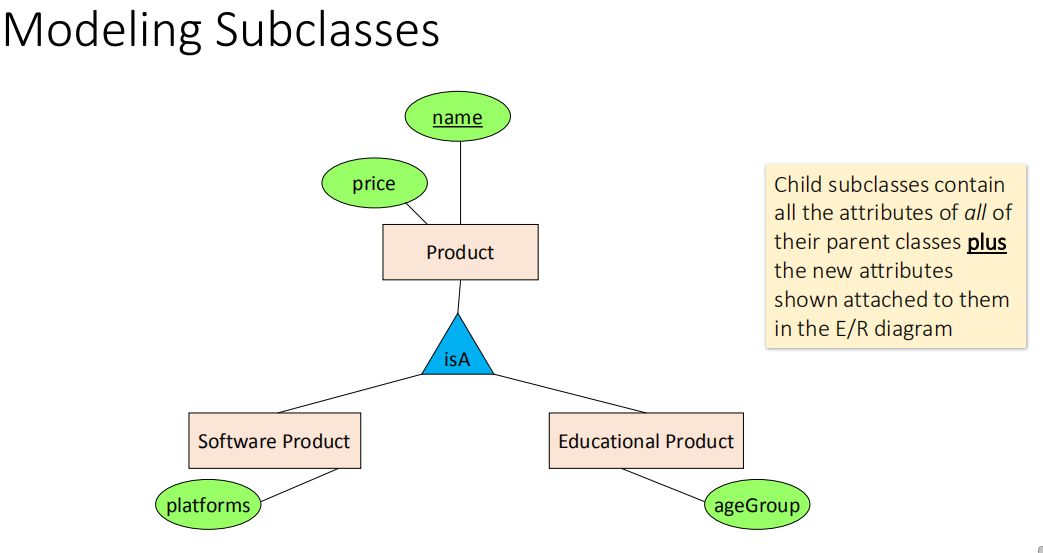
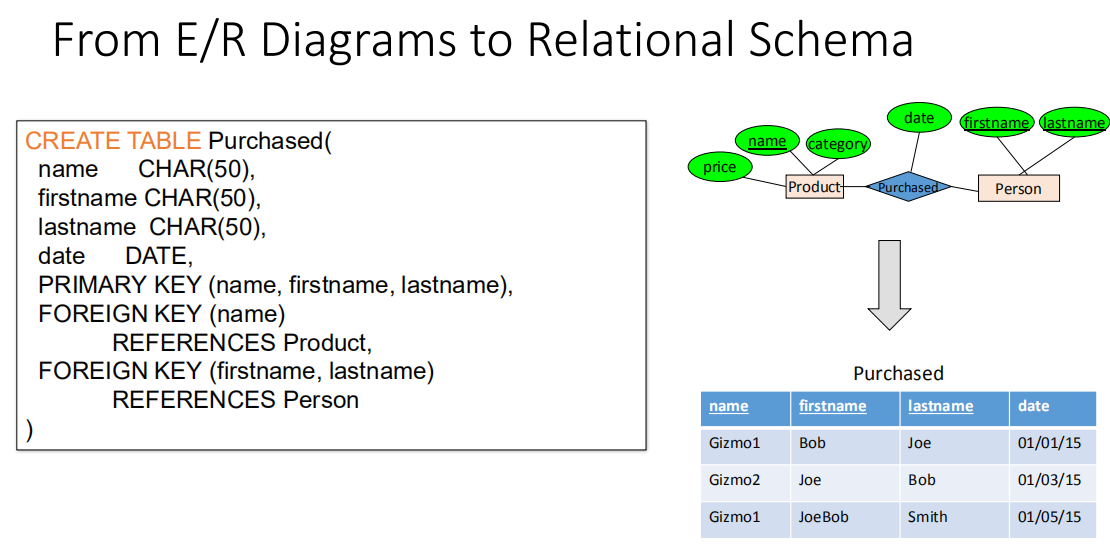
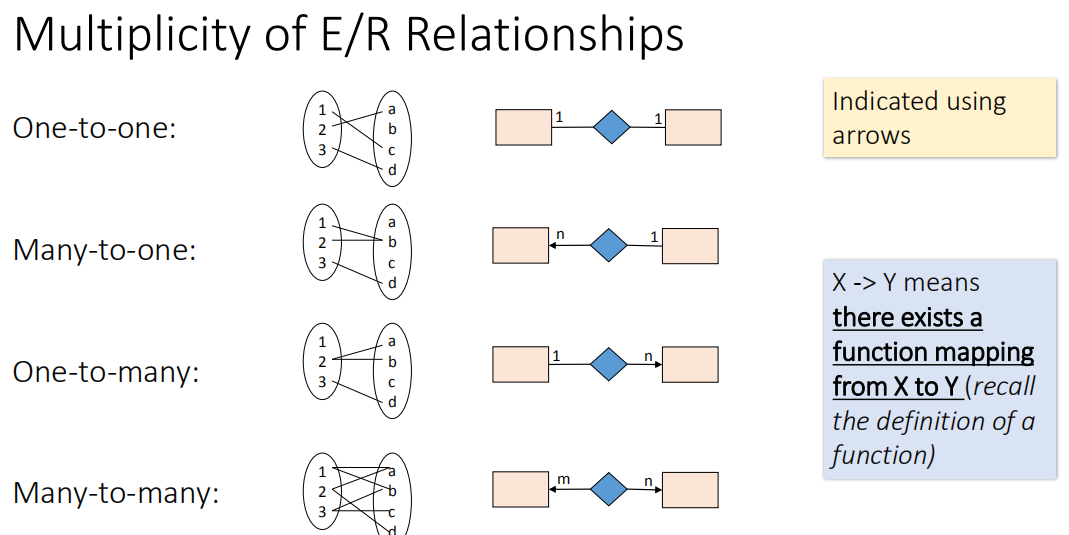
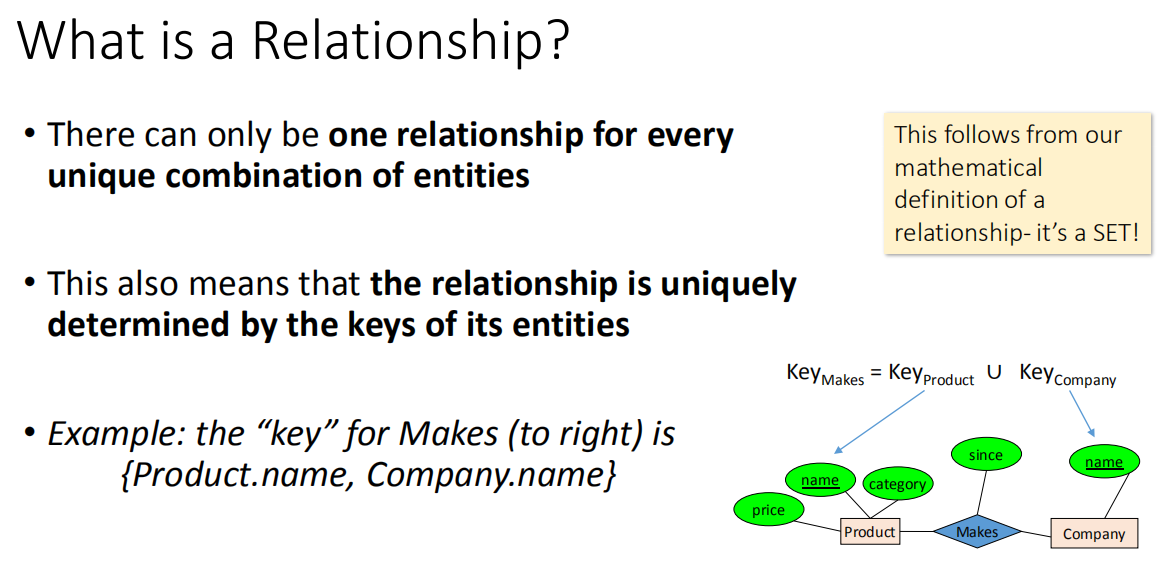
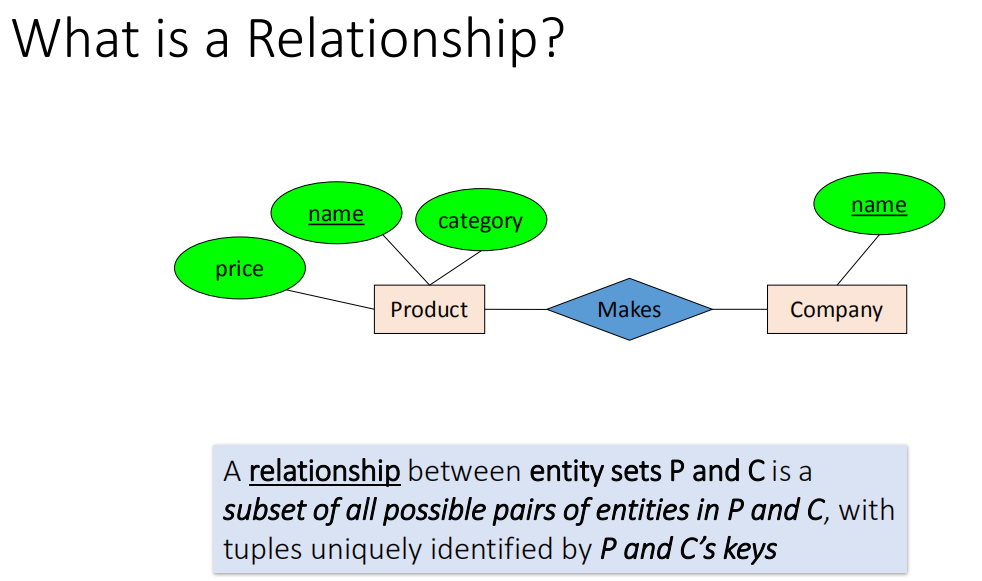
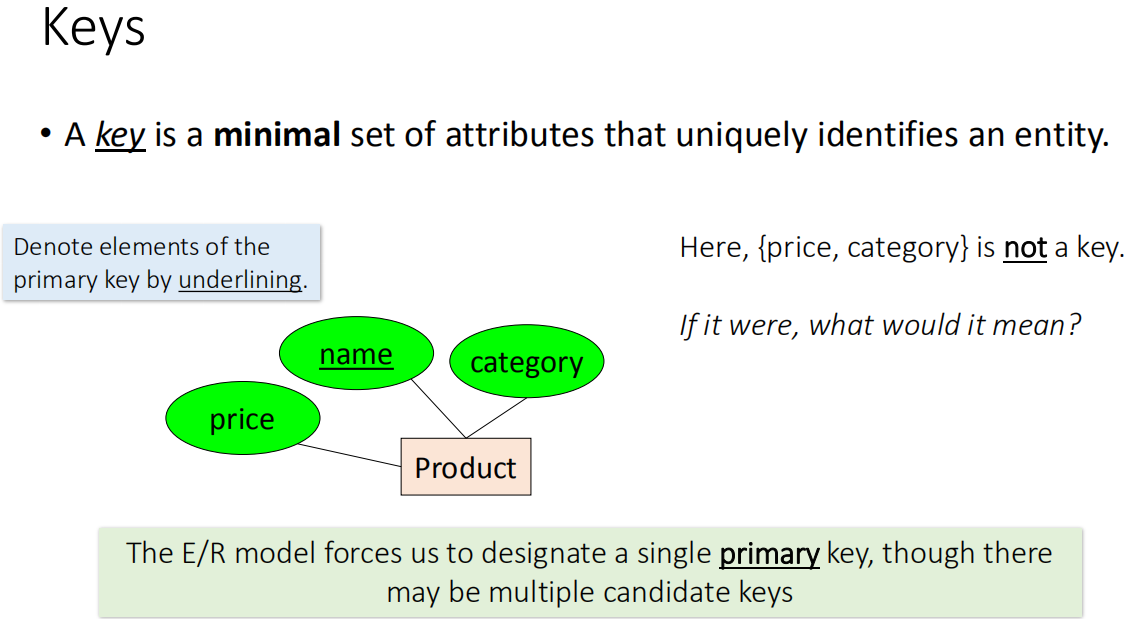
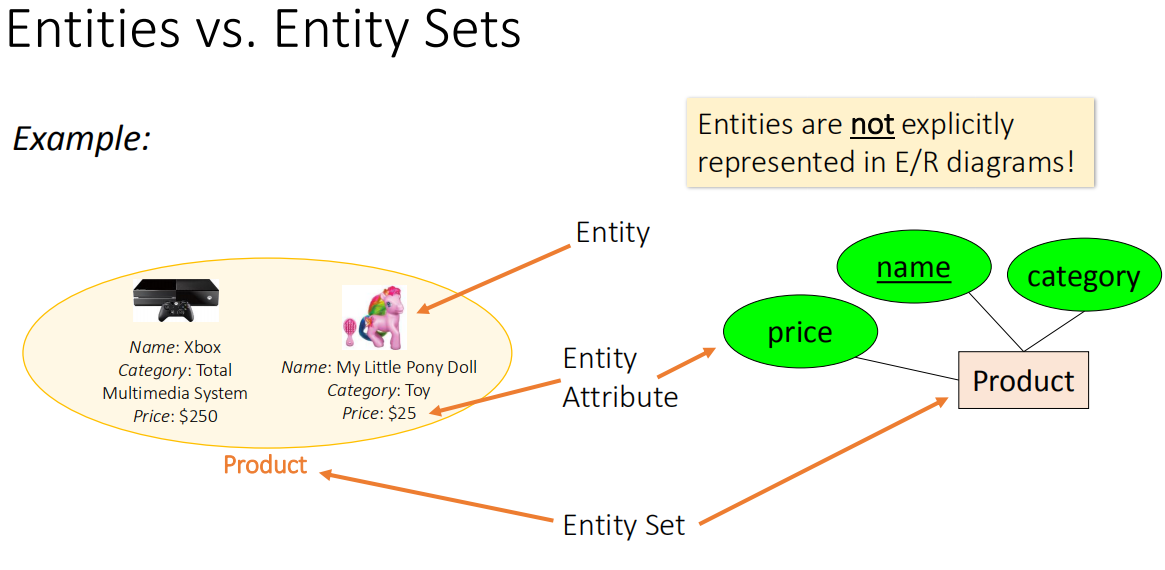
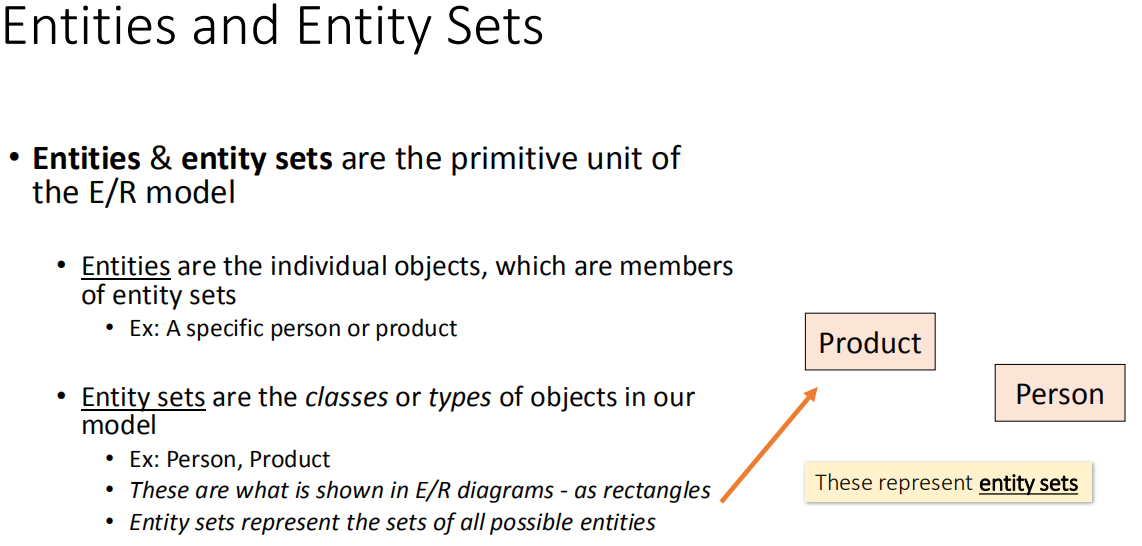
: BBM471 - DATABASE MANAGEMENT SYSTEMS :

**Ders02 - ER Model:**



**Der03 - SQL:**

* **SQL is a...**
