Dated: 31/03/2018

Statistical Analysis System: Class 12

::

Functions:

Expression for a function is written as:

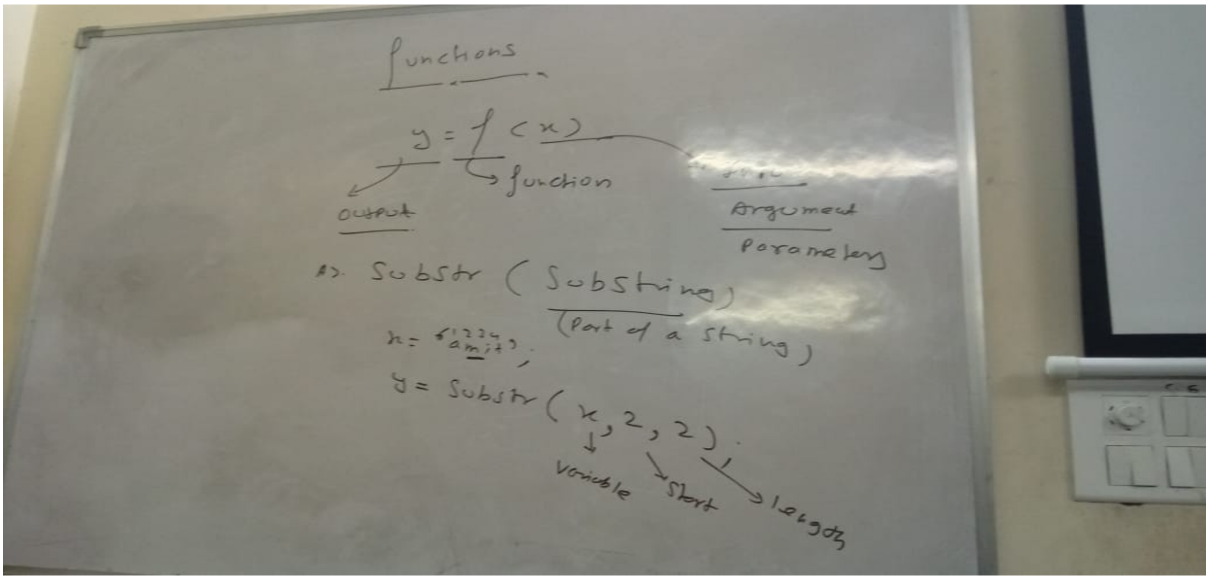
Y = f (x), where

Y = Output

F = function

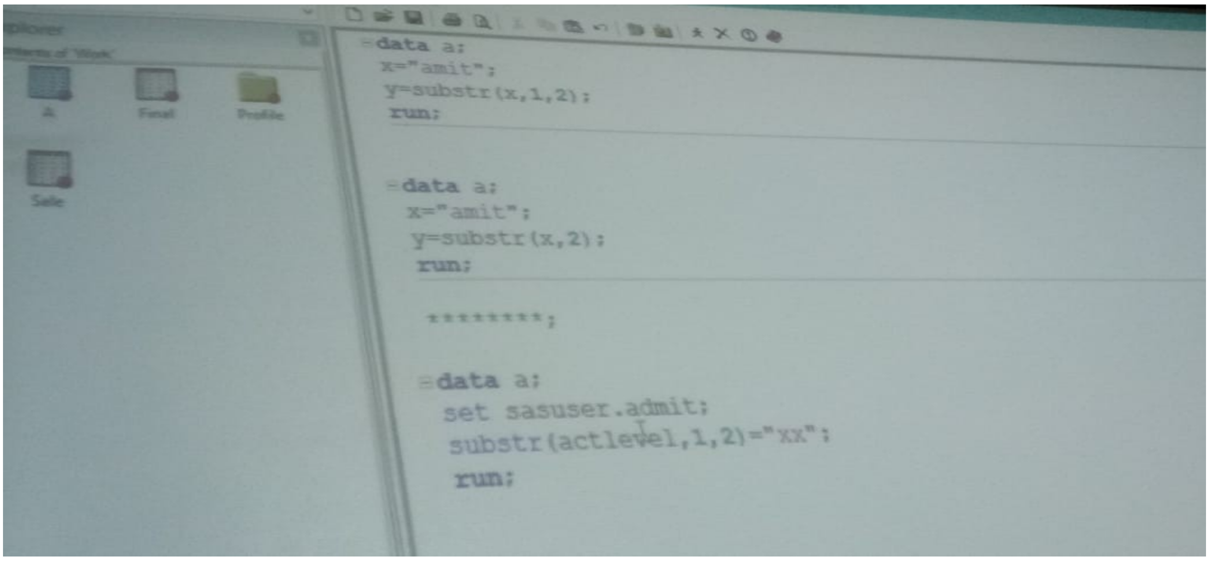
X = Input/ argument/parameter

Functions in SAS:



SUBSTR: This is called as sub-string function.

It returns a part of a string, it’s a character function, it can be used as left sub-string and right sub-string. A Substr function can have minimum 2 arguments and maximum 3 arguments.



```
data a;
 x="amit";
 y=substr(x,1,2);
 run;

 data a;
  x="amit";
  y=substr(x,2);
  run;

  ********;

  data a;
   set sasuser.admit;
   substr(actlevel,1,2)="xx";
