Chapter 10: Combining Data Sets

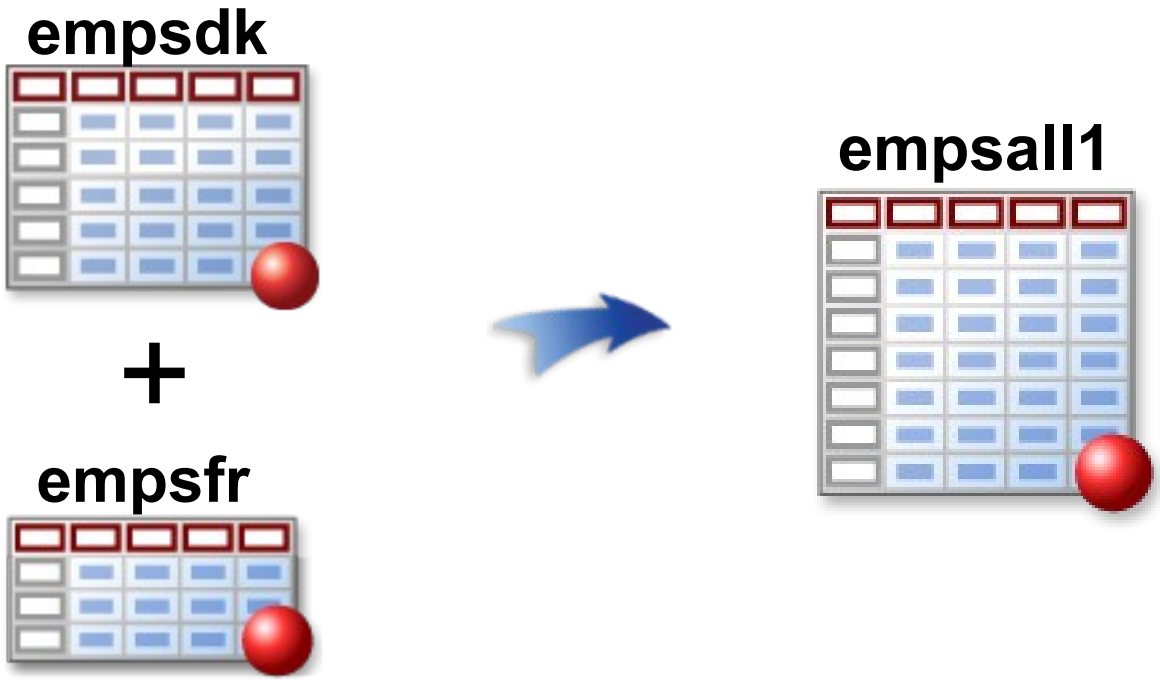
Chapter 10: Combining Data Sets

Objectives

- Concatenate two or more SAS data sets using the SET statement in a DATA step.
- Change the names of variables using the RENAME= data set option.

Business Scenario

You have been asked to combine the data sets containing information about Orion Star employees from Denmark and France into a new data set.



Considerations

Concatenate like-structured data sets **empsdk** and **empsfr** to create a new data set named **empsall1**.

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

empsall1

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark
Pierre	M	France
Sophie	F	France

Both data sets contain the same variables.

Using a DATA Step

Use a DATA step to concatenate the data sets. List the data sets in the SET statement.

```
data empsall1;
   set empsdk empsfr;
run;
```

SET *SAS-data-set1 SAS-data-set2 . . .*;

- The SET statement reads observations from each data set in the order in which they are listed.
- Any number of data sets can be included in the SET statement.

Compilation

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
   set empsdk empsfr;
run;
```

PDV

First	Gender	Country

empsall1

First	Gender	Country

p110d01

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

Initialize PDV

```
data empsall1;
    set empsdk empsfr;
run;
```

PDV

First	Gender	Country

empsall1

First	Gender	Country

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
    set empsdk empsfr;
run;
```

PDV

First	Gender	Country
Lars	M	Denmark

empsall1

First	Gender	Country

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
    set empsdk empsfr;
run;
```

Implicit OUTPUT;
Implicit RETURN;

PDV

First	Gender	Country
Lars	M	Denmark

→

empsall1

First	Gender	Country
Lars	M	Denmark

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
    set empsdk empsfr;
run;
```

PDV

First	Gender	Country
Lars	M	Denmark

empsall1

First	Gender	Country
Lars	M	Denmark

Data set variables are not reinitialized.

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
    set empsdk empsfr;
run;
```

PDV

First	Gender	Country
Kari	F	Denmark

empsall1

First	Gender	Country
Lars	M	Denmark

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
    set empsdk empsfr;
run;
```

Implicit OUTPUT;
Implicit RETURN;

PDV

First	Gender	Country
Kari	F	Denmark

empsall1

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
    set empsdk empsfr;
run;
```

PDV

First	Gender	Country
Kari	F	Denmark

empsall1

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark

PDV is not reinitialized.

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
    set empsdk empsfr;
run;
```

PDV

First	Gender	Country
Jonas	M	Denmark

empsall1

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
    set empsdk empsfr;
run;
```

Implicit OUTPUT;
Implicit RETURN;

PDV

First	Gender	Country
Jonas	M	Denmark

empsall1

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
    set empsdk empsfr;
run;
```

PDV

First	Gender	Country
Jonas	M	Denmark

PDV is not reinitialized.

empsall1

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
nas	M	Denmark

EOF

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
    set empsdk empsfr;
run;
```

PDV

First	Gender	Country
Jonas	M	Denmark

empsall1

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
    set empsdk empsfr;
run;
```

Reinitialize PDV

PDV

First	Gender	Country

PDV is reinitialized before processing the next data set.

empsall1

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
    set empsdk empsfr;
run;
```

PDV

First	Gender	Country
Pierre	M	France

empsall1

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
    set emps
run;
```

Implicit OUTPUT;
Implicit RETURN;

PDV

First	Gender	Country
Pierre	M	France

empsall1

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark
Pierre	M	France

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
    set empsdk empsfr;
run;
```

PDV

First	Gender	Country
Pierre	M	France

PDV is not reinitialized.

empsall1

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark
Pierre	M	France

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
    set empsdk empsfr;
run;
```

PDV

First	Gender	Country
Sophie	F	France

empsall1

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark
Pierre	M	France

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
    set empsdk empsfr;
run;
```

Implicit OUTPUT;
Implicit RETURN;

PDV

First	Gender	Country
Sophie	F	France

empsall1

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark
Pierre	M	France
Sophie	F	France

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

```
data empsall1;
    set empsdk empsfr;
run;
```

PDV

First	Gender	Country
Sophie	F	France

PDV is not reinitialized.

empsall1

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark
Pierre	M	France
Sophie	F	France

...

Execution

empsdk

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark

empsfr

First	Gender	Country
Pierre	M	France
Sophie	F	France

EOF

```
data empsall1;
    set empsdk empsfr;
run;
```

PDV

First	Gender	Country
Sophie	F	France

empsall1

First	Gender	Country
Lars	M	Denmark
Kari	F	Denmark
Jonas	M	Denmark
Pierre	M	France
Sophie	F	France

Viewing the Log

Partial SAS Log

```
145  data empsall1;
146     set empsdk empsfr;
147  run;

NOTE: There were 3 observations read from the data set WORK.EMPSDK.
NOTE: There were 2 observations read from the data set WORK.EMPSFR.
NOTE: The data set WORK.EMPSALL1 has 5 observations and 3 variables.
```

Unlike-Structured Data Sets

Concatenate **empscn** and **empsjp** to create a new data set named **empsall2**.

empscn

First	Gender	Country
Chang	M	China
Li	M	China
Ming	F	China

empsjp

First	Gender	Region
Cho	F	Japan
Tomi	M	Japan

The data sets do not contain the same variables.

