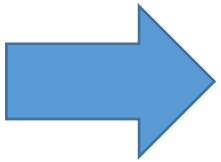
Class : 4

SAS Tutorial

PRESENTED BY : SHASHI KUMAR

Sort / Order the dataset

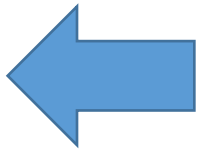
```
data a;
set sashelp.class;
run;


proc sort data=a out=b;
by age;
run;
```



Obs	Name	Sex	Age
1	Alfred	M	14
2	Alice	F	13
3	Barbara	F	13
4	Carol	F	14
5	Henry	M	14
6	James	M	12
7	Jane	F	12
8	Janet	F	15
9	Jeffrey	M	13
10	John	M	12
11	Joyce	F	11
12	Judy	F	14
13	Louise	F	12
14	Mary	F	15
15	Philip	M	16
16	Robert	M	12
17	Ronald	M	15
18	Thomas	M	11
19	William	M	15

	Obs	Name	Sex	Age	Height
1	11	Joyce	F	11	
2	18	Thomas	M	11	
3	6	James	M	12	
4	7	Jane	F	12	
5	10	John	M	12	
6	13	Louise	F	12	
7	16	Robert	M	12	
8	2	Alice	F	13	
9	3	Barbara	F	13	
10	9	Jeffrey	M	13	
11	1	Alfred	M	14	
12	4	Carol	F	14	
13	5	Henry	M	14	
14	12	Judy	F	14	
15	8	Janet	F	15	
16	14	Mary	F	15	
17	17	Ronald	M	15	
18	19	William	M	15	
19	15	Philip	M	16	

If can also alter the default ascending order by using descending followed by var name.



Age
11
11
12
12
12
12
12
13
13
13
14
14
14
14
15
15
15
15
16

	Age
1	16
2	15
3	15
4	15
5	15
6	14
7	14
8	14
9	14
10	13
11	13
12	13
13	12
14	12
15	12
16	12
17	12
18	11
19	11

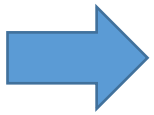
```
Proc sort data=a(keep= age) out=c(keep=age);
by descending age;
run;
```

Caution: If you don't specify out= then parent dataset will be changed

nodupkey, nodup & nouniquekey options

Three

	eid	name	sex
1	1	A	M
2	1	A	F
3	2	B	M
4	2	C	M
5	3	D	F
6	3	D	M
7	1	E	M
8	1	A	M
9	2	F	M
10	2	C	M
11	4	G	M
12	5	H	M
13	5	H	F



Threes

	eid	name	sex
1	1	A	M
2	1	A	F
3	1	E	M
4	1	A	M
5	2	B	M
6	2	C	M
7	2	F	M
8	2	C	M
9	3	D	F
10	3	D	M
11	4	G	M
12	5	H	M
13	5	H	F

Four

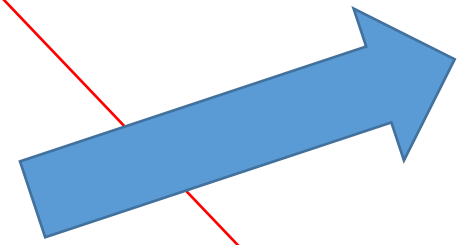
	eid	name	sex
1	1	A	M
2	2	B	M
3	3	D	F
4	4	G	M
5	5	H	M

proc sort data=three out=four nodupkey;
by eid;
run;

proc sort data=three out=threes;
by eid;**run**;

nodupkey= it keep the first unique observation according to key variable(variable present in the by statement)

	eid	name	sex
1	1	A	M
2	1	A	F
3	1	E	M
4	1	A	M
5	2	B	M
6	2	C	M
7	2	F	M
8	2	C	M
9	3	D	F
10	3	D	M
11	4	G	M
12	5	H	M
13	5	H	F



FIVE

	eid	name	sex
1	1	A	F
2	1	A	M
3	1	E	M
4	2	B	M
5	2	C	M
6	2	F	M
7	3	D	F
8	3	D	M
9	4	G	M
10	5	H	F
11	5	H	M

