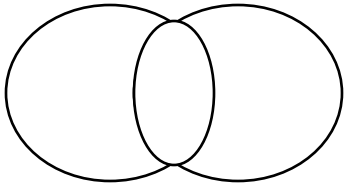
Chapter 9

Advanced Query Formulation with SQL

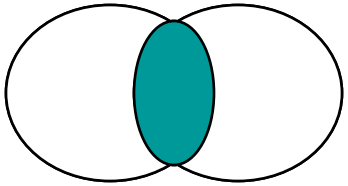
Outline

- Set Operations
- Outer join problems
- Nested queries and difference problems

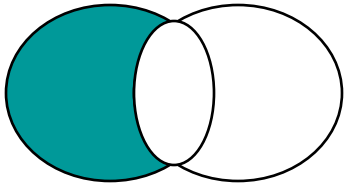
Traditional Set Operators



A UNION B



A INTERSECT B



A MINUS B

SQL UNION Example

Example 21: Retrieve basic data about all university people

```
SELECT FacSSN AS SSN, FacFirstName AS FirstName,
      FacLastName AS LastName
FROM Faculty
UNION
SELECT StdSSN AS SSN, StdFirstName AS FirstName,
      StdLastName AS LastName
FROM Student
```

Union

- Creates a table that combines rows from both SELECTs
- Both SELECTs must have same number of columns
- Each corresponding column must have compatible data type

Other Set Operations

- SQL Standard defines INTERSECT and MINUS operators
- Support for these varies

INTERSECT Example

Example 22: Show faculty who are also students.

```
SELECT FacSSN AS SSN, FacFirstName AS
       FirstName, FacLastName AS LastName
FROM Faculty
   INTERSECT
SELECT StdSSN AS SSN, StdFirstName AS
       FirstName, StdLastName AS LastName
FROM Student
```

If INTERSECT is not supported, it can be emulated using IN and EXISTS operators

Review: INNER JOIN

```
SELECT OfferNo, CourseNo,Offering.FacSSN,
        FacFirstName, FacLastName
  FROM Offering INNER JOIN Faculty
       ON Offering.FacSSN = Faculty.FacSSN
  WHERE CourseNo LIKE 'IS%'
```

OfferNo	CourseNo	FacSSN	FacFirstName	FacLastName
1234	IS320	98765432	LEONARD	VINCE
3333	IS320	98765432	LEONARD	VINCE
4321	IS320	98765432	LEONARD	VINCE
4444	IS320	543210987	VICTORIA	EMMANUEL

Outer Join Overview

- INNER JOIN excludes non matching rows
- Outer Joins include non matching rows
- Three types of Outer Joins
  - Full outer join
  - Left join
  - Right join
- LEFT JOIN is the most common

LEFT JOIN

```
SELECT OfferNo, CourseNo,Offering.FacSSN,
        FacFirstName, FacLastName
  FROM Offering LEFT JOIN Faculty
       ON Offering.FacSSN = Faculty.FacSSN
  WHERE CourseNo LIKE 'IS%'
```

OfferNo	CourseNo	FacSSN	FacFirstName	FacLastName
1111	IS320	*NULL*	*NULL*	*NULL*
1234	IS320	98765432	LEONARD	VINCE
2222	IS460	*NULL*	*NULL*	*NULL*
3333	IS320	98765432	LEONARD	VINCE
4321	IS320	98765432	LEONARD	VINCE

- LEFT JOIN yields all rows from the left-hand table
- RIGHT JOIN yields all rows from the right-hand table
- FULL OUTER JOIN yields all rows from both tables (rare)

Nested Queries

- Query inside a query
- Use in WHERE and HAVING conditions

Nested Query Examples I

Example 6: List faculty who teach IS courses.

```
SELECT FacSSN, FacLastName, FacDept
 FROM Faculty
 WHERE FacSSN IN
  ( SELECT FacSSN FROM Offering
     WHERE CourseNo LIKE 'IS%'  )
```

- Nested query used with IN must have a single column in SELECT
- Could we do this with a JOIN?