```
data empsall2;
   set empscn empsjp;
run;
```

p110d02

10.01 Quiz

How many variables will be in **empsall2** after concatenating **empscn** and **empsjp**?

empscn

First	Gender	Country
Chang	M	China
Li	M	China
Ming	F	China

empsjp

First	Gender	Region
Cho	F	Japan
Tomi	M	Japan

```
data empsall2;
    set empscn empsjp;
run;
```

10.01 Quiz – Correct Answer

How many variables will be in **empsall2** after concatenating **empscn** and **empsjp**?

empscn

First	Gender	Country
Chang	M	China
Li	M	China
Ming	F	China

empsjp

First	Gender	Region
Cho	F	Japan
Tomi	M	Japan

Four variables: First, Gender, Country, and Region

Compilation

empscn

First	Gender	Country
Chang	M	China
Li	M	China
Ming	F	China

empsjp

First	Gender	Region
Cho	F	Japan
Tomi	M	Japan

```
data empsall2;
    set empscn empsjp;
run;
```

PDV

First	Gender	Country

p110d02

Compilation

empscn

First	Gender	Country
Chang	M	China
Li	M	China
Ming	F	China

empsjp

First	Gender	Region
Cho	F	Japan
Tomi	M	Japan

```
data empsall2;
    set empscn empsjp;
run;
```

PDV

First	Gender	Country	Region

Final Results

empsall2

First	Gender	Country	Region
Chang	M	China	
Li	M	China	
Ming	F	China	
Cho	F		Japan
Tomi	M		Japan

- **Region** has missing values due to PDV initialization.
- **Country** has missing values due to PDV reinitialization before processing the second data set.

Business Scenario

Rename variables in one or more data sets to align columns.

empscn

First	Gender	Country
Chang	M	China
Li	M	China
Ming	F	China

empsjp

First	Gender	Region
Cho	F	Japan
Tomi	M	Japan

Rename **Region** to **Country**.

RENAME= Data Set Option

The *RENAME= data set option* changes the name of a variable.

```
data empsall2;
   set empscn empsjp(rename=(Region=Country));
run;
```

SAS-data-set **(RENAME=(***old-name-1***=***new-name-1*
*old-name-2***=***new-name-2*
...
*old-name-n***=***new-name-n***))**

- The RENAME= option must be specified in parentheses immediately after the appropriate SAS data set name.
- The name change affects the PDV and the output data set. It has no effect on the input data set.

RENAME= Data Set Option

Multiples variables can be renamed in one or more data sets.

```
set empscn(rename=(Country=Region))
    empsjp;
```

```
set empscn(rename=(First=Fname
                   Country=Region))
    empsjp(rename=(First=Fname));
```

```
set empscn
    empsjp(rename=(Region=Country));
```

10.02 Quiz

Which statement has correct syntax?

```
set empscn(rename(Country=Location))
    empsjp(rename(Region=Location));
```

```
set empscn(rename=(Country=Location))
    empsjp(rename=(Region=Location));
```

```
set empscn rename=(Country=Location)
    empsjp rename=(Region=Location);
```

10.02 Quiz – Correct Answer

Which statement has correct syntax?

```
set empscn(rename(Country=Location))
    empsjp(rename(Region=Location));
```

```
set empscn(rename=(Country=Location))
    empsjp(rename=(Region=Location));
```

c.

```
set empscn rename=(Country=Location)
    empsjp rename=(Region=Location);
```

Compilation

empscn

First	Gender	Country
Chang	M	China
Li	M	China
Ming	F	China

empsjp

First	Gender	Region
Cho	F	Japan
Tomi	M	Japan

```
data empsall2;
   set empscn empsjp(rename=(Region=Country));
run;
```

PDV

First	Gender	Country

p110d03

Compilation

empscn

First	Gender	Country
Chang	M	China
Li	M	China
Ming	F	China

empsjp

First	Gender	Region
Cho	F	Japan
Tomi	M	Japan

```
data empsall2;
   set empscn empsjp(rename=(Region=Country));
run;
```

PDV

First	Gender	Country

...

Compilation

empscn

First	Gender	Country
Chang	M	China
Li	M	China
Ming	F	China

empsjp

First	Gender	Region
Cho	F	Japan
Tomi	M	Japan

```
data empsall2;
   set empscn empsjp(rename=(Region=Country));
run;
```

PDV

First	Gender	Country

...

Compilation

empscn

First	Gender	Country
Chang	M	China
Li	M	China
Ming	F	China

empsjp

First	Gender	Region
Cho	F	Japan
Tomi	M	Japan

```
data empsall2;
   set empscn empsjp(rename=(Region=Country));
run;
```

PDV

First	Gender	Country

...

Compilation

empscn

First	Gender	Country
Chang	M	China
Li	M	China
Ming	F	China

empsjp

First	Gender	Region
Cho	F	Japan
Tomi	M	Japan

```
data empsall2;
   set empscn empsjp(rename=(Region=Country));
run;
```

PDV

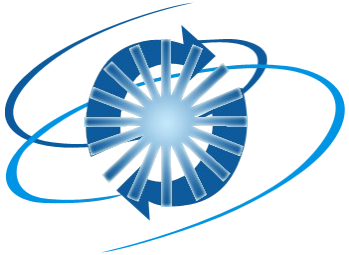
First	Gender	Country

Final Results

The **Region** values are stored in **Country**.

empsall2

First	Gender	Country
Chang	M	China
Li	M	China
Ming	F	China
Cho	F	Japan
Tomi	M	Japan



Exercise

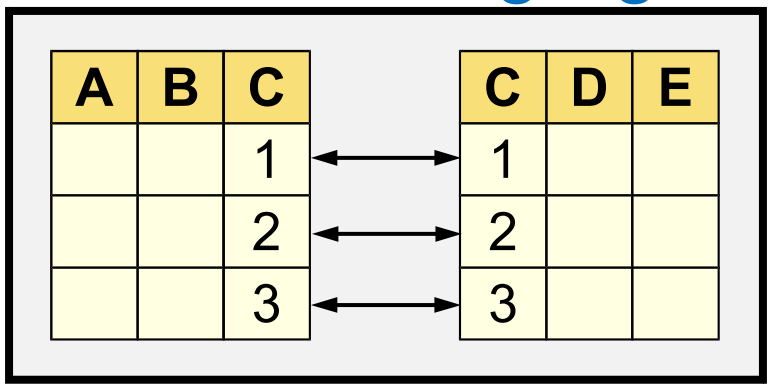
This exercise reinforces the concepts discussed previously.

Chapter 10: Combining Data Sets

Objectives

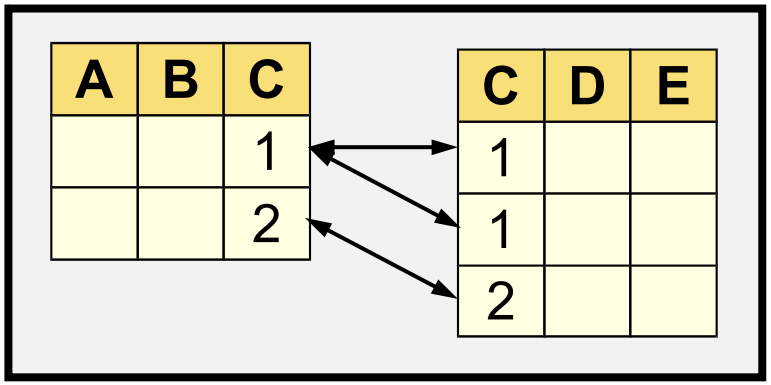
- Prepare data sets for merging using the SORT procedure.
- Merge SAS data sets one-to-one based on a common variable.

Match-Merging



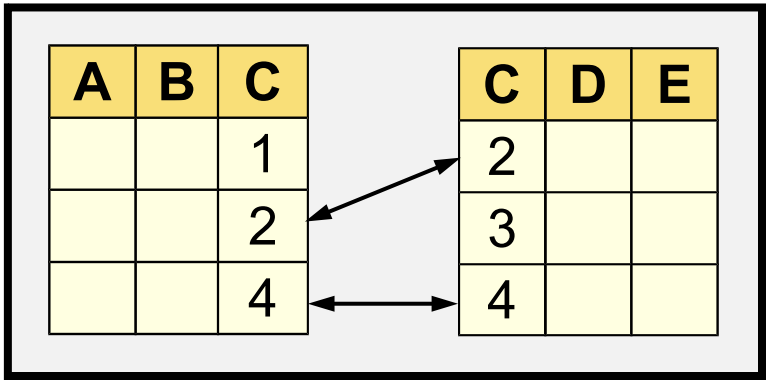
One-to-One

A single observation in one data set is related to exactly one observation in another data set based on the values of one or more selected variables.



One-to-Many

A single observation in one data set is related to more than one observation in another data set based on the values of one or more selected variables.



Non-matches

At least one observation in one data set is unrelated to any observation in another data set based on the values of one or more selected variables.

Match-Merging: Sorting the Data Sets

The data sets in a match-merge must be sorted by the common variable or variables being matched.

