WORKSHOP SLIDES

DATE:30 APRIL 2025

KEY POINTS ON INSY6212 TEST

Key Concepts in

1. Entity Relationship Diagrams
2. Normalization
3. SQL

NB:These slides are based on DBAS6211_Testpaper_2024

ERD: BRIDGE (JUNCTION) TABLES - WHERE DO THE MULTIPLICITIES GO?

- Scenario: Many-to-Many relationship
  - Example: A product can have many sizes, and each size can apply to many products.
- Use a bridge table (e.g., ProductSize)

Multiplicity:

Table	**Multiplicity side**
Bridge table (ProductSize)	* Always gets the * side (e.g. 1.* or 0.* according to the case given) * All the * (many) go toward the bridge table.
Original tables (Product, Size)	Get the 1.1 side

ERD: WHERE DO FOREIGN KEYS GO?

- **In 1:M relationships: FK goes on the * side**
  - Example: Designer → Products
    - Product table gets FK: DesignerID

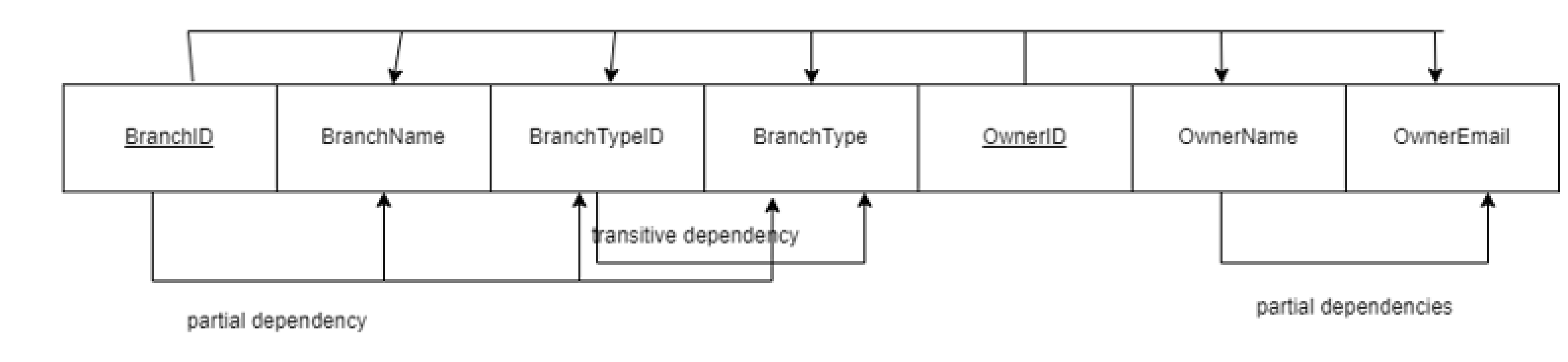
- **In Many to many relationships: FK goes in the bridge table**

- **Pro Tip:** "Follow the many: the FK always goes to the side where you can have many of the other."

NORMALIZATION TO 3NF

First Normal Form (1NF)

