DBMS/SQL

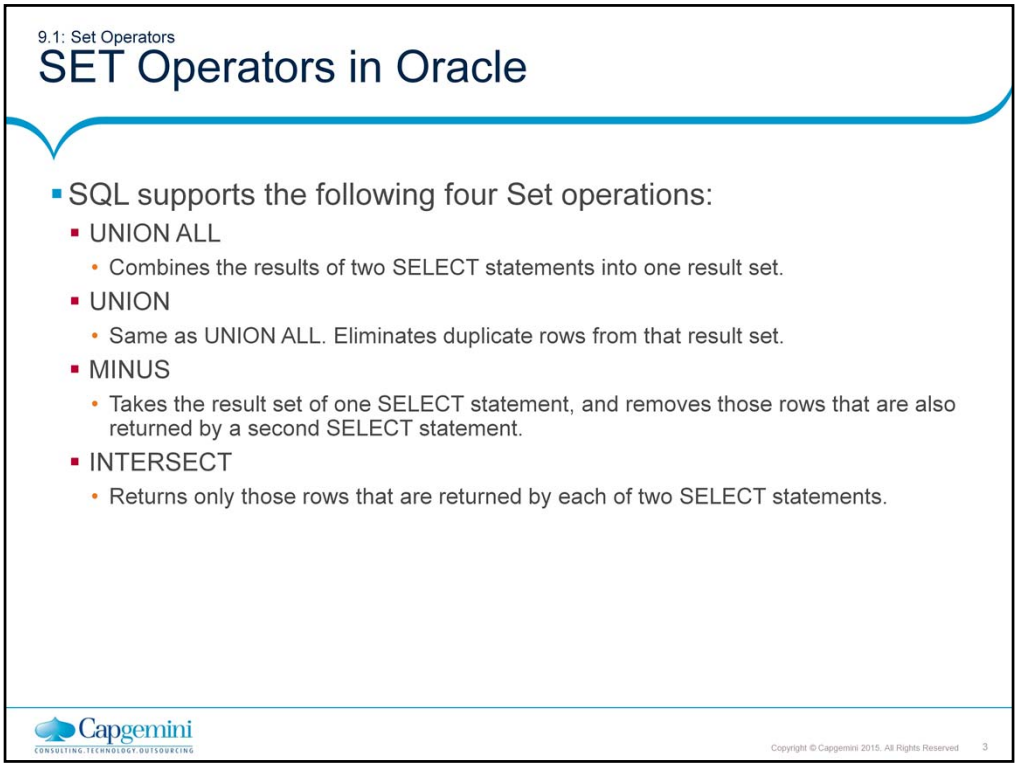
Lesson 09: Set Operators

Lesson Objectives

- To understand the following topics:
  - Set Operators
  - UNION operator
  - INTERSECT operator
  - MINUS operator
  - Tips and Tricks

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 2



Set Operators:

There are situations when we need to “combine the results” from two or more SELECT statements. SQL enables us to handle these requirements by using “Set operations”.

The result of each SELECT statement can be treated as a set. SQL set operations can be applied on these sets to arrive at a final result.

