Chapter 4 COMBINING TABLES VERTICALLY

USING PROC SQL

PROC SQL may be used to select data from multiple tables and to combine the tables vertically (that is, pasting one after another). The following four set operations are distinguished:

* EXCEPT (selects unique rows from the first table that are not found in the second table)

\*\*\*\*\*\*\*

\*\*\* \*\*\*\*\*\*\*\*

\*\* \*\*\*

Syntax

proc sql;

select \*

from *table1*

except

select \*

from *table2*;

quit;

* INTERSECT (selects unique rows that are common to both tables)

\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*

Syntax

proc sql;

select \*

from *table1*

intersect

select \*

from *table2*;

quit;

* UNION (selects unique rows from one or both tables)

\*\*\*\*\*\*\*

\*\*\* \*\*\*\*\*\*\*\*

\*\* \*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*

Syntax

proc sql;

select \*

from *table1*

union

select \*

from *table2*;

quit;

* OUTER UNION (selects all rows from both tables)

\*\*\*\*\*\*\*

\*\*\* \*\*\*\*\*\*\*\*

\*\* \*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*

Syntax

proc sql;

select \*

from *table1*

outer union

select \*

from *table2*;

quit;

Example. Consider two sets

Six

|  |  |
| --- | --- |
| id | varB |
| 1 | x |
| 2 | y |
| 3 | z |
| 3 | v |
| 5 | w |

Five

|  |  |
| --- | --- |
| id | varA |
| 1 | a |
| 1 | a |
| 1 | b |
| 2 | c |
| 3 | v |
| 4 | e |
| 6 | g |

Below are queries for the four set operators and their relative outputs.

1. Set operator EXCEPT

data five;

input id varA $ @@;

datalines;

1 a 1 a 1 b 2 c 3 v 4 e 6 g

;

data six;

input id varB $ @@;

datalines;

1 x 2 y 3 z 3 v 5 w

;

options nodate nonumber;

title;

proc sql;

select \*

from five

except

select \*

from six;

quit;

The output is

id varA

ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ

1 a

1 b

2 c

4 e

6 g

1. Set operator INTERSECT

proc sql;

select \*

from five

intersect

select \*

from six;

quit;

The output is

id varA

ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ

3 v

1. Set operator UNION

proc sql;

select \*

from five

union

select \*

from six

union

select \* from seven

union select \* from eight

quit;

The output is

id varA

ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ

1 a

1 b

1 x

2 c

2 y

3 v

3 z

4 e

5 w

6 g

1. Set operator OUTER UNION

proc sql;

select \*

from five

outer union

select \*

from six;

quit;

The output is

id varA id varB

ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ

1 a .

1 a .

1 b .

2 c .

3 v .

4 e .

6 g .

. 1 x

. 2 y

. 3 z

. 3 v

. 5 w

* To output *all* duplicated rows, use the keyword ALL after EXCEPT, INTERSECT, or UNION.

proc sql;

select \*

from five

except all

select \*

from six;

quit;

The output is

id varA

ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ

1 a

1 a

1 b

2 c

4 e

6 g