- Eliminate repeating groups
- Make sure each field holds only atomic values

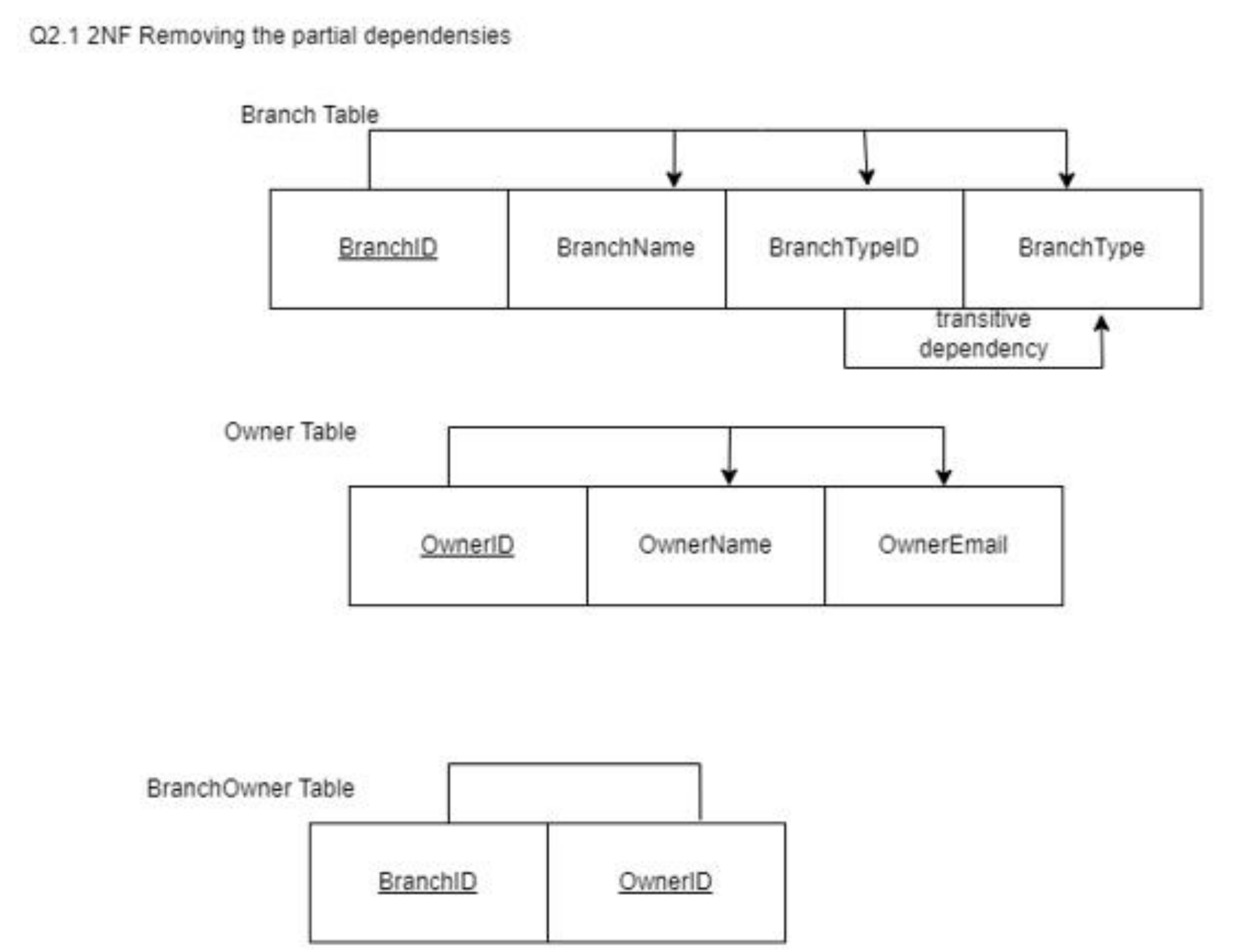


Second Normal Form (2NF)

- Remove partial dependencies
- Separate tables and their dependencies

SECOND NORMAL FORM

Split Example:



