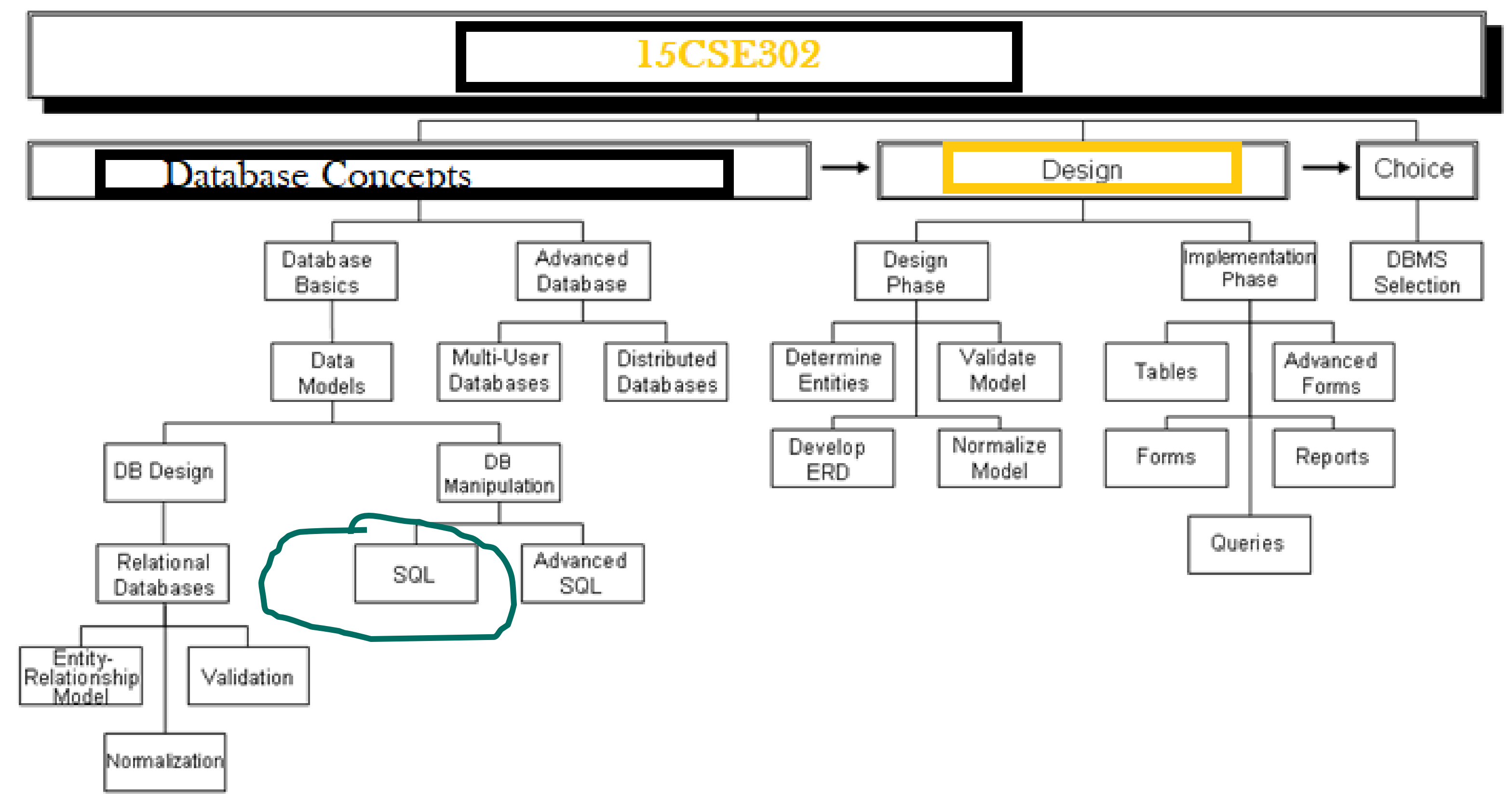
15CSE302 Database Management Systems

SQL -Subqueries

B.Tech /III Year CSE/V Semester L T P C 2 0 2 3

DBMS Team
Dr G Jeyakumar
Bindu K R
Dr Priyanka Kumar
R. Manjusha
Department of CSE
Amrita School of Engineering