* Data Definition Language (DDL)
* Define relational schemata
* Create/alter/delete tables and their attributes
* Data Manipulation Language (DML)
* Insert/delete/modify tuples in tables
* Query one or more tables – discussed next!
* **Data Types in SQL**

Atomic types:

* Characters: CHAR(20), VARCHAR(50)
* Numbers: INT, BIGINT, SMALLINT, FLOAT
* Others: MONEY, DATETIME, …

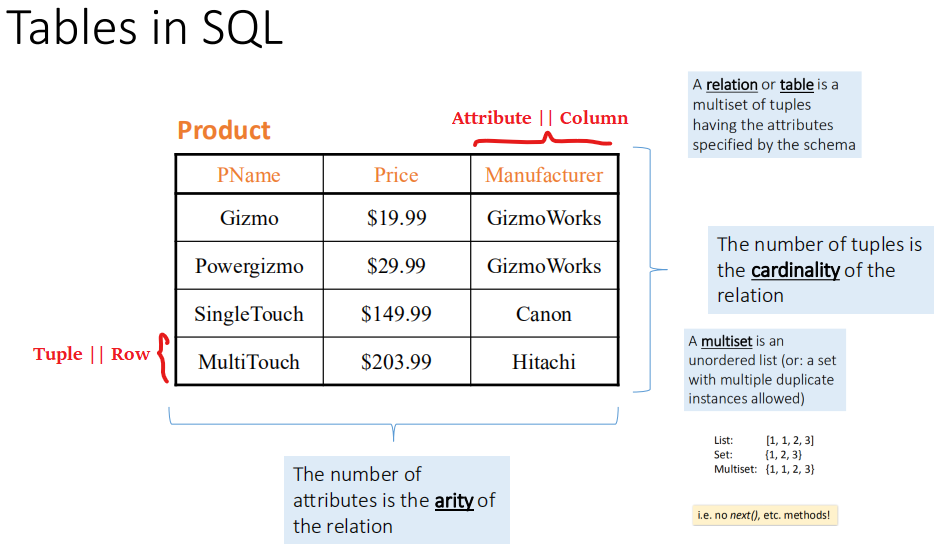