```
PROC SORT  DATA=input-SAS-data-set
           <OUT=output-SAS-data-set>;
     BY <DESCENDING> by-variable(s);
RUN;
```

10.03 Multiple Choice Poll

Which of the following BY statements correctly sorts by descending salary within gender?

a. by descending salary within gender;
b. by descending salary gender;
c. by gender descending salary;
d. by gender salary descending;

10.03 Multiple Choice Poll – Correct Answer

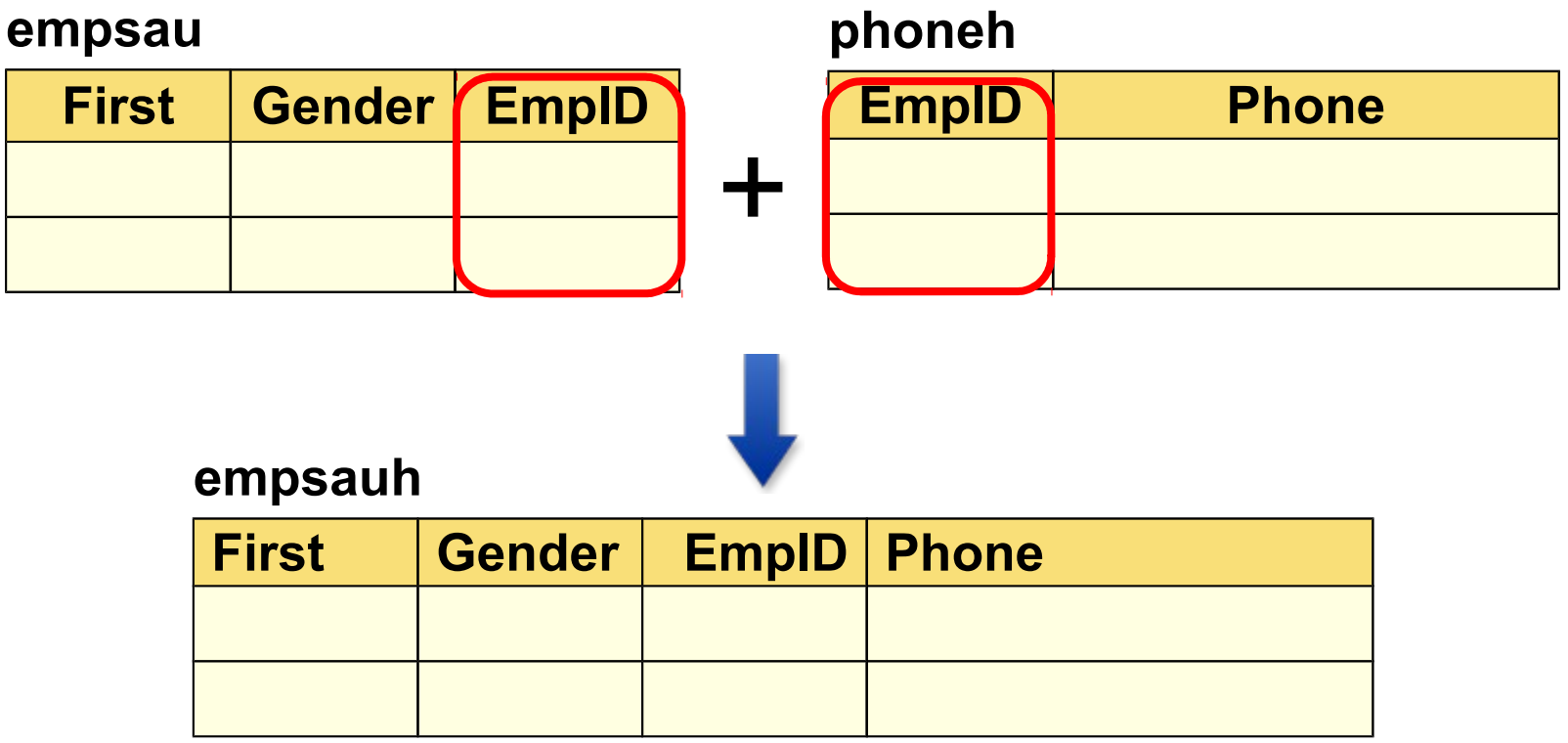
Which of the following BY statements correctly sorts by descending salary within gender?

a. by descending salary within gender;
b. by descending salary gender;
c. by gender descending salary;
d. by gender salary descending;

The keyword descending is placed before the variable to which it applies.

Business Scenario

Merge the Australian employee data set with a phone data set to obtain each employee's home phone number, storing the results in a new data set.



Match-Merging

The *MERGE statement* in a DATA step joins observations from two or more SAS data sets into single observations.

```
data empsauh;
    merge empsau phoneh;
    by EmpID;
run;
```

MERGE *SAS-data-set1 SAS-data-set2 . . .*;
BY <DESCENDING> *BY-variable(s)*;

A *BY statement* indicates a match-merge and lists the variable or variables to match.

MERGE and BY Statements

Requirements for match-merging:

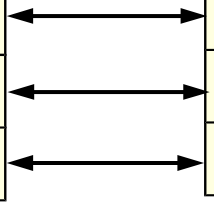
- Two or more data sets are listed in the MERGE statement.
- The variables in the BY statement must be common to all data sets.
- The data sets must be sorted by the variables listed in the BY statement.

One-to-One Merge

One observation in **empsau** matches exactly one observation in **phoneh**.

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152



phoneh

EmpID	Phone
121150	+61(2)5555-1793
121151	+61(2)5555-1849
121152	+61(2)5555-1665

The data sets are sorted by **EmpID**.

Final Results

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phoneh

EmpID	Phone
121150	+61(2)5555-1793
121151	+61(2)5555-1849
121152	+61(2)5555-1665

empsauh

First	Gender	EmpID	Phone
Togar	M	121150	+61(2)5555-1793
Kylie	F	121151	+61(2)5555-1849
Birin	M	121152	+61(2)5555-1665

10.04 Quiz

- Retrieve program **p110a01**.
- Complete the program to match-merge the sorted SAS data sets referenced in the PROC SORT steps.
- Submit the program. Correct and resubmit, if necessary.

What are the modified, completed statements?

10.04 Quiz – Correct Answer

What are the modified, completed statements?

```
proc sort data=orion.employee_payroll
          out=work.payroll;
   by Employee_ID;
run;

proc sort data=orion.employee_addresses
          out=work.addresses;
   by Employee_ID;
run;

data work.payadd;
   merge work.payroll work.addresses;
   by Employee_ID;
run;
```

p110a01s

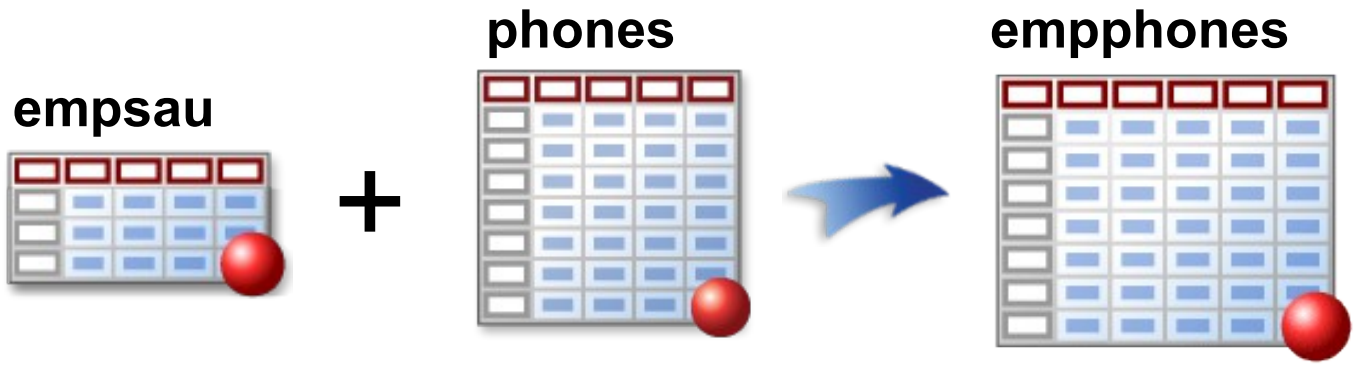
Chapter 10: Combining Data Sets

Objectives

- Merge SAS data sets one-to-many based on a common variable.

Business Scenario

Merge the Australian employee information data set with the phone data set to obtain the phone numbers for each employee.



Considerations

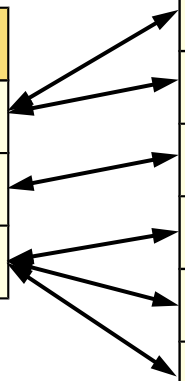
In this one-to-many merge, one observation in **empsau** matches one or more observations in **phones**.

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666



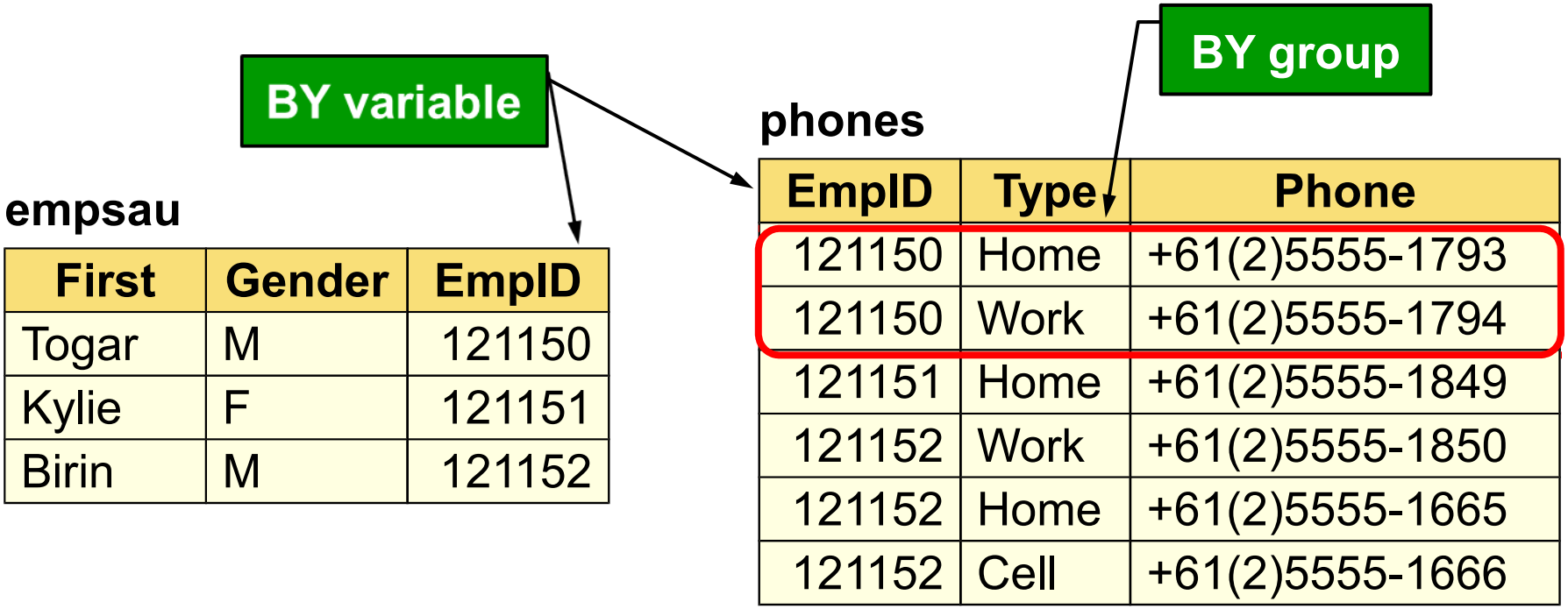
The data sets are sorted by **EmpID**.

Match-Merging

Merge the two data sets by **EmpID** and create a new data set named **empphones**.

