

SAS®

Initiation et manipulation de données Importation de données par l'étape DATA

Nicolas Poulin



Données

- Les données sont stockées dans une table SAS.
- Une table SAS peut être créée :
 - ▶ en rentrant manuellement les données via une étape **DATA**
 - ▶ en important des données brutes depuis un autre format (xlsx, txt,...) via une étape **DATA** ou une procédure **IMPORT**

Importation d'un fichier Excel via l'étape DATA

Remarque

- *importer un fichier Excel via une étape DATA nécessite le module SAS/ACCESS to PC files.*
- SAS est capable de considérer un fichier Excel comme étant une table SAS.
- Pour cela, il faut créer une référence bibliographique au fichier Excel grâce à l'instruction **LIBNAME**.

Assignation d'une référence bibliographique

```
LIBNAME libref XLSX "chemin";
```

où :

- libref : nom donné à la bibliothèque :
- "chemin" du fichier **physique** sur l'ordinateur ou le réseau.

Remarque

- Le "chemin" doit se terminer par le nom complet du fichier (avec son extension *.xlsx*).
- Chaque feuille du fichier Excel sera considérée comme une table SAS.

Manipulation d'un fichier Excel

Une fois l'instruction **LIBNAME** utilisée, chaque feuillet du fichier Excel étant vu comme une table SAS, l'étape **DATA** s'utilise de façon classique.

```
DATA libref.filename2;  
    SET libref.filename1;  
    ...  
RUN;
```

Remarque

- les instructions **WHERE**, **KEEP**, **LABEL**, **FORMAT** peuvent être utilisées.
- dans le fichier Excel, une nouvelle feuille nommée comme *filename2* sera créée.



Dissociation d'un Libref

- Lorsqu'une référence bibliographique est assignée à un fichier Excel, ce dernier ne peut plus être ouvert dans Excel.
- Il faut donc désactiver ce lien pour avoir de nouveau accès au fichier via Excel

`LIBNAME libref CLEAR;`

Fichier avec délimiteur

Il s'agit de fichiers de données dont la séparation entre les données est marquée par un délimiteur défini.

- Fichiers *.txt* : tabulation
- Fichiers *.csv* :
 - ▶ anglo-saxon : virgule
 - ▶ européen : point-virgule

Fichier avec délimiteur

```
DATA libref.filename2;  
    INFILE "chemin";  
    INPUT spécifications;  
    ...  
RUN;
```

Remarque

- Les instructions *KEEP*, *LABEL*, *FORMAT* peuvent être utilisées.
- Le *"chemin"* doit se terminer par le nom complet du fichier (avec son extension).
- L'instruction *WHERE* ne peut pas être utilisée.

Fichier avec délimiteur

- Un espace vide est le délimiteur par défaut.
- L'option **DLM=** doit être ajoutée à l'instruction **INFILE** pour spécifier tout autre délimiteur.
- SAS ne peut pas extraire les noms des variables de la première ligne d'un tel fichier. Si la première ligne contient les noms des variables il faut utiliser l'option **FIRSTOBS=2** de l'instruction **INFILE**

```
DATA libref.filename2;  
    INFILE "chemin" DLM="délimiteur" FIRSTOBS=2;  
    INPUT spécifications;  
    ...  
RUN;
```

Données standard et non standard

- Les données standard sont les données que SAS peut reconnaître comme étant numérique.
- Les données numériques non standard seront vues comme des données de type texte.
- Exemples de données standard : 27 -27 27,09 2,7E7 ...
- Exemples de données non standard : 27.09 (27) \$27
27/09/2016 27SEPTEMBER2016 ...

Remarque

*27,09 est une donnée standard pour la localisation French_France
mais pas pour la localisation English_UnitedStates*

Spécifications données standard

- Il s'agit des spécifications pour l'instruction **INPUT**.

INPUT liste des variables et nature;

- Les variables doivent être déclarées dans l'ordre du tableau de données.
- Il faut mettre \$ après le nom de chaque variable de type caractère.
- La longueur par défaut des variables, quel que soit leur type, est de 8 octets.