Every attribute must have an atomic type. Hence tables are flat.

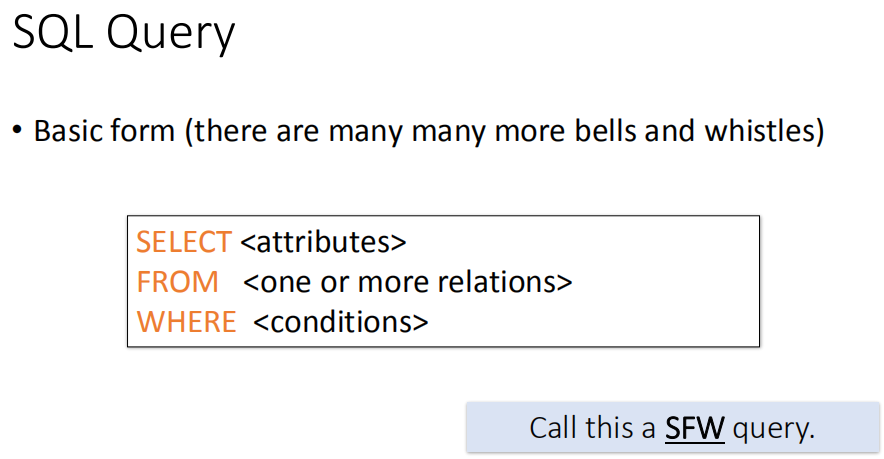
* **Table Schemas**
* The schema of a table is the table name, its attributes, and their types:

Product(Pname: string, Price: float, Category: string, Manufacturer: string)

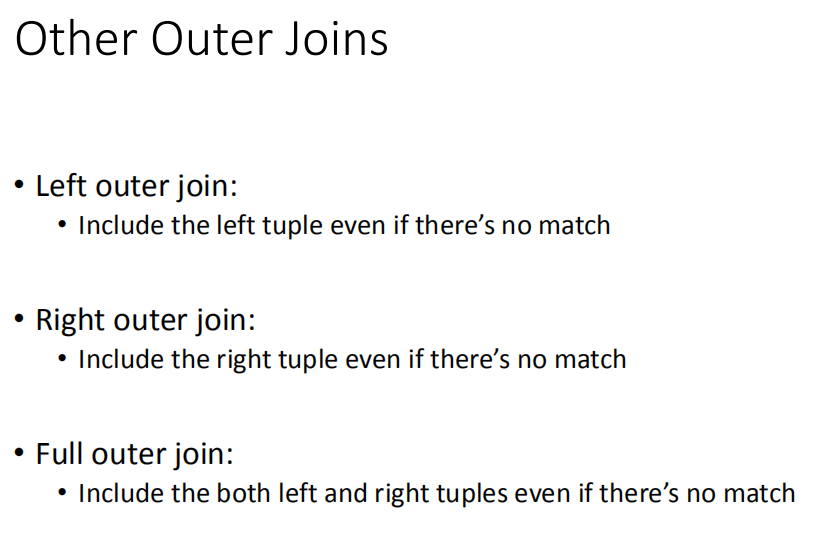
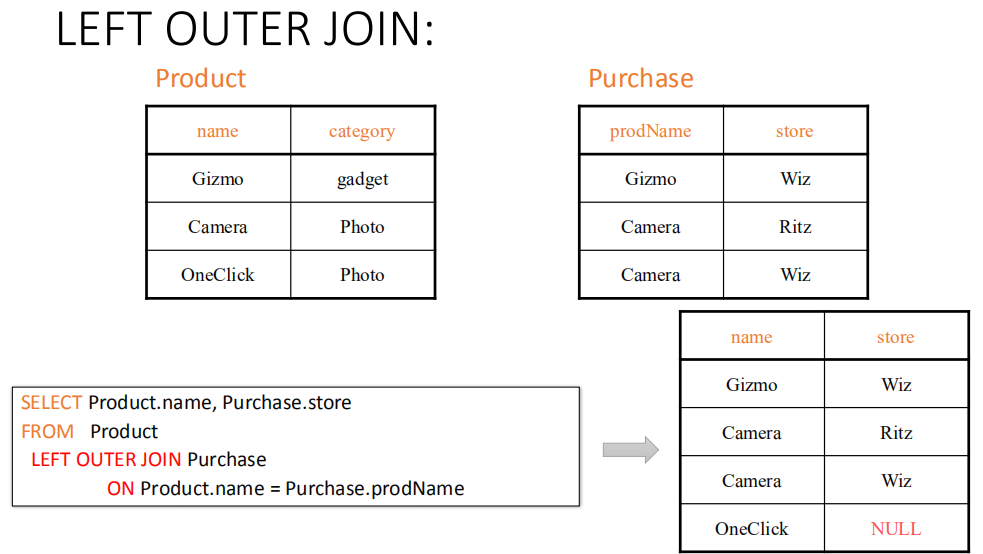
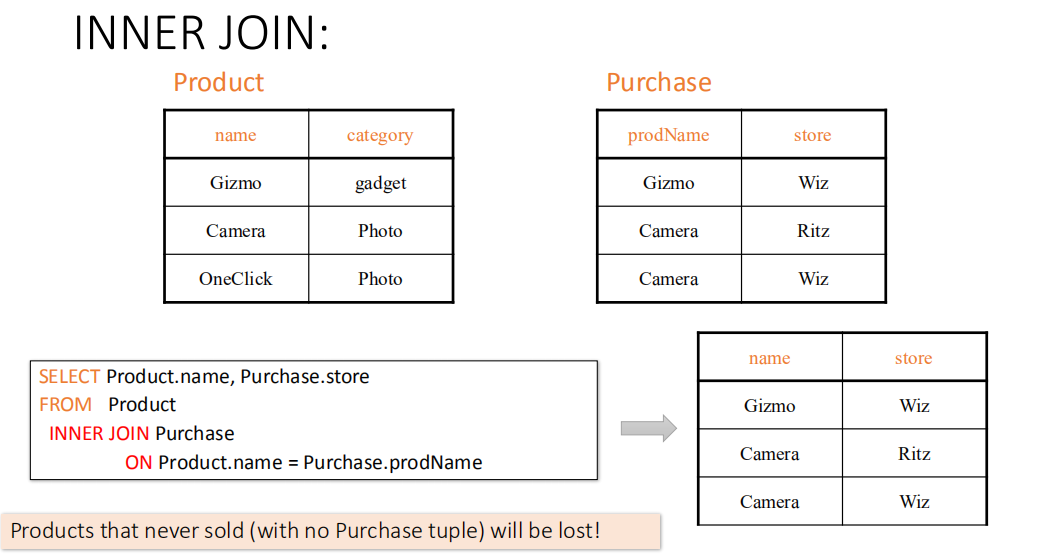
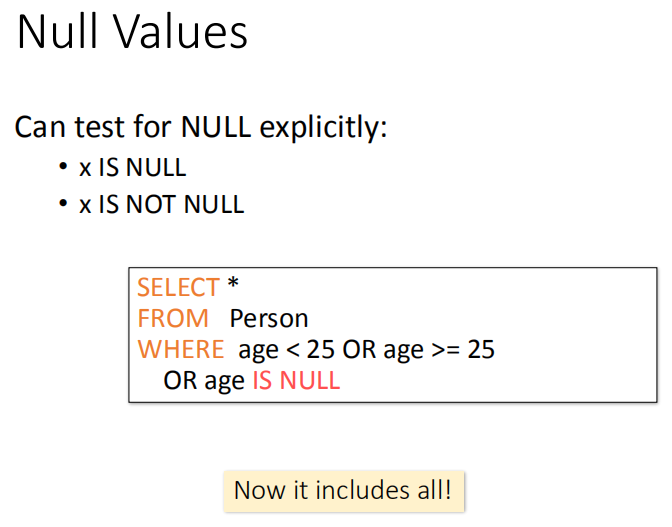