Q2.1 2NF Removing the partial dependensies

Branch Table

BranchID	BranchName	BranchTypeID	BranchType

transitive dependency

Owner Table

OwnerID	OwnerName	OwnerEmail

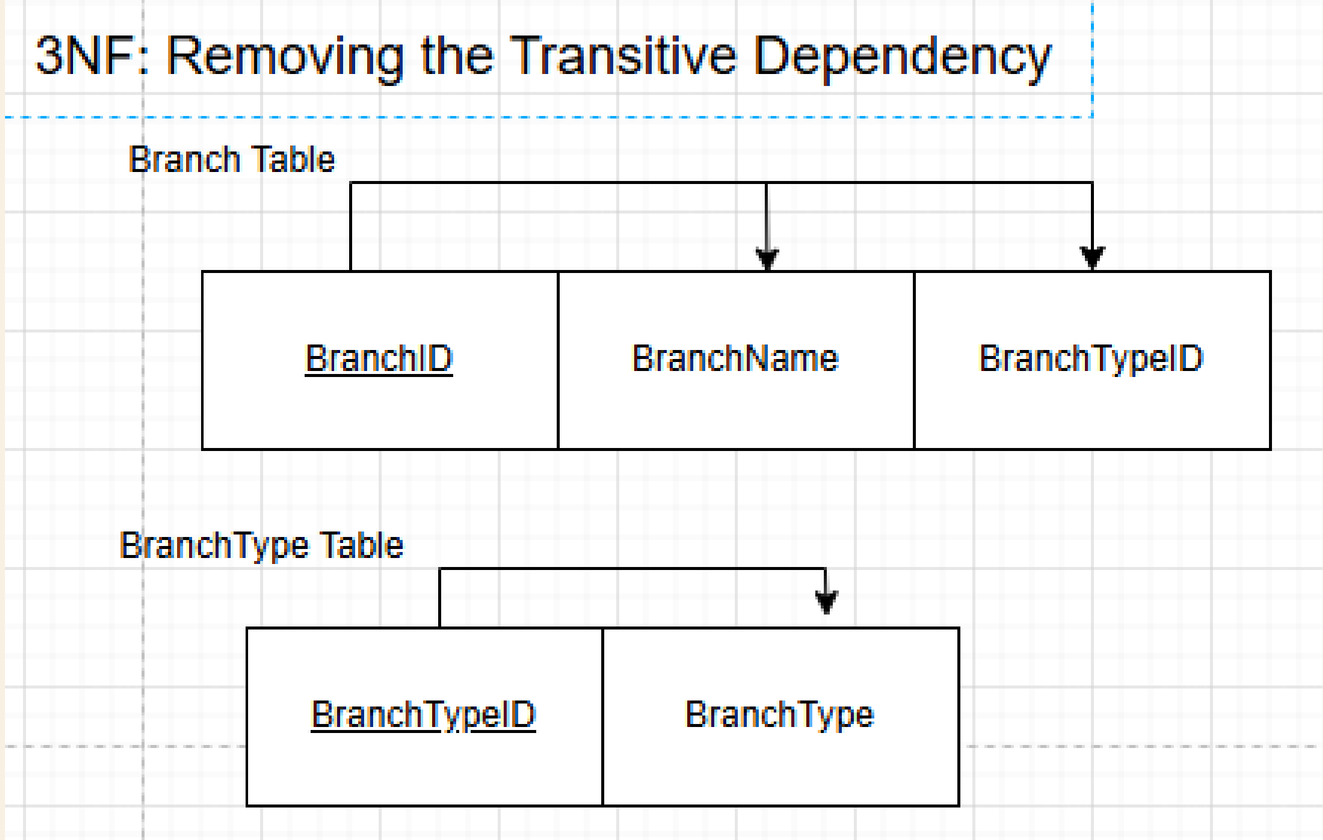
BranchOwner Table

BranchID	OwnerID

THIRD NORMAL FORM 3NF

Third Normal Form (3NF)

- Remove transitive dependencies
- **Updated Split:**



3NF: Removing the Transitive Dependency

Branch Table

BranchID	BranchName	BranchTypeID

BranchType Table

BranchTypeID	BranchType

SQL

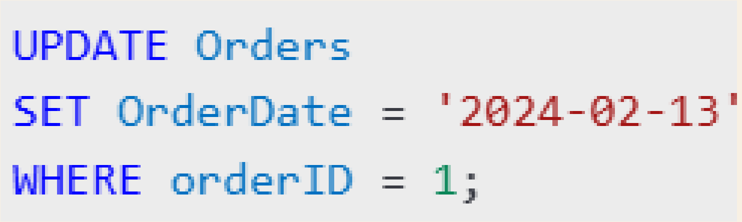
SQL Command

1. UPDATE
2. ALTER
3. DELETE

4 .DROP

1. UPDATE

- Used to modify existing data in a table
- Always use a WHERE clause!
- **EXAMPLE CODE**



```sql
UPDATE Orders
SET OrderDate = '2024-02-13'
WHERE orderID = 1;
```

SQL- ALTER

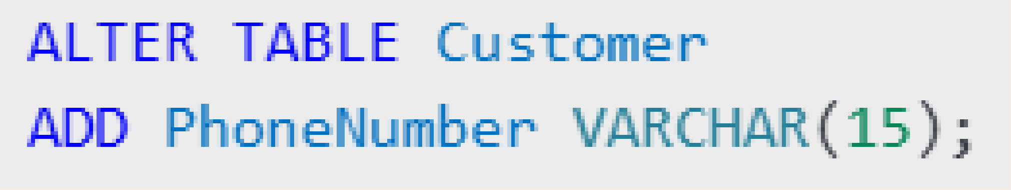
ALTER

•Used to change table structure

•Common Use Cases:

1. **Add a column:**

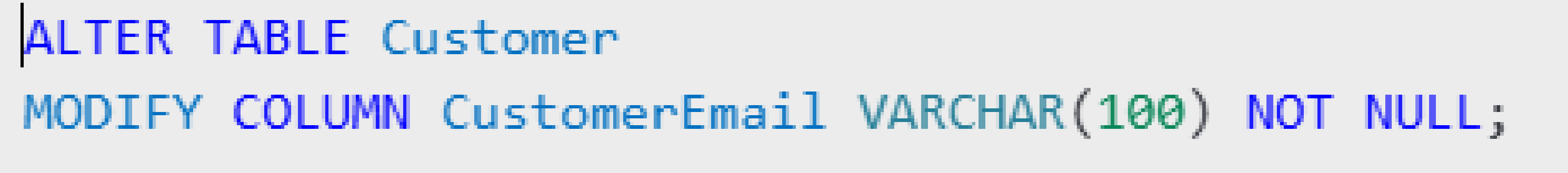
#EXAMPLE CODE: Add Column PhoneNumber in Customer Table



```
ALTER TABLE Customer
ADD PhoneNumber VARCHAR(15);
```