Exemple : fichier avec délimiteur

```
DATA work.maishort1;  
    INFILE "/folders/myfolders/UEdata/maishort.csv"  
        DLM=";" FIRSTOBS=2;  
    INPUT Individu Hauteur Masse Couleur $ Parcelle $;  
RUN;
```

Remarque

*SAS ne réussit pas, par défaut, à importer, via une étape **DATA** un fichier avec des données manquantes (cellules vides). Pour cela, il faut ajouter l'option **DSD** à l'instruction **INFILE**.*

Exemple : fichier avec délimiteur

```
DATA work.maishort1;  
  INFILE "/folders/myfolders/UEdata/maishortNA.csv"  
        DLM=";" FIRSTOBS=2;  
  INPUT Individu Hauteur Masse Couleur $ Parcelle $;  
RUN;
```

```
DATA work.maishort1;  
  INFILE "/folders/myfolders/UEdata/maishortNA.csv"  
        DLM=";" FIRSTOBS=2 DSD;  
  INPUT Individu Hauteur Masse Couleur $ Parcelle $;  
RUN;
```

Vecteur de données de programme

- Lors d'une étape **DATA**, SAS va copier successivement les lignes du fichier source dans le vecteur de données de programme (PDV).
- Une fois rempli, le PDV sera copié dans une ligne de la table SAS.
- De manière générale, le PDV est utilisé pour stocker l'observation en cours de traitement (par exemple lors de l'exécution d'une **PROC**). Ce traitement ligne par ligne explique généralement la nécessité de trier les tables SAS.

Phase de compilation

Pendant la phase de compilation, SAS :

- vérifie la syntaxe du code,
- crée un tampon d'entrée pour stocker les données brutes en cours de traitement,
- crée le PDV,
- crée le bloc descripteur de la table.

Remarque

La phase d'exécution est le moment où SAS remplit à chaque itération le PDV qu'il va copier dans la table SAS de sortie.

Compilation

```
DATA work.maishort1;  
    INFILE "/folders/myfolders/UEdata/maishort.csv"  
        DLM=";" FIRSTOBS=2;  
INPUT Individu Hauteur Masse Couleur $ Parcelle $;  
RUN;
```


Compilation

```
DATA work.maishort1;  
    INFILE "/folders/myfolders/UEdata/maishort.csv"  
        DLM=";" FIRSTOBS=2;  
INPUT Individu Hauteur Masse Couleur $ Parcelle $;  
RUN;
```

Tampon d'entrée

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Compilation

```
DATA work.maissshort1;  
  INFILE "/folders/myfolders/UEdata/maishort.csv"  
        DLM=";" FIRSTOBS=2;  
  INPUT Individu Hauteur Masse Couleur $ Parcelle $;  
RUN;
```

Tampon d'entrée

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

PDV

Individu N 8

Compilation

```
DATA work.maissshort1;  
  INFILE "/folders/myfolders/UEdata/maishort.csv"  
        DLM=";" FIRSTOBS=2;  
  INPUT Individu Hauteur Masse Couleur $ Parcelle $;  
RUN;
```

Tampon d'entrée

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

PDV

Individu N 8	Hauteur N 8

Compilation

```
DATA work.maissshort1;  
  INFILE "/folders/myfolders/UEdata/maishort.csv"  
        DLM=";" FIRSTOBS=2;  
  INPUT Individu Hauteur Masse Couleur $ Parcelle $;  
RUN;
```

Tampon d'entrée

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

PDV

Individu N 8	Hauteur N 8	Masse N 8

Compilation

```
DATA work.maissshort1;  
  INFILE "/folders/myfolders/UEdata/maishort.csv"  
        DLM=";" FIRSTOBS=2;  
  INPUT Individu Hauteur Masse Couleur $ Parcelle $;  
RUN;
```

Tampon d'entrée

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

PDV

Individu N 8	Hauteur N 8	Masse N 8	Couleur \$ 8

Compilation

```
DATA work.maissshort1;  
  INFILE "/folders/myfolders/UEdata/maishort.csv"  
        DLM=";" FIRSTOBS=2;  
  INPUT Individu Hauteur MasseCouleur $ Parcelle $;  
RUN;
```

Tampon d'entrée

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

PDV

Individu N 8	Hauteur N 8	Masse N 8	Couleur \$ 8	Parcelle \$ 8

Compilation

```
DATA work.maissshort1;  
  INFILE "/folders/myfolders/UEdata/maishort.csv"  
        DLM=";" FIRSTOBS=2;  
INPUT Individu Hauteur MasseCouleur $ Parcelle $;  
RUN;
```

Tampon d'entrée

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Bloc descripteur

Individu N 8	Hauteur N 8	Masse N 8	Couleur \$ 8	Parcelle \$ 8

Exécution

```
DATA work.maishort1;  
  INFILE "/folders/myfolders/UEdata/maishort.csv"  
        DLM=";" FIRSTOBS=2;  
  INPUT Individu Hauteur MasseCouleur $ Parcelle $;  
RUN;
```

Tampon d'entrée

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Initialisation du PDV

Individu N 8	Hauteur N 8	Masse N 8	Couleur \$ 8	Parcelle \$ 8
.	.	.		