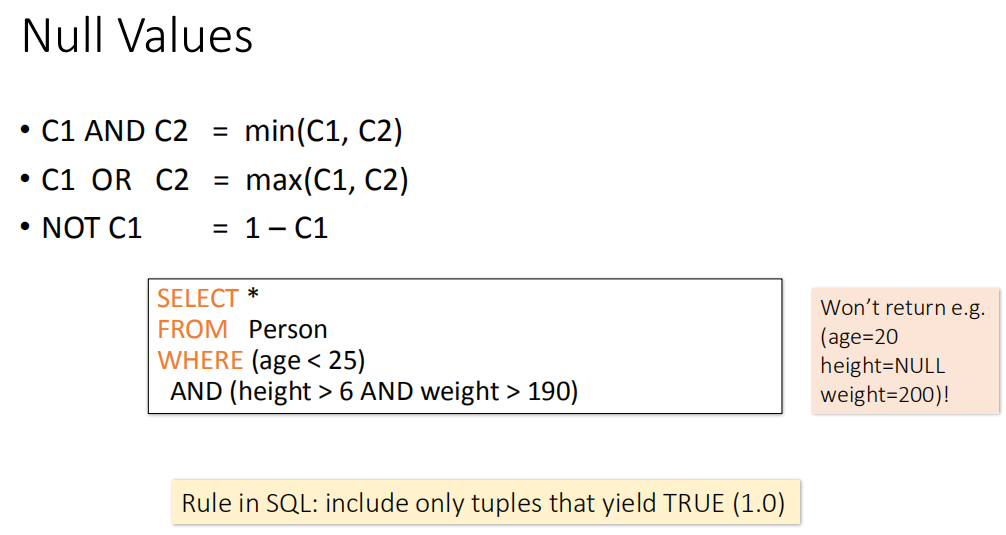
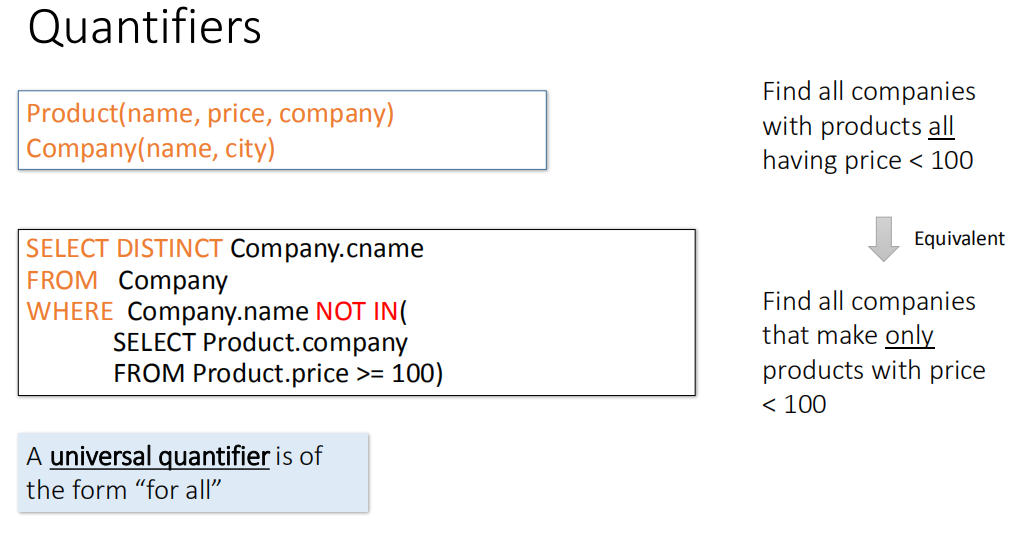
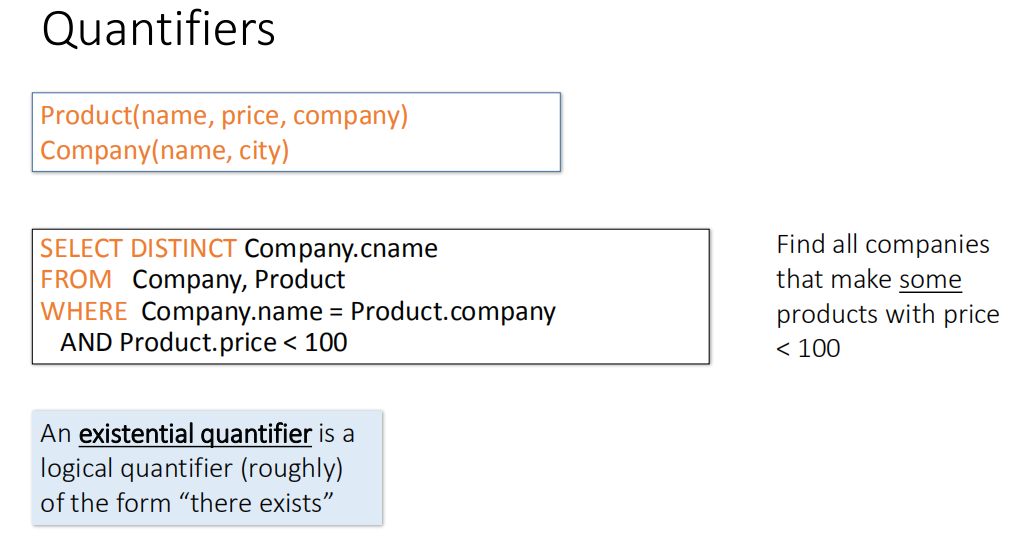
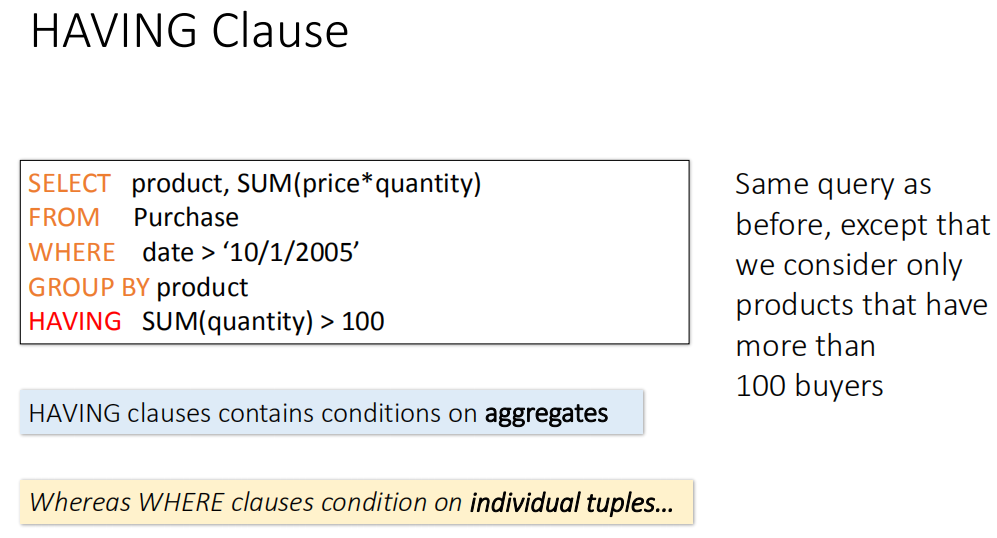
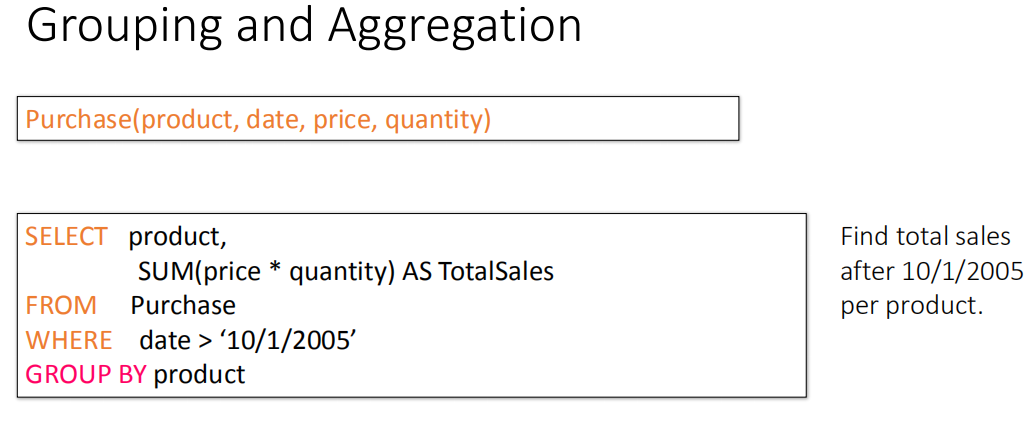
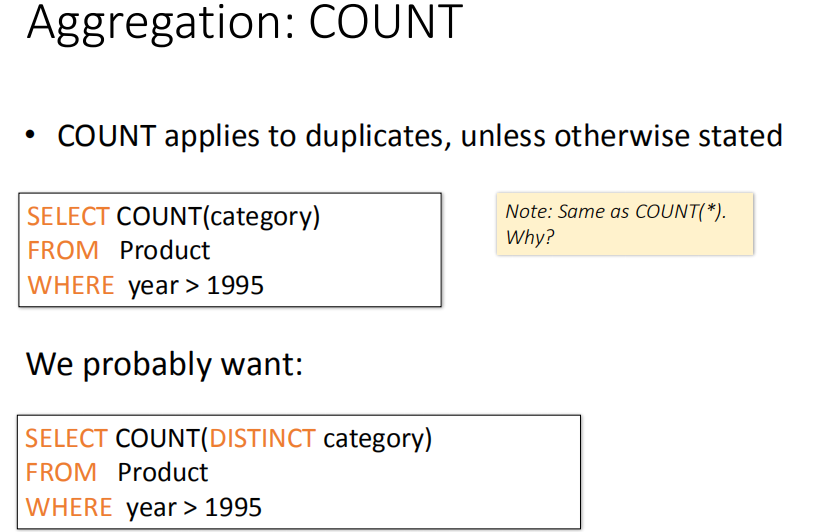
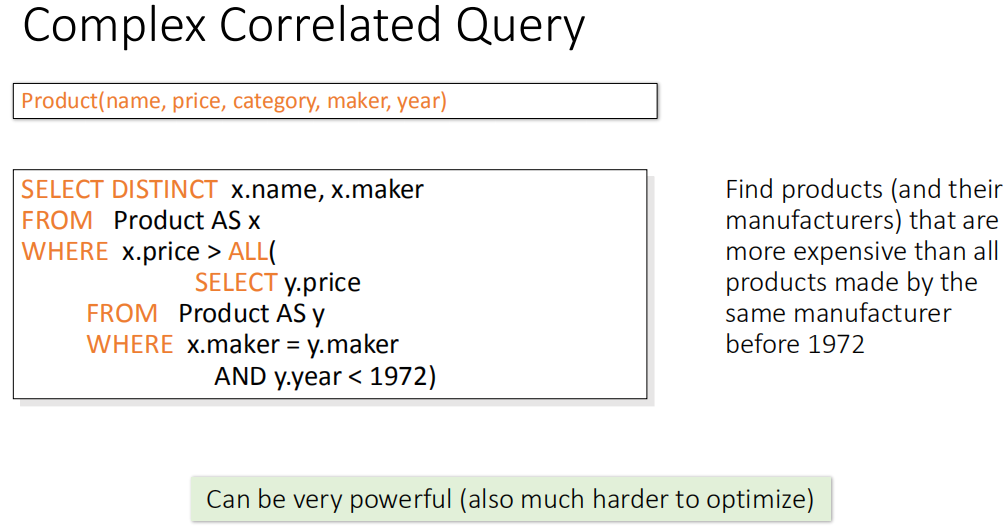
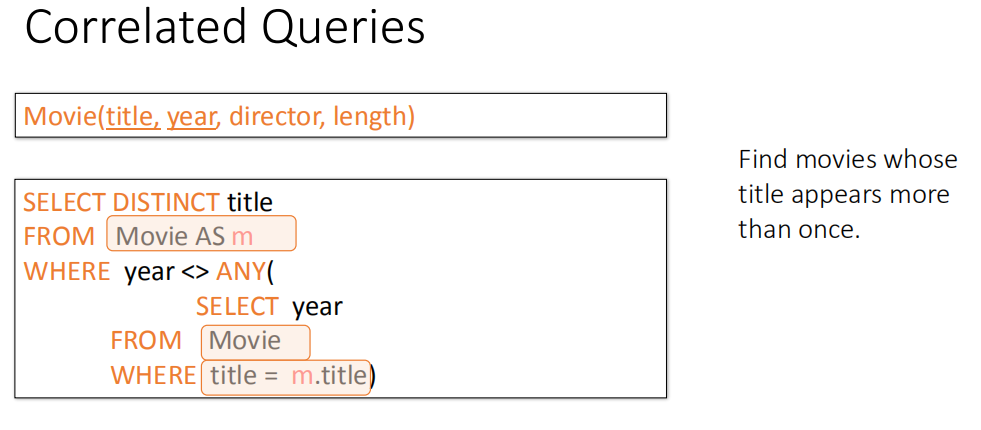
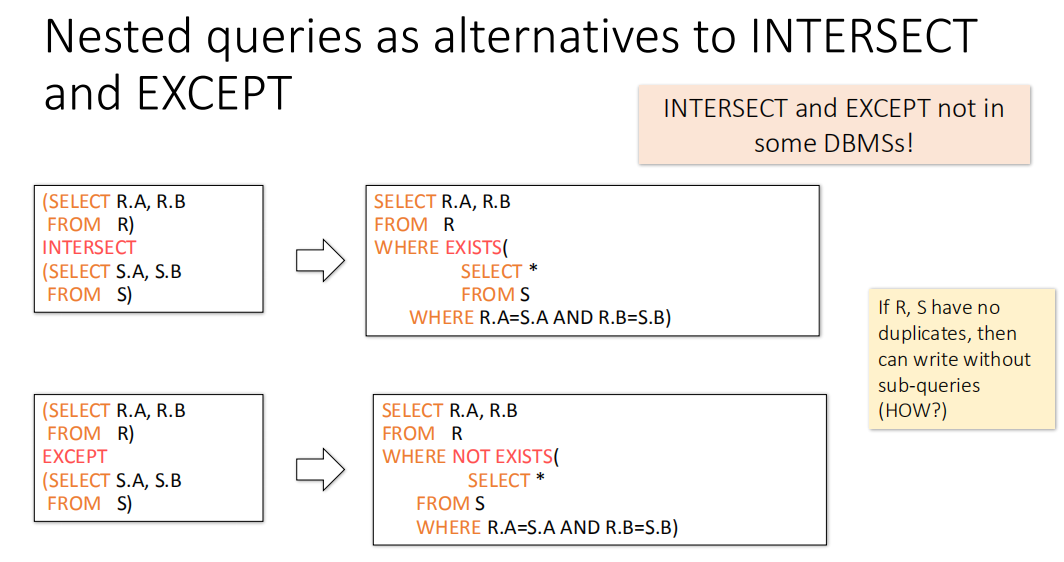
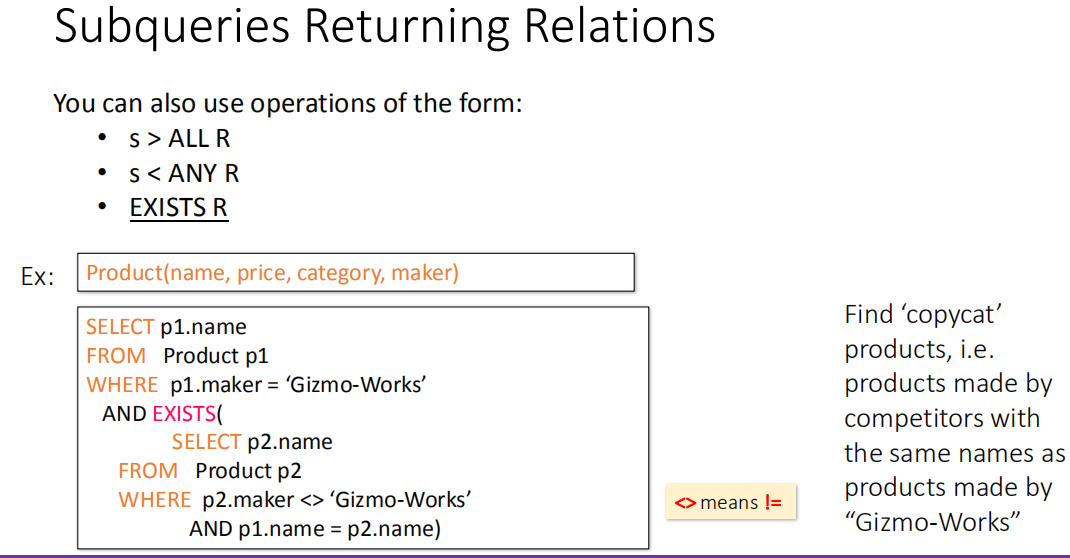
