PROC APPEND

This procedure is used when you want to append observations from one dataset on the end of another dataset.

Syntax:-

```
PROC APPEND BASE= SAS DATASET  DATA=SASDATASET <OPTIONS>;
RUN;
```

Note: - Datasets can be appending by using set statement in dataset, but proc append can be efficient sometimes when large datasets are involved.

Options:-

Force – This is an optional specification that forces proc append to concatenate datasets. When we used this append the datasets force by without checking any conditions like present of variables, differences in the variable datasets and length.

Examples:-

```
Data ds1;
Input P_id Drug_name$ Sex$ Visit_date date9. ;
Format Visit_date date9.;
Datalines;
101 asp-05mg f 10Jan2011
102 asp-10mg m 10Jan2011
101 asp-05mg f 10Jan2011
102 asp-10mg m 10Jan2011
101 asp-05mg f 11Jan2011
103 asp-15mg f 11Jan2011
101 asp-05mg f 11Jan2011
102 asp-10mg m 12Jan2011
101 asp-05mg m 12Jan2011
102 asp-10mg m 12Jan2011
;
Run;
```

```
Data ds2;
Input P_id Drug_name$ Sex$ Visit_date date9.;
Format Visit_date date9.;
Datalines;
101 asp-05mg f 13Jan2011
102 asp-10mg m 13Jan2011
103 asp-15mg f 13Jan2011
104 asp-20mg m 13Jan2011
;
Run;
```

```
Proc append base=ds1 data=ds2;
Run;
```

Rule (1) Base dataset contain the variables but not in data
Prints results and gives warning

```
NOTE: Appending WORK.DS2 to WORK.DS1.
WARNING: Variable Sex was not found on DATA file.
NOTE: There were 4 observations read from the data set WORK.DS2.
NOTE: 4 observations added.
NOTE: The data set WORK.DS1 has 14 observations and 4 variables.
```

Without force or with force it gives same result.

```
Data ds1;
Infile Datalines;
Input P_id Drug_name$ Sex$ Visit_date;
Informat Visit_date date9. ;
Format Visit_date date9. ;
Datalines;
101 asp-05mg f 10Jan2011
102 asp-10mg m 10Jan2011
101 asp-05mg f 10Jan2011
102 asp-10mg m 10Jan2011
101 asp-05mg f 11Jan2011
103 asp-15mg f 11Jan2011
101 asp-05mg f 11Jan2011
102 asp-10mg m 12Jan2011
101 asp-05mg m 12Jan2011
102 asp-10mg m 12Jan2011
;
Run;
Data ds2;
Infile Datalines;
Input P_id Drug_name$ Visit_date;
Informat Visit_date date9. ;
Format Visit_date date9. ;
Datalines;
101 asp-05mg 13Jan2011
102 asp-10mg 13Jan2011
103 asp-15mg 13Jan2011
104 asp-20mg 13Jan2011
;
Run;
```

Without force –

```
Proc append base=ds1 data=ds2;
Run;
```

With force –

```
Proc append base=ds1 data=ds2 force;
Run;
```

Rule (2) Base dataset doesn't contain the variables but data dataset contain the variables

```
NOTE: Appending WORK.DS2 to WORK.DS1.
WARNING: Variable age was not found on BASE file. The variable will not be added to the
BASE file.
ERROR: No appending done because of anomalies listed above.
       Use FORCE option to append these files.
NOTE: 0 observations added.
NOTE: The data set WORK.DS1 has 10 observations and 4 variables.
NOTE: Statements not processed because of errors noted above.
```

After using force option appending done but Variable age was not found on BASE file. The variable will not be added to the BASE file.

```
Data ds1;
Infile Datalines;
Input P_id Drug_name$ Sex$ Visit_date;
Informat Visit_date date9. ;
Format Visit_date date9. ;
Datalines;
101 asp-05mg f 10Jan2011
102 asp-10mg m 10Jan2011
101 asp-05mg f 10Jan2011
102 asp-10mg m 10Jan2011
101 asp-05mg f 11Jan2011
103 asp-15mg f 11Jan2011
101 asp-05mg f 11Jan2011
102 asp-10mg m 12Jan2011
101 asp-05mg m 12Jan2011
102 asp-10mg m 12Jan2011
;
Run;
Data ds2;
Infile Datalines;
Input P_id Drug_name$ Sex$ Visit_date age;
Informat Visit_date date9. ;
Format Visit_date date9. ;
```

```
Datalines;
101 asp-05mg f 13Jan2011 23
102 asp-10mg m 13Jan2011 25
103 asp-15mg f 13Jan2011 28
104 asp-20mg m 13Jan2011 30
;
Run;
```