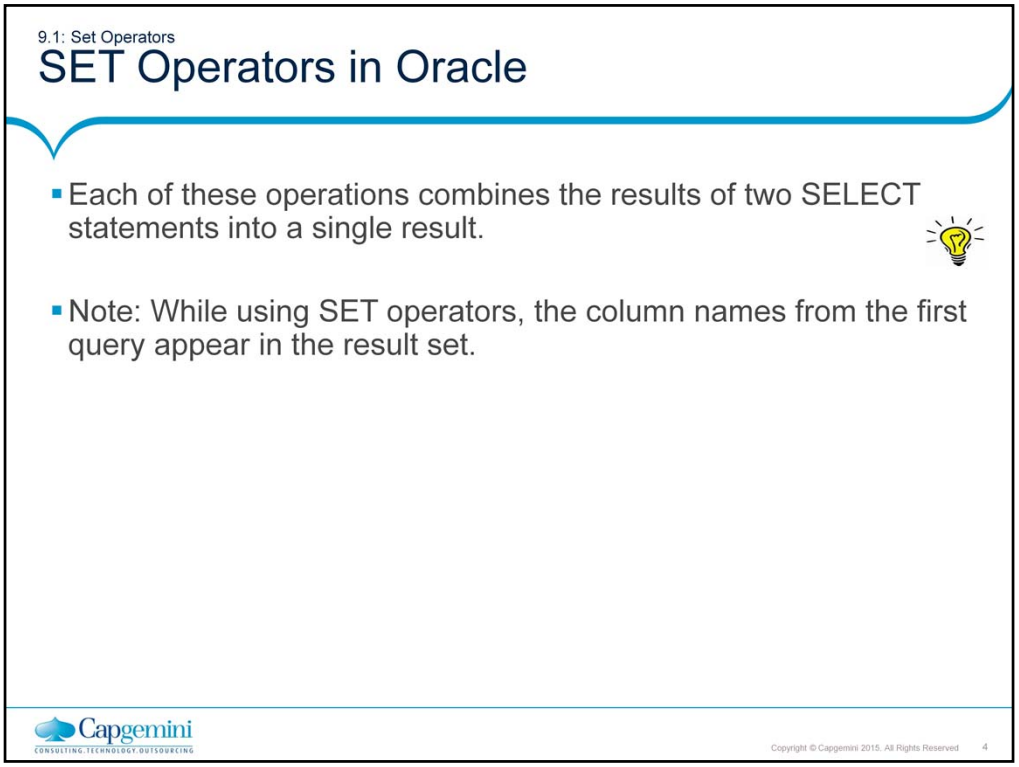
SQL supports the following four Set operations:

- UNION ALL
- UNION
- MINUS
- INTERSECT

All set operators have equal precedence.

If a SQL statement contains multiple set operators, Oracle evaluates them from the left to right if there are no parentheses explicitly specifying another order.

contd.



SQL statements containing the Set operators are referred to as “compound queries”. Each SELECT statement in a compound query is referred to as a “component query”.

Two SELECT statements can be combined into a compound query by a set operation only if they satisfy the following two conditions:

The “result sets” of both the queries must have the “same number of columns”.

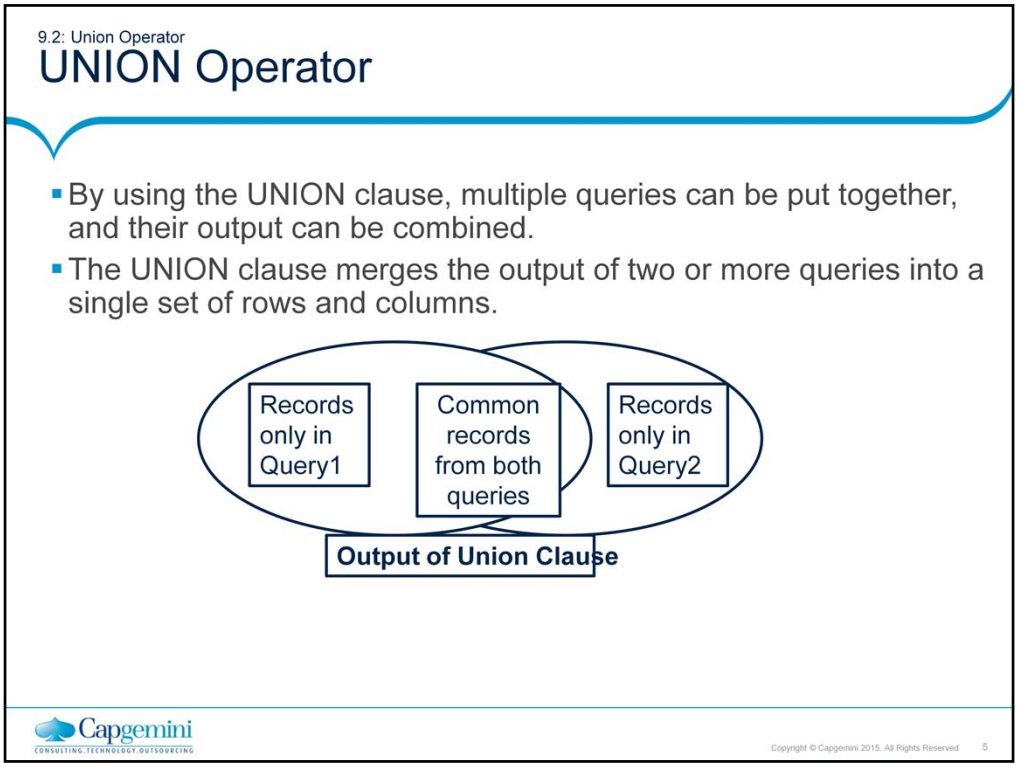
The “datatype” of each column in the “second result set” must match the “datatype” of its corresponding column in the “first result set”.