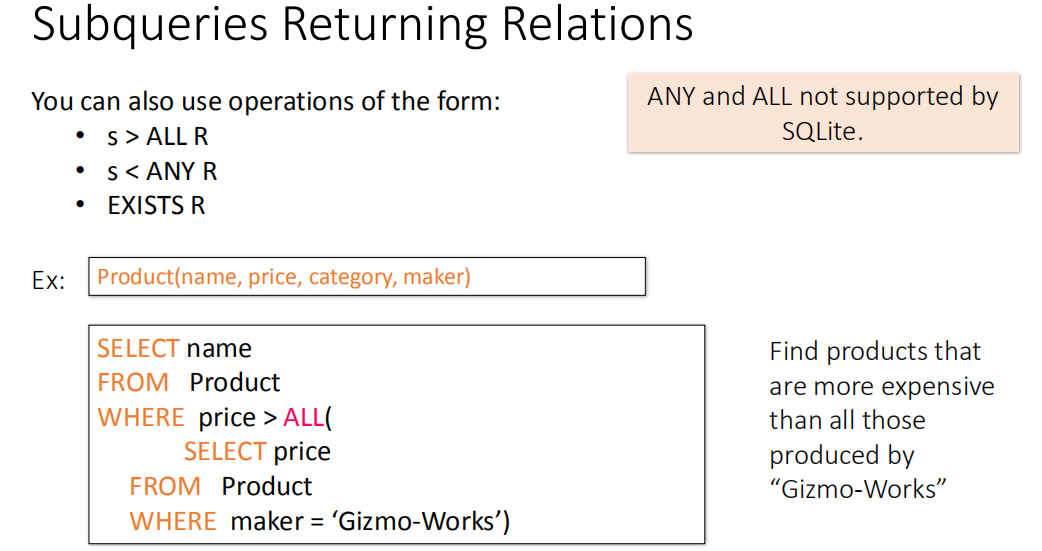
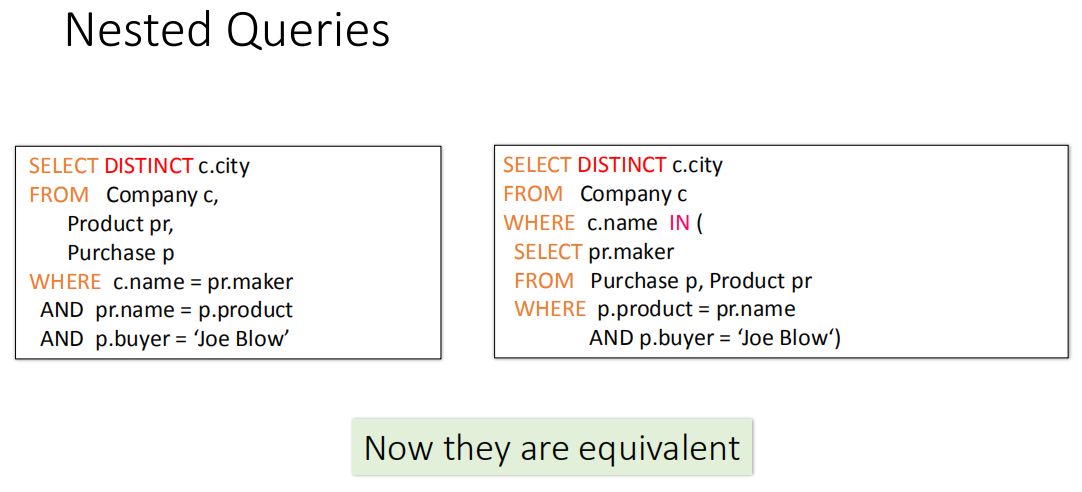
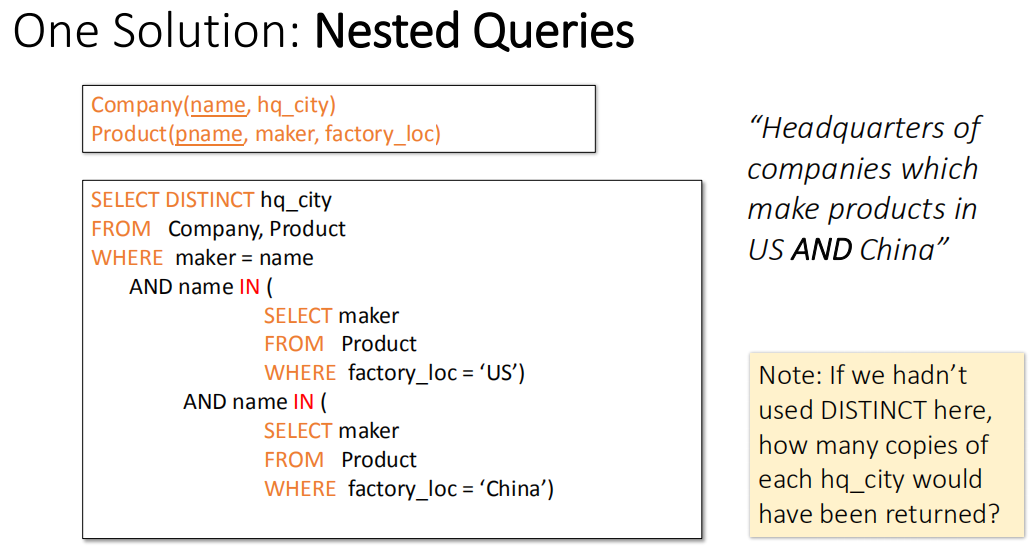
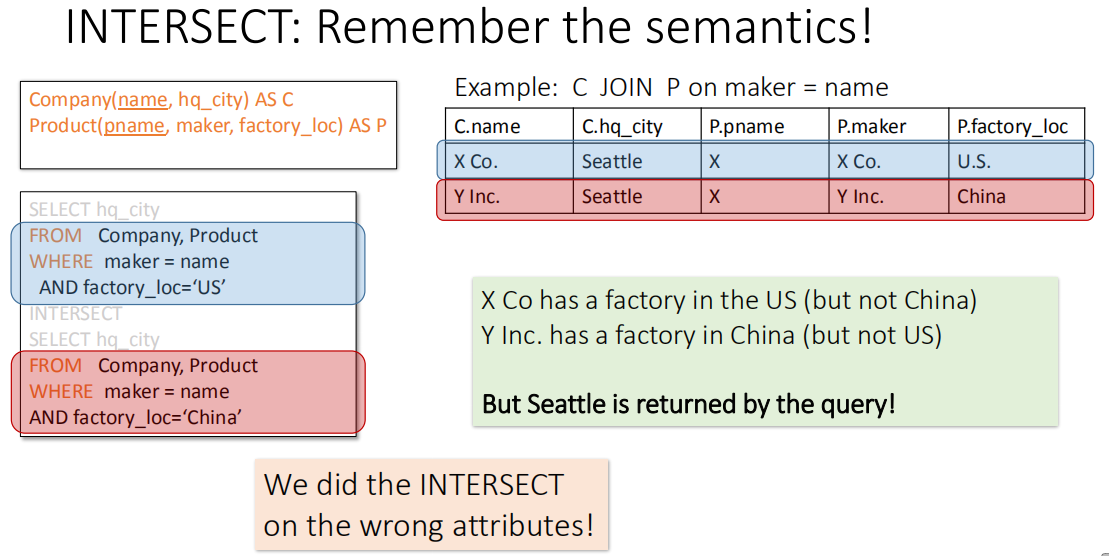
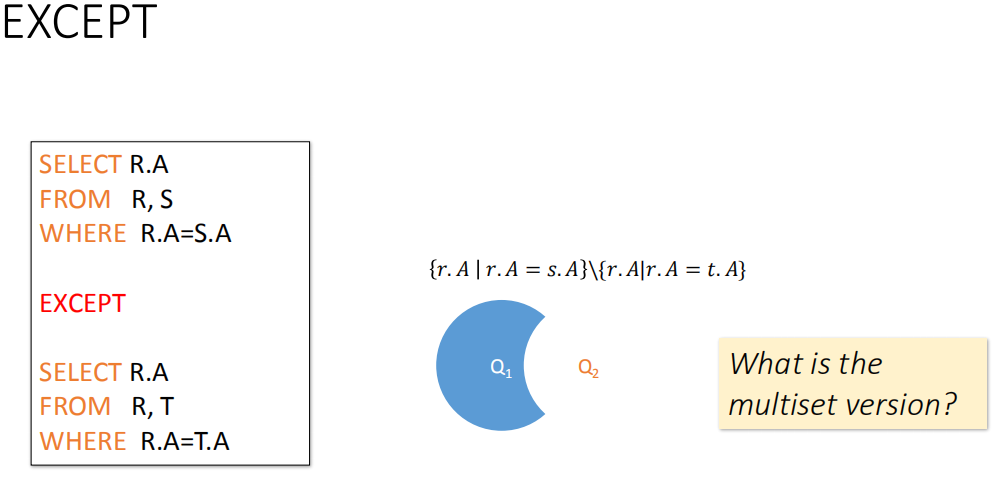
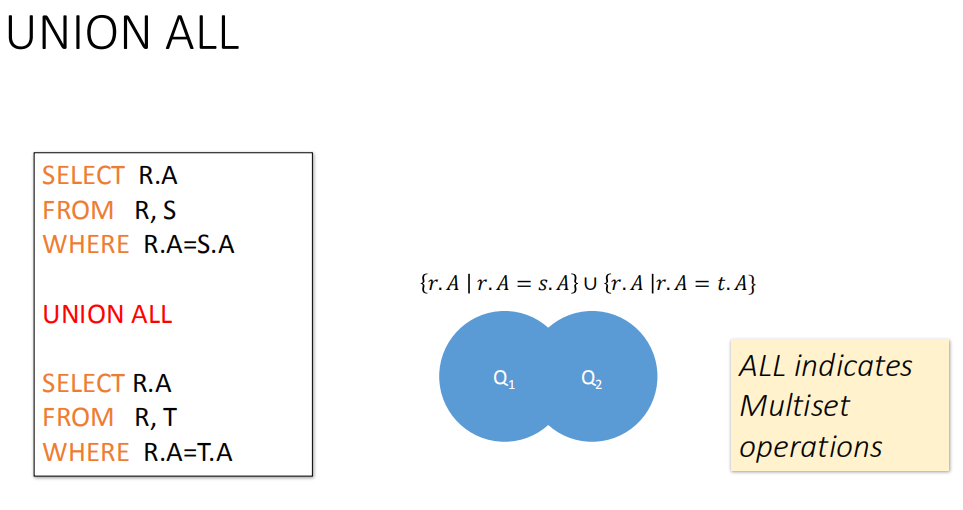
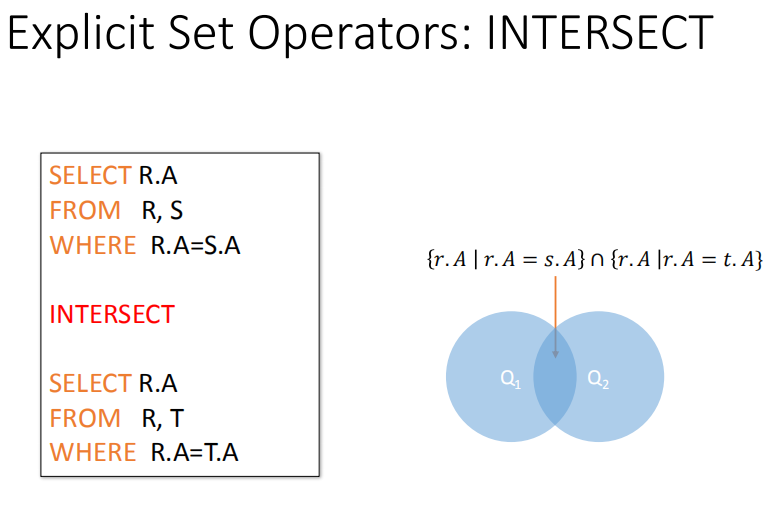
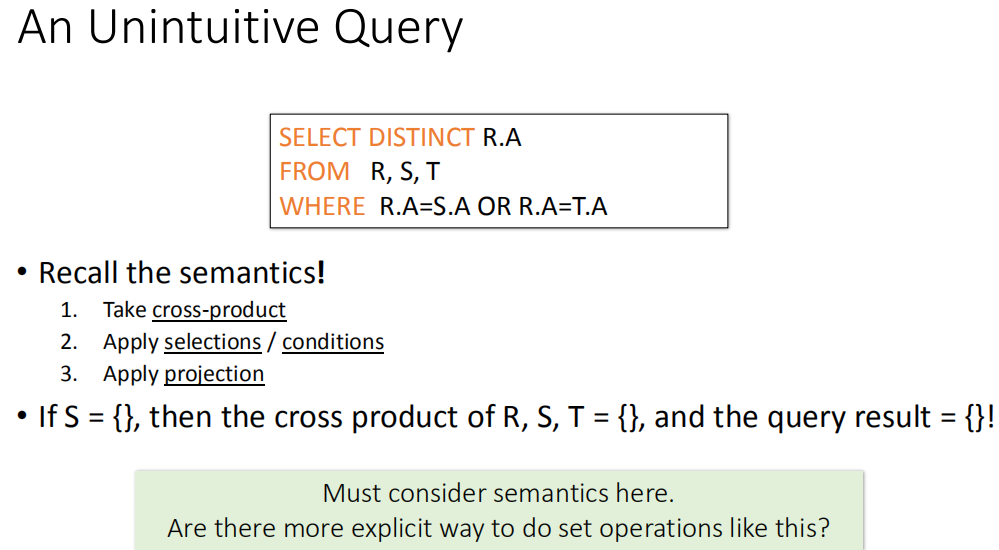
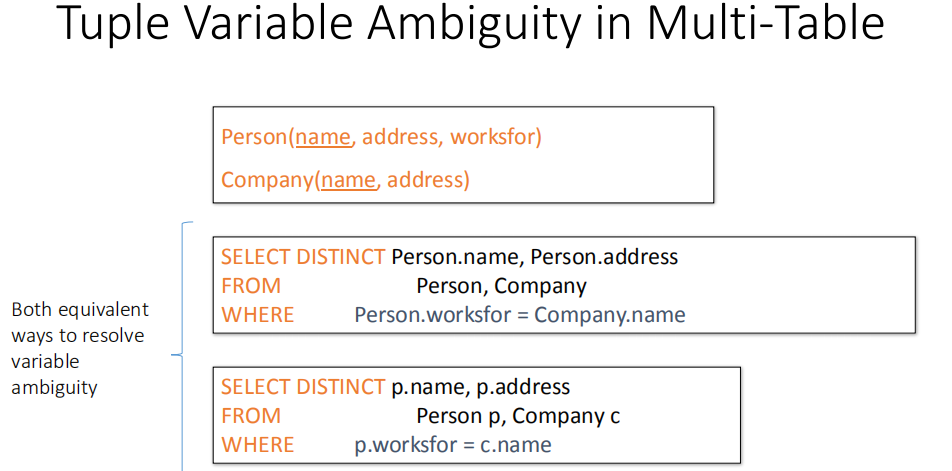
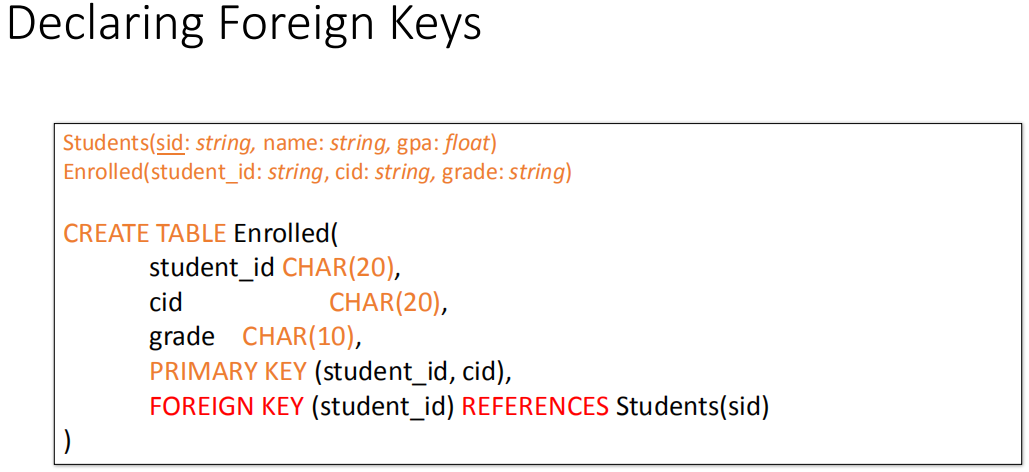
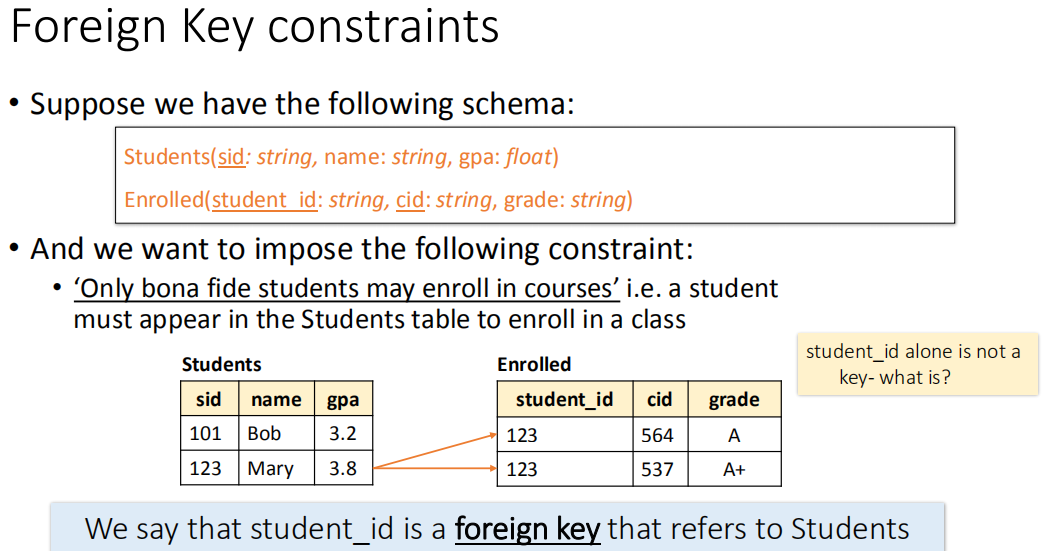
