

# LA GESTION DES FICHIERS EN LOTUS SCRIPT

## LE TRAITEMENT DES FICHIERS EN LOTUSSCRIPT



**Cette création est mise à disposition selon le Contrat Paternité**

- **Pas d'Utilisation Commerciale**
- **Pas de Modification disponible en ligne**

<http://creativecommons.org/licenses/by-nc-nd/2.0/fr/>

**ou par courrier postal à Creative Commons, 559 Nathan Abbott Way,  
Stanford, California 94305, USA.**

<b>1</b>	<b>LES FICHIERS TEXTE</b>	<b>5</b>
1.1	Ouverture	5
1.2	Les pointeurs de fichiers	7
1.3	La fermeture d'un fichier	8
1.4	Ecriture dans un fichier	8
1.5	Lecture d'un fichier	14
1.6	Synthèse – Enregistrement de longueur variable avec séparateur sans delimiteur	21
1.7	Conclusion	23
<b>2</b>	<b>LES FICHIERS A ACCES DIRECT</b>	<b>24</b>
2.1	Ouverture	24
2.2	Déclaration du format d'enregistrement	24
2.3	Ouverture du fichier en fonction de la structure d'enregistrement	25
2.4	Ecriture d'enregistrements	25
2.5	Ecriture à un rang donné	25
2.6	Exemple complet	26
2.7	Le pointeur d'enregistrement	27
2.8	Lecture d'un enregistrement	28
2.9	Accès direct à un enregistrement	28
2.10	Lecture séquentielle	28

### **PREAMBULE**

Pour la gestion des fichiers en LotusScript il est inutile de passer par les objets WScript de Windows ou autres méthodes.

Depuis les premières versions du langage Basic il existe des ordres pour traiter des fichiers.

Pour rappel, LotusScript est un dérivé du Langage Basic adapté au développement sous Lotus Notes.

# 1 LES FICHIERS TEXTE

## 1.1 OUVERTURE

**Open *fileName* [For {Input | Output | Append} ] As *fileNumber***

**FileName** : Correspond au nom du fichier sous forme de chaîne de caractère.  
Filename peut aussi représenter une variable déclarée par un ordre Dim ou post fixée par le signe \$.

**Input** : Le fichier est ouvert en mode **Lecture**.

**Output** : Le fichier est ouvert en mode **Ecriture**

**Append** : Le fichier est ouvert en mode **Ajout en fin de fichier**

**FileNumber** : Entier représentant le pointeur du fichier

### 1.1.1 OUVERTURE EN LECTURE

```
Open "C:\DATA\FICLI.TXT" For Input As 1
```

### 1.1.2 OUVERTURE EN ECRITURE

```
Open "C:\DATA\FICLI.TXT" For Output As 1
```

☞ **Si le fichier existe celui-ci est écrasée.**

### 1.1.3 OUVERTE EN MODE AJOUT (FIN DE FICHIER)

```
Open "C:\DATA\FICLI.TXT" For Append As 1
```

☞ **Le fichier n'est pas écrasé s'il existe et il est créé s'il n'existe pas.**

## LE TRAITEMENT DES FICHIERS EN LOTUSSCRIPT

### REMARQUE

Le nom de fichier aurait très bien pu être passé via une variable afin d'avoir une ouverture dynamique sur le nom du fichier.

### Exemple

```
Dim Fic as String
. . .
. . .
Fic="C:\DATA\FICLI.TXT"
. . .
. . .
Open Fic For Output As 1
```

## 1.2 LES POINTEURS DE FICHIERS

La référence au fichier s'effectue via son pointeur ; celui-ci correspond à l'entier suivant la clause **As**.

Cette valeur peut être laissée à la liberté du programmeur ou définie automatiquement par le programme. Le programmeur peut utiliser la fonction **FreeFile** qui renvoie automatiquement le premier pointeur disponible.

### EXEMPLE

```
Dim Fp1 as Integer
Dim Fp2 as integer
. . .
. . .
Fp1=FreeFile()
Open "C:\DATA\FICLI1.TXT" For Input As Fp1
. . .
. . .
Fp2=FreeFile()
Open "C:\DATA\FICLI2.TXT" For Output As Fp2
```

⚠ **Attention de ne pas faire cette erreur ci-dessous**

```
Dim Fp1 as Integer
Dim Fp2 as integer
. . .
. . .
Fp1=FreeFile()
Fp2=FreeFile()
Open "C:\DATA\FICLI1.TXT" For Input As Fp1
. . .
. . .
Open "C:\DATA\FICLI2.TXT" For Output As Fp2
```

☞ En effet, **Fp1** et **Fp2** auront la même valeur, soit **1**. Le calcul du prochain pointeur disponible **se fait après ouverture du fichier** car cette fonction recherche automatiquement le suivant qui n'a pas été ouvert.

## 1.3 LA FERMETURE D'UN FICHIER

La fermeture d'un fichier se réalise par l'instruction **Close FileNumber**. Une instruction **Close** sans FileNumber fermera tous les fichiers ouverts.

### EXEMPLES

```
'// Ferme le fichier n° 1
Close 1

'// Ferme le fichier associé au pointeur
Fp1
Close Fp1

'// Ferme tous les fichiers ouverts
Close
```

## 1.4 ECRITURE DANS UN FICHIER

### 1.4.1 ECRITURE EN MODE CREATION

#### Rappel du mode Ouverture

#### **Open FileName For Output As FileNumber**

L'écriture dans un fichier peut se faire sous différentes formes avec différents types d'instructions.