For example: If component queries select character data, then the datatype of the return values are determined as follows:

If both queries select values of datatype CHAR, then the returned values have datatype CHAR.

If either or both of the queries select values of datatype VARCHAR2, then the returned values have datatype VARCHAR2.

Tip: The datatypes do not need to be the same, if those in the second result set can be automatically converted by Oracle (using implicit casting) to types that are compatible with those in the first result set.



UNION Operator:

The UNION operator returns the records retrieved by either of the queries.

By default, the UNION operator eliminates duplicate records.

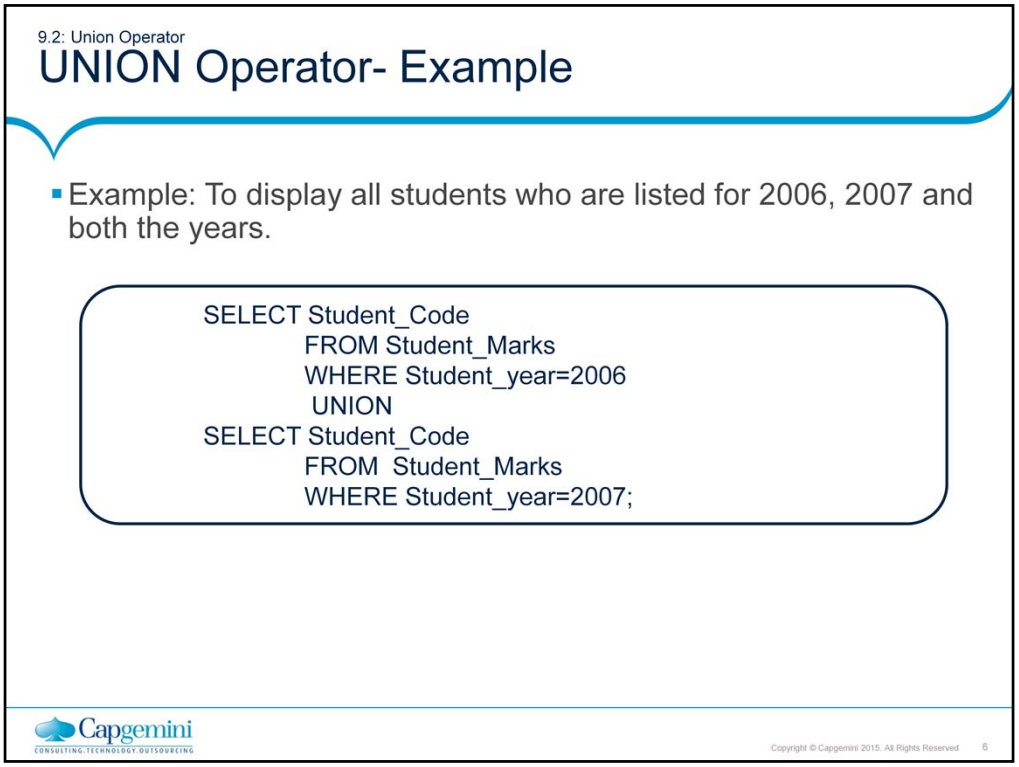
If however we want to retain duplicates, we use UNION ALL instead of UNION.