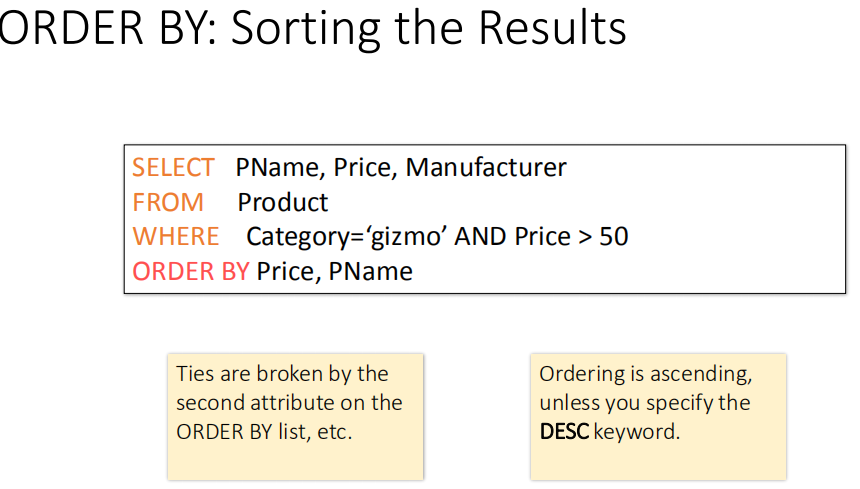
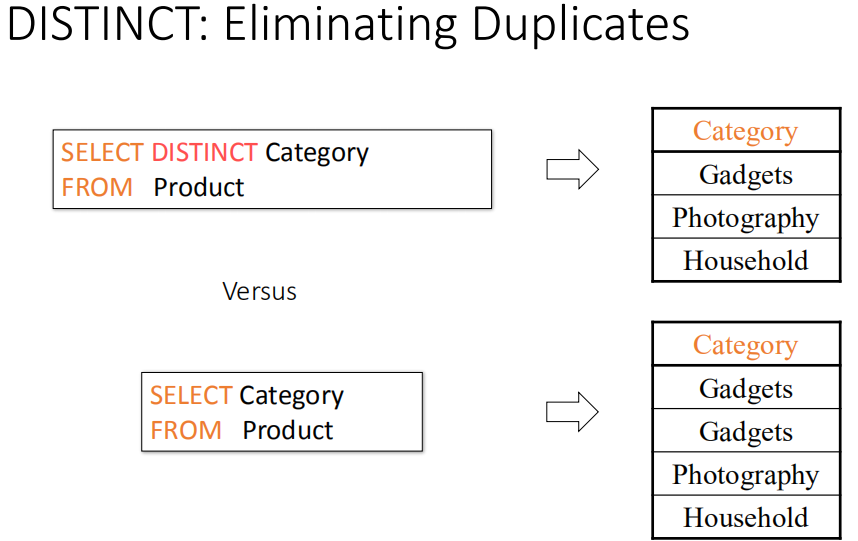
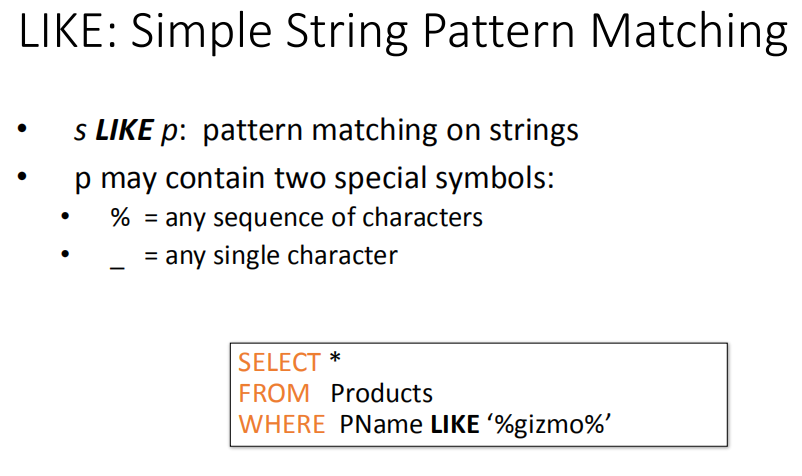
* A key is an attribute whose values are unique; we underline a key:

Product(Pname: string, Price: float, Category: string, Manufacturer: string)

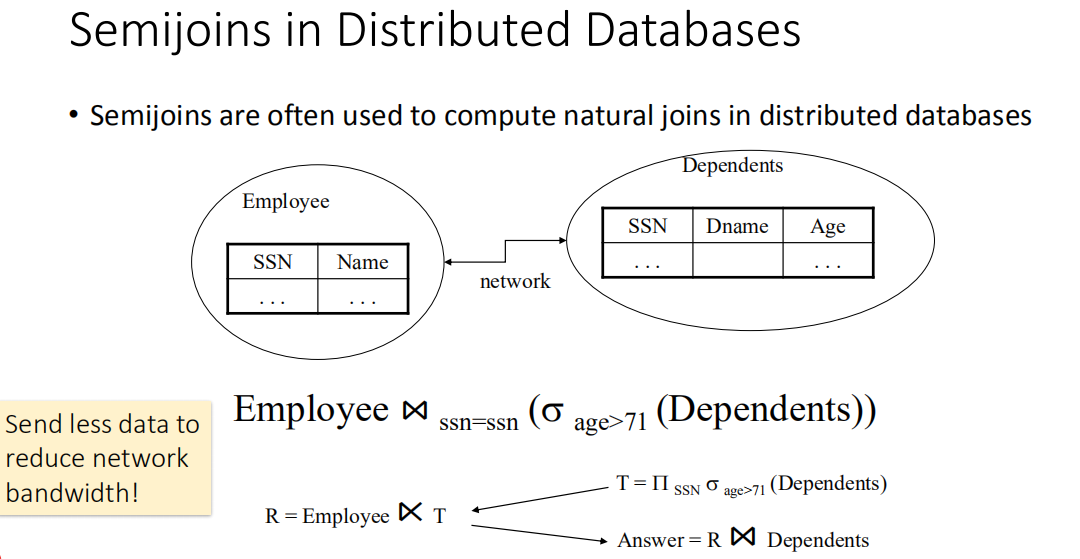
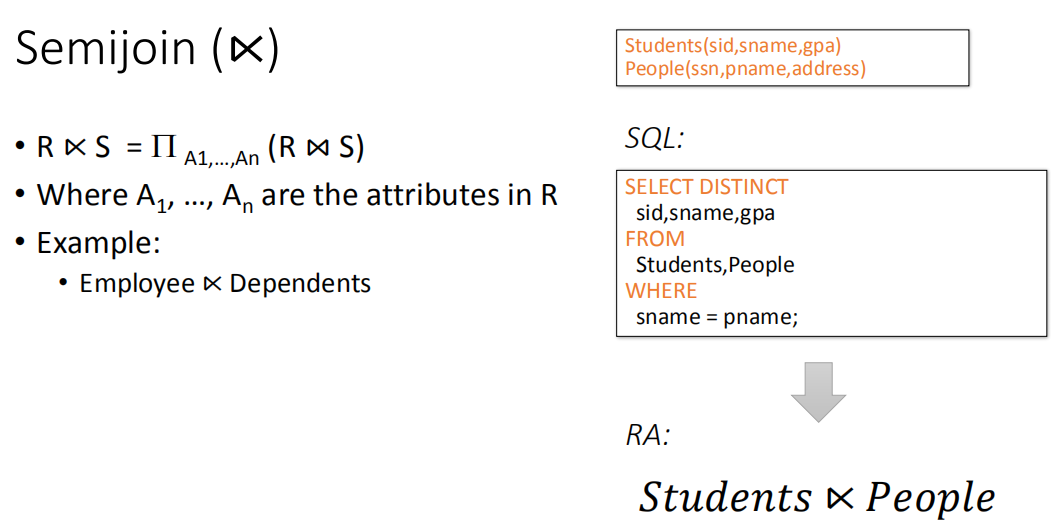
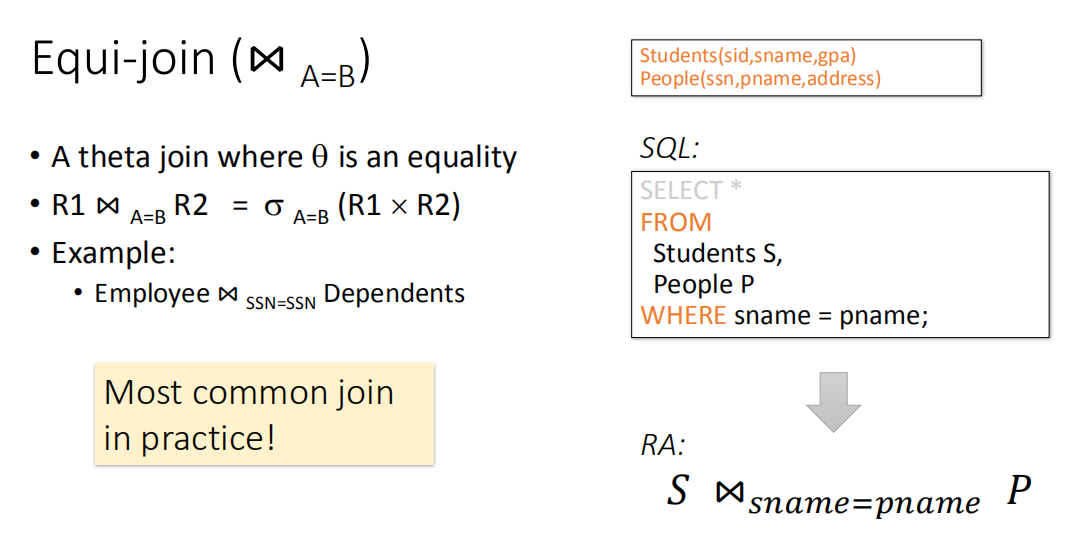
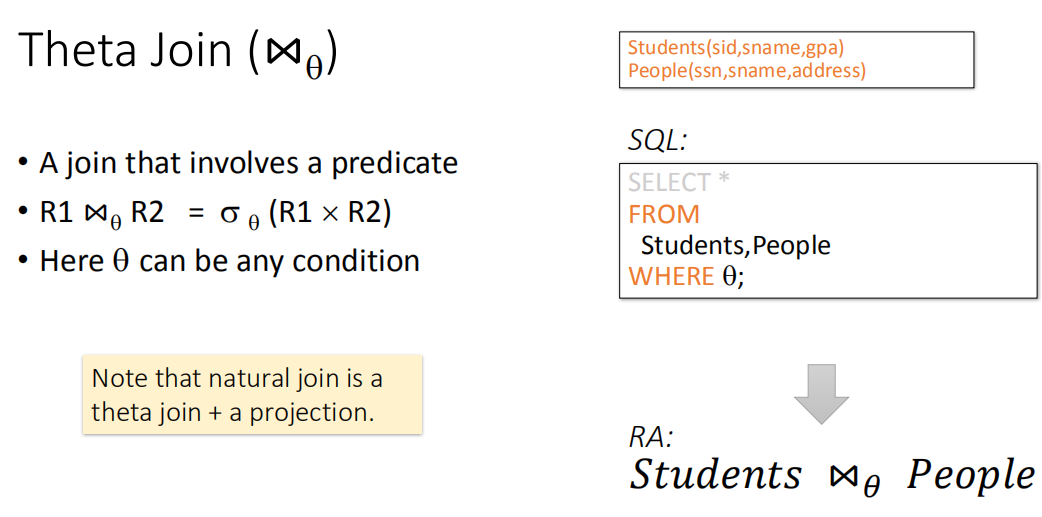