Without force –

```
Proc append base=ds1 data=ds2;
Run;
```

With force –

```
Proc append base=ds1 data=ds2 force;
Run;
```

Rule (3) Variables of different lengths.

Base dataset variable length is more than Data dataset variable length

With force or without force appending done but length is equal to base dataset variable length.

```
NOTE: Appending WORK.DS2 to WORK.DS1.
WARNING: Variable Drug_name has different lengths on BASE and DATA files
      (BASE 10 DATA 8).
NOTE: There were 4 observations read from the data set WORK.DS2.
NOTE: 4 observations added.
NOTE: The data set WORK.DS1 has 14 observations and 4 variables.
```

```
Data ds1;
Length Drug_name$10. ;
Input P_id Drug_name$ Sex$ Visit_date;
Informat Visit_date date9. ;
Format Visit_date date9. ;
Datalines;
101 asp-05mg f 10Jan2011
102 asp-10mg m 10Jan2011
101 asp-05mg f 10Jan2011
102 asp-10mg m 10Jan2011
101 asp-05mg f 11Jan2011
103 asp-15mg f 11Jan2011
101 asp-05mg f 11Jan2011
102 asp-10mg m 12Jan2011
101 asp-05mg m 12Jan2011
102 asp-10mg m 12Jan2011
;
Run;
```

```
Data ds2;
Infile Datalines;
Length Drug_name$8. ;
Input P_id Drug_name$ Sex$ Visit_date;
Informat Visit_date date9. ;
Format Visit_date date9. ;
Datalines;
101 asp-05mg f 13Jan2011
102 asp-10mg m 13Jan2011
103 asp-15mg f 13Jan2011
104 asp-20mg m 13Jan2011
;
Run;
```

Without force –

```
Proc append base=ds1 data=ds2;
Run;
```

With force –

```
Proc append base=ds1 data=ds2 force;
Run;
```

Base dataset variable length is less than Data dataset variable length

Without force appending not done

```
NOTE: Appending WORK.DS2 to WORK.DS1.
WARNING: Variable Drug_name has different lengths on BASE and DATA files (BASE 8 DATA 10).
ERROR: No appending done because of anomalies listed above.
       Use FORCE option to append these files.
```

With force appending done but length is equal to base dataset variable length

```
NOTE: Appending WORK.DS2 to WORK.DS1.
WARNING: Variable Drug_name has different lengths on BASE and DATA files (BASE 8 DATA 10).
NOTE: FORCE is specified, so dropping/truncating will occur.
NOTE: There were 4 observations read from the data set WORK.DS2.
NOTE: 4 observations added.
```

```
Data ds1;
Infile Datalines;
Length Drug_name$10. ;
Input P_id Drug_name$ Sex$ Visit_date;
Informat Visit_date date9. ;
Format Visit_date date9. ;
Datalines;
101 asp-05mg f 10Jan2011
102 asp-10mg m 10Jan2011
```

```
101 asp-05mg f 10Jan2011
102 asp-10mg m 10Jan2011
101 asp-05mg f 11Jan2011
103 asp-15mg f 11Jan2011
101 asp-05mg f 11Jan2011
102 asp-10mg m 12Jan2011
101 asp-05mg m 12Jan2011
102 asp-10mg m 12Jan2011
;
Run;
```

```
Data ds2;
Infile Datalines;
Length Drug_name$8. ;
Input P_id Drug_name$ Sex$ Visit_date;
Informat Visit_date date9. ;
Format Visit_date date9. ;
Datalines;
101 asp-05mg f 13Jan2011
102 asp-10mg m 13Jan2011
103 asp-15mg f 13Jan2011
104 asp-20mg m 13Jan2011
;
Run;
```