UNION operates over all of the columns being selected.

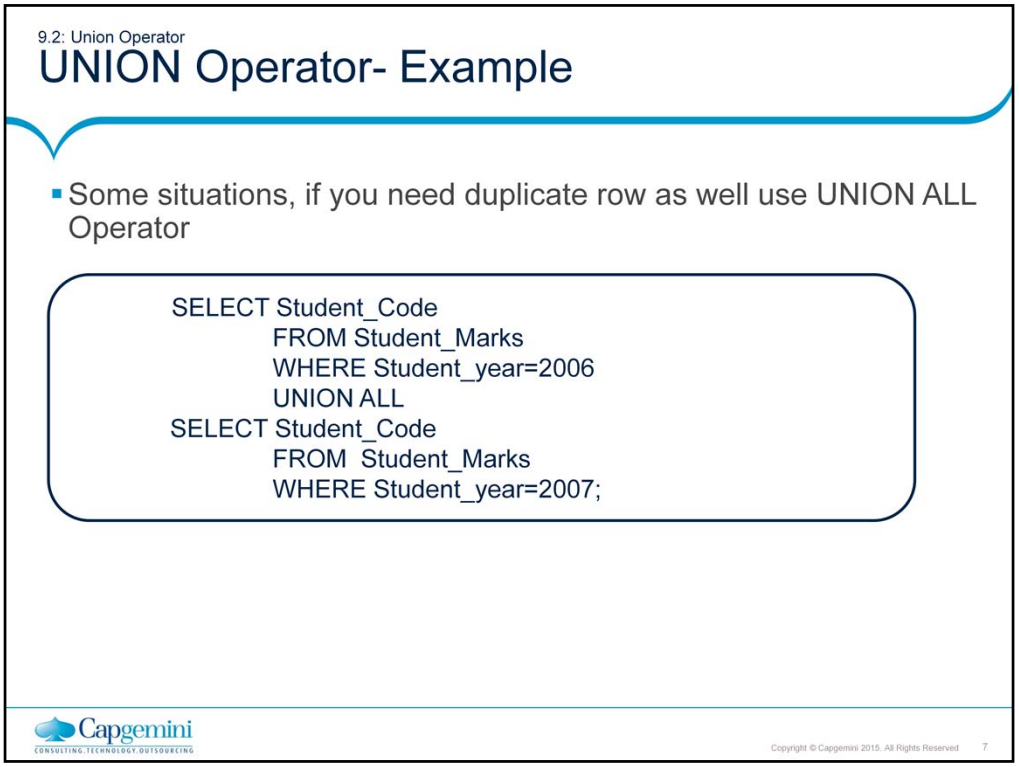
NULL values are not ignored during duplicate checking.

The IN operator has a higher precedence than the UNION operator.