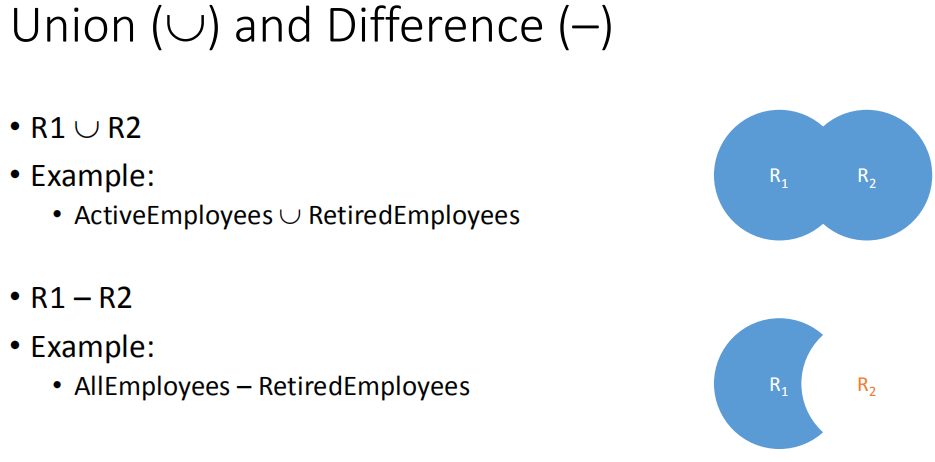
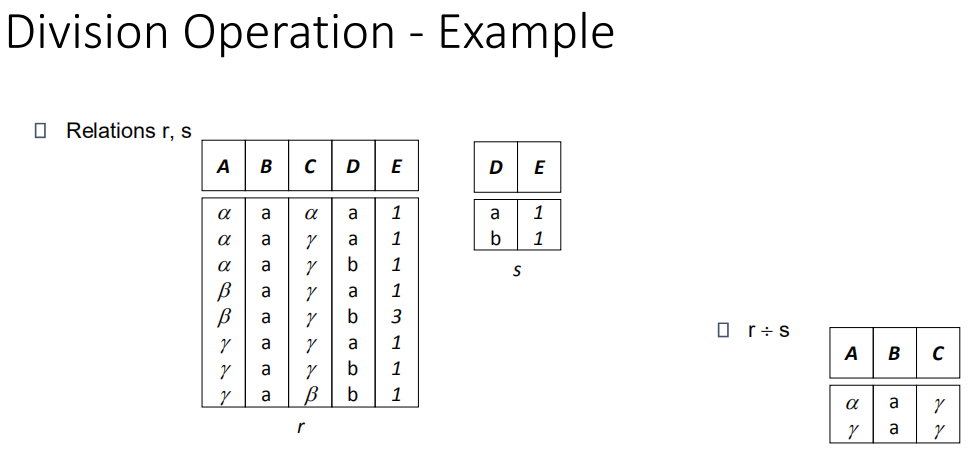
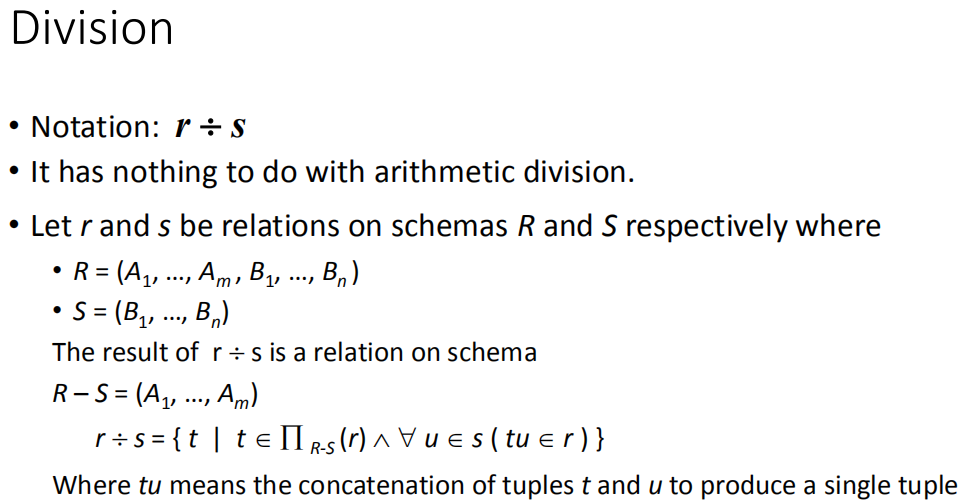
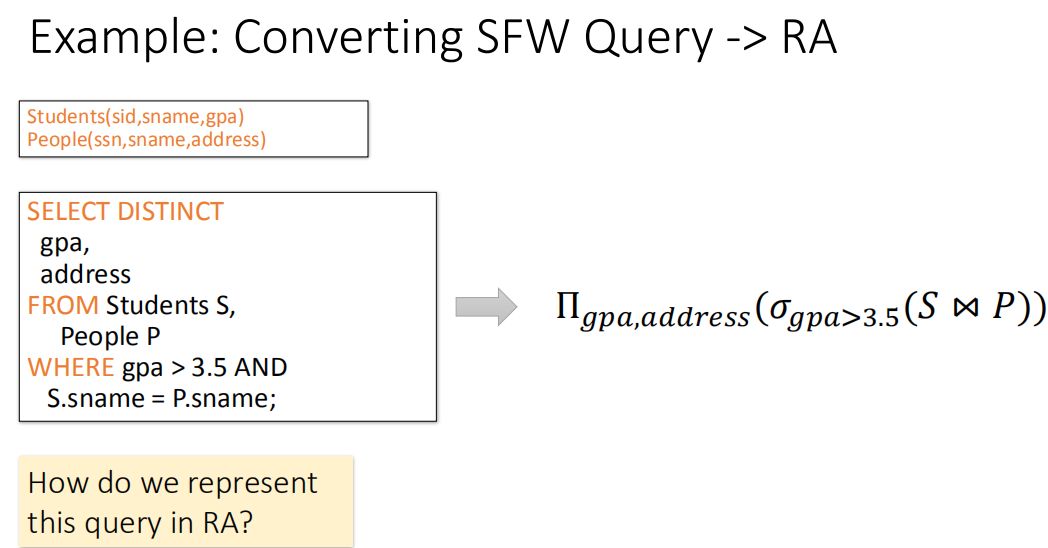
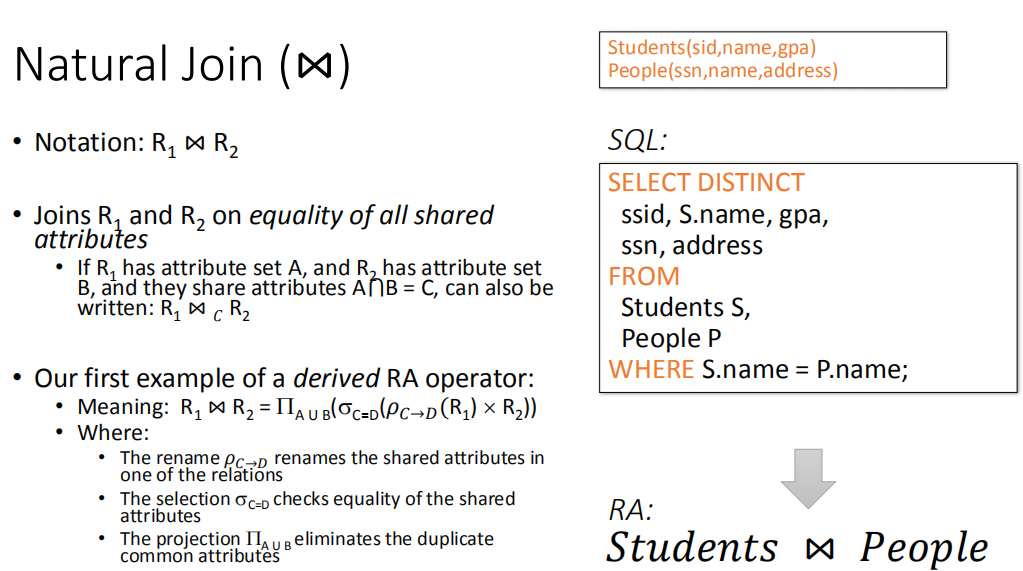
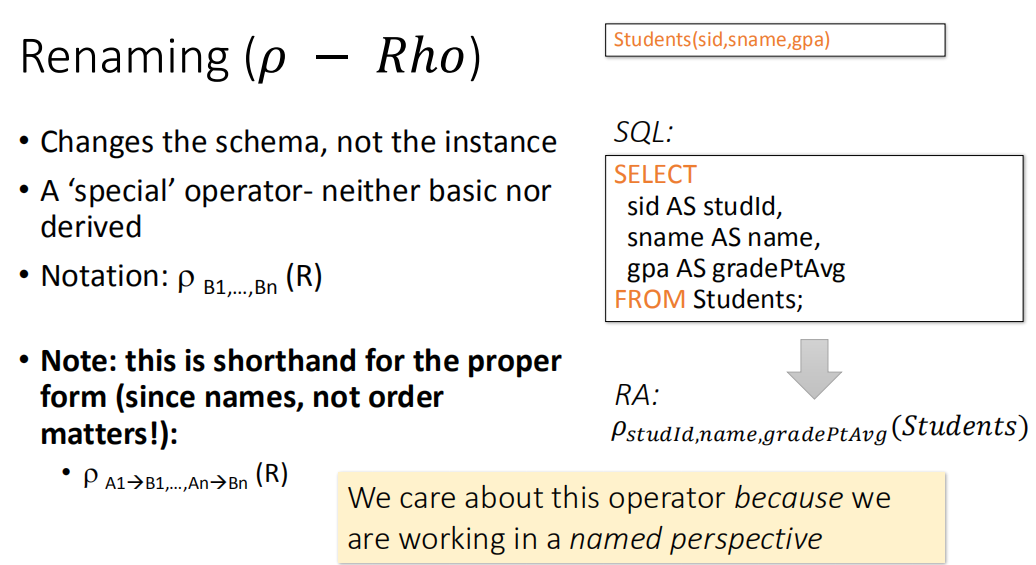
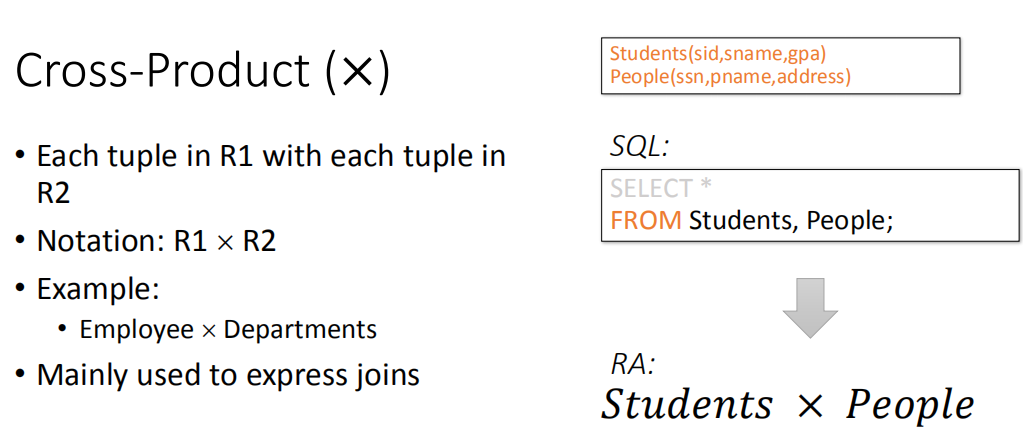
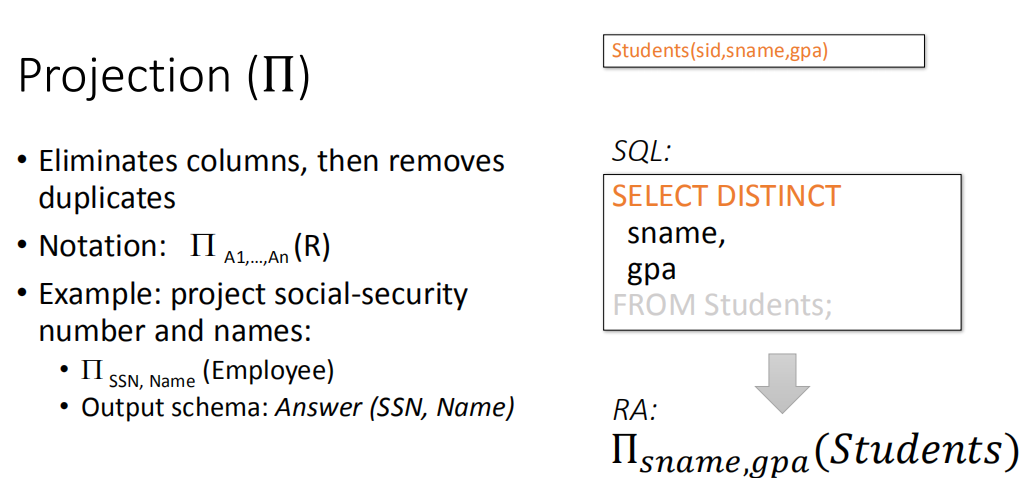
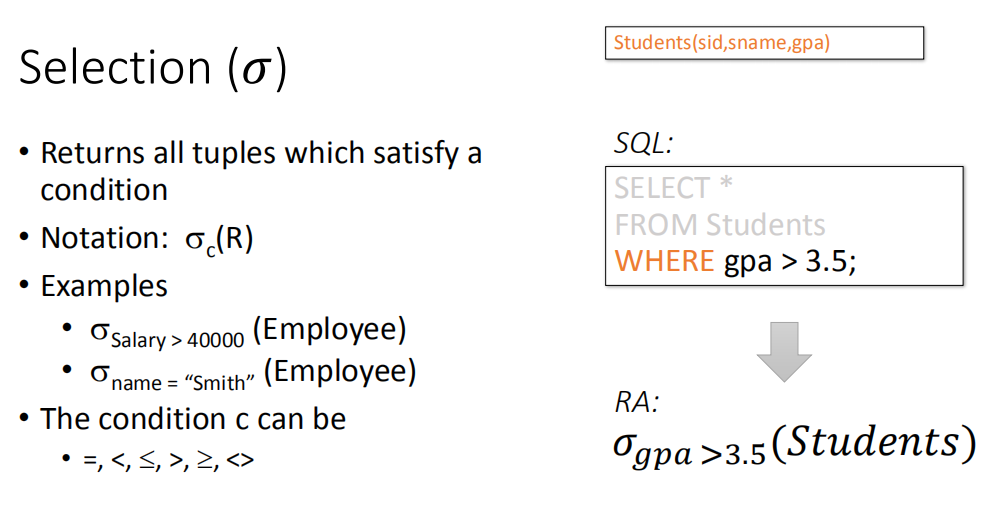
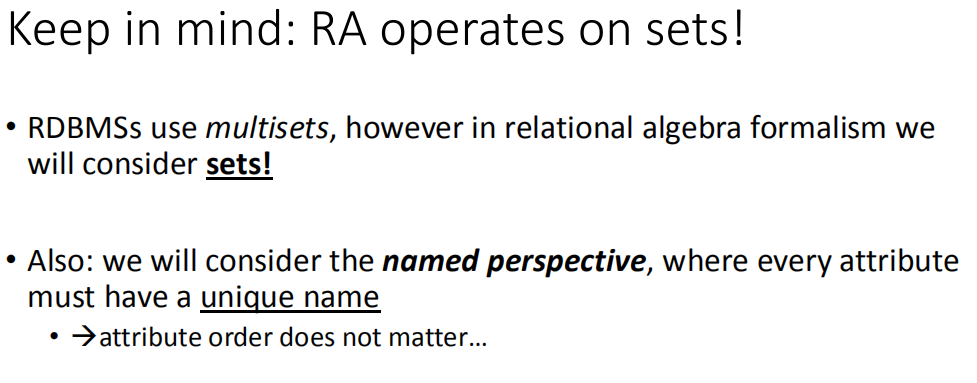