2. **Modify a column:**

#EXAMPLE CODE:

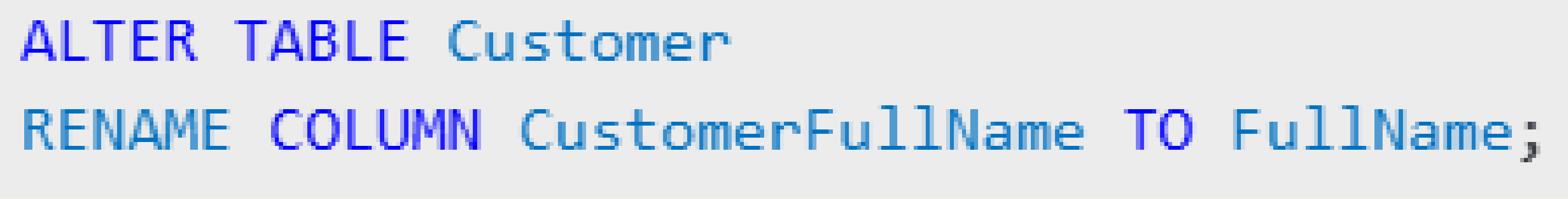


```
ALTER TABLE Customer
MODIFY COLUMN CustomerEmail VARCHAR(100) NOT NULL;
```

SQL- ALTER

3.Rename a column:

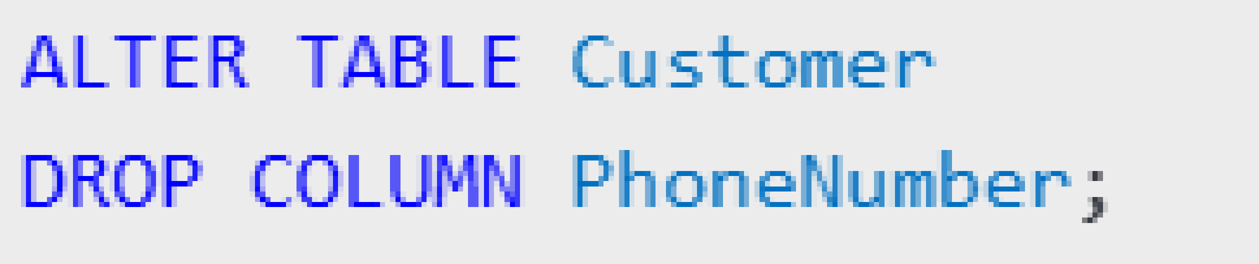
EXAMPLE CODE:



```
ALTER TABLE Customer
RENAME COLUMN CustomerFullName TO FullName;
```

4. Drop a column:

EXAMPLE CODE:



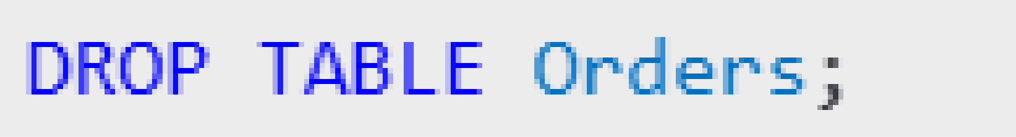
```
ALTER TABLE Customer
DROP COLUMN PhoneNumber;
```

SQL- DROP AND DELETE

DROP

Used to permanently remove tables or databases

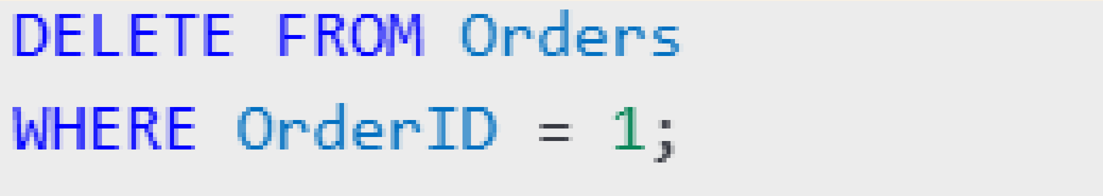
#EXAMPLE CODE:



```sql
DROP TABLE Orders;
```

DELETE

- Deletes data from a table, not the table itself
- Always use WHERE clause to target specific rows
- #EXAMPLE CODE:



```sql
DELETE FROM Orders
WHERE OrderID = 1;
```

THANK YOU

ALL THE BEST IN YOUR TEST!!!