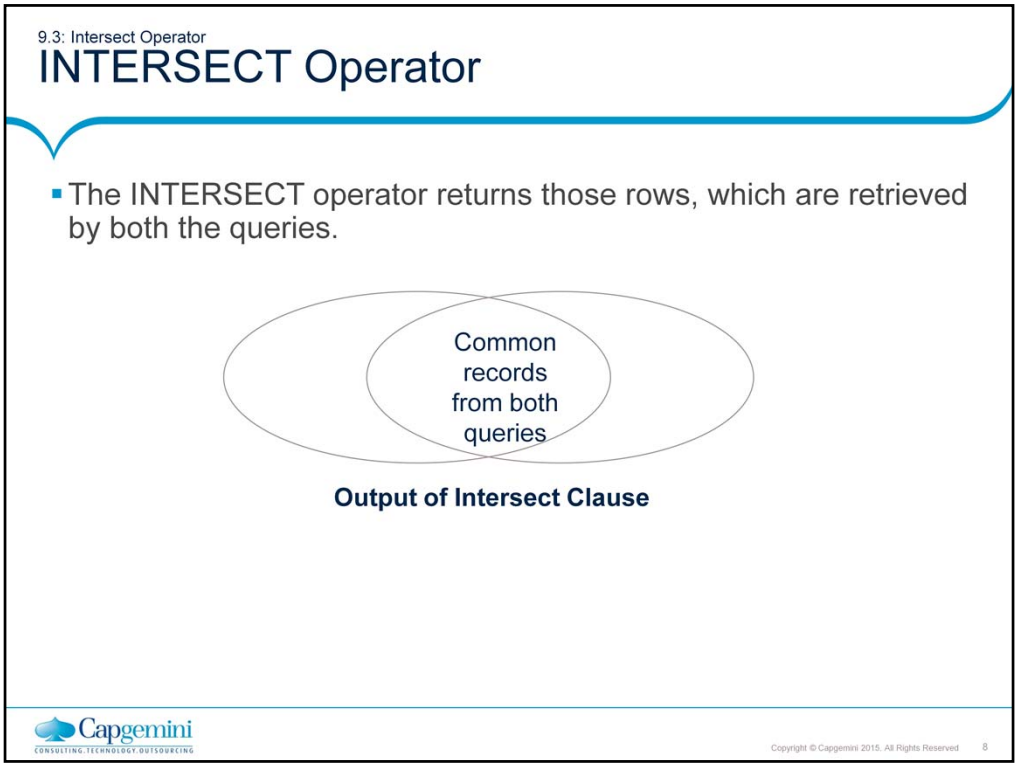
By default, the output is sorted in ascending order of the first column of the SELECT clause.



The query on the slide retrieves unique values from first query and second query. It also retrieves common values in both the queries by removing the duplicate records.



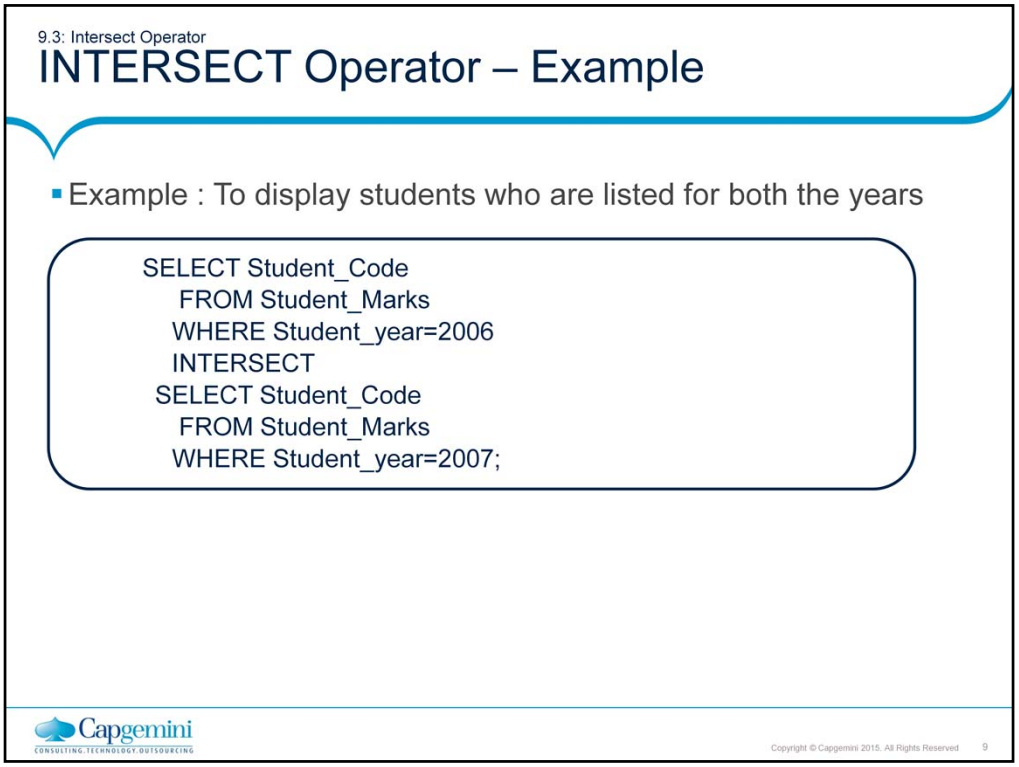
The above query since is using UNION ALL will remove results returned by both the queries. The common values will be duplicate.



