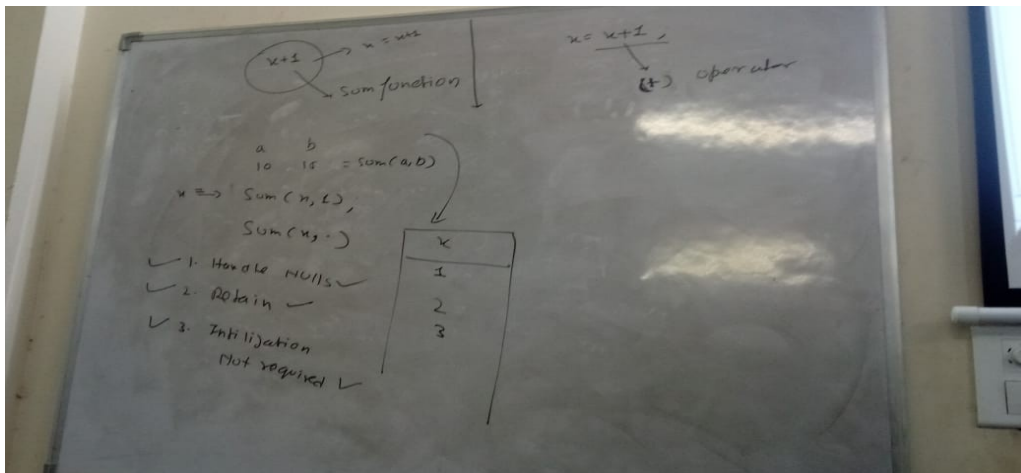


Dated: 24/03/2018

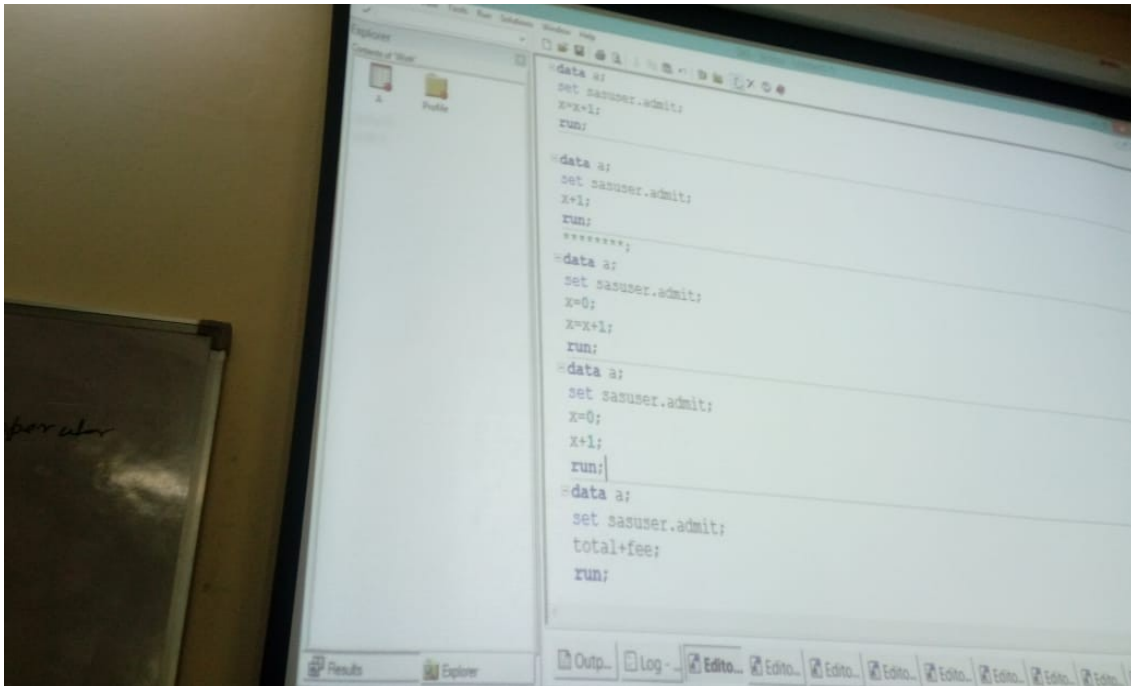
Statistical Analysis System: Class 10

.....
.....

Differentiating $x=x+1$ & $x+1$ statement:



$X=x+1$	$X+1$
<ul style="list-style-type: none">• It's an arithmetic (+) operator.	<ul style="list-style-type: none">• It's a SUM function.
<ul style="list-style-type: none">• Can't handle null value.	<ul style="list-style-type: none">• Can handle null value.
<ul style="list-style-type: none">• Doesn't retain the value	<ul style="list-style-type: none">• Can retain the value
<ul style="list-style-type: none">• Requires initialization of variable.	<ul style="list-style-type: none">• Doesn't require initialization of variable.



Example 1:

Data a;

Set sasuser.admit;

$X = x + 1$;

Run;

Explained: X, here is not defined explicitly and $x = x + 1$, does not retain value, also X here will be a null value by default therefore any operation with X will also result in a null value.

Output (S.A dataset):

ID	Name	Sex	Age	Date	Height	Weight	AcclLevel	Fee	x
1	2458 Murray, W	M	27		1	72	123 HIGH	85.20	
2	2462 Almers, C	F	34		3	66	152 HIGH	124.80	
3	2501 Bonaventura, T	F	31		17	61	123 LOW	149.75	
4	2523 Johnson, R	F	43		31	63	137 MOD	149.75	
5	2539 LaMance, K	M	51		4	71	158 LOW	124.80	
6	2544 Jones, M	M	29		6	76	193 HIGH	124.80	
7	2552 Reberson, P	F	32		9	67	151 MOD	149.75	
8	2555 King, E	M	35		13	70	173 MOD	149.75	
9	2563 Pitts, D	M	34		22	73	154 LOW	124.80	
10	2568 Eberhardt, S	F	49		27	64	172 LOW	124.80	
11	2571 Nunnally, A	F	44		19	66	140 HIGH	149.75	
12	2573 Almaraz, M	F	58		17	63	118 LOW	149.75	
13	2574 Peterson, V	M	30		6	69	147 LOW	149.75	
14	2575 Gungley, M	F	40		8	69	163 HIGH	124.80	
15	2578 Cameron, L	M	47		5	72	173 NA	124.80	
16	2579 Underwood, K	M	60		22	71	191 LOW	149.75	
17	2594 Takahashi, Y	F	43		29	65	123 MODY	124.80	
18	2596 Derber, B	M	25		23	75	188 HIGH	85.20	
19	2598 Ivan, H	F	22		20	63	139 LOW	85.20	
20	2599 Wilcox, E	F	41		16	67	141 HIGH	149.75	
21	2595 Warren, C	M	54		7	71	183 MOD	149.75	

