Business School
UNIVERSITY OF COLORADO DENVER

Information Systems Program

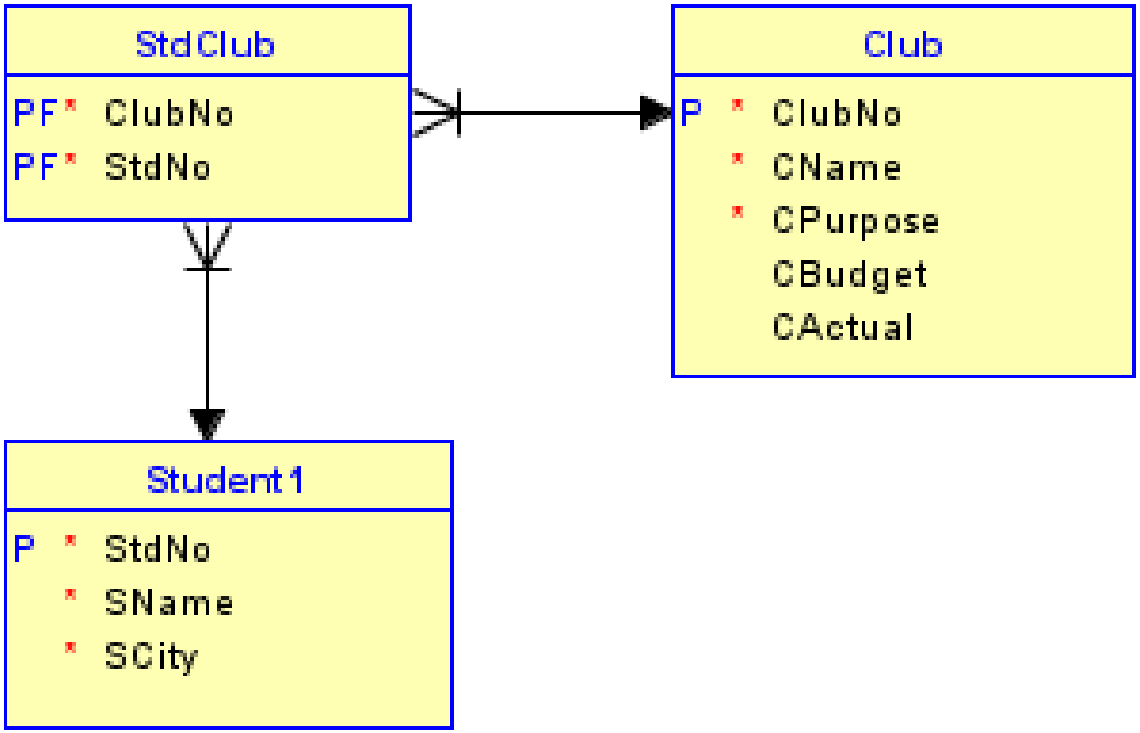
Module 12
Advanced Query Formulation with SQL

Lesson 4 (Part 2): Containment Exceptions in the SELECT statement

Lesson Objectives

- Define containment exception as related to the relational division operator
- Apply the text pattern to detect problem statements with containment exceptions
- Apply the statement pattern to write SELECT statements for containment exception problems
- Write SELECT statements for basic and extended containment exception problems

Simplified University Database Diagram



Sample Rows in the Containment Exception Tables

Student1

StdNo	SName	SCity
S1	JOE	SEATTLE
S2	SALLY	SEATTLE
S3	SUE	PORTLAND

StdClub

StdNo	ClubNo
S1	C1
S1	C2
S1	C3
S1	C4
S2	C1
S2	C4
S3	C3

Club

ClubNo	CName	CPurpose	CBudget	CActual
C1	DELTA	SOCIAL	1000.00	1200.00
C2	BITS	ACADEMIC	500.00	350.00
C3	HELPS	SERVICE	300.00	330.00
C4	SIGMA	SOCIAL	[null]	150.00

Overview of the Containment Exception Problem

- Derived from relational division operator with subset matching
- Containment exceptions with most entities not matching a subset of entities
- Most customers do not order all suggested items in the same shopping cart.

Text Pattern

- Containment adjective connecting prominent nouns
- Every or all typical containment adjectives
- Students who belong to all social clubs
- Students who have taken every finance course

Count Statement Pattern

```
SELECT <ChildTableColumns> [, <RelatedTableColumns>]
 FROM <ChildTable> [, <RelatedTables>]
 WHERE [<JoinConditions>]
   AND [<ChildTableConditions>]
   AND [<RelatedTableConditions>]
 GROUP BY <ChildTableColumns>,
          [<RelatedTableColumns>]
 HAVING COUNT(*) =
    ( SELECT COUNT(*)
       FROM <ParentTable>
       [ WHERE <ParentTableConditions> ] );
```

Basic Example

Example 1: List the student number of students who belong to all clubs.

```
SELECT StdNo
   FROM StdClub
   GROUP BY StdNo
   HAVING COUNT(*) =
     ( SELECT COUNT(*) FROM Club );
```

StdNo
S1

Extended Example

- Compare student club membership to an interesting subset of Club rows using identical conditions in the outer and nested queries
- Add columns of a related table

Example 2: List details of students who belong to all social clubs.

```
SELECT Student1.StdNo, SName
 FROM StdClub, Club, Student1
 WHERE StdClub.ClubNo = Club.ClubNo
   AND StdClub.StdNo = Student1.StdNo
   AND CPurpose = 'SOCIAL'
 GROUP BY Student1.StdNo
 HAVING COUNT(*) =
  ( SELECT COUNT(*) FROM Club
     WHERE CPurpose = 'SOCIAL' );
```

StdNo	SName
S1	JOE
S2	SALLY

Advanced Example

- COUNT(DISTINCT …) to eliminate duplicate courses
- Using the original university database tables

Example 3 : List the number and the name of faculty who teach at least one offering of all fall 2019, IS courses.

```
SELECT Faculty.FacNo, FacFirstName, FacLastName
 FROM Faculty, Offering
 WHERE Faculty.FacNo = Offering.FacNo
   AND OffTerm = 'FALL' AND CourseNo LIKE 'IS%'
   AND OffYear = 2019
 GROUP BY Faculty.FacNo, FacFirstName, FacLastName
 HAVING COUNT(DISTINCT CourseNo) =
  ( SELECT COUNT(DISTINCT CourseNo)
     FROM Offering
     WHERE OffTerm = 'FALL' AND OffYear = 2019
       AND CourseNo LIKE 'IS%' );
```

Summary

- Subset matching related to the relational division operator
- Most entities not matching an interesting subset of rows
- Text pattern with containment adjective connecting prominent nouns
- COUNT statement pattern with a Type I nested query in the HAVING clause
- Other statement patterns but more complex