```
data empphones;
   merge empsau phones;
   by EmpID;
run;
```

Match-Merging



- The common variable is called the *BY variable*.
- The value of the BY variable is the *BY value*.
- A group of observations with the same BY value is a *BY group*.

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

Initialize PDV

```
data empphones;
   merge empsau phones;
   by EmpID;
run;
```

PDV

First	Gender	EmpID	Type	Phone
		.		

p110d05
...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
    merge empsau phones;
    by EmpID;
run;
```

Do the **EmpID** values match?

Yes

PDV

First	Gender	EmpID	Type	Phone
		.		

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
    merge empsau phones;
    by EmpID;
run;
```

Reads both observations into the PDV.

PDV

First	Gender	EmpID	Type	Phone
Togar	M	121150	Home	+61(2)5555-1793

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
   merge empsau phones;
   by EmpID;
run;
```

Implicit OUTPUT;
Implicit RETURN;

PDV

First	Gender	EmpID	Type	Phone
Togar	M	121150	Work	+61(2)5555-1793

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
    merge empsau phones;
    by EmpID;
run;
```

PDV

Data set variables are not reinitialized.

First	Gender	EmpID	Type	Phone
Togar	M	121150	Work	+61(2)5555-1793

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
    merge empsau phones;
    by EmpID;
run;
```

Do the **EmpID** values match?

No

PDV

First	Gender	EmpID	Type	Phone
Togar	M	121150	Home	+61(2)5555-1793

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
    merge empsau phones;
    by EmpID;
run;
```

Did **EmpID** change?

No

PDV

First	Gender	EmpID	Type	Phone
Togar	M	121150	Home	+61(2)5555-1793

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
    merge empsau phones;
    by EmpID;
run;
```

Reads the matching observation into the PDV.

PDV

First	Gender	EmpID	Type	Phone
Togar	M	121150	Work	+61(2)5555-1794

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
   merge empsau phonehW;
   by EmpID;
run;
```

Implicit OUTPUT;
Implicit RETURN;

PDV

First	Gender	EmpID	Type	Phone
Togar	M	121150	Work	+61(2)5555-1794

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
    merge empsau phones;
    by EmpID;
run;
```

Do the **EmpID** values match?

Yes

PDV

First	Gender	EmpID	Type	Phone
Togar	M	121150	Work	+61(2)5555-1794

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
    merge empsau phones;
    by EmpID;
run;
```

Did **EmpID** change?

Yes

PDV

First	Gender	EmpID	Type	Phone
Togar	M	121150	Work	+61(2)5555-1794

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
    merge empsau phones;
    by EmpID;
run;
```

Reinitialize PDV

PDV

First	Gender	EmpID	Type	Phone
		.		

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
   merge empsau phones;
   by EmpID;
run;
```

Reads both observations into the PDV.

PDV

First	Gender	EmpID	Type	Phone
Kylie	F	121151	Home	+61(2)5555-1849

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
   merge empsau phones;
   by EmpID;
run;
```

Implicit OUTPUT;
Implicit RETURN;

PDV

First	Gender	EmpID	Type	Phone
Kylie	F	121151	Home	+61(2)5555-1849

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
    merge empsau phones;
    by EmpID;
run;
```

Do the **EmpID** values match?

Yes

PDV

First	Gender	EmpID	Type	Phone
Kylie	F	121151	Home	+61(2)5555-1849

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
    merge empsau phones;
    by EmpID;
run;
```

Did **EmpID** change?

Yes

PDV

First	Gender	EmpID	Type	Phone
Kylie	F	121151	Home	+61(2)5555-1849

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

Reinitialize PDV

```
data empphones;
    merge empsau phones;
    by EmpID;
run;
```

PDV

First	Gender	EmpID	Type	Phone
		.		

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
   merge empsau phones;
   by EmpID;
run;
```

Reads both observations into the PDV.

PDV

First	Gender	EmpID	Type	Phone
Birin	M	121152	Work	+61(2)5555-1850

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
   merge empsau phones;
   by EmpID;
run;
```

Implicit OUTPUT;
Implicit RETURN;

PDV

First	Gender	EmpID	Type	Phone
Birin	M	121152	Work	+61(2)5555-1850

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

EOF

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
    merge empsau phones;
    by EmpID;
run;
```

Did **EmpID** change?

No

PDV

First	Gender	EmpID	Type	Phone
Birin	M	121152	Work	+61(2)5555-1850

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

EOF

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
    merge empsau phones;
    by EmpID;
run;
```

Reads the matching observation into the PDV.

PDV

First	Gender	EmpID	Type	Phone
Birin	M	121152	Home	+61(2)5555-1665

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

EOF

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
   merge empsau phones;
   by EmpID;
run;
```

Implicit OUTPUT;
Implicit RETURN;

PDV

First	Gender	EmpID	Type	Phone
Birin	M	121152	Home	+61(2)5555-1665

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

EOF

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
    merge empsau phones;
    by EmpID;
run;
```

Did **EmpID** change?