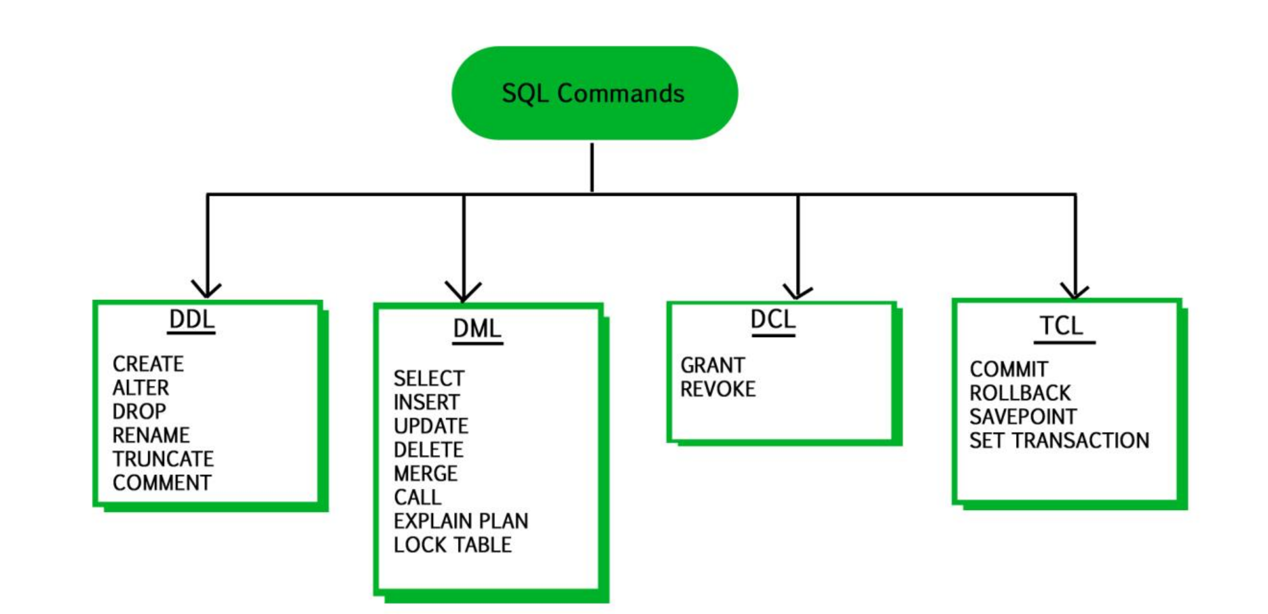
Syllabus


15CSE302
Database Concepts
Design
Choice
Database Basics
Advanced Database
Data Models
Multi-User Databases
Distributed Databases
DB Design
DB Manipulation
Relational Databases
SQL
Advanced SQL
Entity-Relationship Model
Validation
Normalization
Design Phase
Implementation Phase
DBMS Selection
Determine Entities
Validate Model
Develop ERD
Normalize Model
Tables
Advanced Forms
Forms
Reports
Queries

Contents

- SQL Subqueries
  - Relational operators
  - IN
  - ALL
  - ANY

SQL Structured Query Language



SQL Commands

DDL

CREATE
ALTER
DROP
RENAME
TRUNCATE
COMMENT

DML

SELECT
INSERT
UPDATE
DELETE
MERGE
CALL
EXPLAIN PLAN
LOCK TABLE

DCL

GRANT
REVOKE

TCL

COMMIT
ROLLBACK
SAVEPOINT
SET TRANSACTION

Renaming Columns

Product

PName	Price	Category	Manufacturer
Gizmo	$19.99	Gadgets	GizmoWorks
Powergizmo	$29.99	Gadgets	GizmoWorks
SingleTouch	$149.99	Photography	Canon
MultiTouch	$203.99	Household	Hitachi

```
SELECT   Pname AS prodName,
Price AS askPrice
FROM     Product
WHERE    Price > 100
```

prodName	askPrice
SingleTouch	$149.99
MultiTouch	$203.99

Query with renaming

Subqueries

A subquery is a SQL query nested inside a larger query.

A **subquery** may occur in :

- The **subquery** can be nested inside a SELECT, INSERT, UPDATE, or DELETE statement or inside another **subquery**.

- A **subquery** is usually added within the WHERE Clause of another **SQL** SELECT statement.

Subqueries

Types of Subqueries

- Single-row subquery
- Multiple-row subquery
- Multiple-column subquery

Subqueries

Types of Subqueries

Single-row subqueries: Queries that return only one row from the inner SELECT statement

Multiple-row subqueries: QUERIES that return more than one rows from the inner SELECT statement

Multiple-column subqueries: QUERIES that return more than one column from the inner SELECT statement.