   run;
```

Right sub-string function: function, when used in the form, **Y = substr (a,b,c).**

where,Y = output
substr = function
a = input variable
b = start point
c = implies length

Example 1:

Data a;

x="amit";

Y = substr (X,1,2);

Run;

Output :

X	Y
amit	am

Example 2 (with 2 arguments):

Data a;

x="amit";

Y = substr (x,2);

Run;

Output : Y = "mit" \\ The substr function considers entire string as output after the specified position (here 2nd position).

X	Y
amit	mit

Left sub-string function: function, when used in the form: **substr (a,b,c).**

Example 3:

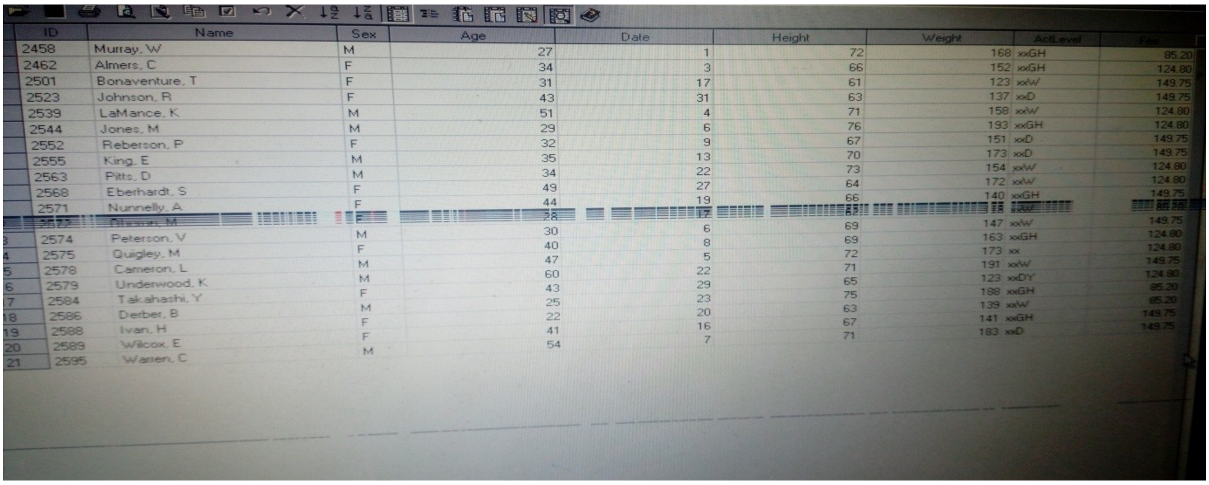
Data a;

Set sasuser.admit;

Substr (actlevel,1,2) = “xx”;

Run;