No

PDV

First	Gender	EmpID	Type	Phone
Birin	M	121152	Home	+61(2)5555-1665

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

EOF

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
    merge empsau phones;
    by EmpID;
run;
```

Reads the matching observation into the PDV.

PDV

First	Gender	EmpID	Type	Phone
Birin	M	121152	Cell	+61(2)5555-1666

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

EOF

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

```
data empphones;
   merge empsau phones;
   by EmpID;
run;
```

Implicit OUTPUT;
Implicit RETURN;

PDV

First	Gender	EmpID	Type	Phone
Birin	M	121152	Cell	+61(2)5555-1666

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

EOF

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666

EOF

```
data empphones;
    merge empsau phones;
    by EmpID;
run;
```

PDV

First	Gender	EmpID	Type	Phone
Birin	M	121152	Cell	+61(2)5555-1666

...

Final Results

empphones

First	Gender	EmpID	Type	Phone
Togar	M	121150	Home	+61(2)5555-1793
Togar	M	121150	Work	+61(2)5555-1794
Kylie	F	121151	Home	+61(2)5555-1849
Birin	M	121152	Work	+61(2)5555-1850
Birin	M	121152	Home	+61(2)5555-1665
Birin	M	121152	Cell	+61(2)5555-1666

10.05 Quiz

In a one-to-many merge, does it matter which data set is listed first in the MERGE statement?

- Open **p110a02** and submit
- Reverse the order of the data sets and submit again.
- Observe the results. How are they different?

10.05 Quiz – Correct Answer

In a one-to-many merge, does it matter which data set is listed first in the MERGE statement?

- Open **p110a02** and submit
- Reverse the order of the data sets and submit again.
- Observe the results. How are they different?

The results are the same, but the order of the variables is different.

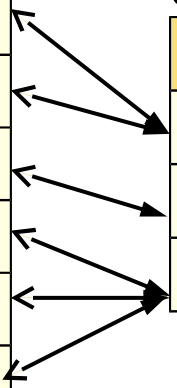
p110a02s

Many-to-One Merge

One or more rows in one data set match exactly one row in the other data set.

phones

EmpID	Type	Phone
121150	Home	+61(2)5555-1793
121150	Work	+61(2)5555-1794
121151	Home	+61(2)5555-1849
121152	Work	+61(2)5555-1850
121152	Home	+61(2)5555-1665
121152	Cell	+61(2)5555-1666



empsau

EmpID	First	Gender
121150	Togar	M
121151	Kylie	F
121512	Birin	M

```
data phones;
   merge phones empsau;
   by EmpID;
run;
```

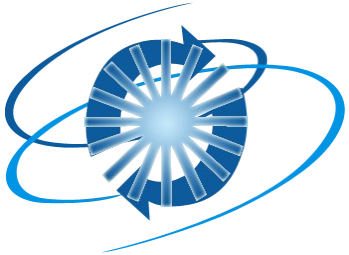
p110d06

Viewing the Output

PROC PRINT Output

Obs	EmpID	Type	Phone	First	Gender
1	121150	Home	+61(2)5555-1793	Togar	M
2	121150	Work	+61(2)5555-1794	Togar	M
3	121151	Home	+61(2)5555-1849	Kylie	F
4	121152	Work	+61(2)5555-1850	Birin	M
5	121152	Home	+61(2)5555-1665	Birin	M
6	121152	Cell	+61(2)5555-1666	Birin	M

- The results are the same as the one-to-many merge.
- The order of variables is different.



Exercise

This exercise reinforces the concepts discussed previously.

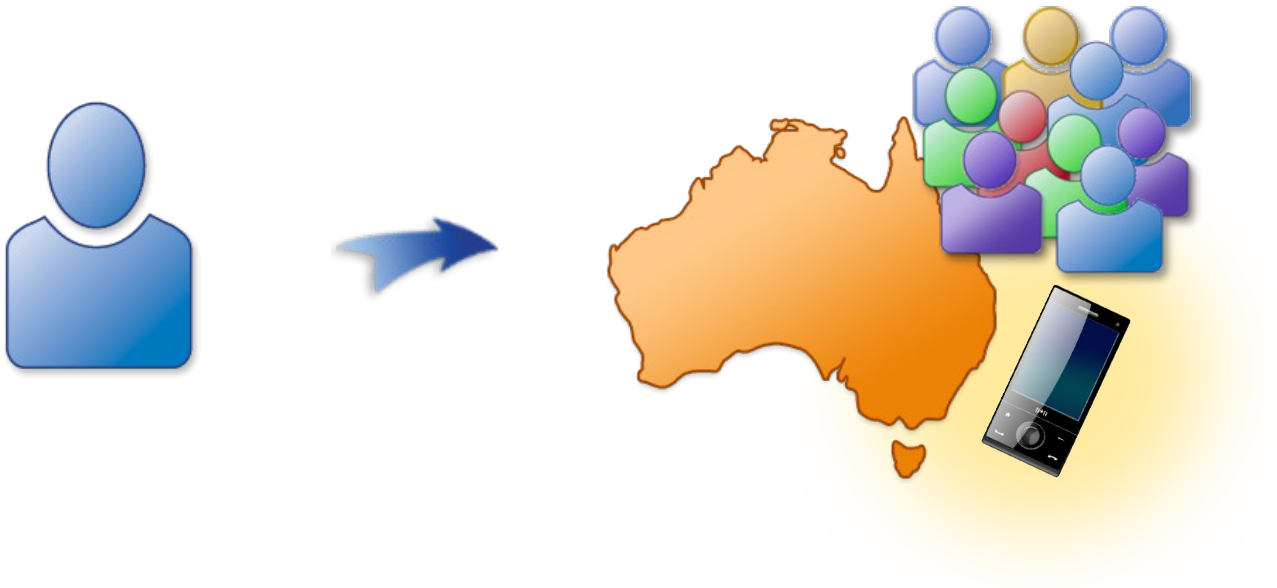
Chapter 10: Combining Data Sets

Objectives

- Control the observations in the output data set by using the IN= option.

Business Scenario

An Orion Star manager in Australia requested an inventory of company phone numbers.



Merge with Non-Matches

There are observations in **empsau** that do not have a match in **phonec**, and some in **phonec** that do not match any in **empsau**.

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

The data sets are sorted by **EmpID**.

Match-Merging

Merge **empsau** and **phonec** by **EmpID** to create a new data set named **empsauc**.

```
data empsauc;
   merge empsau phonec;
   by EmpID;
run;
```

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

Initialize PDV

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

PDV

First	Gender	EmpID	Phone
		.	

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

Do the **EmpID** values match?

Yes

PDV

First	Gender	EmpID	Phone
		.	

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

Reads both observations into the PDV.

PDV

First	Gender	EmpID	Phone
Togar	M	121150	+61(2)5555-1795

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

```
data empsauc;
   merge empsau phonec;
   by EmpID;
run;
```

Implicit OUTPUT;
Implicit RETURN;

PDV

First	Gender	EmpID	Phone
Togar	M	121150	+61(2)5555-1795

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

Do the **EmpID** values match?

No

PDV

First	Gender	EmpID	Phone
Togar	M	121150	+61(2)5555-1795

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

Did **EmpID** change?

Yes

PDV

First	Gender	EmpID	Phone
Togar	M	121150	+61(2)5555-1795

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

Reinitialize PDV

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

PDV

First	Gender	EmpID	Phone
		.	

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

Which **EmpID** value sequentially comes first?

```
121151
```

PDV

First	Gender	EmpID	Phone
		.	

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

Reads that observation into the PDV.

PDV

First	Gender	EmpID	Phone
Kylie	F	121151	

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

Implicit OUTPUT;
Implicit RETURN;

PDV

First	Gender	EmpID	Phone
Kylie	F	121151	

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

Do the **EmpID** values match?

Yes

PDV

First	Gender	EmpID	Phone
Kylie	F	121151	

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

Did **EmpID** change?

Yes

PDV

First	Gender	EmpID	Phone
Kylie	F	121151	

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

Reinitialize PDV

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

PDV

First	Gender	EmpID	Phone
		.	

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

Reads both observations into the PDV.

PDV

First	Gender	EmpID	Phone
Birin	M	121152	+61(2)5555-1667

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

Implicit OUTPUT;
Implicit RETURN;

PDV

First	Gender	EmpID	Phone
Birin	M	121152	+61(2)5555-1667

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

EOF

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

Did **EmpID** change?

Yes

PDV

First	Gender	EmpID	Phone
Birin	M	121152	+61(2)5555-1667

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

EOF

Reinitialize PDV

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

PDV

First	Gender	EmpID	Phone
		.	

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

EOF

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

Reads the observation into the PDV.

PDV

First	Gender	EmpID	Phone
		121153	+61(2)5555-1348

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

EOF

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

Implicit OUTPUT;
Implicit RETURN;

PDV

First	Gender	EmpID	Phone
		121153	+61(2)5555-1348

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

EOF

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

EOF

