Set Operations

CSC 256, SQL Programming

Basic Set Theory Review

- A set is well-defined, unordered collection of objects
- The objects in a set are called the elements, or members, of the set
- A set is said to contain its elements
- The notation $a \in A$ denotes that a is an element of the set A
- The notation $a \notin A$ denotes that a is not an element of the set A

Set Operators

- Union: The union of the sets A and B, denoted by $A \cup B$, is the set of elements in A OR in B.
- Intersection: The intersection of sets A and B, denoted by $A \cap B$, is the set of elements that are in A AND in B.
- Difference: The difference of A and B, denoted by $A - B$, is the set of elements in A not including any elements that are also in B.

SQL Set Operators

- Set operators in (ANSI) SQL:
  - `UNION`
  - `INTERSECT`
  - `EXCEPT`
- Set operators work on tables; the columns must have the same number and type (cardinality), but names do not need to be the same. If the column names are not the same, the result uses the names from the first table.

UNION

- UNION returns all the rows of both tables with duplicates removed.
- Example:

```
( select n
  from (values (1), (1), (2)) as v(n) )
union
( select n
  from (values (2), (3), (4)) as v(n) );

 n
---
 1
 3
 4
 2
```

Example: Different Column Names

```
( select a, b
  from (values (1, 2), (1, 5), (3, 4)) as v(a, b) )
union
( select c, d
  from (values (1, 5), (2, 3), (4, 5)) as v(c, d) );

 a | b
---+---
 1 | 5
 1 | 2
 2 | 3
 4 | 5
 3 | 4
```

UNION ALL

- UNION ALL returns all the rows of both tables (including duplicates)
- Example

```
( select n
  from (values (1), (1), (2)) as v(n) )
union all
( select n
  from (values (2), (3)) as v(n) );

 n
---
 1
 1
 2
 2
 3
```

INTERSECT

- INTERSECT return rows common to both tables
- Example

```
( select n
  from (values (1), (1), (3)) as v(n) )
intersect
( select n
  from (values (1), (2), (4)) as v(n) );

 n
---
 1
```

INTERSECT ALL

- INTERSECT ALL returns matching rows in the input and output using a one-to-one correspondence; this is possibly non-intuitive behavior.
- Example:

```
( select n
  from (values (1), (1), (3)) as v(n) )
intersect all
( select n
  from (values (1), (1), (4)) as v(n) );

 n
---
 1
 1
```

EXCEPT

- EXCEPT is the set difference operator
- Example

```
( select n
  from (values (1), (1), (3)) as v(n) )
except
( select n
  from (values (1), (7), (8)) as v(n));

 n
---
 3
```

EXCEPT ALL

- EXCEPT ALL returns the set difference of matching rows in the input and output using a one-to-one correspondence; this is possibly non-intuitive behavior.
- Example

```
( select n
  from (values (1), (1), (3)) as v(n) )
except all
( select n
  from (values (1), (7), (8)) as v(n) );

 n
---
 1
 3
```