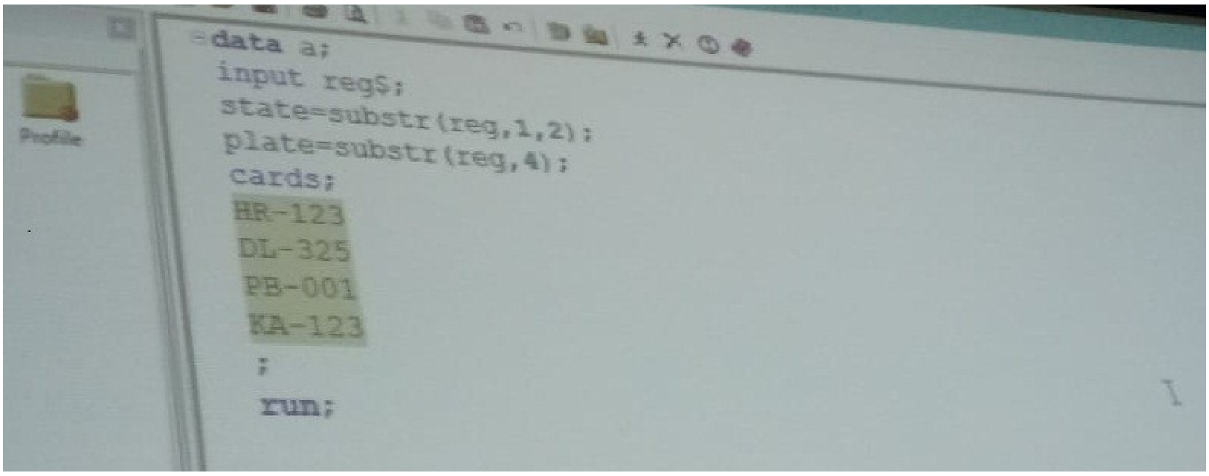
Output:



ID	Name	Sex	Age	Date	Height	Weight	ActLevel	Fee
2458	Murray, W	M	27	1	72	168	xxGH	85.20
2462	Almers, C	F	34	3	66	152	xxGH	124.80
2501	Bonaventure, T	F	31	17	61	123	xxW	149.75
2523	Johnson, R	F	43	31	63	137	xxD	149.75
2539	LaMance, K	M	51	4	71	158	xxW	124.80
2544	Jones, M	M	29	6	76	193	xxGH	124.80
2552	Reberson, P	F	32	9	67	151	xxD	149.75
2555	King, E	M	35	13	70	173	xxD	149.75
2563	Pitts, D	M	34	22	73	154	xxW	124.80
2568	Eberhardt, S	F	49	27	64	172	xxW	124.80
2571	Nunnelly, A	F	44	19	66	140	xxGH	149.75
2572	[illegible]	F	28	17	62	[illegible]	xxW	85.20
2574	Peterson, V	M	30	6	69	147	xxW	149.75
2575	Quigley, M	F	40	8	69	163	xxGH	124.80
2578	Cameron, L	M	47	5	72	173	xx	124.80
2579	Underwood, K	M	60	22	71	191	xxW	149.75
2584	Takahashi, Y	F	43	29	65	123	xxDY	124.80
2586	Derber, B	M	25	23	75	188	xxGH	85.20
2588	Ivan, H	F	22	20	63	139	xxW	85.20
2589	Wilcox, E	F	41	16	67	141	xxGH	149.75
2595	Warren, C	M	54	7	71	183	xxD	149.75

NOTE: this can also be used to update or perform masking in a variable, as seen in the above output.

Example 4:



```
data a;
input reg$;
state=substr(reg,1,2);
plate=substr(reg,4);
cards;
HR-123
DL-325
PB-001
KA-123
;
run;
```

Explained: This code here separates out information from input variable “reg” into two desired variable “state”, starts with character from 1st position & ends at 2nd position and “plate” starts with character from 4th position to the end.