INTERSECT Operator

INTERSECT result is same on reversing the order

INTERSECT does not ignore NULL values



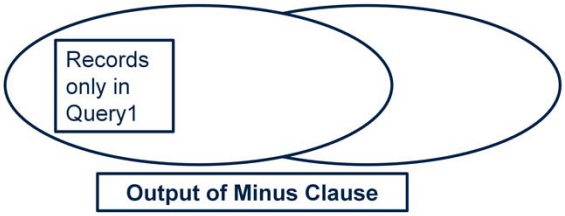
Example of INTERSECT operator:

The example on the slide will only display the common records retrieved by both the queries

9.4: Minus Operator

MINUS Operator

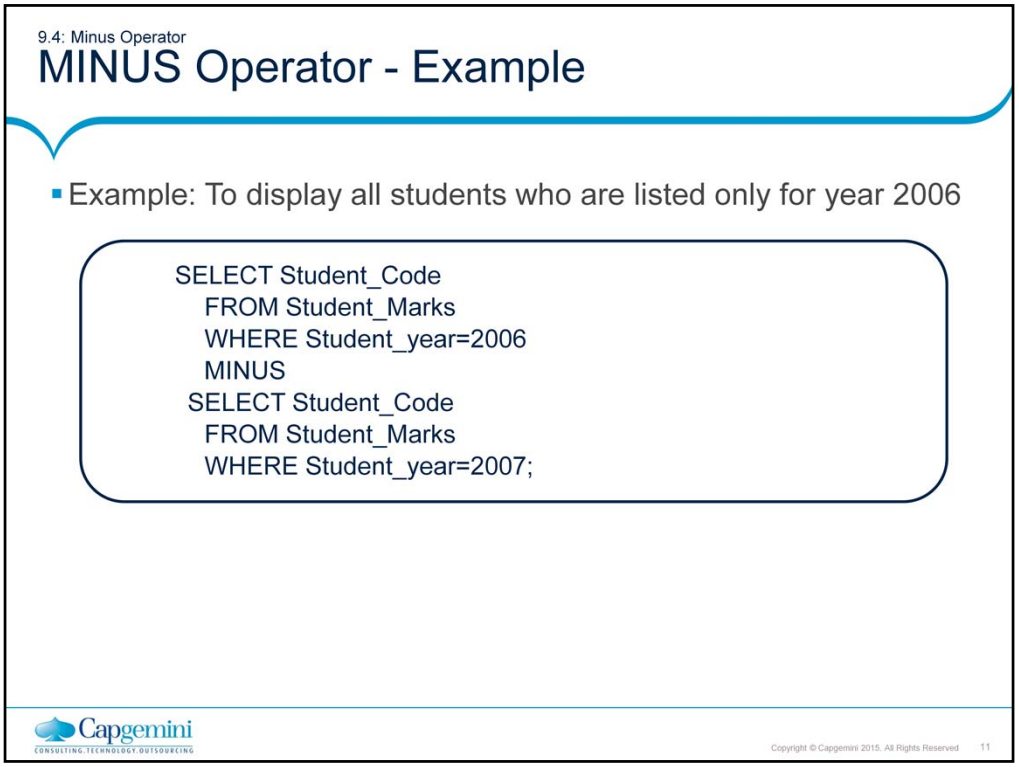
- The MINUS operator returns all rows retrieved by the first query but not by the second query.