Example 2:

Data a;

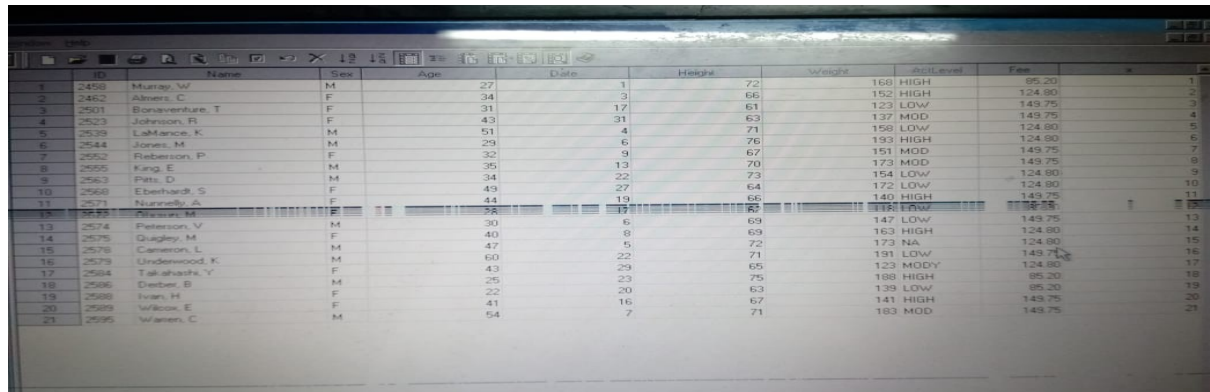
Set sasuser.admit;

```
x+1;
```

```
Run;
```

Explained: x+1, statement here works as a function, retains value, will act as a counter, will have values ranging from 1 to 21 wrt S.A dataset.

Output (S.A dataset):



	Name	Sex	Age	Date	Height	Weight	Fee
1	Murray, W.	M	27	1	72	168 HIGH	95.20
2	Almers, C.	F	34	3	66	152 HIGH	124.90
3	Bonaventure, T.	F	31	17	61	123 LOW	149.75
4	Johnson, R.	F	43	31	63	137 MOD	149.75
5	LaMarche, K.	M	51	4	71	158 LOW	124.80
6	Jones, M.	M	29	6	76	193 HIGH	124.90
7	Reberson, P.	F	32	9	67	151 MOD	149.75
8	King, E.	F	35	13	70	173 MOD	149.75
9	Pitts, D.	M	34	22	73	154 LOW	124.90
10	Eberhardt, S.	F	49	27	64	172 LOW	124.90
11	Nunnally, A.	F	44	19	66	140 HIGH	149.75
12	Almer, M.	F	52	15	81	181 LOW	95.20
13	Peterson, V.	M	30	6	69	147 LOW	149.75
14	Gugley, M.	F	40	8	69	163 HIGH	124.90
15	Cameron, L.	M	47	5	72	173 NA	124.80
16	Underwood, K.	M	60	22	71	191 LOW	149.75
17	Takahashi, Y.	F	43	29	65	123 MOD	124.90
18	Dexter, B.	M	25	23	75	188 HIGH	95.20
19	Ivan, H.	F	22	20	63	139 LOW	95.20
20	Wilcox, E.	F	41	16	67	141 HIGH	149.75
21	Waters, C.	M	54	7	71	183 MOD	149.75

Example 3:

```
Data a;
```

```
Set sasuser.admit;
```

```
X=0;
```

```
X=x+1;
```

```
Run;
```

Explained: X = 0, is defined explicitly and will get executed everytime in the dataset. However, X= x+1 still doesn't retain the value (characteristic of this statement) therefore will act as x=0+1 everytime resulting into X = 1 as output everywhere.

Example 4:

```
Data a;
```

```
Set sasuser.admit;
```

```
X=0;
```

```
X+1;
```

```
Run;
```

Explained: X = 0, is defined explicitly and will get executed everytime in the dataset. Here, X+1 retains the value (characteristic of this statement) but since X=0 is defined therefore x+1 will return 0+1 = 1 everytime as output.

Output (S.A dataset): for example 3 & example 4

	ID	Name	Sex	Age	Date	Height	Weight	ActLevel	Fee	total
1	2458	Murray, W	M	27	1	72	168	HIGH	85.20	85.2
2	2462	Almers, C	F	34	3	66	152	HIGH	124.80	210
3	2501	Bonaventure, T	F	31	17	61	123	LOW	149.75	359.75
4	2523	Johnson, R	F	43	31	63	137	MOD	149.75	509.5
5	2539	LaMance, K	M	51	4	71	158	LOW	124.80	634.3
6	2544	Jones, M	M	29	6	76	193	HIGH	124.80	759.1
7	2552	Reberson, P	F	32	9	67	151	MOD	149.75	908.85
8	2555	King, E	M	35	13	70	173	MOD	149.75	1058.6
9	2563	Pitts, D	M	34	22	73	154	LOW	124.80	1183.4
10	2568	Eberhardt, S	F	49	27	64	172	LOW	124.80	1308.2
11	2571	Nunnally, A	F	44	19	66	140	HIGH	149.75	1457.95
12	2573	Altmann, M	F	38	17	63	118	LOW	85.20	1543.15
13	2574	Peterson, V	M	30	6	69	147	LOW	149.75	1692.9
14	2575	Gugley, M	F	40	8	69	163	HIGH	124.80	1817.7
15	2576	Cameron, L	F	47	5	72	173	NA	124.80	1942.5
16	2579	Underwood, K	M	60	22	71	191	LOW	149.75	2092.25
17	2584	Takahashi, Y	F	43	29	65	123	MODY	124.80	2217.05
18	2586	Darbee, B	M	25	23	75	198	HIGH	85.20	2302.25
19	2588	Ivan, H	F	22	20	63	139	LOW	85.20	2387.45
20	2589	Wilcox, E	F	41	16	67	141	HIGH	149.75	2537.2
21	2595	Warren, C	M	54	7	71	183	MOD	149.75	2686.95

Example 5:

Data a;

Set sasuser.admit;

total + fee;

Run;

