

# SQL Processing with SAS® Tip Sheet

This tip sheet is associated with the SAS® Certified Professional Prep Guide Advanced Programming Using SAS® 9.4. For more information, visit [www.sas.com/certify](http://www.sas.com/certify)

## Basic Queries

```
PROC SQL <options>;
SELECT column-1 <, ...column-n>
FROM input-table
<WHERE expression>
<GROUP BY col-name>
<HAVING expression>
<ORDER BY col-name> <DESC> <,...col-name>;
```

SQL Query Order of Execution:

Clause	Description
SELECT	Retrieve data from a table
FROM	Choose and join tables
WHERE	Filter the data
GROUP BY	Aggregate the data
HAVING	Filter the aggregate data
ORDER BY	Sort the final data

## Managing Tables

CREATE TABLE	<b>CREATE TABLE</b> <i>table-name</i> ( <i>column-specification-1</i> <, ... <i>column-specification-n</i> >);
DESCRIBE TABLE	<b>DESCRIBE TABLE</b> <i>table-name-1</i> <,... <i>table-name-n</i> >;
DROP TABLE	<b>DROP TABLE</b> <i>table-name-1</i> <,... <i>table-name-n</i> >;

## Managing Views

CREATE VIEW	<b>CREATE VIEW</b> <i>table-name</i> <b>AS</b> <i>query</i> ;
DESCRIBE VIEW	<b>DESCRIBE VIEW</b> <i>view-name-1</i> <,... <i>view-name-n</i> >;
DROP VIEW	<b>DROP VIEW</b> <i>view-name-1</i> <,... <i>view-name-n</i> >;

## Modifying Columns

LABEL=	<b>SELECT</b> <i>col-name</i> <b>LABEL=</b> ' <i>column label</i> '
FORMAT=	<b>SELECT</b> <i>col-name</i> <b>FORMAT=</b> <i>format</i> .
Creating a new column	<b>SELECT</b> <i>col-name</i> <b>AS</b> <i>new-col-name</i>
Filtering new columns	<b>WHERE CALCULATED</b> <i>new-col-name</i>

## Modifying Rows

Inserting rows into tables	<b>INSERT INTO</b> <i>table</i> <b>SET</b> <i>column-name=value</i> <,... <i>column-name=value</i> >;  <b>INSERT INTO</b> <i>table</i> <(column-list)> <b>VALUES</b> ( <i>value</i> <,... <i>value</i> >);  <b>INSERT INTO</b> <i>table</i> <(column-list)> <b>SELECT</b> <i>column-1</i> <,... <i>column-n</i> > <b>FROM</b> <i>input-table</i> ;
Eliminating duplicate rows	<b>SELECT DISTINCT</b> <i>col-name</i> <,... <i>col-name</i> >
Filtering rows	<b>WHERE</b> <i>col-name</i> <b>IN</b> ( <i>value1</i> , <i>value2</i> , ...) <b>WHERE</b> <i>col-name</i> <b>LIKE</b> "_ <i>string</i> %" <b>WHERE</b> <i>col-name</i> <b>BETWEEN</b> <i>value</i> <b>AND</b> <i>value</i> <b>WHERE</b> <i>col-name</i> <b>IS NULL</b> <b>WHERE</b> <i>date-value</i> "<01JAN2019>" <b>d</b> <b>WHERE</b> <i>time-value</i> "<14:45:35>" <b>t</b> <b>WHERE</b> <i>datetime-value</i> "<01JAN201914:45:35>" <b>dt</b>

## Remerging Summary Statistics

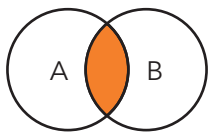
```
SELECT col-name, summary function(argument)
FROM input table;
```

# SQL Processing with SAS® Tip Sheet

This tip sheet is associated with the SAS® Certified Professional Prep Guide Advanced Programming Using SAS® 9.4. For more information, visit [www.sas.com/certify](http://www.sas.com/certify)

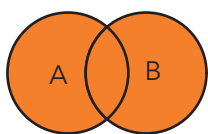
## Joins Summary

Inner Join



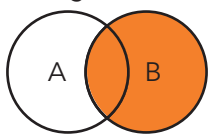
```
SELECT <list>  
FROM table-A INNER JOIN table-B  
ON A.Key=B.Key;
```

Full Join



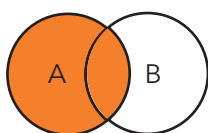
```
SELECT <list>  
FROM table-A FULL JOIN table-B  
ON A.Key=B.Key;
```

Right Join



```
SELECT <list>  
FROM table-A RIGHT JOIN table-B  
ON A.Key=B.Key;
```

Left Join



```
SELECT <list>  
FROM table-A LEFT JOIN table-B  
ON A.Key=B.Key;
```

## Creating Macro Variables

Storing a value in a macro variable using SQL:

```
SELECT col-name-1 <,...col-name-n>  
INTO:macvar_1<,...macvar-n>  
FROM input-table;
```

Storing a list of values in a macro variable using SQL:

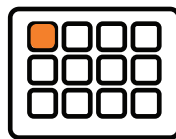
```
SELECT col-name-1 <,...col-name-n>  
INTO:macvar_1 SEPARATED BY 'delimiter'  
FROM input-table;
```

Viewing the value of the macro variable in the SAS Log:

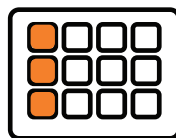
```
%PUT &=macvar;
```

## Subqueries

```
SELECT col-name,  
       (SELECT function(argument)  
        FROM input-table)  
FROM input-table;
```

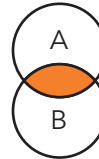


```
SELECT col-name, <,...col-name>  
FROM input-table  
WHERE col-name  
      (SELECT function(argument)  
       FROM input-table)
```



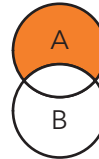
## Set Operators

The INTERSECT operator selects unique rows that are common to both tables.



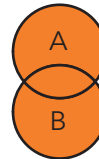
```
SELECT <list>  
FROM table-A INTERSECT  
SELECT <list>  
FROM table-B;
```

The EXCEPT operator selects unique rows from table A that are not found in table B.



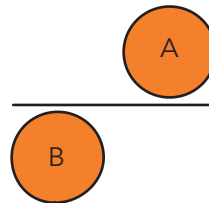
```
SELECT <list>  
FROM table-A EXCEPT  
SELECT <list>  
FROM table-B;
```

The UNION operator selects unique rows from both tables.



```
SELECT <list>  
FROM table-A UNION  
SELECT <list>  
FROM table-B;
```

The OUTER UNION operator selects all rows from both tables.



```
SELECT <list>  
FROM table-A OUTER UNION  
SELECT <list>  
FROM table-B;
```

## Accessing DBMS Data

The SQL pass-through facility enables you to code in the native DBMS SQL syntax and pass the query to the database.

```
PROC SQL;  
CONNECT TO DBMS-name <AS alias>  
           (DBMS-connection-options);
```

```
SELECT col-name  
FROM CONNECTION TO DBMS-name|alias (dbms-query);  
DISCONNECT FROM DBMS-name|alias;  
QUIT;
```