Output of Minus Clause

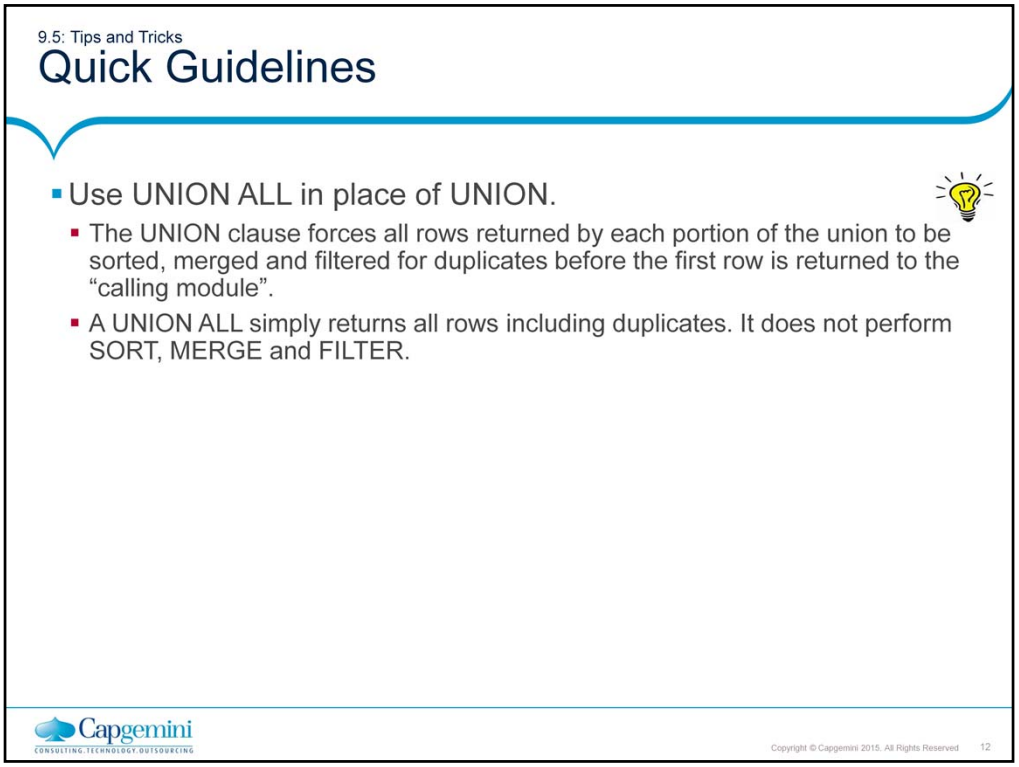
Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 10



Example of MINUS operator:

The query on the slide will show results which are unique to first query



Tips and Tricks:

When using the UNION statement, keep in mind, that the UNION statement performs the equivalent of a SELECT DISTINCT on the final result set, by default.

In other words, UNION takes the results of two like record sets, combines them, and then performs a SELECT DISTINCT in order to eliminate any duplicate rows.

> This process occurs even if there are no duplicate records in the final record set.

Hence,

> If you know that there are duplicate records, and it creates a problem for your application, then use the UNION statement "to eliminate the duplicate rows".
>
> If you know that there will never be any duplicate rows, or if there are duplicates, and it does not create problems in your application, then you should use the UNION ALL statement instead of the UNION statement.

The advantage of the UNION ALL statement is that is does not perform the SELECT DISTINCT function. This saves a lot of server resources from being unnecessarily used.

Summary

- In this lesson you have learnt,
  - Use of Set Operations
  - Use of UNION operator
    - Advantage of UNION ALL over UNION operator
  - Use of INTERSECT operator
  - Use of MINUS operator

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 13

Review – Questions

- Question 1: The Set operation that will show all the rows from both the resultsets including duplicates is ____.
  - Option 1: Union All
  - Option 2: Union
  - Option 3: Intersect
  - Option 4: Minus

- Question 2: The Intersect operator returns ___.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 14

Review – Questions

- Question 3: The output of set operators shows the columns names from ___.
- Question 4: The Union ALL clause performs SORT, MERGE and FILTER to give the required output.
  - True / False