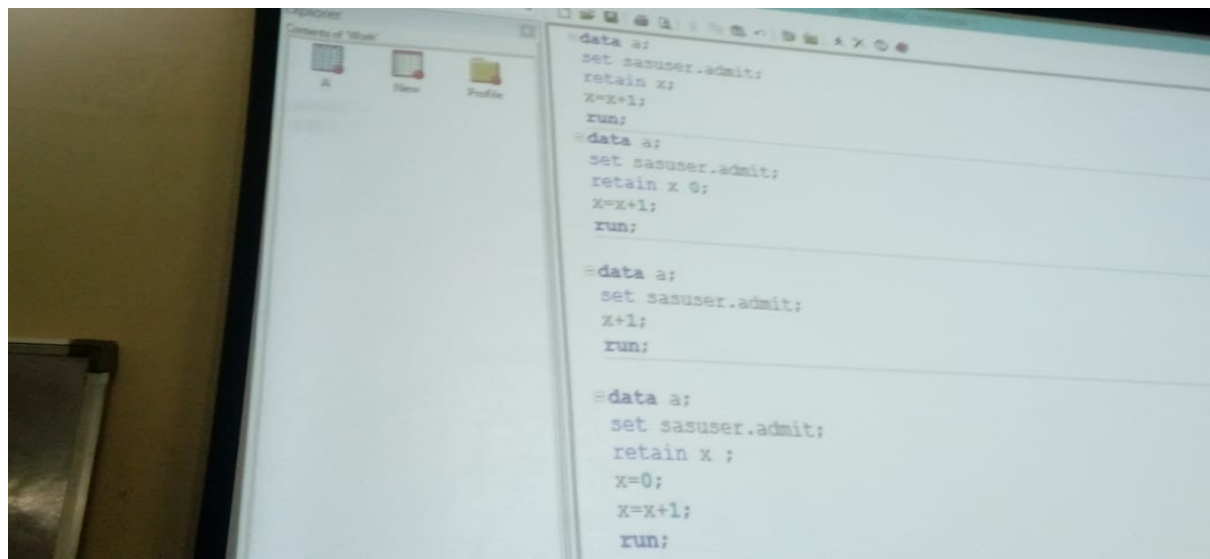
Explained: This creates a new variable “total” in the S.A dataset which will retain its value. Also “total” will have values starting from 1st value of fee and then adding 2nd value of fee to existing “total” to get the 2nd value of total and likewise for all 21 observations in S.A dataset.

Output (S.A dataset):

	ID	Name	Sex	Age	Date	Height	Weight	ActLevel	Fee	total
1	2458	Murray, W	M	27	1	72	168	HIGH	85.20	85.2
2	2462	Almers, C	F	34	3	66	152	HIGH	124.80	210
3	2501	Bonaventure, T	F	31	17	61	123	LOW	149.75	359.75
4	2523	Johnson, R	F	43	31	63	137	MOD	149.75	509.5
5	2539	LaMance, K	M	51	4	71	158	LOW	124.80	634.3
6	2544	Jones, M	M	29	6	76	193	HIGH	124.80	759.1
7	2552	Reberson, P	F	32	9	67	151	MOD	149.75	908.85
8	2555	King, E	M	35	13	70	173	MOD	149.75	1058.6
9	2563	Pitts, D	M	34	22	73	154	LOW	124.80	1183.4
10	2568	Eberhardt, S	F	49	27	64	172	LOW	124.80	1308.2
11	2571	Nunnally, A	F	44	19	66	140	HIGH	149.75	1457.95
12	2573	Altmann, M	F	38	17	63	118	LOW	85.20	1543.15
13	2574	Peterson, V	M	30	6	69	147	LOW	149.75	1692.9
14	2575	Gugley, M	F	40	8	69	163	HIGH	124.80	1817.7
15	2576	Cameron, L	F	47	5	72	173	NA	124.80	1942.5
16	2579	Underwood, K	M	60	22	71	191	LOW	149.75	2092.25
17	2584	Takahashi, Y	F	43	29	65	123	MODY	124.80	2217.05
18	2586	Darbee, B	M	25	23	75	198	HIGH	85.20	2302.25
19	2588	Ivan, H	F	22	20	63	139	LOW	85.20	2387.45
20	2589	Wilcox, E	F	41	16	67	141	HIGH	149.75	2537.2
21	2595	Warren, C	M	54	7	71	183	MOD	149.75	2686.95

Working with Retain (keyword):

Using “retain” will always store the value of the variable used along with it.



Example 6:

Data a;

Set sasuser.admit;

retain x;

x = x+1;

Run;

Explained: retain x, retains value of x, in the statement $x = x+1$, x is not defined so has null value therefore output will also be null everywhere.

Output (S.A dataset):

ID	Name	Sex	Age	Date	Height	Weight	AcclLevel	Fee
1	2458 Murray, W	M	27	1	72	152	HIGH	85.20
2	2462 Almers, C	F	34	3	66	123	LOW	124.80
3	2501 Bonaventure, T	F	31	17	61	137	MOD	149.75
4	2523 Johnson, R	F	43	31	63	156	LOW	124.80
5	2539 LaMance, K	M	51	4	71	193	HIGH	124.80
6	2544 Jones, M	M	29	6	76	151	MOD	149.75
7	2552 Rebersen, P	F	32	9	67	173	MOD	149.75
8	2555 King, E	M	35	13	70	154	LOW	124.80
9	2563 Pitts, D	M	34	22	73	172	LOW	124.80
10	2569 Eberhardt, S	F	49	27	64	140	HIGH	149.75
11	2571 Nunnelly, A	F	44	19	66	116	LOW	149.75
12	2573 Alvaraz, M	F	58	12	63	163	HIGH	124.80
13	2574 Peterson, V	M	30	6	69	173	NA	124.80
14	2575 Gungley, M	F	40	8	69	191	LOW	149.75
15	2578 Cameron, L	M	47	5	72	188	HIGH	85.20
16	2579 Underwood, K	M	60	22	71	123	MOD	124.80
17	2584 Takahashi, Y	F	25	23	75	139	LOW	85.20
18	2586 Derber, B	F	22	20	63	141	HIGH	149.75
19	2589 Iwan, H	F	41	16	67	183	MOD	149.75
20	2599 Wilcox, E	F	41	16	67	183	MOD	149.75
21	2595 Warren, C	M	54	7	71	183	MOD	149.75

Example 7:

Data a;

Set sasuser.admit;