```
data empsauc;
    merge empsau phonec;
    by EmpID;
run;
```

PDV

First	Gender	EmpID	Phone
		121153	+61(2)5555-1348

Final Results

empsauc

First	Gender	EmpID	Phone
Togar	M	121150	+61(2)5555-1795
Kylie	F	121151	
Birin	M	121152	+61(2)5555-1667
		121153	+61(2)5555-1348

The final results include both matches and non-matches.

10.06 Quiz

Which data set contributed information to the last observation in the output data set?

- empsau
- phonec
- both empsau and phonec
- There is insufficient information.

empsauc

First	Gender	EmpID	Phone
Togar	M	121150	+61(2)5555-1795
Kylie	F	121151	
Birin	M	121152	+61(2)5555-1667
		121153	+61(2)5555-1348

10.06 Quiz – Correct Answer

Which data set contributed information to the last observation in the output data set?

- empsau
- phonec
- both empsau and phonec
- There is insufficient information.

empsauc

First	Gender	EmpID	Phone
Togar	M	121150	+61(2)5555-1795
Kylie	F	121151	
Birin	M	121152	+61(2)5555-1667
		121153	+61(2)5555-1348

Business Scenario

An Orion Star manager requested three phone inventory reports.

- employees with company phones
- employees without company phones
- phones with an invalid employee ID

IN= Data Set Option

The *IN= data set option* creates a variable that indicates whether the data set contributed to building the current observation.

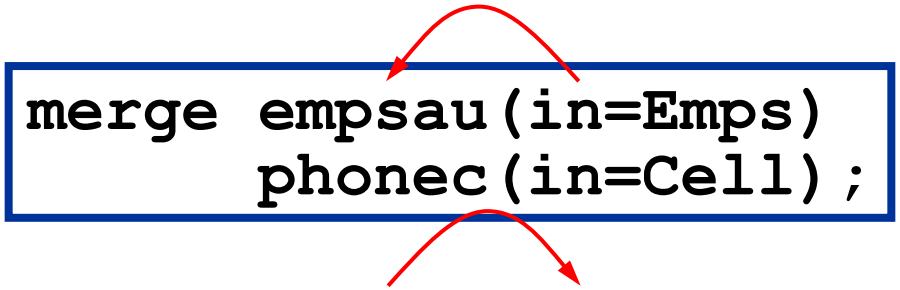
MERGE *SAS-data-set* **(IN=***variable***) …**

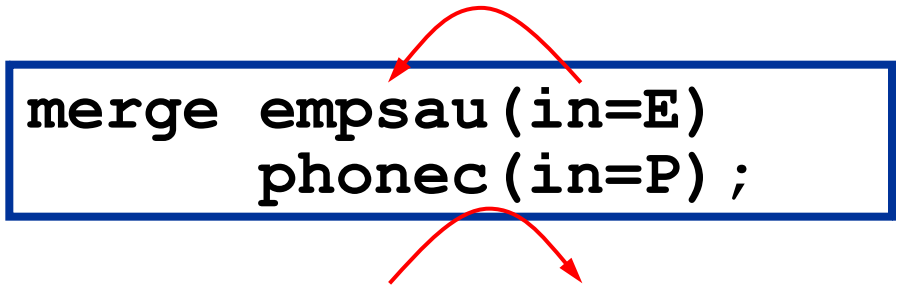
variable is a temporary numeric variable that has two possible values:

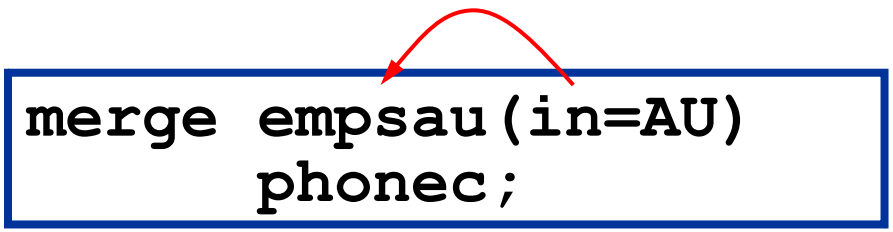
0	Indicates that the data set did ***not*** contribute to the current observation.
1	Indicates that the data set ***did*** contribute to the current observation.

IN= Data Set Option

MERGE statement examples:







Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

```
data empsauc;
   merge empsau(in=Emps)
         phonec(in=Cell);
   by EmpID;
run;
```

match

PDV

First	Gender	EmpID	Emps	Phone	Cell
Togar	M	121150	1	+61(2)5555-1795	1

p110d08

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

```
data empsauc;
   merge empsau(in=Emps)
         phonec(in=Cell);
   by EmpID;
run;
```

non-match

PDV

First	Gender	EmpID	Emps	Phone	Cell
Kylie	F	121151	1		0

...

Execution

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

```
data empsauc;
   merge empsau(in=Emps)
         phonec(in=Cell);
   by EmpID;
run;
```

match

PDV

First	Gender	EmpID	Emps	Phone	Cell
Birin	M	121152	1	+61(2)5555-1667	1

...

10.07 Quiz

What are the values of **Emps** and **Cell**?

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

EOF

```
data empsauc;
   merge empsau(in=Emps)
         phonec(in=Cell);
   by EmpID;
run;
```

PDV

First	Gender	EmpID	D Emps	Phone	D Cell
		121153		+61(2)5555-1348	

10.07 Quiz – Correct Answer

What are the values of **Emps** and **Cell**?

empsau

First	Gender	EmpID
Togar	M	121150
Kylie	F	121151
Birin	M	121152

phonec

EmpID	Phone
121150	+61(2)5555-1795
121152	+61(2)5555-1667
121153	+61(2)5555-1348

EOF

```
data empsauc;
   merge empsau(in=Emps)
         phonec(in=Cell);
   by EmpID;
run;
```

non-match

PDV

First	Gender	EmpID	Emps	Phone	Cell
		121153	0	+61(2)5555-1348	1

PDV Results

PDV

First	Gender	EmpID	d Emps	Phone	d Cell
Togar	M	121150	1	+61(2)5555-1795	1
Kylie	F	121151	1		0
Birin	M	121152	1	+61(2)5555-1667	1
		121153	0	+61(2)5555-1348	1

The variables created with the IN= data set option are only available during DATA step execution.

- They are not written to the SAS data set.
- Their value can be tested using conditional logic.

Matches Only

Add a subsetting IF statement to select the employees that have company phones.

```
data empsauc;
   merge empsau(in=Emps)
         phonec(in=Cell);
   by EmpID;
   if Emps=1 and Cell=1;
run;
```

empsauc

First	Gender	EmpID	Phone
Togar	M	121150	+61(2)5555-1795
Birin	M	121152	+61(2)5555-1667

Non-Matches from empsau

Select the employees that do not have company phones.

```
data empsauc;
    merge empsau(in=Emps)
          phonec(in=Cell);
    by EmpID;
    if Emps=1 and Cell=0;
run;
```

empsauc

First	Gender	EmpID	Phone
Kylie	F	121151	

Non-Matches from phonec

Select the phones associated with an invalid employee ID.

```
data empsauc;
    merge empsau(in=Emps)
          phonec(in=Cell);
    by EmpID;
    if Emps=0 and Cell=1;
run;
```

empsauc

First	Gender	EmpID	Phone
		121153	+61(2)5555-1348

All Non-Matches

```
data empsauc;
   merge empsau(in=Emps)
         phonec(in=Cell);
   by EmpID;
   if Emps=0 or Cell=0;
run;
```

empsauc

First	Gender	EmpID	Phone
Kylie	F	121151	
		121153	+61(2)5555-1348

Use the OR operator, not the AND operator.

Alternate Syntax

When checking a variable for a value of *1* or *0* as in the previous scenario, you can use the following syntax:

Instead of	You can use
if Emps=1 and Cell=1;	if Emps and Cell;
if Emps=1 and Cell=0;	if Emps and not Cell;
if Emps=0 and Cell=1;	if not Emps and Cell;
if Emps=0 or Cell=0;	If not Emps or not Cell;

Alternate Syntax

Both programs create a report of employees without cell phones.

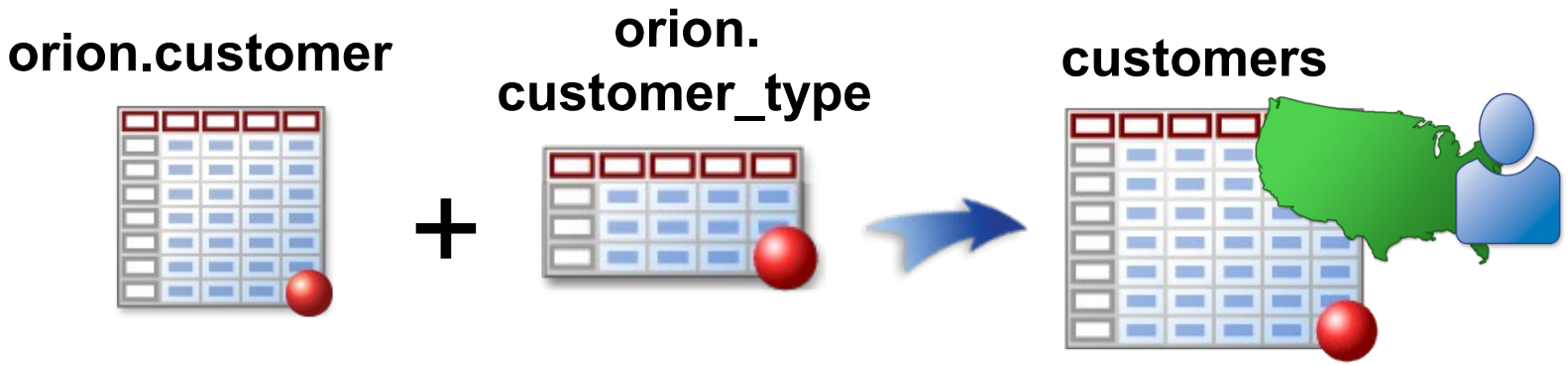
```
data empsphone;
   merge empsact(in=inEmps)
         phoneact(in=inCell);
   by EmpID;
   if inEmps=1 and inCell=0;
run;
```