Exécution

```
DATA work.maissshort1;  
  INFILE "/folders/myfolders/UEdata/maishort.csv"  
  DLM=";" FIRSTOBS=2;  
INPUT Individu Hauteur MasseCouleur $ Parcelle $;  
RUN;
```

Tampon d'entrée

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
2	;	1	9	9	;	1	4	3	1	;	R	o	u	g

PDV

Individu N 8	Hauteur N 8	Masse N 8	Couleur \$ 8	Parcelle \$ 8
.	.	.		

Exécution

```
DATA work.maissshort1;  
    INFILE "/folders/myfolders/UEdata/maishort.csv"  
        DLM=";" FIRSTOBS=2;  
INPUT Individu Hauteur MasseCouleur $ Parcelle $;  
RUN;
```

Tampon d'entrée

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
2	;	1	9	9	;	1	4	3	1	;	R	o	u	g

PDV

Individu N 8	Hauteur N 8	Masse N 8	Couleur \$ 8	Parcelle \$ 8
2	199	1431	Rouge	Nord

Table SAS après la première itération

```
DATA work.maissshort1;  
    INFILE "/folders/myfolders/UEdata/maishort.csv"  
          DLM=";" FIRSTOBS=2;  
    INPUT Individu Hauteur MasseCouleur $ Parcelle $;  
RUN;
```

work.maissshort1

Individu	Hauteur	Masse	Couleur	Parcelle
2	199	1431	Rouge	Nord

Exécution

```
DATA work.maishort1;  
  INFILE "/folders/myfolders/UEdata/maishort.csv"  
        DLM=";" FIRSTOBS=2;  
  INPUT Individu Hauteur MasseCouleur $ Parcelle $;  
RUN;
```

Tampon d'entrée

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Réinitialisation du PDV

Individu N 8	Hauteur N 8	Masse N 8	Couleur \$ 8	Parcelle \$ 8
.	.	.		

Exécution

```
DATA work.maissshort1;  
  INFILE "/folders/myfolders/UEdata/maishort.csv"  
        DLM=";" FIRSTOBS=2;  
  INPUT Individu Hauteur MasseCouleur $ Parcelle $;  
RUN;
```

Tampon d'entrée

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
3	;	2	0	5	;	1	4	6	8	;	J	a	u	n

PDV

Individu N 8	Hauteur N 8	Masse N 8	Couleur \$ 8	Parcelle \$ 8
.	.	.		

Exécution

```
DATA work.maissshort1;  
  INFILE "/folders/myfolders/UEdata/maishort.csv"  
        DLM=";" FIRSTOBS=2;  
  INPUT Individu Hauteur MasseCouleur $ Parcelle $;  
RUN;
```

Tampon d'entrée

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
3	;	2	0	5	;	1	4	6	8	;	J	a	u	n

PDV

Individu N 8	Hauteur N 8	Masse N 8	Couleur \$ 8	Parcelle \$ 8
3	205	1468	Jaune	Nord

Table SAS après la deuxième itération

```
DATA work.maissshort1;  
    INFILE "/folders/myfolders/UEdata/maishort.csv"  
          DLM=";" FIRSTOBS=2;  
    INPUT Individu Hauteur MasseCouleur $ Parcelle $;  
RUN;
```

work.maissshort1

Individu	Hauteur	Masse	Couleur	Parcelle
2	199	1431	Rouge	Nord
3	205	1468	Jaune	Nord

Et ainsi de suite jusqu'à la fin du fichier.

Instruction LENGTH

- Par défaut, toutes les variables créées sont de longueur 8.
- L'instruction **LENGTH** permet de définir la longueur des variables textes.
- Cette instruction doit se trouver avant l'instruction **INPUT**.

```
DATA work.maishort1;  
    LENGTH Couleur $ 11 Parcelle $ 5;  
    INFILE "/folders/myfolders/UEdata/maishort.csv"  
          DLM=";" FIRSTOBS=2;  
    INPUT Individu Hauteur Masse Couleur $ Parcelle $;  
RUN;
```


Instruction LENGTH

- SAS prendra l'ordre de déclaration des variables sur l'ensemble de l'étape **DATA**.
- Les premières colonnes seront donc les variables déclarées dans l'instruction **LENGTH**.
- Ensuite viendront les variables déclarées dans l'instruction **INPUT** en omettant celles déjà déclarées dans l'instruction **LENGTH**.