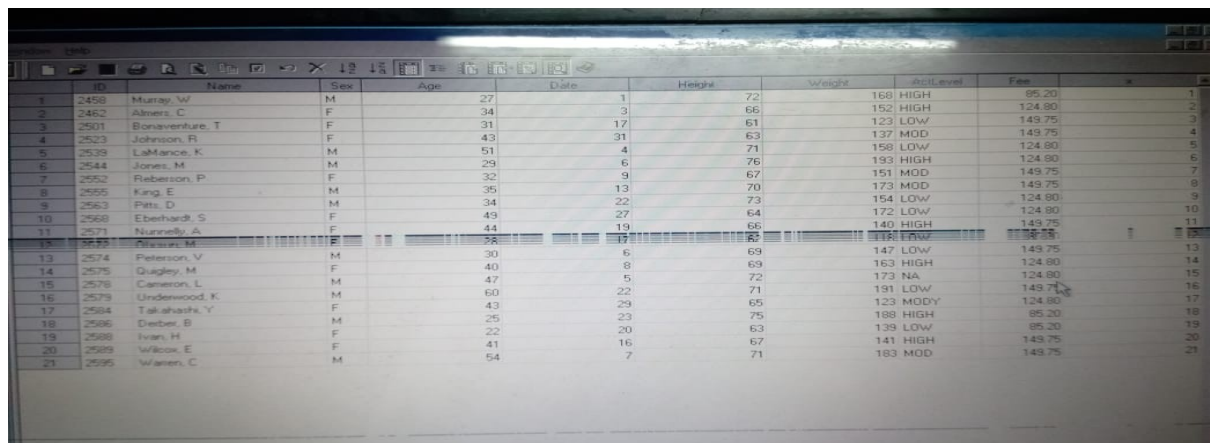
retain x 0;

x = x+1;

Run;

Explained: Retain x, retains value of x, also defines initial value of X = 0. Now since X is defined, X=x+1 will act as counter giving output values of X ranging from 1-21.

Output (S.A dataset):



x	ID	Name	Sex	Age	Date	Height	Weight	actLevel	Fee	x
1	2458	Murray, W.	M	27	1	72	168	HIGH	95.20	1
2	2462	Almers, C.	F	34	3	66	152	HIGH	124.80	2
3	2501	Bonavenure, T.	F	31	17	61	123	LOW	149.75	3
4	2523	Johnson, R.	F	43	31	63	137	MOD	149.75	4
5	2539	LaMance, K.	M	51	4	71	158	LOW	124.80	5
6	2544	Jones, M.	M	29	6	76	193	HIGH	124.80	6
7	2552	Reberson, P.	F	32	9	67	151	MOD	149.75	7
8	2555	King, E.	M	35	13	70	173	MOD	149.75	8
9	2563	Pati, D.	M	34	22	73	154	LOW	124.80	9
10	2568	Eberhardt, S.	F	49	27	64	172	LOW	124.80	10
11	2571	Nunnally, A.	F	44	19	66	140	HIGH	149.75	11
12	2575	Olsson, M.	F	55	17	65	184	LOW	149.75	12
13	2574	Peterson, V.	M	30	6	69	147	LOW	149.75	13
14	2575	Gugley, M.	F	40	8	69	163	HIGH	124.80	14
15	2576	Cameron, L.	M	47	5	72	173	NA	124.80	15
16	2579	Underwood, K.	M	60	22	71	191	LOW	149.75	16
17	2584	Takahashi, Y.	F	43	29	65	123	MOD	124.80	17
18	2586	Detber, B.	M	25	23	75	188	HIGH	95.20	18
19	2588	Ivan, H.	F	22	20	63	139	LOW	95.20	19
20	2589	Wilcox, E.	F	41	16	67	141	HIGH	149.75	20
21	2595	Waters, C.	M	54	7	71	183	MOD	149.75	21

Example 8:

Data a;

Set sasuser.admit;

retain x ;

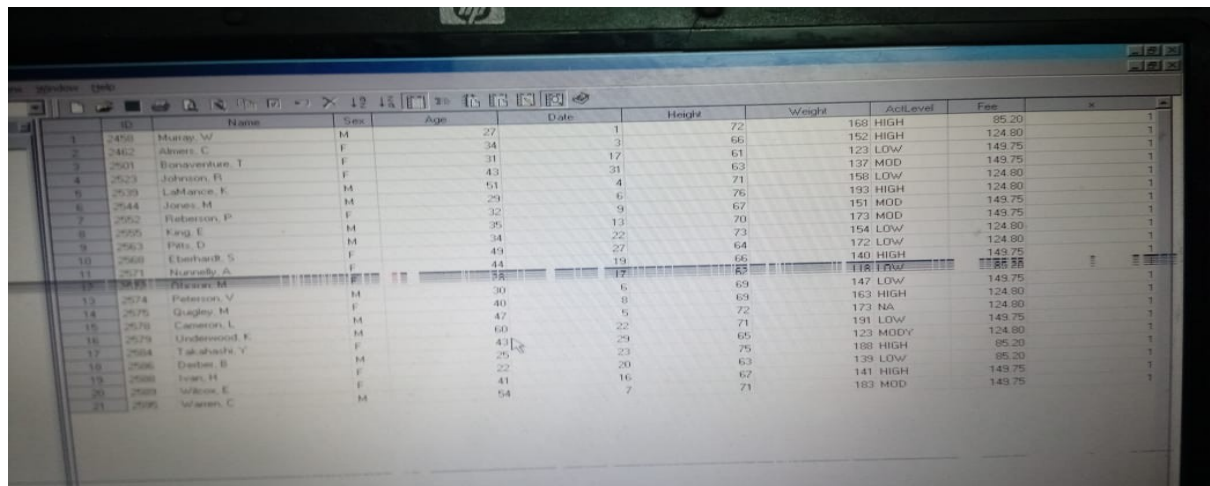
x=0;

x = x+1;

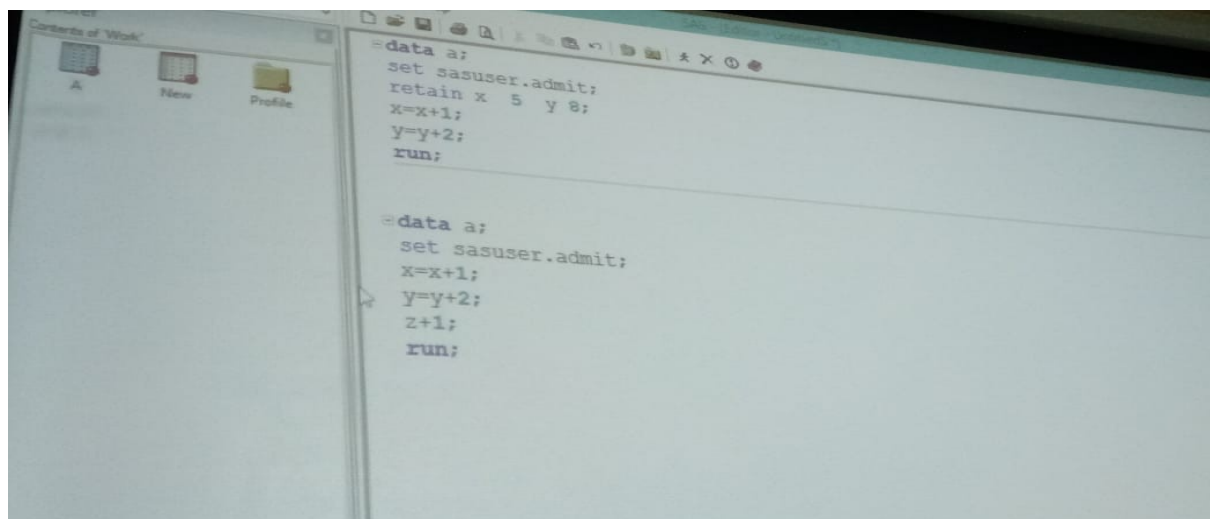
Run;