```
data empsphone;
   merge empsact(in=inEmps)
         phoneact(in=inCell);
   by EmpID;
   if inEmps and not inCell;
run;
```

p110d09

Business Scenario

Merge Orion Star customer information with customer type data to obtain a customer description. The new data set should include only US customers.



Considerations

The **orion.customer** data set is not sorted by **Customer_Type_ID**, the common variable. The subsetting variable, **Country**, is defined in only one data set.

orion.customer

Customer_ ID	Country	Customer_ Name	. . .	Birth_ Date	Customer_Type_ ID

orion.customer_type

Customer_ Group	Customer_Group_ ID	Customer_ Type	Customer_Type_ ID

10.08 Quiz

Open and submit **p110a03**. Correct the program and resubmit. What change is needed to correct the error?

```
proc sort data=orion.customer
          out=cust_by_type;
   by Customer_Type_ID;
run;

data customers;
   merge cust_by_type orion.customer_type;
   by Customer_Type_ID;
   where Country='US';
run;
```

10.08 Quiz – Correct Answer

What change is needed to correct the error?

```
397  proc sort data=orion.customer
398           out=cust_by_type;
399     by Customer_Type_ID;
400  run;
NOTE: There were 77 observations read from the data set ORION.CUSTOMER.
NOTE: The data set WORK.CUST_BY_TYPE has 77 observations and 12 variables.

401
402  data customers;
403     merge cust_by_type orion.customer_type;
404     by Customer_Type_ID;
405     where Country='US';
ERROR: Variable Country is not on file ORION.CUSTOMER_TYPE.
406  run;

NOTE: The SAS System stopped processing this step because of errors.
WARNING: The data set WORK.CUSTOMERS may be incomplete.  When this step was stopped
there were 0 observations and 15 variables.
```

Country is not defined in both data sets. Replace the WHERE statement with a subsetting IF.

Subsetting IF

Use a subsetting IF when the subsetting variable is not in all data sets named in the MERGE statement.

```
proc sort data=orion.customer
          out=cust_by_type;
   by Customer_Type_ID;
run;

data customers;
   merge cust_by_type  orion.customer_type;
   by Customer_Type_ID;
   if Country='US';
run;
```

p110a03s

Viewing the Output

Partial SAS Log

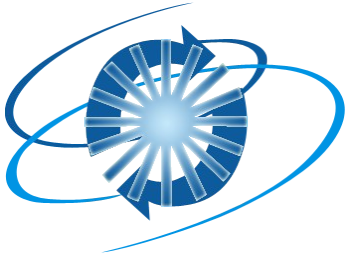
```
407  proc sort data=orion.customer
408            out=cust_by_type;
409     by Customer_Type_ID;
410  run;
NOTE: There were 77 observations read from the data set ORION.CUSTOMER.
NOTE: The data set WORK.CUST_BY_TYPE has 77 observations and 12 variables.

411
412  data customers;
413     merge cust_by_type orion.customer_type;
414     by Customer_Type_ID;
415     if Country='US';
416  run;

NOTE: There were 77 observations read from the data set WORK.CUST_BY_TYPE.
NOTE: There were 8 observations read from the data set ORION.CUSTOMER_TYPE.
NOTE: The data set WORK.CUSTOMERS has 28 observations and 15 variables.
```

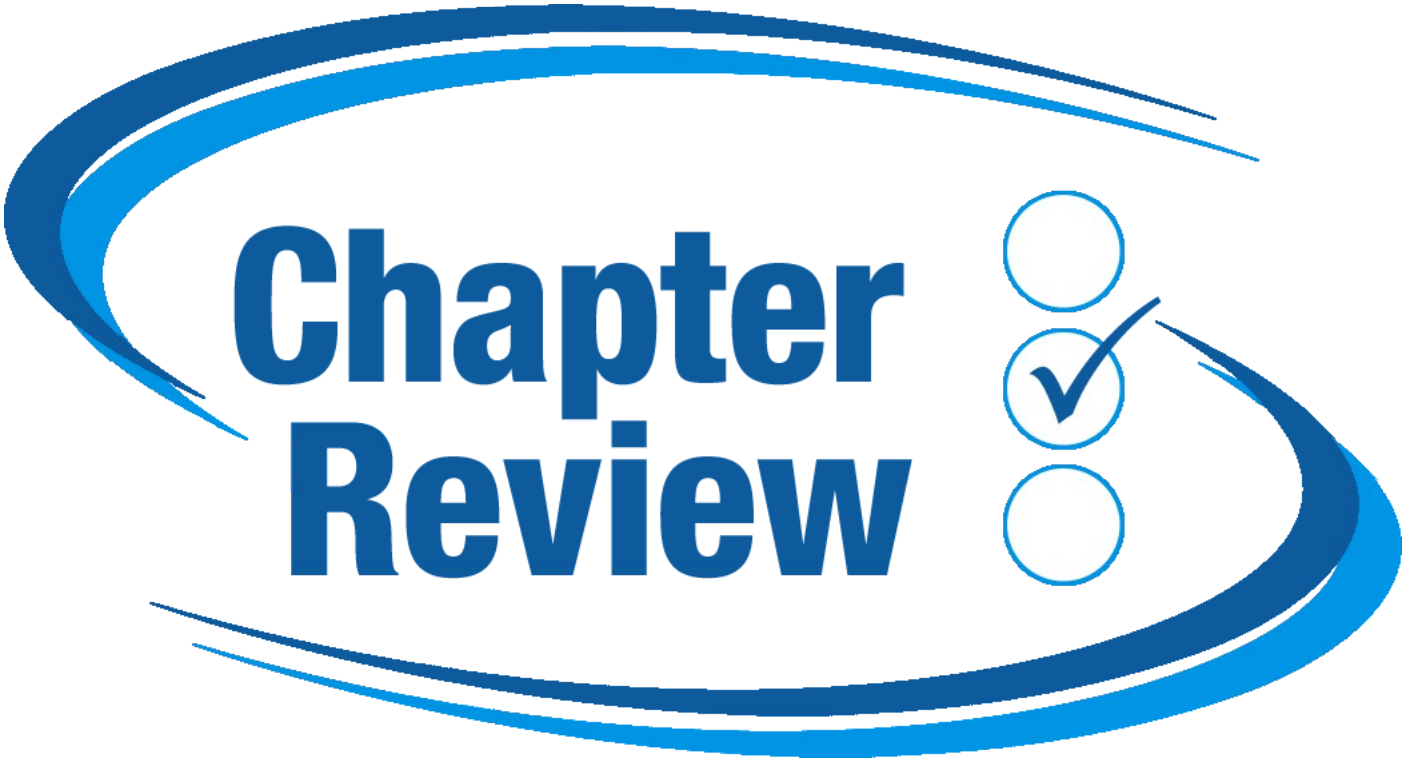
WHERE versus Subsetting IF Statement

Step and Usage	WHERE	IF
PROC step	Yes	No
DATA step (source of variable)		
SET statement	Yes	Yes
assignment statement	No	Yes
INPUT statement	No	Yes
SET/MERGE statement (multiple data sets)		
Variable in ALL data sets	Yes	Yes
Variable not in ALL data sets	No	Yes



Exercise

This exercise reinforces the concepts discussed previously.



1. Which of the following statements is true about merging SAS data sets by using the DATA step?

   a. Merging combines observations from two or more data sets into a single observation in new data set.
   b. SAS can merge data sets based on the position of observations in the original data set or by the values of one or more common variables.
   c. Match-merging is merging by values of one or more common variables.
   d. To match-merge data sets, all input data sets must be sorted or indexed on the BY variable or variables.
   e. all of the above

1. Which of the following statements is true about merging SAS data sets by using the DATA step?

   a. Merging combines observations from two or more data sets into a single observation in new data set.
   b. SAS can merge data sets based on the position of observations in the original data set or by the values of one or more common variables.
   c. Match-merging is merging by values of one or more common variables.
   d. To match-merge data sets, all input data sets must be sorted or indexed on the BY variable or variables.
   e. all of the above

2. Which of the following programs concatenates the data sets **sales** and **products**, in that order?

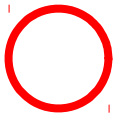
a.
```
data newsales;
   set products sales;
run;
```

```
data newsales;
   set sales products;
run;
```

b.
```
data newsales;
   set sales;
   set products;
run;
```

2. Which of the following programs concatenates the data sets **sales** and **products**, in that order?

a.
```
data newsales;
   set products sales;
run;
```



```
data newsales;
   set sales products;
run;
```

b.
```
data newsales;
   set sales;
   set products;
run;
```

3. If you run this DATA step, what observations does the data set **bonuses** contain?