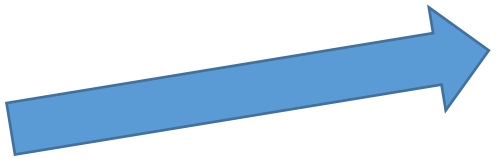
	eid	name	sex
1	1	A	M
2	2	C	M

SIX

```
proc sort data=three out=five nodup dupout=six;
by _all_;run;
```

nodup=It keep the first unique observation, if entire row is duplicate

	eid	name	sex
1	1	A	M
2	1	A	F
3	1	E	M
4	1	A	M
5	2	B	M
6	2	C	M
7	2	F	M
8	2	C	M
9	3	D	F
10	3	D	M
11	4	G	M
12	5	H	M
13	5	H	F



Seven

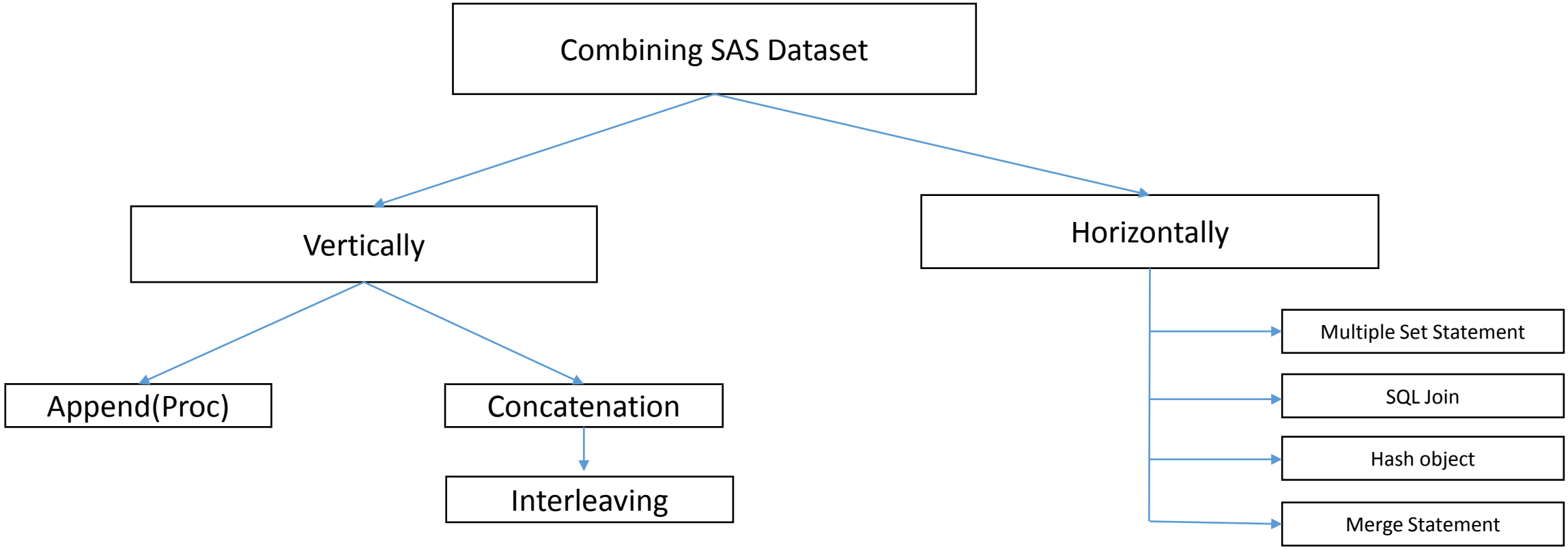
	eid	name	sex
1	1	A	M
2	1	A	F
3	1	E	M
4	1	A	M
5	2	B	M
6	2	C	M
7	2	F	M
8	2	C	M
9	3	D	F
10	3	D	M
11	5	H	M
12	5	H	F

	eid	name	sex
1	4	G	M

Eight

proc sort data=three out=seven nouniquekey uniqueout=eight;
by eid;**run**;

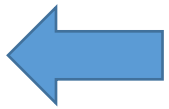
- nouniquekey = It eliminate unique records based on by variable.
- Duplicate record deleted from original dataset can be saved in another dataset by using the option uniqueout.


Combining SAS Dataset
Vertically
Horizontally
Append(Proc)
Concatenation
Interleaving
Multiple Set Statement
SQL Join
Hash object
Merge Statement

Appending & Interleaving

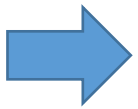
- Appending is joining two datasets vertically.
- Mention dataset name/s in single set statement to append.
- Make sure the variable/s names should be same otherwise unwanted result would come.
- Interleaving combines individual sorted dataset into one sorted dataset by specified variable in the by statement.

card	amount
1	100
1	200
1	100
2	200
2	300
1	100
1	200
1	100
2	200
2	300



```
data append;
set a a;
run;
```

```
data interleave;
set a a;
by card;
run;
```



card	amount
1	100
1	200
1	100
1	100
1	200
1	100
2	200
2	300
2	200
2	300

Merging

What's need to have a successful merging??