Explained: Retain x, retains value of x, explicitly defines initial value of X = 0 which gets executed everytime in the dataset. Here, X=x+1 will work as x = 0+1 everytime because x=0, resets x everytime. Therefore S.A will have X as 1 everywhere.

Output (S.A dataset):



	ID	Name	Sex	Age	Date	Height	Weight	ActLevel	Fee	
1	2450	Murray, W	M	27		72	168	HIGH	95.20	1
2	2462	Adams, C	F	34		66	152	HIGH	124.80	1
3	2501	Bondaventura, T	F	31		17	61	123 LOW	149.75	1
4	2523	Johnson, R	F	43		31	63	137 MOD	149.75	1
5	2539	LaManco, K	M	51		4	71	158 LOW	124.80	1
6	2544	Jones, M	M	29		6	76	193 HIGH	149.75	1
7	2552	Roberson, P	F	32		9	67	151 MOD	149.75	1
8	2555	King, E	M	35		13	70	173 MOD	124.80	1
9	2563	Pitts, D	M	34		22	73	154 LOW	124.80	1
10	2568	Eberhardt, S	F	49		27	64	172 LOW	124.80	1
11	2571	Nunnally, A	F	44		19	66	140 HIGH	149.75	1
12	2583	Alfonso, M	M	38		17	63	118 LOW	149.75	1
13	2574	Peterson, V	M	30		6	69	147 LOW	149.75	1
14	2575	Quigley, M	F	40		8	69	163 HIGH	124.80	1
15	2576	Cameron, L	M	47		5	72	173 NA	124.80	1
16	2579	Underwood, K	M	60		22	71	191 LOW	149.75	1
17	2584	Takahashi, Y	F	43		29	65	123 MODY	124.80	1
18	2586	Darbin, E	F	25		23	75	188 HIGH	95.20	1
19	2588	Dean, H	F	22		20	63	139 LOW	95.20	1
20	2589	Wahne, E	F	41		16	67	141 HIGH	149.75	1
21	2595	Warren, C	M	54		7	71	183 MOD	149.75	1



```
=data a;
set sasuser.admit;
retain x 5 y 8;
x=x+1;
y=y+2;
run;

=data a;
set sasuser.admit;
x=x+1;
y=y+2;
z+1;
run;
```

Example 9:

Data a;

Set sasuser.admit;

retain x 5 y 8;

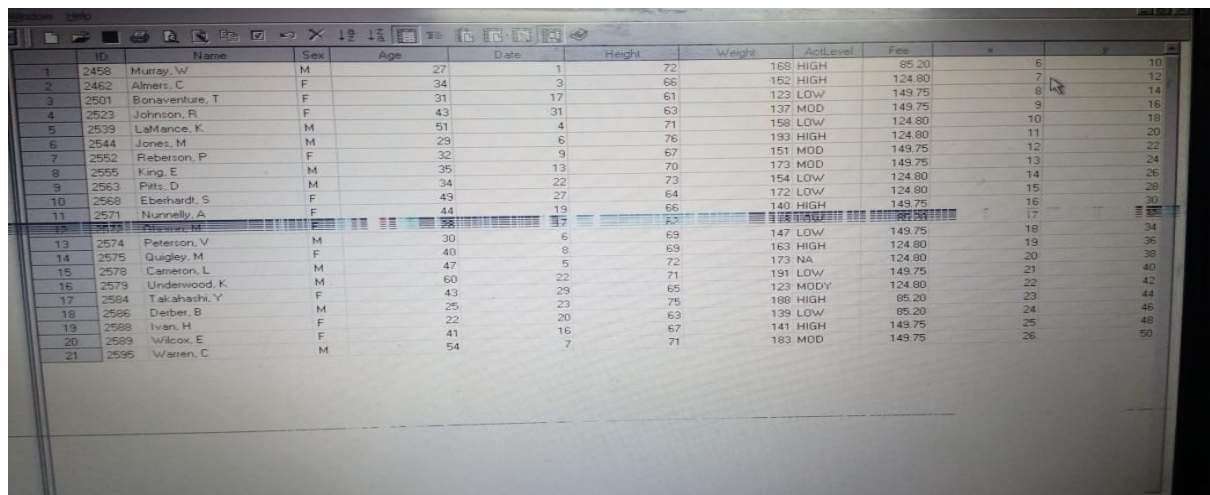
x = x+1;

y= y+2;

Run;

Explained: Retains value of x & y, also explicitly defines initial value of X = 5, y = 8, in the output dataset X will increase by 1 everytime starting with 6, Y will increase by 2 everytime starting with 10 in the output dataset.

Output (S.A dataset):



	ID	Name	Sex	Age	Date	Height	Weight	ActLevel	Fee	x	y	z
1	2458	Murray, W	M	27	1	72	168	HIGH	85.20	6	10	10
2	2462	Almers, C	F	34	3	66	152	HIGH	124.80	7	12	12
3	2501	Bonaventure, T	F	31	17	61	123	LOW	149.75	8	14	14
4	2523	Johnson, R	F	43	31	63	137	MOD	149.75	9	16	16
5	2539	LaMance, K	M	51	4	71	158	LOW	124.80	10	18	18
6	2544	Jones, M	M	29	6	76	193	HIGH	124.80	11	20	20
7	2552	Reberson, P	F	32	9	67	151	MOD	149.75	12	22	22
8	2555	King, E	M	35	13	70	173	MOD	149.75	13	24	24
9	2563	Pitts, D	M	34	22	73	154	LOW	124.80	14	26	26
10	2568	Eberhardt, S	F	49	27	64	172	LOW	124.80	15	28	28
11	2571	Nunnally, A	F	44	19	66	140	HIGH	149.75	16	30	30
12	2573	Blissner, M	F	52	17	67	173	MOD	149.75	17	32	32
13	2574	Peterson, V	M	30	6	69	147	LOW	149.75	18	34	34
14	2575	Quigley, M	F	40	8	69	163	HIGH	124.80	19	36	36
15	2578	Cameron, L	M	47	5	72	173	NA	124.80	20	38	38
16	2579	Underwood, K	F	60	22	71	191	LOW	149.75	21	40	40
17	2584	Takahashi, Y	F	43	29	65	123	MODV	124.80	22	42	42
18	2586	Derber, B	M	25	23	75	188	HIGH	85.20	24	46	46
19	2588	Ivan, H	F	22	20	63	139	LOW	85.20	25	48	48
20	2589	Wilcox, E	F	41	16	67	141	HIGH	149.75	26	50	50
21	2595	Warren, C	M	54	7	71	183	MOD	149.75			