Alternate Formulation with Join

```
SELECT DISTINCT Faculty.FacSSN,
          FacLastName, FacDept
  FROM Faculty JOIN Offering ON
          Faculty.FacSSN = Offering.FacSSN
  WHERE CourseNo LIKE 'IS%'
```

Why is DISTINCT required here?

Nested Query Examples II

Example 7: List finance faculty who teach 4-unit IS courses.

```
SELECT FacSSN, FacLastName, FacDept
 FROM Faculty
 WHERE FacDept = 'FIN' AND FacSSN IN
  ( SELECT FacSSN FROM Offering
     WHERE CourseNo LIKE 'IS%' AND CourseNo IN
      ( SELECT CourseNo FROM Course
         WHERE CrsUnits = 4 ) )
```

Difference Problems

- Consider two sets of rows: A and B
- A – B = all of the rows in A that are not in B
- Example: Find all faculty who do not teach winter courses

Incorrect Difference Attempt

HOW NOT TO DO IT

Example: Retrieve faculty who are not teaching in 2006.

```
SELECT FacSSN, FacLastName, FacDept
 FROM Faculty INNER JOIN Offering
      ON Faculty.FacSSN = Offering.FacSSN
 WHERE OffYear <> 2006
                ???
```

Difference Example I

Example: Retrieve faculty who are not teaching in 2006.

```
SELECT FacSSN, FacLastName, FacDept
 FROM Faculty
 WHERE FacSSN NOT IN
  ( SELECT FacSSN FROM Offering
     WHERE OffYear = 2006 )
```

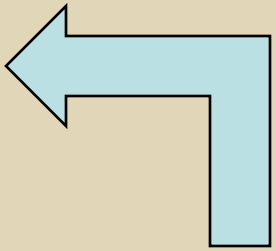
Other Formulations for Difference Problems

- NOT EXISTS Query
- LEFT JOIN with IS NULL condition

Meet EXISTS

- For each row in the outer query, executes the nested query
- If nested query returns any rows, includes the row in the outer query in the results
- Tip: Properly written query that uses EXISTS: Nested query should reference table in outer query in WHERE

```
SELECT *
FROM product
WHERE EXISTS (SELECT *
              FROM ordline
              WHERE ordline.prodno = product.prodno
              AND   qty > 1)
```



NOT EXISTS
Example for a Difference Problem

Example 9: Faculty who are not teaching in winter 2008.

```
SELECT FacSSN, FacLastName, FacDept
 FROM Faculty
 WHERE NOT EXISTS
  ( SELECT * FROM Offering
     WHERE OffTerm = 'WINTER'
       AND OffYear = 2008
       AND Faculty.FacSSN = Offering.FacSSN )
```

Nested SELECT executes one time for each row of outer SELECT

Left Join Difference Formulation

Example 11: Retrieve MS faculty who have never taught a course (research faculty).

```
SELECT FacSSN, FacLastName, FacDept
 FROM Faculty LEFT JOIN Offering
   ON Faculty.FacSSN = Offering.FacSSN
 WHERE FacDept = 'MS'
   AND Offering.FacSSN IS NULL
```

Summary: 3 ways to do Difference queries

- NOT IN
  - select * from product
    where prodno not in (select prodno from orderline)
- NOT EXISTS
  - select * from product
    where not exists (
      select * from orderline where
        orderline.prodno = product.prodno)
- LEFT JOIN
  - select *
    from product left join ordline on
      product.prodno = ordline.prodno
    where ordline.prodno is null

Summary

- Advanced matching problems not common but important when necessary
- Understand outer join, difference, and division operators
- Nested queries important for advanced matching problems
- Lots of practice to master query formulation and SQL

Mixing Inner and Outer Joins

```
SELECT OfferNo, Offering.CourseNo, OffTerm,
        CrsDesc, Faculty.FacSSN, FacLastName
  FROM ( Faculty RIGHT JOIN Offering
       ON Offering.FacSSN = Faculty.FacSSN )
    INNER JOIN Course
       ON Course.CourseNo = Offering.CourseNo
  WHERE Course.CourseNo LIKE 'IS%'
```