Chapter 5

Creating and Managing Views
Using PROC SQL
What is a View?

- A view is a stored query that is executed when the view is used.
- The view does not contain the data, just the logic for accessing the data.
- Views can be used in a SAS procedure, data step, or function.
- Views can be joined with tables or other views.
Why Are Views Useful

- Often save space
- Ensure input data sets are always current, because data is derived from tables at execution time
- Hide confidential columns (e.g., SSN) while allowing access to view other columns in the same table
## Computer Exercise 1

<table>
<thead>
<tr>
<th>SSN</th>
<th>Major Code</th>
<th>Degree</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>123-11-5526</td>
<td>135</td>
<td>Bachelors</td>
<td>9</td>
</tr>
<tr>
<td>206-67-1454</td>
<td>135</td>
<td>Bachelors</td>
<td>10</td>
</tr>
<tr>
<td>323-50-2746</td>
<td>126</td>
<td>Masters</td>
<td>9</td>
</tr>
<tr>
<td>367-47-9580</td>
<td>126</td>
<td>Bachelors</td>
<td>9</td>
</tr>
<tr>
<td>564-94-6508</td>
<td>126</td>
<td>Bachelors</td>
<td>9</td>
</tr>
<tr>
<td>782-57-8284</td>
<td>135</td>
<td>Masters</td>
<td>10</td>
</tr>
<tr>
<td>354-83-9603</td>
<td>126</td>
<td>Masters</td>
<td>10</td>
</tr>
<tr>
<td>590-71-6170</td>
<td>135</td>
<td>Bachelors</td>
<td>8</td>
</tr>
<tr>
<td>060-76-2046</td>
<td>126</td>
<td>Bachelors</td>
<td>9</td>
</tr>
<tr>
<td>983-13-6455</td>
<td>135</td>
<td>Masters</td>
<td>9</td>
</tr>
<tr>
<td>638-03-0350</td>
<td>135</td>
<td>Bachelors</td>
<td>9</td>
</tr>
<tr>
<td>861-59-4706</td>
<td>126</td>
<td>Bachelors</td>
<td>9</td>
</tr>
<tr>
<td>SSN</td>
<td>Major Code</td>
<td>Degree</td>
<td>HW 1</td>
</tr>
<tr>
<td>-----------</td>
<td>------------</td>
<td>-----------</td>
<td>------</td>
</tr>
<tr>
<td>367-47-9580</td>
<td>126</td>
<td>Bachelors</td>
<td>19</td>
</tr>
<tr>
<td>564-94-6508</td>
<td>126</td>
<td>Bachelors</td>
<td>19</td>
</tr>
<tr>
<td>060-76-2047</td>
<td>126</td>
<td>Bachelors</td>
<td>17</td>
</tr>
<tr>
<td>123-11-5526</td>
<td>135</td>
<td>Bachelors</td>
<td>20</td>
</tr>
<tr>
<td>206-67-1454</td>
<td>135</td>
<td>Bachelors</td>
<td>19</td>
</tr>
<tr>
<td>638-03-0350</td>
<td>135</td>
<td>Bachelors</td>
<td>19</td>
</tr>
<tr>
<td>861-59-4706</td>
<td>126</td>
<td>Bachelors</td>
<td>20</td>
</tr>
<tr>
<td>983-13-6455</td>
<td>135</td>
<td>Masters</td>
<td>19</td>
</tr>
<tr>
<td>782-57-8284</td>
<td>135</td>
<td>Masters</td>
<td>19</td>
</tr>
<tr>
<td>354-83-9603</td>
<td>126</td>
<td>Masters</td>
<td>19</td>
</tr>
<tr>
<td>590-71-6170</td>
<td>135</td>
<td>Bachelors</td>
<td>18</td>
</tr>
</tbody>
</table>
Creating a View

proc sql;
create view viewname as
select column1, column2,..., columnn
from table1
<where expression>
<group by column1,...,columnn>
<having expression>
<order by column1,...,columnn>
*The where, group by, having, and order by clauses are optional.
Creating a View

Inner view:

```sql
select cel.code label=“Major Code”, cel.degree label=“Degree”, cel.grade as cel label=“Computer Exercise 1”, hw1.grade as hw1 label=“Homework 1” from cel left join hw1 on cel.ssn=hw1.ssn
```
Creating a View

Outer clause:

```sql
proc sql;
select code, avg(ce1) as ceavg
  label="CE 1" format=4.1, avg(hw1)
  label="HW 1" format=5.1
from . . .
group by code;
quit;
```
Creating a View

Create and execute the view:

```
proc sql;
create view Major_Comp as . . . .;
quit;
```

```
proc sql;
select * Major_Comp;
quit;
```
Using a View

Views can also be used in PROC steps:

```sql
proc sql;
create view JoinGrades as select
  ce1.code label="Major Code",
  ce1.degree label="Degree",
  ce1.grade as ce1 label="Computer Exercise 1",
  hw1.grade as hw1 label="Homework 1"
from ce1 left join hw1 on
  ce1.ssn=hw1.ssn
order by code;quit;
```
Views can also be used in PROC steps:

```
proc means data=JoinGrades
   maxdec=2; by code;
var cel hw1;
run;
```
Describe View Statement

- Use a DESCRIBE VIEW statement to display the definition of a view in the SAS log.

```
proc sql;
   describe view Major_Comp JoinGrades;
quit;
```
Guidelines for Using Views

- Avoid using the ORDER BY clause in a view definition, otherwise the data will have to be sorted each time the view is executed.
- It is more efficient to create a table if the same data is used many times in one program.
Guidelines for Using Views

- Avoid creating views that are based on tables whose structure (e.g., columns in table) may change
- Specify a one level name (e.g., claims, not work.claims) in the FROM clause if a view resides in the same SAS library as the contributing table(s)
USING LIBNAME

- References to a single-level table name assume the table is in the same library as the view
- USING LIBNAME can be appended to the CREATE VIEW clause to resolve confusion in table references
USING LIBNAME

proc sql;
create view viewname as
select * from libname.table1
using libname 'directory';
Updating a View

- Underlying tables can be updated with UPDATE, INSERT, and DELETE

- Limitations
  - Only a single table can be updated
  - Views with WHERE clauses can be updated
  - Views with ORDER BY, HAVING or GROUP BY cannot be updated
Dropping a View

- To drop (delete) a view, use the drop view statement.

```sql
proc sql;
  drop view viewname;
quit;
```