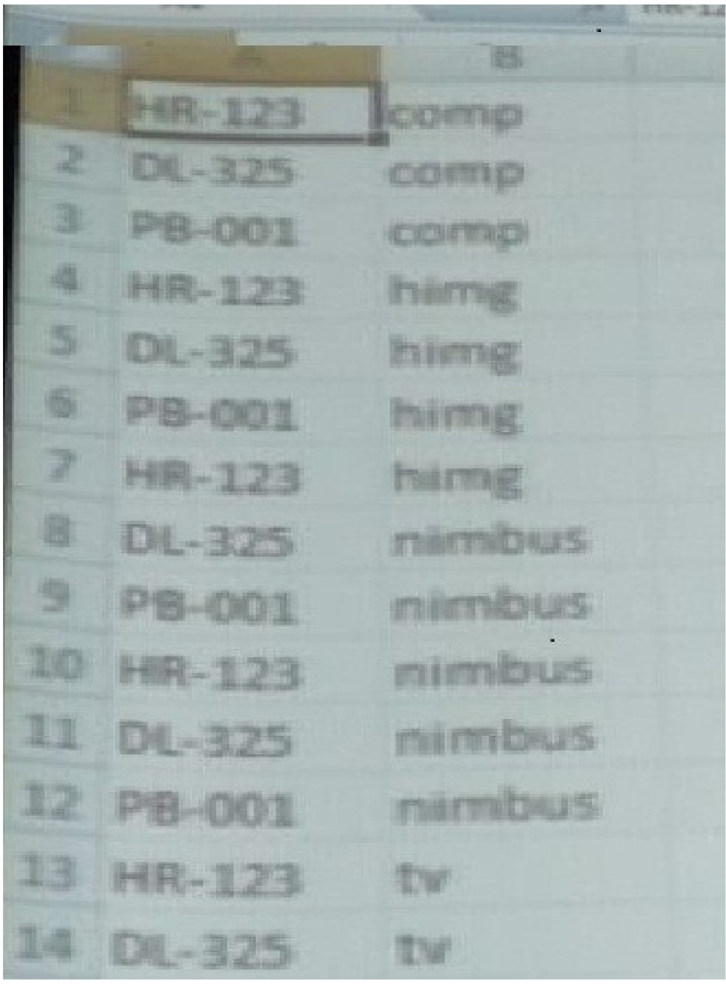
Output:

reg	state	Plate
HR-123	HR	123
DL-325	DL	325
PB-001	PB	001
KA-123	KA	123

SUBSTR with conditional formatting (if-else-if)

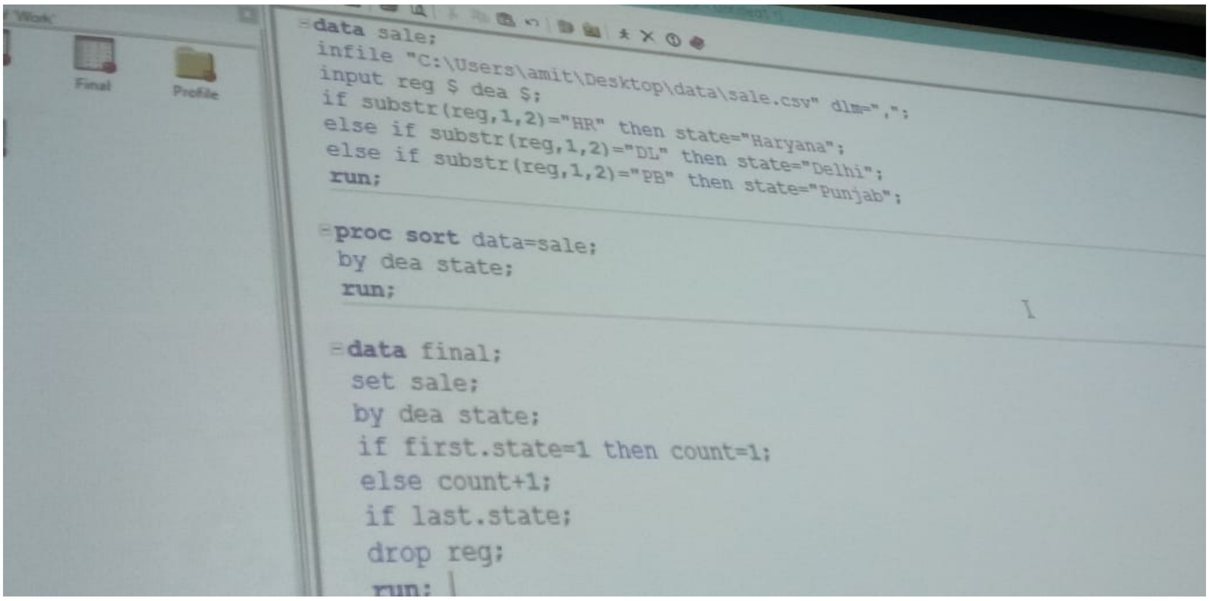
Example 5:

Input data:--



	A	B
1	HR-123	comp
2	DL-325	comp
3	PB-001	comp
4	HR-123	himg
5	DL-325	himg
6	PB-001	himg
7	HR-123	himg
8	DL-325	nimbus
9	PB-001	nimbus
10	HR-123	nimbus
11	DL-325	nimbus
12	PB-001	nimbus
13	HR-123	tv
14	DL-325	tv

Code:--



```
data sale;
infile "C:\Users\amit\Desktop\data\sale.csv" dlm=",";
input reg $ dea $;
if substr(reg,1,2)="HR" then state="Haryana";
else if substr(reg,1,2)="DL" then state="Delhi";
else if substr(reg,1,2)="PB" then state="Punjab";
run;

proc sort data=sale;
 by dea state;
 run;

data final;
 set sale;
 by dea state;
 if first.state=1 then count=1;
 else count+1;
 if last.state;
 drop reg;
 run;
```

Explained:

1st importing the input data with variable "reg" & "dea". Also using Substr function to update and filter out state using if-else-if statement.

2nd sorting data by dealer ,state as per requirement.

3rd creating final dataset, using first. & last. to get the total count of state a dealer is operating in.

Output:

Dealer	State	Count
comp	Haryana	1
comp	Delhi	2
comp	Punjab	3
himg	Haryana	1
himg	Delhi	2
himg	Punjab	3
himg	Haryana	4
nimbus	Delhi	1
nimbus	Punjab	2
nimbus	Haryana	3
nimbus	Delhi	4
nimbus	Punjab	5
tv	Haryana	1
tv	Delhi	2

SCAN: This function scans for a delimiter, breaks a input string into chunks / parts.

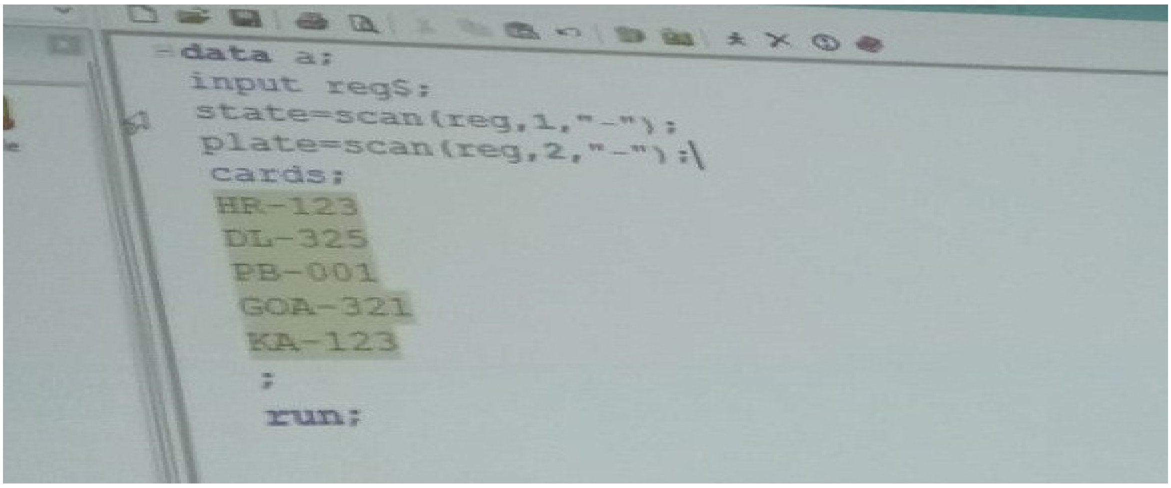
Scan (a,b,c);

a = input variable

b = start of chunk / part

c =specified delimiter

Example 6:



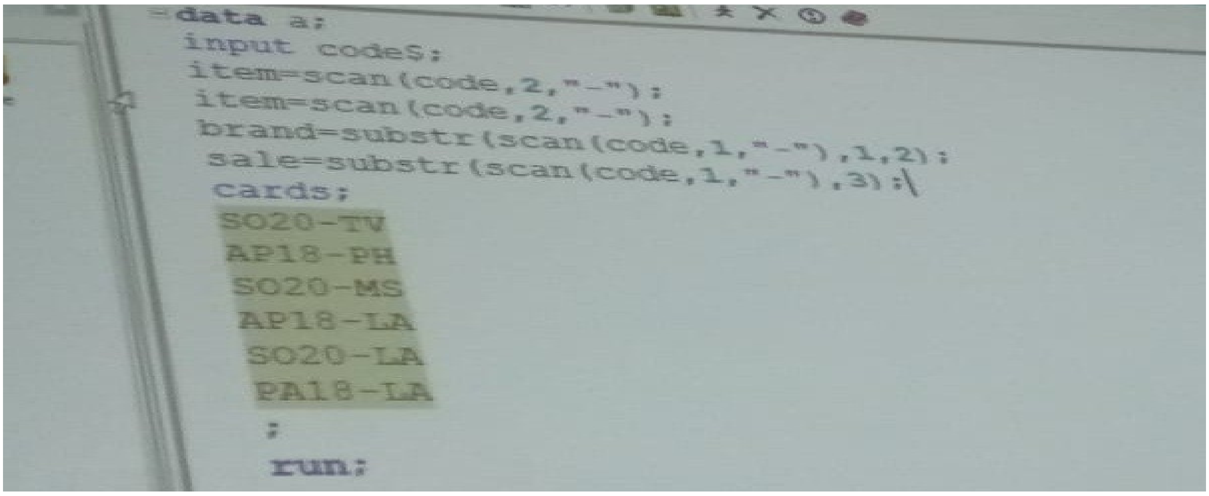
```
data a;
input reg$;
state=scan(reg,1,"-");
plate=scan(reg,2,"-");
cards;
HR-123
DL-325
PB-001
GOA-321
KA-123
;
run;
```

Explained: for state variable, input variable “reg” will be read in the first chunk separated by   dlm = “-” and plate will be read in the second chunk.

Output:

reg	**state**	**plate**
HR-123	HR	123
DL-325	DL	325
PB-001	PB	001
GOA-321	GOA	321
KA-123	KA	123

Example 7:



```
data a;
input code$;
item=scan(code,2,"-");
item=scan(code,2,"-");
brand=substr(scan(code,1,"-"),1,2);
sale=substr(scan(code,1,"-"),3);
cards;
SO20-TV
AP18-PH
SO20-MS
AP18-LA
SO20-LA
PA18-LA
;
run;
```

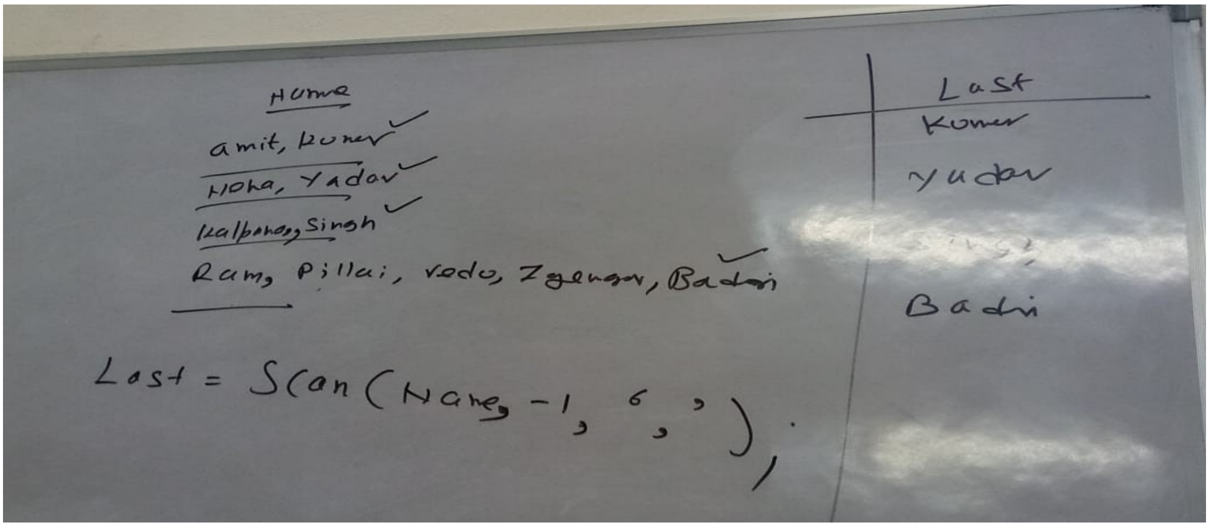
Explained: dataset “a” will have input variable “code” through which we get 3 different variable: “item” from the second chunk, “brand” and “sale” by applying Substr on the first chunk.