```
data bonuses;
   merge managers staff;
   by EmpID;
run;
```

a. all of the observations from **managers**, and only those observations from **staff** with matching values for **EmpID**
b. all of the observations from **staff**, and only those observations from **managers** with matching values for **EmpID**
c. all observations from **staff** and all observations from **managers**, whether or not they have matching values
d. only those observations from **staff** and **managers** with matching values for **EmpID**

3. If you run this DATA step, what observations does the data set **bonuses** contain?

```
data bonuses;
   merge managers staff;
   by EmpID;
run;
```

a. all of the observations from **managers**, and only those observations from **staff** with matching values for **EmpID**

b. all of the observations from **staff**, and only those observations from **managers** with matching values for **EmpID**

c. all observations from **staff** and all observations from **managers**, whether or not they have matching values

d. only those observations from **staff** and **managers** with matching values for **EmpID**

4. If you concatenate the data sets below in the order shown, what is the value of **Sale** in observation 2 of the new data set?

Reps

ID	Name
1	Nay Rong
2	Kelly Windsor
3	Julio Meraz
4	Richard Krabill

Close

ID	Sale
1	$28,000
2	$30,000
2	$40,000
3	$15,000
3	$20,000
3	$25,000
4	$35,000

a. missing
b. $30,000
c. $40,000
d. You cannot concatenate these data sets.

4. If you concatenate the data sets below in the order shown, what is the value of **Sale** in observation 2 of the new data set?

Reps

ID	Name
1	Nay Rong
2	Kelly Windsor
3	Julio Meraz
4	Richard Krabill

Close

ID	Sale
1	$28,000
2	$30,000
2	$40,000
3	$15,000
3	$20,000
3	$25,000
4	$35,000

a. missing
b. $30,000
c. $40,000
d. You cannot concatenate these data sets.

5. What happens if you submit the following program to merge **donors1** and **donors2**, shown below?

```
data merged;
   merge donors1 donors2;
   by ID;
run;
```

donors1

ID	Type	Units
2304	O	16
1129	A	48
1129	A	50
1129	A	57
2486	B	63

donors2

ID	Code	Units
6488	65	27
1129	63	32
5438	62	39
2304	61	45
1387	64	67

a. The **merged** data set contains some missing values because not all observations have matching observations in the other data set.
b. The **merged** data set contains eight observations.
c. The DATA step produces errors.

5. What happens if you submit the following program to merge **donors1** and **donors2**, shown below?

```
data merged;
   merge donors1 donors2;
   by ID;
run;
```

donors1

ID	Type	Units
2304	O	16
1129	A	48
1129	A	50
1129	A	57
2486	B	63

donors2

ID	Code	Units
6488	65	27
1129	63	32
5438	62	39
2304	61	45
1387	64	67

a. The **merged** data set contains some missing values because not all observations have matching observations in the other data set.

b. The **merged** data set contains eight observations.

c. The DATA step produces errors.

6. Suppose you want to concatenate these data sets. Which DATA step creates an output data set that combines the values of **Color** and **Hue** in the single variable **Color**?

a.
```
data widgets_all;
   set widget1(rename=(Color=Hue))
       widget2;
run;
```

```
data widgets_all;
   set widget1
       widget2(rename=(Hue=Color));
run;
```

b.
```
data widgets_all;
   set widget1
       widget2(Hue=Color);
run;
```

Widget1

Tag	Color	Model
77904	blue	AB42
56012	red	BA25
35499	orange	FC36

Widget2

Tag	Hue	Model
89325	red	SP17
65888	yellow	BA12
00167	green	PG20

6. Suppose you want to concatenate these data sets. Which DATA step creates an output data set that combines the values of **Color** and **Hue** in the single variable **Color**?

a.
```
data widgets_all;
   set widget1(rename=(Color=Hue))
       widget2;
run;
```

```
data widgets_all;
   set widget1
       widget2(rename=(Hue=Color));
run;
```

b.
```
data widgets_all;
   set widget1
       widget2(Hue=Color);
run;
```

Widget1

Tag	Color	Model
77904	blue	AB42
56012	red	BA25
35499	orange	FC36

Widget2

Tag	Hue	Model
89325	red	SP17
65888	yellow	BA12
00167	green	PG20

7. What is the syntax error in this DATA step?

```
data returns_qtr1;
   set returns_jan(rename=(ID=CustID)
                   (Return=Item))
       returns_feb(rename=(Dt=Date))
       returns_mar;
run;
```

- You cannot specify more than two data sets in the SET statement.
- There are too many sets of parentheses in the RENAME= option.
- You cannot specify multiple variables in the RENAME= option.
- The BY statement is missing.

7. What is the syntax error in this DATA step?

```
data returns_qtr1;
   set returns_jan(rename=(ID=CustID)
                          (Return=Item))
       returns_feb(rename=(Dt=Date))
       returns_mar;
run;
```

- You cannot specify more than two data sets in the SET statement.
- There are too many sets of parentheses in the RENAME= option.
- You cannot specify multiple variables in the RENAME= option.
- The BY statement is missing.

8. In the second iteration of this DATA step, after the data has been merged, what are the values of **C** and **A**?

```
data client_amount;
   merge clients(in=C)
         amounts(in=A);
   by Name;
run;
```

Clients

Name	EmpID
Ankerton	11123
Davis	22298
Masters	33351
Wolmer	44483

Amounts

Name	Date	Amt
Ankerton	08OCT96	92
Ankerton	15OCT96	43
Davis	04OCT96	16
Masters	.	27
Thomas	21OCT96	15

- C=1, A=0
- C=0, A=1
- C=1, A=1
- missing
- unknown

8. In the second iteration of this DATA step, after the data has been merged, what are the values of **C** and **A**?

```
data client_amount;
   merge clients(in=C)
         amounts(in=A);
   by Name;
run;
```

Clients

Name	EmpID
Ankerton	11123
Davis	22298
Masters	33351
Wolmer	44483

Amounts

Name	Date	Amt
Ankerton	08OCT96	92
Ankerton	15OCT96	43
Davis	04OCT96	16
Masters	.	27
Thomas	21OCT96	15

- C=1, A=0
- C=0, A=1
- C=1, A=1
- missing
- unknown

9. If you run this DATA step, what observations does the data set **bonuses** contain?

```
data bonuses;
   merge managers (in=M)
         staff (in=S);
   by EmpID;
   if M=0 and S=1;
run;
```

a. only the observations from **staff** that have no match in **managers**
b. only the observations from **managers** that have no match in **staff**
c. all observations from both **managers** and **staff**, whether or not they match
d. no observations

9. If you run this DATA step, what observations does the data set **bonuses** contain?

```
data bonuses;
   merge managers (in=M)
         staff (in=S);
   by EmpID;
   if M=0 and S=1;
run;
```

a. only the observations from **staff** that have no match in **managers**

b. only the observations from **managers** that have no match in **staff**

c. all observations from both **managers** and **staff**, whether or not they match

d. no observations

10. What is the relationship of the data set **first** to the data set **second** when merged by the variable **ID**?

- one-to-one
- one-to-many
- many-to-one
- many-to-many
- non-matching

first

Name	ID	Age
Togar	121150	39
Kylie	121152	34
Birin	121153	32
Gloria	121154	12
James	121155	36
Gene	121156	28
Tom	121157	27

second

ID	Date
121150	02/15/05
121152	05/22/07
121153	03/04/06
121154	11/22/05
121155	07/08/06
121156	12/15/06
121157	04/30/07

10. What is the relationship of the data set **first** to the data set **second** when merged by the variable **ID**?

- one-to-one
- one-to-many
- many-to-one
- many-to-many
- non-matching

first

Name	ID	Age
Togar	121150	39
Kylie	121152	34
Birin	121153	32
Gloria	121154	12
James	121155	36
Gene	121156	28
Tom	121157	27

second

ID	Date
121150	02/15/05
121152	05/22/07
121153	03/04/06
121154	11/22/05
121155	07/08/06
121156	12/15/06
121157	04/30/07