- There should be at least one key variable else unwanted result will populate.
- Key variable values should be sorted in both dataset.
- Key variable attribute i.e format, length , alignment should be same.

- ❑zero-to-one
- ❑one-to-zero
- ❑ one-to-one
- ❑one-to-many
- ❑ many-to-one
- ❑ few-to-many
- ❑many-to-few
- ❑many-to-many

Types of Matched Merging

Merging

Different Cases of Merging

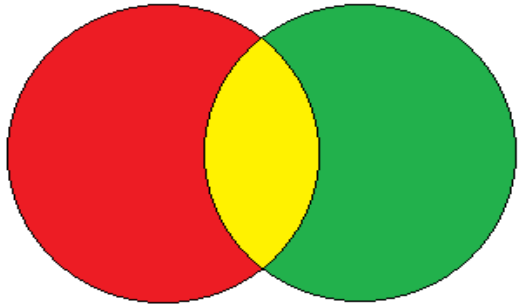
- You can do it by merge statement
- You can do it also with set statement
- Be very careful when you have different # of records in both tables i.e in few to many or few to many cases

Data one			Data Two			Merging Type
ID	Var1		ID	Var2		
A01	53		A01	40		Zero to One
A02	24		A02	62		
A05	86		A04	71		One to Zero
A10	64		A10	40		
A25	18		A25	27		One to One
A25	96		A25	72		
A25	66		A25	90		One to Many
A25	41		A25	58		
A32	63		A32	10		Many to One
A55	16		A32	20		
A55	51		A32	19		Few to Many
A55	61		A55	99		
A92	40		A92	71		Many to Few
A92	34		A92	87		
A96	11		A92	51		Mant to Many
A96	56		A96	94		
A96	50		A96	31		

Full Join

```
data M;
merge A (in = x) B (in = y);
by id;
if x = 1 or y = 1;
run;
```

```
Proc sql;
create table M2 as
select coalesce(a.id, b.id) as id, gender, sex
from A full join B on a.id = b.id;
quit;
```



	ID	Gender
1	1	M
2	2	M
3	3	M
4	4	M
5	5	M
6	6	F
7	7	F
8	8	M
9	9	M
10	10	M

	ID	Sex
1	6	F
2	7	F
3	8	M
4	9	M
5	10	M
6	11	F
7	12	F
8	13	F
9	14	F
10	15	F

	ID	Gender	Sex
1	1	M	
2	2	M	
3	3	M	
4	4	M	
5	5	M	
6	6	F	F
7	7	F	F
8	8	M	M
9	9	M	M
10	10	M	M
11	11		F
12	12		F
13	13		F
14	14		F
15	15		F

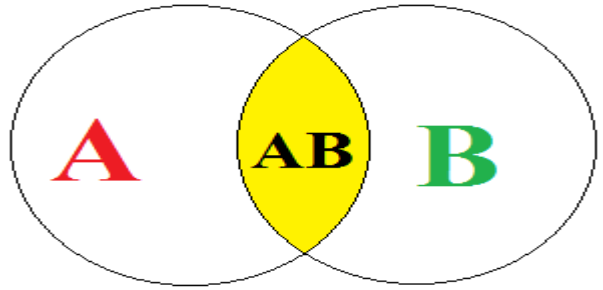
A

B

Inner Join

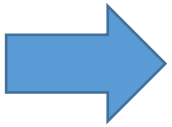
```
data M;
merge A (in = x) B (in = y);
by id;
if x = 1 and y = 1;
run;
```

```
Proc sql;
create table M2 as
select coalesce(a.id, b.id) as id, gender, sex
from A inner join B on a.id = b.id;
quit;
```



