

## Feuille de TP n°1 Mme Malot

### REMARQUE 1

*Cette première feuille de TP a pour but certes de prendre en main le logiciel SAS, mais surtout de se familiariser avec les méthodes existantes pour constituer des bases de données utilisables par ce logiciel. Toutes les manières d'y parvenir ne seront certes pas explorer, mais cela vous donnera une première approche.*

## 1 Création d'une base de données dans SAS

1. Commencer par associer à votre répertoire de travail un nom logique (tp1 pour tout le monde) utilisable uniquement pendant la session SAS ouverte. Utilisation de "LIBNAME".
2. Considérons l'exemple suivant :

```
DATA test;  
INPUT x1 x2 x3 x4 x5 $;  
CARDS;  
78 5 5 1161 ABELIEN  
35 6 3 1336 ALMERIC  
82 1 5 1499 ANIEL  
;  
RUN;
```

Décrire chacune des instructions.

3. En considérant l'exemple précédent, créer une table SAS permanente test2 qui contient les données suivantes :

1	t	2	3	a
4	abc	5	6	b
7	z	8	9	c
10	rt	11	12	d

4. Afficher cette base.
5. Afficher uniquement les observations 2 et 3.

6. Afficher les observations à partir de l'observation 3.

7. Comprendre ce qui se passe dans les programmes suivants :

(a) prog1 :

```
DATA test;
INPUT x1 x2 x3 x4 x5 ;
CARDS;
1 2 3 4 5
6 7 8 9 10
11 12 13 14 15
16 17 18 19 20
21 22 23 24 25
;
RUN;
```

(b) prog2 :

```
DATA test;
INPUT x1 x2 x3;
CARDS;
1 2 3 4 5
6 7 8 9 10
11 12 13 14 15
16 17 18 19 20
21 22 23 24 25
;
RUN;
```

(c) prog3 :

```
DATA test;
INPUT x1 x2 x3;
CARDS;
1
2
3
4
...
...
...
24
25
;
RUN;
```

(d) prog4 :

```

DATA test;
INPUT x1 x2 x3 x4 x5 x6;
CARDS;
1 2 3 4 5
6 7 8 9 10
11 12 13 14 15
16 17 18 19 20
21 22 23 24 25
;
RUN;

```

(e) prog5 :

```

DATA test;
INPUT x1 x2 x3 x4 x5;
CARDS;
1 2 3 4 5
6 7 8 9 10
11 12 13 14
16 17 18 19 20
21 22 23 24 25
;
RUN;

```

8. Comment gérer les données manquantes ? Comparer les instructions suivantes :

(a) prog1 :

```

DATA test;
INFILE CARDS MISSOVER;
INPUT x1 x2 x3 x4 x5;
CARDS;
1 2 3 4 5
6 7 8 9 10
11 12 13 14
16 17 18 19 20
21 22 23 24 25
;
RUN;

```

(b) prog2:

```

DATA test;
INFILE CARDS MISSOVER;
INPUT x1 x2 x3 x4 x5 ;
CARDS;
1 2 3 5
6 8 9 10

```

```

12 13 14 15
17 18 19 20
21 22 24 25
;
RUN;

```

(c) prog3:

```

DATA test;
INPUT x1 x2 x3 x4 x5 ;
CARDS;
1 2 3 . 5
6 . 8 9 10
. 12 13 14 15
. 17 18 19 20
21 22 . 24 25
;
RUN;

```

## 2 Importation de données

1. Récupérer à partir de ma page web, le fichier test22.txt
2. 0 l'aide d'une instruction infile, récupérer les données contenues dans test22.txt
3. Faire de même mais en ne récupérant que les observations allant des lignes 20 à 50.
4. Considérer à présent l'importation de test24.txt
5. Faire de même avec test100.txt
6. Que font les programmes suivants :

(a) prog1 :

```

DATA test;
INPUT x1 $ x2;
CARDS;
aa 1
aa a 1
aa a a 1
;
RUN;

```

(b) prog2 :

```

DATA test;
INPUT x1 & $ x2;
CARDS;
aa 1

```

```

aa a 1
aa a a 1
;
RUN;

```

(c) prog3 :

```

DATA test;
INFILE CARDS DLM='#';
INPUT x1 $ x2;
CARDS;
aa#1
aa a#1
aa a a#1
;
RUN;