Example 10:

Data a;

Set sasuser.admit;

x = x+1;

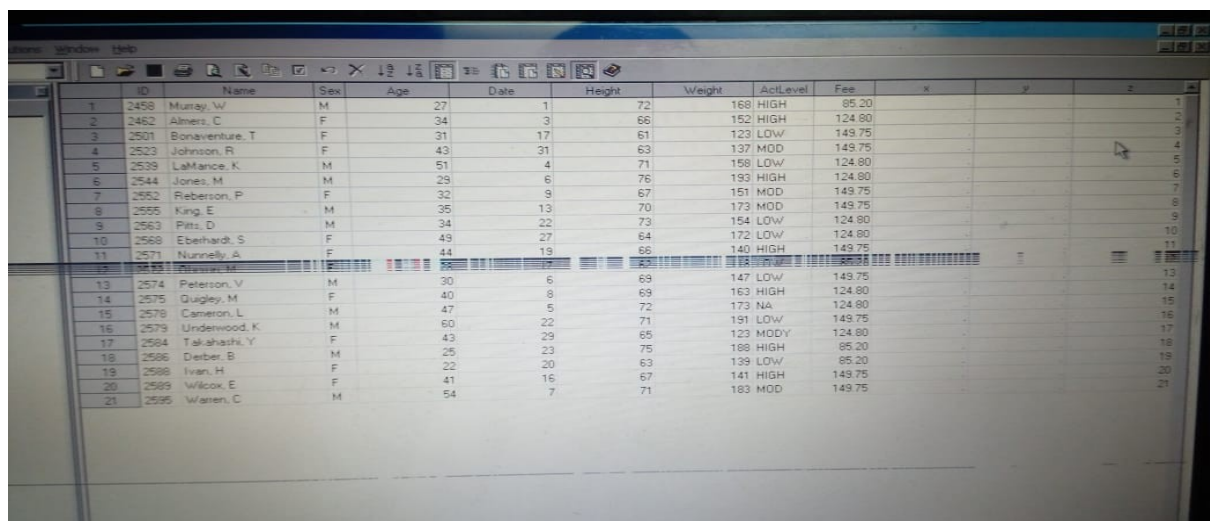
y= y+2;

z+1;

Run;

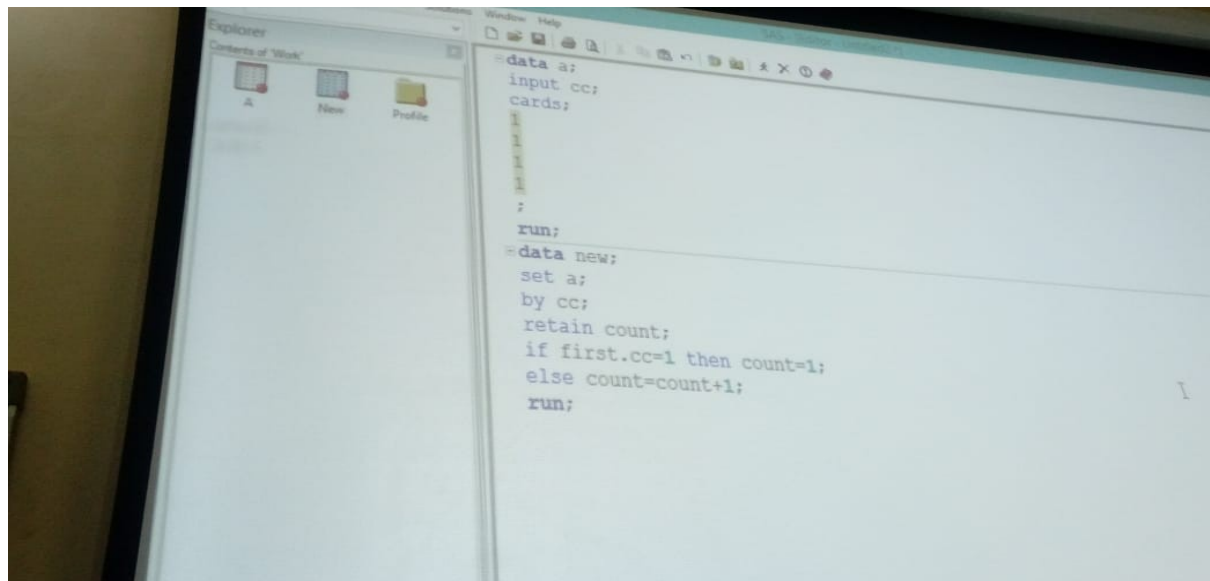
Explained: X,Y not defined so will result into null values in the output dataset, z+1 will act as counter and will have values ranging from 1-21 in the output dataset.