Without force –

```
Proc append base=ds1 data=ds2;
Run;
```

With force –

```
Proc append base=ds1 data=ds2 force;
Run;
```

Rule (4) Variables of different data types.

```
Data ds1;
Infile Datalines;
Input P_id$ Drug_name$ Sex$ Visit_date;
Informat Visit_date date9. ;
Format Visit_date date9. ;
Datalines;
101 asp-05mg f 10Jan2011
102 asp-10mg m 10Jan2011
101 asp-05mg f 10Jan2011
102 asp-10mg m 10Jan2011
101 asp-05mg f 11Jan2011
```

```
103 asp-15mg f 11Jan2011
101 asp-05mg f 11Jan2011
102 asp-10mg m 12Jan2011
101 asp-05mg m 12Jan2011
102 asp-10mg m 12Jan2011
;
Run;
Data ds2;
Infile Datalines;
Input P_id Drug_name$ Sex$ Visit_date;
Informat Visit_date date9. ;
Format Visit_date date9. ;
Datalines;
101 asp-05mg f 13Jan2011
102 asp-10mg m 13Jan2011
103 asp-15mg f 13Jan2011
104 asp-20mg m 13Jan2011
;
Run;
```

Without force –

```
Proc append base=ds1 data=ds2;
Run;
```

```
NOTE: Appending WORK.DS2 to WORK.DS1.
WARNING: Variable P_id not appended because of type mismatch.
ERROR: No appending done because of anomalies listed above. Use FORCE option to append these files.
NOTE: 0 observations added.
NOTE: The data set WORK.DS1 has 10 observations and 4 variables.
NOTE: Statements not processed because of errors noted above.
```

With force –

```
Proc append base=ds1 data=ds2 force;
Run;
```

```
NOTE: Appending WORK.DS2 to WORK.DS1.
WARNING: Variable P_id not appended because of type mismatch.
NOTE: FORCE is specified, so dropping/truncating will occur.
NOTE: There were 4 observations read from the data set WORK.DS2.
NOTE: 4 observations added.
NOTE: The data set WORK.DS1 has 14 observations and 4 variables.
```

So convert both P_id into one data type before appending

```
Data ds1a (drop=P_id rename=(id1=P_id ));
Set ds1;
id1=input (P_id ,best.);
Run;
Proc append base=ds1a data=ds2 force;
Run;
```

Rule (5) Base dataset doesn't exist but Data dataset is there.

In this case data will copy from data dataset to base dataset.

```
Proc append base=ds1 data=sashelp.class;
Run;
```

```
NOTE: Appending SASHELP.CLASS to WORK.DS1.
NOTE: BASE data set does not exist. DATA file is being copied to BASE file.
NOTE: There were 19 observations read from the data set SASHELP.CLASS.
NOTE: The data set WORK.DS1 has 19 observations and 5 variables.
```

Limitation of Proc Append

Suppose if we execute proc append program multiple times data also append multiple times to base data. So handle with care when you are executing Proc append statements.

If it's happened multiple times first take the back up of history (base) data then delete extra added observations from history data. And freshly run append statement once again.

By mistake if anything is wrong with data you can pick from backup table again.

```
Data krishna.dss1;
Set work.ds1;
```

Taking backup

```
Run;
Data work.ds1;
Set work.ds1;
If Visit_date='13Jan2011'd then delete;
Run;
```

Deleting observations

```
Proc append base=ds1 data=ds2;
Run;
```

Append observation once again freshly

When we can use Proc append

To load data into datasets/tables.
To combine two or more datasets into single dataset with large volume of data.

Mainly when we are performing ETL process in that for LOADING we can use Proc append.

An important document which helps to understand Proc append

Appending Datasets! WARNING! This may (

APPEND, EXECUTE and MACRO.pdf

COMBINING SAS DATA SETS.pdf

Efficient Techniques and Tips in Handling L

How Should I Combine My Data, Is

Illustrating Concatenation Opera

Methods of Storing SASÆ Data into Orac

PROC APPEND May Make More Sense.pdf

PROC COPY and PROC APPEND A Dev

PROGRAMMING EFFICIENCIES USING

Same Data Different Attributes Cloning Iss

SAS fastload to Teradata.htm

Smart Import-Append Data i