	ID	Gender
1	1	M
2	2	M
3	3	M
4	4	M
5	5	M
6	6	F
7	7	F
8	8	M
9	9	M
10	10	M

	ID	Sex
1	6	F
2	7	F
3	8	M
4	9	M
5	10	M
6	11	F
7	12	F
8	13	F
9	14	F
10	15	F

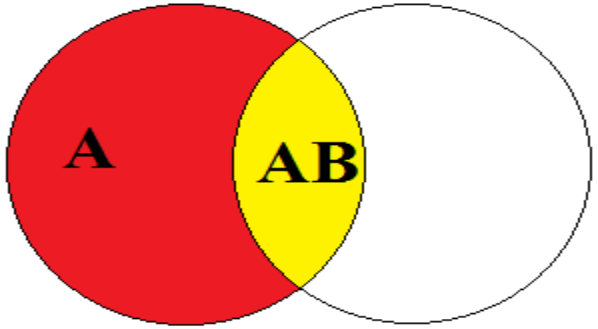


	ID	Gender	Sex
1	6	F	F
2	7	F	F
3	8	M	M
4	9	M	M
5	10	M	M

Left Join

```
data M;
merge A (in = x) B (in = y);
by id;
if x = 1 ;
run;
```

```
Proc sql;
create table M2 as
select coalesce(a.id, b.id) as id, gender, sex
from A left join B on a.id = b.id;
quit;
```



	ID	Gender
1	1	M
2	2	M
3	3	M
4	4	M
5	5	M
6	6	F
7	7	F
8	8	M
9	9	M
10	10	M

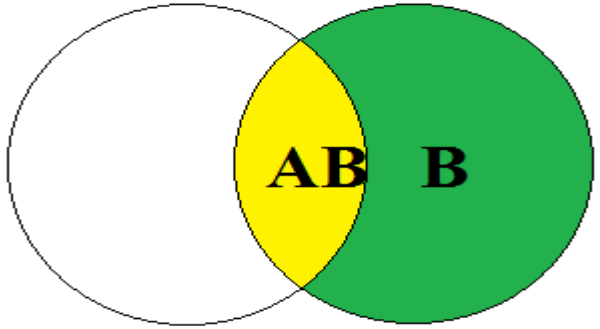
	ID	Sex
1	6	F
2	7	F
3	8	M
4	9	M
5	10	M
6	11	F
7	12	F
8	13	F
9	14	F
10	15	F

	ID	Gender	Sex
1	1	M	
2	2	M	
3	3	M	
4	4	M	
5	5	M	
6	6	F	F
7	7	F	F
8	8	M	M
9	9	M	M
10	10	M	M

Right Join

```
data M;
merge A (in = x) B (in = y);
by id;
if y = 1 ;
run;
```

```
Proc sql;
create table M2 as
select coalesce(a.id, b.id) as id, gender, sex
from A right join B on a.id = b.id;
quit;
```



	ID	Gender
1	1	M
2	2	M
3	3	M
4	4	M
5	5	M
6	6	F
7	7	F
8	8	M
9	9	M
10	10	M

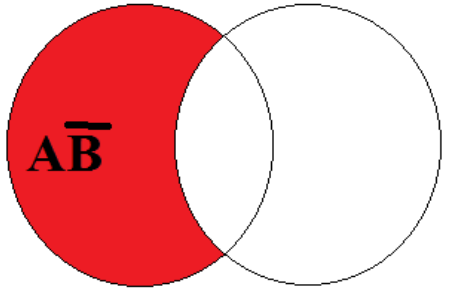
	ID	Sex
1	6	F
2	7	F
3	8	M
4	9	M
5	10	M
6	11	F
7	12	F
8	13	F
9	14	F
10	15	F

	ID	Gender	Sex
1	6	F	F
2	7	F	F
3	8	M	M
4	9	M	M
5	10	M	M
6	11		F
7	12		F
8	13		F
9	14		F
10	15		F

Non Matching From A

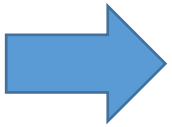
```
data M;
merge A (in = x) B (in = y);
by id;
If  x=1 and y = 0 ;
run;
```

```
Proc Sql;
create table Q2 as
select coalesce(a.id, b.id) as id, gender, sex
from A left join B on a.id = b.id
where b.id is null;
quit;
```