Output (S.A dataset):



	ID	Name	Sex	Age	Date	Height	Weight	ActLevel	Fee	x	y	z
1	2458	Murray, W	M	27	1	72	168	HIGH	85.20			1
2	2462	Almers, C	F	34	3	66	152	HIGH	124.80			2
3	2501	Bonaventure, T	F	31	17	61	123	LOW	149.75			3
4	2523	Johnson, R	F	43	31	63	137	MOD	149.75			4
5	2539	LaMance, K	M	51	4	71	158	LOW	124.80			5
6	2544	Jones, M	M	29	6	76	193	HIGH	124.80			6
7	2552	Reberson, P	F	32	9	67	151	MOD	149.75			7
8	2555	King, E	M	35	13	70	173	MOD	149.75			8
9	2563	Pitts, D	M	34	22	73	154	LOW	124.80			9
10	2568	Eberhardt, S	F	49	27	64	172	LOW	124.80			10
11	2571	Nunnally, A	F	44	19	66	140	HIGH	149.75			11
12	2573	Blissner, M	F	52	17	67	173	MOD	149.75			12
13	2574	Peterson, V	M	30	6	69	147	LOW	149.75			13
14	2575	Quigley, M	F	40	8	69	163	HIGH	124.80			14
15	2578	Cameron, L	M	47	5	72	173	NA	124.80			15
16	2579	Underwood, K	F	60	22	71	191	LOW	149.75			16
17	2584	Takahashi, Y	F	43	29	65	123	MODV	124.80			17
18	2586	Derber, B	M	25	23	75	188	HIGH	85.20			18
19	2588	Ivan, H	F	22	20	63	139	LOW	85.20			19
20	2589	Wilcox, E	F	41	16	67	141	HIGH	149.75			20
21	2595	Warren, C	M	54	7	71	183	MOD	149.75			21

Example 11:



Explained:

CC	First.cc	Last.cc	Count
1	1	0	1
1	0	0	2
1	0	0	3
1	0	1	4

Output (with - retain count):

CC	Count
1	1
1	2
1	3
1	4

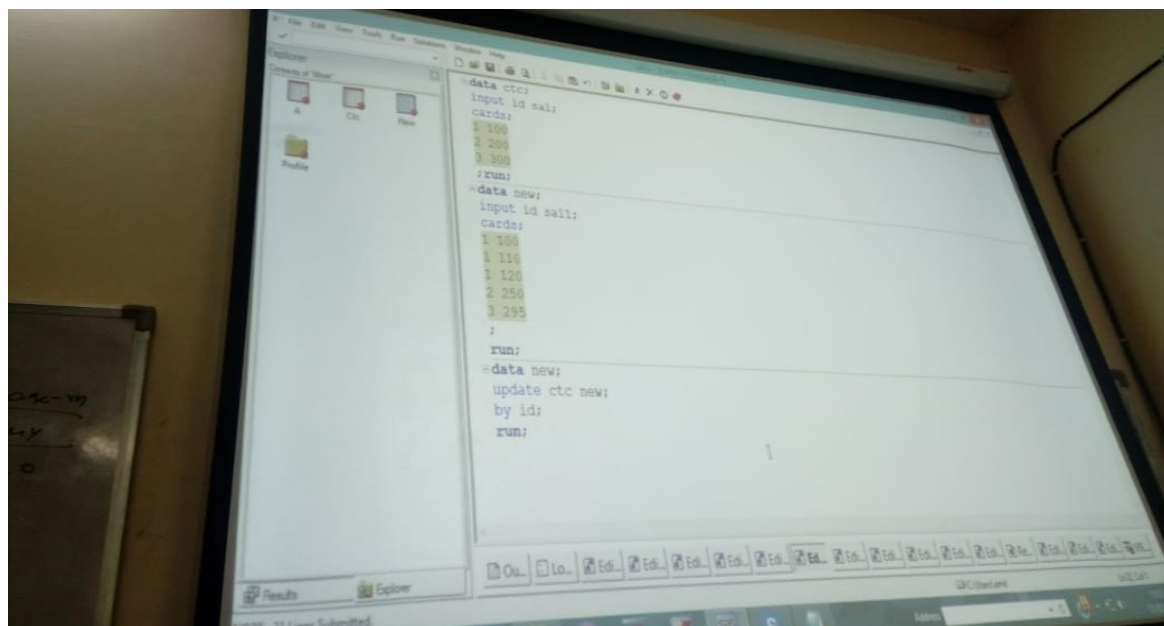
Output (without - retain count):

CC	Count
1	1
1	.
1	.
1	.

Because count = count+1, statement cannot operate on null value.

Update Revised:

Example 12:



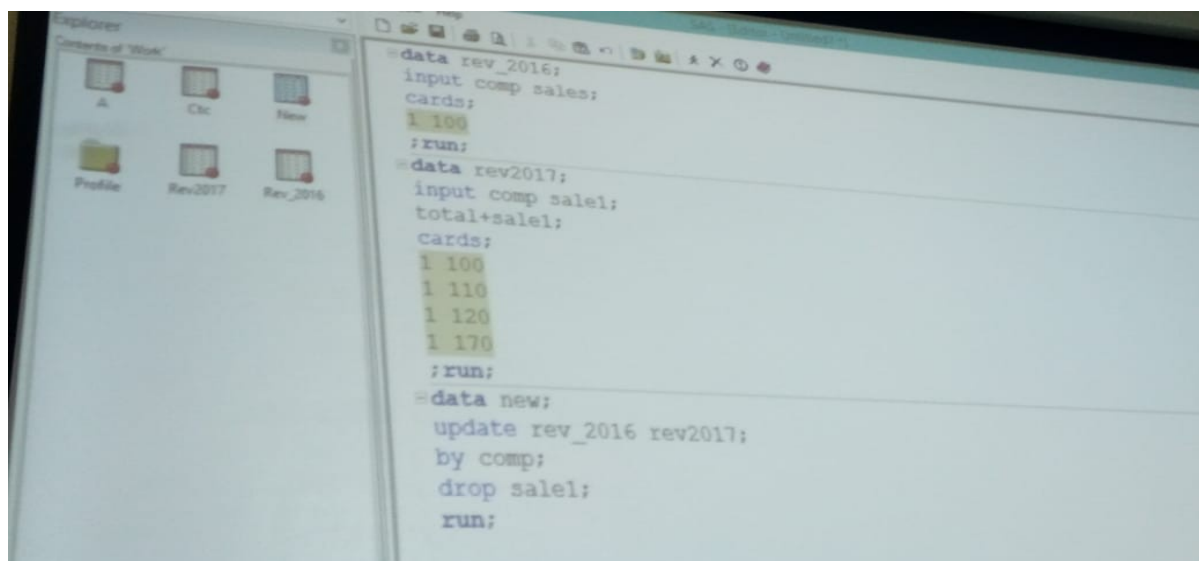
Explained:

- Dataset **ctc**, has variables id, sal
- Dataset **new**, has variables id sal1.
- Dataset **new** (latter one) will update dataset **ctc** with the dataset new (previous one).

Output (dataset New):

ID	Sal	Sal1
1	100	120
2	200	250
3	300	295

Example 13:



Explained:

- Dataset rev_2016 has variables comp, sales.
- Dataset rev2017 has variables comp sale1.
- Dataset rev2017 has another variable "total " which acts on the statement (total + sale1)."Total" will retain values and will have values in it adding in a cumulative manner wrt variable sale1.
- Dataset new will update rev_2016 with rev2107 (adding variable "total" into new dataset) where sale1 will be dropped.