Output:

Code	item	brand	sale
So20-tv	tv	So	20
Ap18-ph	ph	Ap	18
So20-ms	ms	So	20
Ap18-la	la	Ap	18
So20-la	la	So	20
Ap18-la	la	Ap	18

NOTE: if we have to find last name in a variable of varying length, define length of variable initially to any optimum value (max 200), then use

Ouput variable = scan (input variable name, -1,”dlm”)

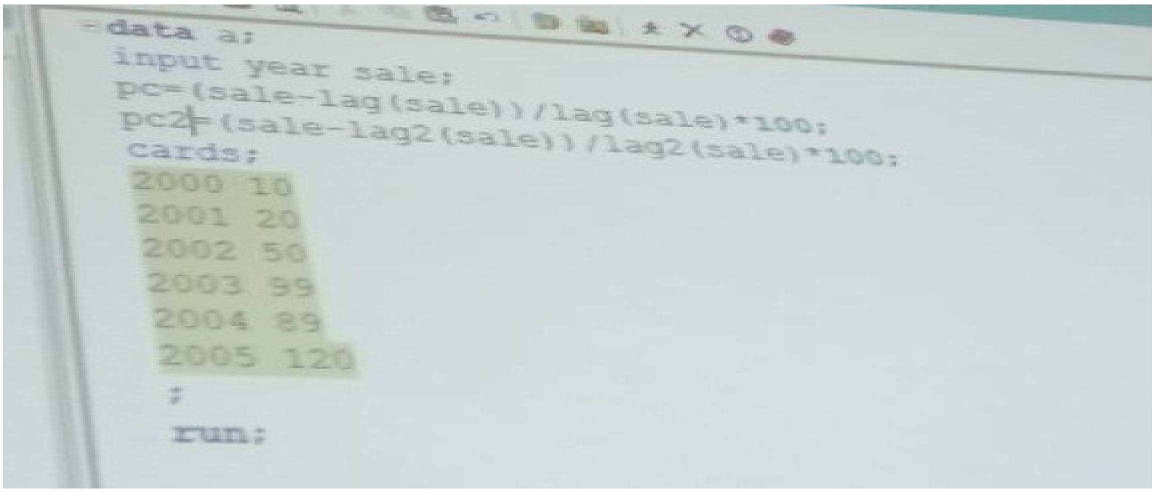


Name	Last
amit, kumar ✓	Kumar
Neha, Yadav ✓	yadav
kalpana, Singh ✓	Singh
Ram, Pillai, vedu, Iyengar, Badri ✓	Badri

Last = Scan(Name, -1, ',');