Subqueries

A subquery producing a single value:

Find the product bought by Person whose ssn no is 123456789

```
SELECT Purchase.product
FROM      Purchase
WHERE   buyer =
            (SELECT  name
              FROM      Person
              WHERE    ssn = ‘123456789‘);
```

In this case, the subquery returns one value.
If it returns more, it’s a run-time error.

Same query with Join

Find the product bought by Person whose ssn no is 123456789

Can say the same thing without a subquery:

```
SELECT Purchase.product
FROM     Purchase,  Person
WHERE  buyer = name  AND ssn  = ‘123456789‘
```

This is equivalent to the previous one when the ssn is a key and

‘123456789’ exists in the database; otherwise they are different.

Subqueries Returning Relations-- IN

Find companies that manufacture products bought by Joe Blow.

```
SELECT  Company.name
FROM    Company, Product
WHERE   Company.name=Product.maker
            AND  Product.name  IN
                 (SELECT Purchase.product
                   FROM   Purchase
                   WHERE Purchase .buyer = ‘Joe Blow‘);
```

Here the subquery returns a set of values: no more runtime errors.

Subqueries Returning Relations

Equivalent to:

```
SELECT  Company.name
FROM       Company,  Product,  Purchase
WHERE    Company.name=  Product.maker
              AND  Product.name   = Purchase.product
              AND  Purchase.buyer  = ‘Joe Blow’
```

Is this query equivalent to the previous one ?

Beware of duplicates !

Removing Duplicates

```
SELECT DISTINCT  Company.name
FROM      Company,  Product
WHERE    Company.name=  Product.maker
         AND  Product.name   IN
                 (SELECT  Purchase.product
                   FROM     Purchase
                   WHERE  Purchase.buyer  = 'Joe Blow')
```

```
SELECT DISTINCT Company.name
FROM      Company, Product, Purchase
WHERE   Company.name= Product.maker
          AND  Product.name  = Purchase.product
          AND  Purchase.buyer = 'Joe Blow'
```

Now they are equivalent

Subqueries Returning Relations --ALL

You can also use: s > ALL R
s > ANY R
EXISTS R

Product (pname, price, category, maker)

Find products that are more expensive than all those produced By “Gizmo-Works”

```
SELECT  name
FROM    Product
WHERE   price >  ALL (SELECT  price
                      FROM    Purchase
                      WHERE   maker=‘Gizmo-Works’)
```

GROUP BY vs Nested Quereis

```
SELECT      product, Sum(price*quantity) AS TotalSales
FROM        Purchase
WHERE       date > “9/1”
GROUP BY  product
```

```
SELECT DISTINCT  x.product, (SELECT Sum(y.price*y.quantity)
                             FROM     Purchase y
                             WHERE x.product = y.product
                                   AND y.date > ‘9/1’)
                            AS TotalSales
FROM          Purchase x
WHERE         x.date > “9/1”
```

Subquery in insert

```
INSERT INTO Purchase_new  (product)  values
SELECT Purchase.product
FROM     Purchase
WHERE  buyer =
           (SELECT  name
             FROM     Person
             WHERE   ssn = ‘123456789‘);
```

Subquery in update

Let's assume we have an EMPLOYEE_BKP table available which is backup of EMPLOYEE table.
The given example updates the SALARY by .25 times in the EMPLOYEE table for all employee whose AGE is greater than or equal to 29.

```
UPDATE EMPLOYEE
   SET SALARY = SALARY * 0.25
   WHERE AGE IN (SELECT AGE FROM CUSTOMERS_BKP
      WHERE AGE >= 29);
```

Subquery in delete

Let's assume we have an EMPLOYEE_BKP table available which is backup of EMPLOYEE table.

The given example deletes the records from the EMPLOYEE table for all EMPLOYEE whose AGE is greater than or equal to 29.

```
DELETE FROM EMPLOYEE
   WHERE AGE IN (SELECT AGE FROM EMPLOYEE_BKP
      WHERE AGE >= 29 );
```

Guidelines for Using Subqueries

- ❑ Enclose subqueries in parentheses.
- ❑ Place subqueries on the right side of the comparison operator.
- ❑ Do not add an ORDER BY clause to a subquery.
- ❑ Use single-row operators with singlerow subqueries.
- ❑ Use multiple-row operators with multiple-row subqueries.

Summary

- SQL Subqueries

Next Session

- Correlated Subqueries

References

- https://www.geeksforgeeks.org/sql-subquery/

About Me

Bindu K R

Assistant Professor

Areas of Interests:

1. NLP
2. Information Retrieval
3. Deep Learning

E-mail:j_bindu@cb.amrita.edu

Thank You

Happy to answer any questions ! ! !