Compilation

```
DATA work.maissshort1;  
  LENGTH Couleur $ 11 Parcelle $ 5;  
  INFILE "/folders/myfolders/UEdata/maishort.csv"  
        DLM=";" FIRSTOBS=2;  
  INPUT Individu Hauteur Masse Couleur $ Parcelle $;  
RUN;
```

Bloc descripteur

Parcelle \$ 11	Couleur \$ 5	Individu N 8	Hauteur N 8	Masse N 8

Spécifications données non standard

- Il faut rajouter des déclarations pour l'instruction **INPUT**.
- Il faut déclarer des informats pour signifier à SAS quel format est utilisé dans le fichier d'entrée pour chaque donnée non standard.

Informats SAS

Les informats SAS se présentent sous la forme :

$\langle \$ \rangle \text{ informat } \langle w \rangle . \langle d \rangle$

\$	présent uniquement pour les variables de type caractère
<i>informat</i>	informat SAS
<i>w</i>	spécifie le nombre de colonnes à lire dans les données d'entrée
.	délimiteur obligatoire
d	nombre de décimales pour les informats numériques

Remarque

La notation $\langle \cdot \rangle$ signifie qu'il s'agit d'un argument facultatif.

Informats SAS

Informat	Effet
\$w.	données de type caractère standard
w.d	données de type numérique standard
COMMAw.d	lit des données numériques non standard et enlève tous les caractères non conformes
DOLLARXw.d	lit des données numériques non standard et enlève tous les caractères non conformes
EUROw.d	lit des données numériques non standard et enlève tous les caractères non conformes

Informats SAS

Informat	Valeur de données brutes	Valeur de données SAS
COMMA7.0 DOLLAR7.0	\$12,345	12345
COMMAX7.0 DOLLARX7.0	\$12,345	12345
EUROX7.0	€ 12,345	12345

Informats SAS

$\langle \$ \rangle \text{ informat } \langle w \rangle . \langle d \rangle$

Informat	Valeur de données brutes	Valeur de données SAS
COMMA6.2 DOLLAR6.2	\$12345	123,45
COMMAX10.2 DOLLARX10.2	\$12345.567	12345,567

Remarque

Si la valeur de données contient une décimale, la valeur de d est ignorée.

Informats de date SAS

Informat	Valeur de données brutes	Valeur de données SAS
MMDDYY6.	123160	365
DDMMYY6.	311260	365
MMDDYY8.	12/31/60	365
DDMMYY8.	31/12/60	365
MMDDYY10.	12/31/1960	365
DDMMYY10.	31/12/1960	365
DATE7.	31DEC60	365
DATE9.	31DEC1960	365

Modificateur :

- Le modificateur : permet d'ignorer la longueur w et de laisser la priorité au délimiteur du fichier source.
- Cela permet aussi d'avoir un informat plus souple.

:DDMMYY10. sera capable de lire :

- 03/07/2008
- 3/07/2008
- 03/7/2008
- 3/7/2008
- 03/07/08
- 3/7/08

Informat : exemple

```
DATA work.sales;  
  LENGTH First_Name $ 12 Last_Name $ 18 Gender $ 1  
         Job_Title $25 Country $ 2;  
  INFILE "/folders/myfolders/UEdata/sales.csv"  
         DLM=",";  
  INPUT Employee_ID First_Name $ Last_Name $  
         Gender $ Salary Job_Title $ Country $  
         Birth_Date :date9. Hire_Date :MMDDYY10.;  
RUN;
```

Informat : exemple

```
DATA work.sales;  
  LENGTH First_Name $ 12 Last_Name $ 18 Gender $ 1  
         Job_Title $25 Country $ 2;  
  INFILE "/folders/myfolders/UEdata/sales.csv"  
         DLM=",";  
  INPUT Employee_ID First_Name $ Last_Name $  
         Gender $ Salary Job_Title $ Country $  
         Birth_Date :date9. Hire_Date :MMDDYY10.;  
  FORMAT Salary dollar12. Birth_Date DDMMYY10.  
         Hire_Date MONYY7.;  
  
RUN;
```