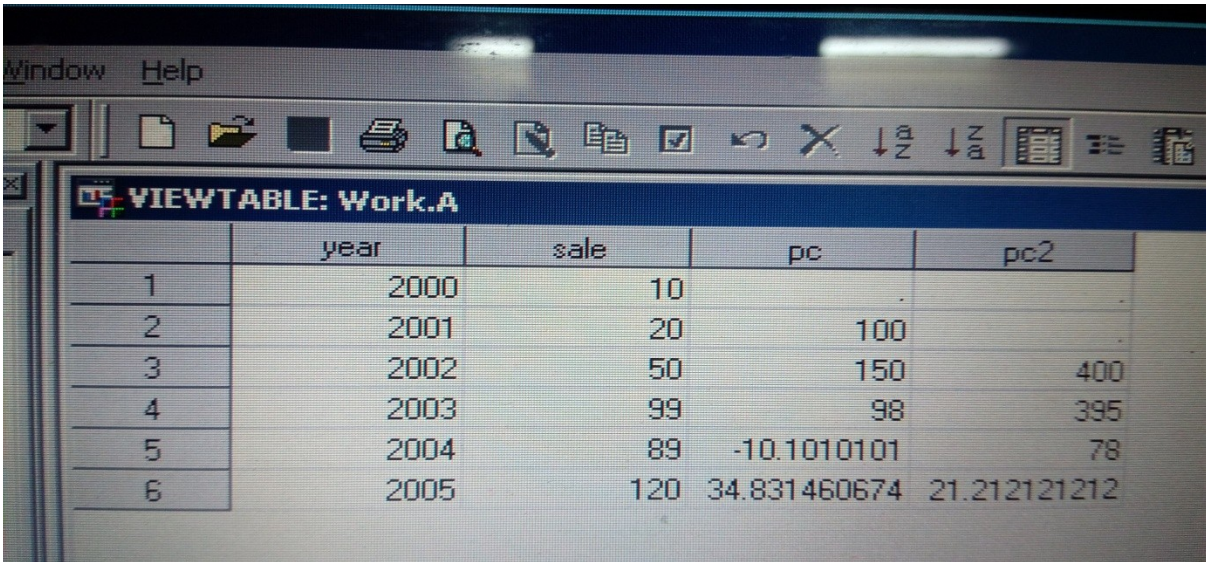
LAG: This function is used to get the desired lag in between data. It is helpful in calculating %change.

Example 8:



```
data a;
input year sale;
pc=(sale-lag(sale))/lag(sale)*100;
pc2=(sale-lag2(sale))/lag2(sale)*100;
cards;
2000 10
2001 20
2002 50
2003 99
2004 89
2005 120
;
run;
```

Output:

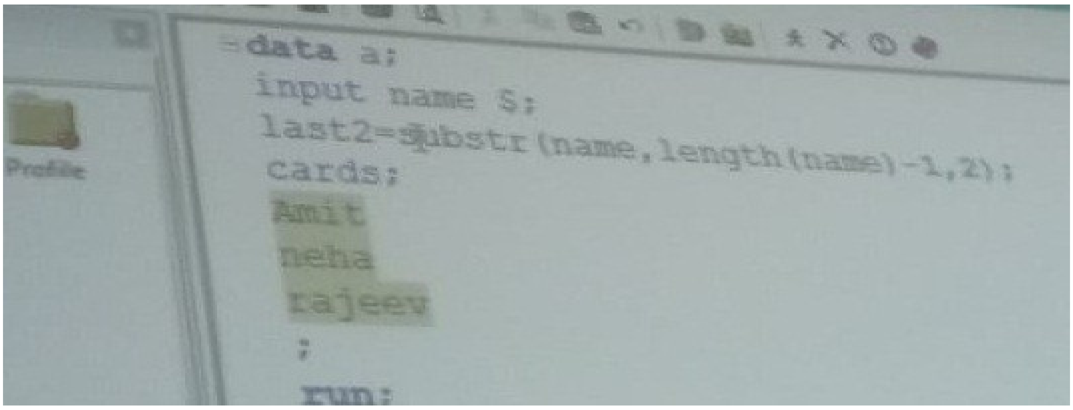


VIEWTABLE: Work.A

	year	sale	pc	pc2
1	2000	10	.	.
2	2001	20	100	.
3	2002	50	150	400
4	2003	99	98	395
5	2004	89	-10.1010101	78
6	2005	120	34.831460674	21.21212121

Length: length (var_name) -1

Example 9:



```
data a;
input name $;
last2=substr(name,length(name)-1,2);
cards;
Amit
neha
rajeev
;
run;
```

Explained: this will get the 2 characters from the end as output.

Output:

name	**Last2**
amit	**It**
neha	**Ha**
rajeev	**ev**