Comp	sales
1	100

Dataset rev_2016

Comp	Sale1	Total
1	100	100
1	110	210
1	120	330
1	170	500

Dataset rev2017

Output (dataset New):

Comp	Sales	Total
1	100	500

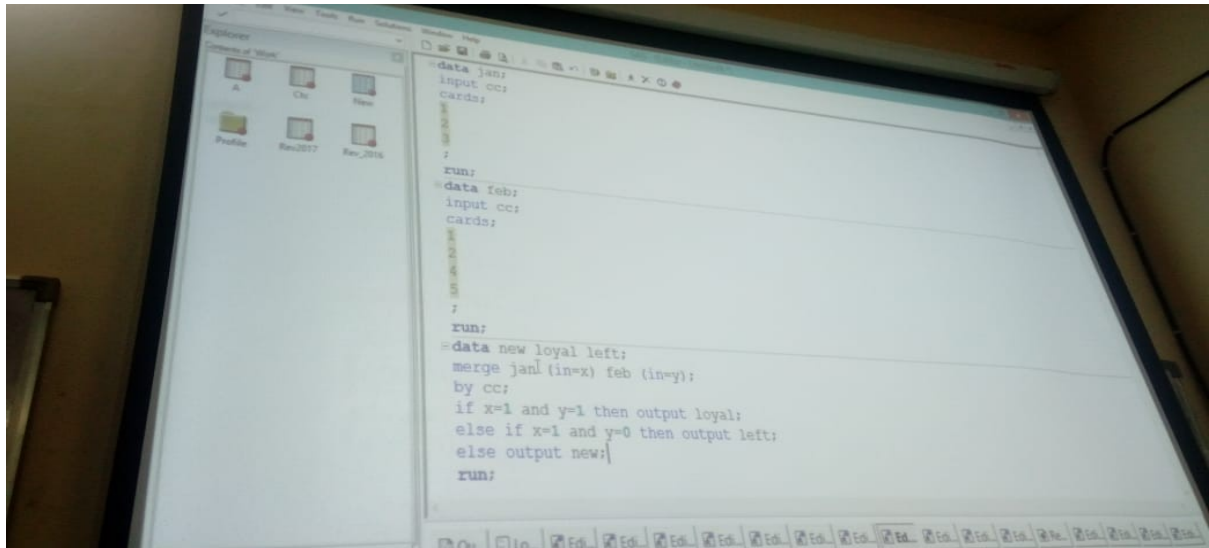
If dataset new does not drops sale1 variable from dataset rev2017, then output dataset is:

Comp	Sales	Sale1	Total
1	100	170	500

Merge (Special merging):

IN keyword: In used with merge creates intermediate (automatic, Boolean) variables at the backend.

Syntax: merge dataset_name (in = new_variable_name) dataset_name (in = new_variable_name);

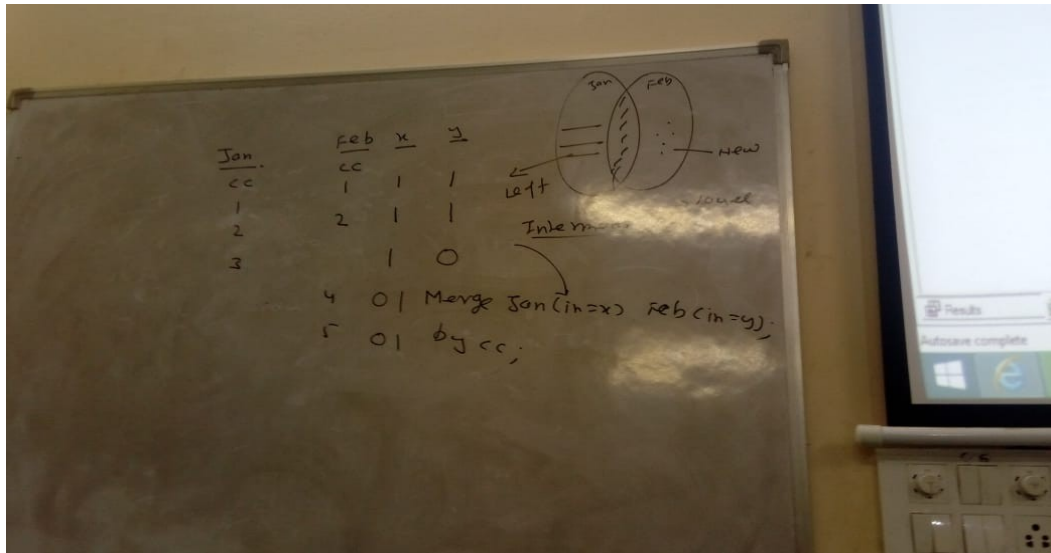


Example 14: From the given data of Jan & Feb based on the CC -id, create 3 separate datasets:

- 1st for newly joined customers
- 2nd for loyal customers who did not left in the feb month.
- 3rd for the customers who left in the feb month.

Explained:

- Dataset Jan has the cc for customer in Jan month.
- Dataset Feb has the cc for customer in Feb month.
- Dataset **new** is created for newly added customer, **loyal** for those who did not leave and dataset **left** for the ones who left in the feb month.
- **Using IN with merge, creates two automatic & Boolean variables with given / assumed names (x for Jan,y for Feb).**
- Looking at the picture below, x=1,y=1 implies loyal customer
- x=1, y=0 implies customer left, as customer is not present in the feb month implied by y=0.
- x=0, y=1 implies a new customer, as x=0 implies customer not initially present in the Jan month.



Output datasets:

CC
1
2

Loyal Customer (X=1, Y=1)

CC
4
5

New Customer (X=0, Y=1)

CC
3

Left Customer (X=1, Y=0)


```

File Edit Format View Help
data model;
infile "C:\Users\amit\Desktop\model.csv" dlm="," firstobs=2;
input vno      model$ Brand $;
run;

data city;
infile "C:\Users\amit\Desktop\city.csv" dlm="," firstobs=2;
input vno      city$ income;
run;

data final;
merge model city;
by vno;
if city="GGM" then city="Gurgaon";
else if city="NOI" then city="Noida";
else if city="DEL" then city="Delhi";
length Income_grp $4;
if 0<=income<=4.999999 then Income_grp="LOW";
else if 5<=income<=9.999999 then Income_grp="MID";
else Income_grp="HIGH";
run;

proc sort data=final;
by city brand Income_grp;
run;

data report;
set final;
by city brand Income_grp;
if first.Income_grp=1 then count=1;
else count+1;
if last.Income_grp;
keep city brand Income_grp count;
run;

```