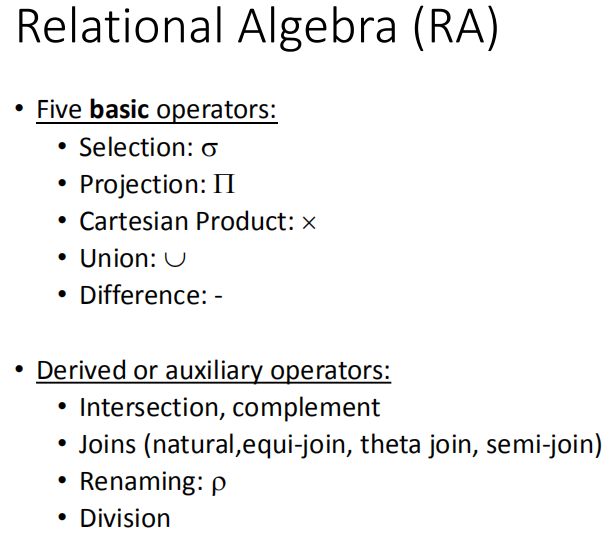
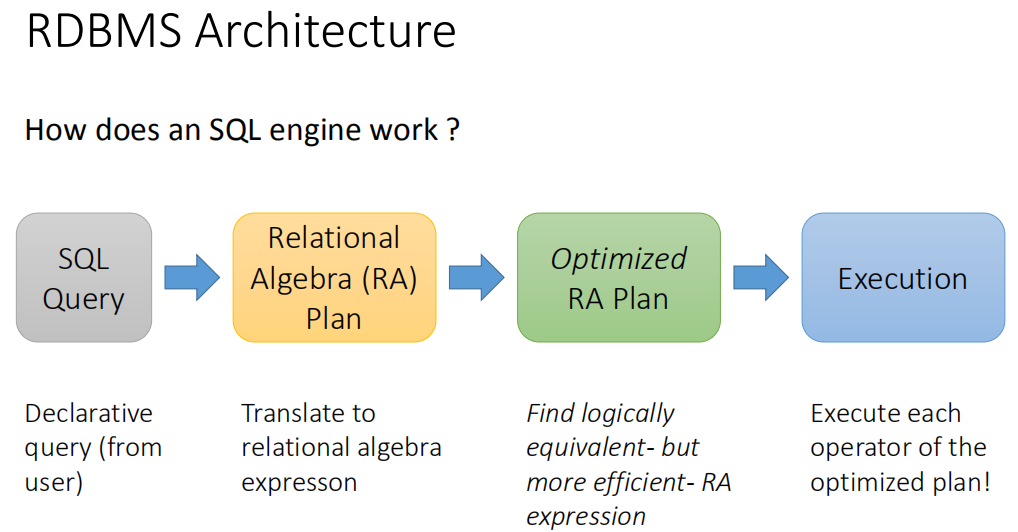
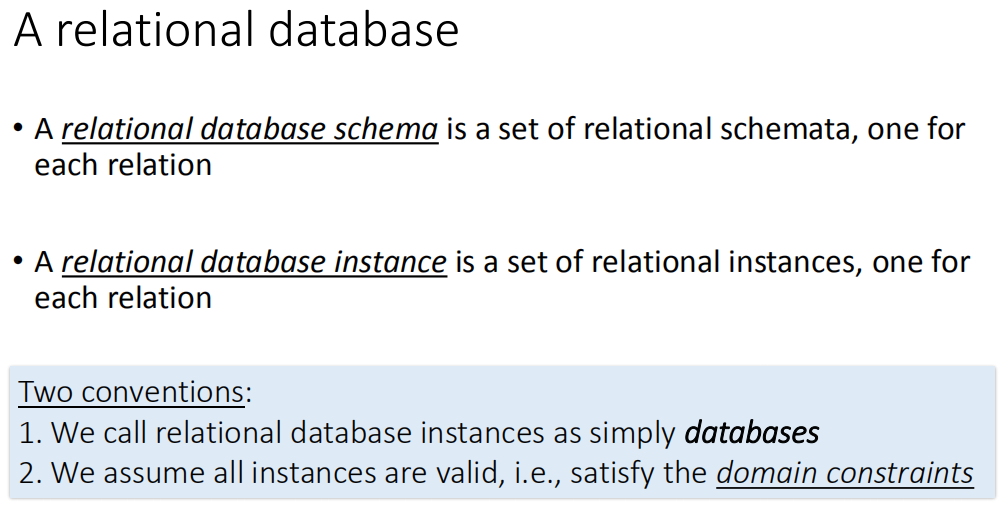
****



* **A Few Details**
* SQL statements are case insensitive: “Same: SELECT, Select, select” or “Same: Product, product”
* Values are not: “Different: ‘Seattle’, ‘seattle’ ”
* Use single quotes for constants: ‘abc’ - yes but “abc” - no



**Ders04 - Relational Algebra:**



**EXAMPLE RELATIONAL ALGEBRA**