```

7. Que se passe t'il avec les instructions suivantes et pourquoi?

```

DATA clint;
INPUT titre_us $ titre_fra $ acteur $ annee note;
CARDS;
Hoover Hoover Leonardo DiCaprio 2012 N8.6
Hereafter Hereafter Matt Damon 2010 N8.3
Invictus Invictus Morgan Freeman 2009 N7.5
Gran Torino Gran Torino Clint Eastwood 2008 N8.4
Changeling L'échange Angelina Jolie 2008 N8.0
Letters from Iwo Jima Lettres d'Iwo Jima Ken Watanabe 2006 N8.1
Flags of Our Fathers Mémoires de nos pères Ryan Phillippe 2006 N7.2
;
RUN;

```

8. Comment en insérant la commande INPUT annee 64-67 titre\_us \$ 1-21 titre\_fra \$ 23-43 note 71-73 acteur \$ 46-62; en lieu et place de la commande INPUT précédente peut-on avoir le tableau correct?

9. Que fait ce qui suit?

```

DATA floride;
INFILE 'C:\intro_SAS\ fichiers\floride.txt' LRECL=38000;
INPUT base annee pluie @@;
IF pluie=-9999 THEN pluie=.;
RUN;

```

10. Résoudre l'exercice suivant : Le fichier orleans.txt contient les températures journalières maximales observées entre le 1er janvier 2005 et fin juillet 2009 sur Orléans. Les données sont présentées sur 168 lignes et présentes 10 données de températures. Chaque température est précédée de l'année d'observation, puis du mois et enfin du jour. Rédiger un programme qui construira la table.

### 3 Gestion des données manquantes

1. Quelle est la différence entre ce qui suit :

(a) progmi1 :

```
DATA portable;
INPUT ope x1 $;
CARDS;
1 abc
X def
2 ghi
1 jkl
. mno
2 pqr
;
RUN;
```

(b) progmi2 :

```
MISSING X;
DATA portable;
INPUT ope x1 $;
CARDS;
1 abc
X def
2 ghi
1 jkl
. mno
2 pqr
;
RUN;
```

(c) progmi3 :

```
DATA portable;
INPUT x0 x1 $;
CARDS;
1 abc
2 def
.A ghi
;B jkl
.ccc mno
2 pqr
;
RUN;
```

(d) progmi4 :

```
DATA test;
infile "... missing2.txt" DLM="?" DSD;
INPUT x1 $ x2 $ x3 $ n1 n2 n3;
RUN;
```

## 4 Exécution

1. l'instruction OUTPUT :

(a) Comprendre les instructions suivantes :

```
data capital;
capital=1000;
annee=0;
OUTPUT;
annee=1;
capital=capital*1.05;
OUTPUT;
annee=2;
capital=capital*1.05;
OUTPUT;
annee=3;
capital=capital*1.05;
OUTPUT;
annee=4;
capital=capital*1.05;
;
run;

data femme homme;
set ensemble;
if sexe='femme' then output femme;
else output homme;
run;
```

2. résoudre l'exercice suivant :

- (a) Créer une table test comprenant 30 observations et une variable x qui prend la valeur 1 pour les 10 premières observations, puis la valeur 2 pour les 10 suivantes et la valeur 3 pour les dernières observations.
- (b) Que fait ce qui suit :

```
data tab1 tab2 tab3;
set test;
if x=1 then output tab1;
if x=2 then output tab2;
```

```
run;
```

(c) Et si l'on introduit une instruction output; juste avant run; ?

3. l'instruction drop/keep/delete :

- Construire à partir de la base de données cross une base cross1 où l'on ne conserve que les variables annee et sexe.
- Construire à partir de la base de données cross une base cross2 où l'on retire les variables annee et sexe.
- Retour sur la base ensemble. Créer la base femme mais en utilisant delete, output puis where.
- Que fait ce programme ?

```
data femme;  
set ensemble (where=(sexe='femme' and cheveux='blond'));  
run;
```

4. l'instruction select/when/otherwise :

(a) Comprendre ce qui suit :

```
data reg1 reg2 reg3 autre;  
SET dep1;  
SELECT (depnum);  
WHEN (45,37,36,41,18,28) OUTPUT reg1;  
WHEN (10,52,51) OUTPUT reg2;  
WHEN (62,59) OUTPUT reg3;  
OTHERWISE OUTPUT autre;  
END;  
RUN;
```