	ID	Gender
1	1	M
2	2	M
3	3	M
4	4	M
5	5	M
6	6	F
7	7	F
8	8	M
9	9	M
10	10	M

	ID	Sex
1	6	F
2	7	F
3	8	M
4	9	M
5	10	M
6	11	F
7	12	F
8	13	F
9	14	F
10	15	F

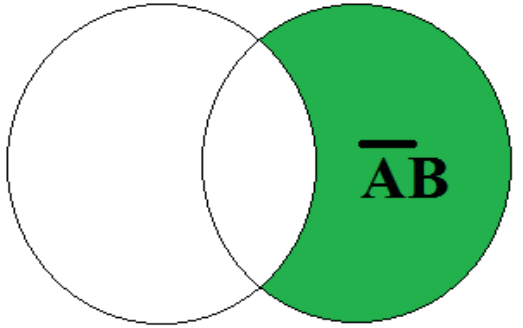


	ID	Gender	Sex
1	1	M	
2	2	M	
3	3	M	
4	4	M	
5	5	M	

Non Matching From B

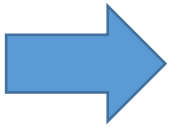
```
data M;
merge A (in = x) B (in = y);
by id;
If  x=0 and y = 1 ;
run;
```

```
Proc Sql;
create table Q2 as
select coalesce(a.id, b.id) as id, gender, sex
from A right join B on a.id = b.id
where a.id is null;
quit;
```



	ID	Gender
1	1	M
2	2	M
3	3	M
4	4	M
5	5	M
6	6	F
7	7	F
8	8	M
9	9	M
10	10	M

	ID	Sex
1	6	F
2	7	F
3	8	M
4	9	M
5	10	M
6	11	F
7	12	F
8	13	F
9	14	F
10	15	F

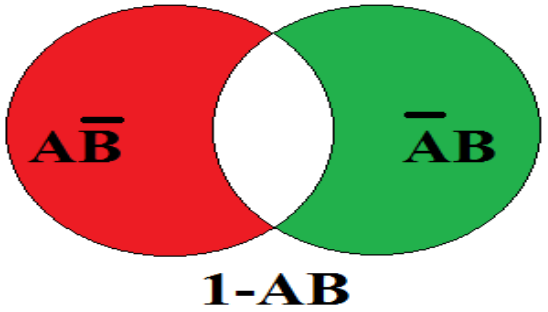


	ID	Gender	Sex
1	11		F
2	12		F
3	13		F
4	14		F
5	15		F

Non Matching From A and B

```
data M;
merge A (in = x) B (in = y);
by id;
If  (x=0 and y = 1) or (y=0 and x = 1) ;
run;
```

```
Proc Sql;
create table Q2 as
select coalesce(a.id, b.id) as id, gender, sex
from A right join B on a.id = b.id
where a.id is null or b.id is null ;
quit;
```



	ID	Gender
1	1	M
2	2	M
3	3	M
4	4	M
5	5	M
6	6	F
7	7	F
8	8	M
9	9	M
10	10	M

	ID	Sex
1	6	F
2	7	F
3	8	M
4	9	M
5	10	M
6	11	F
7	12	F
8	13	F
9	14	F
10	15	F

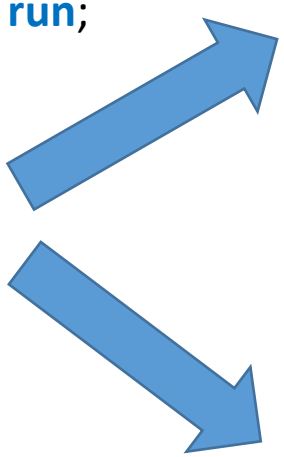
	ID	Gender	Sex
1	1	M	
2	2	M	
3	3	M	
4	4	M	
5	5	M	
6	11		F
7	12		F
8	13		F
9	14		F
10	15		F

Many to Many

	ID	Info
1	1	3123
2	1	1234
3	2	7482
4	2	8912
5	3	1284

	ID	Info2
1	1	4444
2	1	5555
3	1	8989
4	2	9099
5	2	8888
6	3	8989

```
data combined;
merge dat1 dat2 ;
by ID;
run;
```



	ID	Info	Info2
1	1	3123	4444
2	1	1234	5555
3	1	1234	8989
4	2	7482	9099
5	2	8912	8888
6	3	1284	8989

	ID	Info	Info2
1	1	3123	4444
2	1	1234	4444
3	1	3123	5555
4	1	1234	5555
5	1	3123	8989
6	1	1234	8989
7	2	7482	9099
8	2	8912	9099
9	2	7482	8888
10	2	8912	8888
11	3	1284	8989

```
proc sql ;
create table combined2 as
select coalesce(dat1.id, dat2.id) as id ,info,info2
from dat1  full join dat2  on dat1.ID = dat2.ID;
quit;
```