Il existe deux instructions :

- **Print #FileNumber, Chaine\$**
- **Write #FileNumber, Chaine\$**

### 1.4.1.1 L'INSTRUCTION WRITE

#### 1.4.1.1.1 ECRITURE RAPIDE AVEC FORMATAGE AUTOMATIQUE

```
Dim Fp As Integer
Dim Fic As String
Fp=Freefile()
Fic="C:\ALPHA\FICLI.TXT"
Open Fic For Output As Fp
Write #Fp, "CLI01", "MARCHAND", "LILLE", 10, "ROUTE"
Write #Fp, "CLI02", "GOURMET", "PARIS", 50, "MER"
Write #Fp,
"CLI03", "ATTCHOUM", "MARSEILLE", 140, "AIR"
Write #Fp, "CLI04", "PETIT", "RENNES", 130, "MER"
Write #Fp, "CLI05", "LEGRAND", "TROYES", 146, "ROUTE"
Close Fp
```

#### RESULTAT OBTENU DANS LE FICHIER

```
"CLI01", "MARCHAND", "LILLE", 10, "ROUTE"
"CLI02", "GOURMET", "PARIS", 50, "MER"
"CLI03", "ATTCHOUM", "MARSEILLE", 140, "AIR"
"CLI04", "PETIT", "RENNES", 130, "MER"
"CLI05", "LEGRAND", "TROYES", 146, "ROUTE"
```

#### Remarques

Cette instruction a placé automatiquement la virgule comme séparateur de champ et a encadré les champs par des guillemets. Il est à noter que les valeurs numériques ne sont pas encadrées par des guillemets.

En revanche le programmeur n'est pas libre sur le format des enregistrements et les séparateurs de champs. Ce formatage est imposé par cette instruction.

### 1.4.1.2 L'INSTRUCTION PRINT

Cette instruction laisse plus de liberté sur le formatage des enregistrements.

En effet le programmeur peut imposer son délimiteur et son séparateur de champ. Il peut construire des enregistrements de taille fixe ou variable.

**Print #FileNumber, Chaine\$**

## LE TRAITEMENT DES FICHIERS EN LOTUSSCRIPT

### 1.4.1.2.1 ECRITURE AVEC ALIGNEMENT AUTOMATIQUE

```
Dim Fp As Integer
Dim Fic As String
Fp=Freefile()
Fic="C:\ALPHA\FICLI2.TXT"
Open Fic For Output As Fp
Print #Fp, "CLI01", "MARCHAND", "LILLE", 10, "ROUTE"
Print #Fp, "CLI02", "GOURMET", "PARIS", 50, "MER"
Print #Fp,
"CLI03", "ATTCHOUM", "MARSEILLE", 140, "AIR"
Print #Fp, "CLI04", "PETIT", "RENNES", 130, "MER"
Print #Fp,
"CLI05", "LEGRAND", "TROYES", 146, "ROUTE"
Close Fp1
```

#### RESULTAT OBTENU DANS LE FICHER

CLI01	MARCHAND	LILLE	10	ROUTE
CLI02	GOURMET	PARIS	50	MER
CLI03	ATTCHOUM	MARSEILLE	140	AIR
CLI04	PETIT	RENNES	130	MER
CLI05	LEGRAND	TROYES	146	ROUTE

### 1.4.1.2.2 ECRITURE AVEC CONCATENATION PARTIELLE

```
Dim Fp As Integer
Dim Fic As String
Fp=Freefile()
Fic="C:\ALPHA\FICLI3.TXT"
Open Fic For Output As Fp
Print #Fp,
"CLI01";"MARCHAND";"LILLE";10;"ROUTE"
Print #Fp, "CLI02";"GOURMET";"PARIS";50;"MER"
Print #Fp,
"CLI03";"ATTCHOUM";"MARSEILLE";140;"AIR"
Print #Fp, "CLI04";"PETIT";"RENNES";130;"MER"
Print #Fp,
"CLI05";"LEGRAND";"TROYES";146;"ROUTE"
Close Fp1
```

## LE TRAITEMENT DES FICHIERS EN LOTUSSCRIPT

### RESULTAT OBTENU DANS LE FICHIER

```
CLI01MARCHANDLILLE 10 ROUTE
CLI02GOURMETPARIS 50 MER
CLI03ATTCHOUMMARSEILLE 140 AIR
CLI04PETITRENNES 130 MER
CLI05LEGRANDTROYES 146 ROUTE
```

#### Remarque

Des espaces sont laissés pour la justification des valeurs numériques.  
Ce fichier est difficilement exploitable à la relecture.

### 1.4.1.2.3 ECRITURE AVEC CONCATENATION TOTALE

```
Dim Fp As Integer
Dim Fic As String
Fp=Freefile()
Fic="C:\ALPHA\FICLI3.TXT"
Open Fic For Output As Fp
Print #Fp,
"CLI01"&"MARCHAND"&"LILLE"&10&"ROUTE"
Print #Fp,
"CLI02"&"GOURMET"&"PARIS"&50&"MER"
Print #Fp,
"CLI03"&"ATTCHOUM"&"MARSEILLE"&140&"AIR"
Print #Fp,
"CLI04"&"PETIT"&"RENNES"&130&"MER"
Print #Fp,
"CLI05"&"LEGRAND"&"TROYES"&146&"ROUTE"
Close Fp1
```

### RESULTAT OBTENU DANS LE FICHIER

```
CLI01MARCHANDLILLE10ROUTE
CLI02GOURMETPARIS50MER
CLI03ATTCHOUMMARSEILLE140AIR
CLI04PETITRENNES130MER
CLI05LEGRANDTROYES146ROUTE
```

#### Remarques

Toutes les zones sont mises bout à bout. Ce type de fichier aussi difficilement exploitable à la relecture (pas délimiteurs et séparateurs de champs).

### 1.4.1.2.4 ECRITURE AVEC SEPARATEUR PERSONNALISE

```
Dim Fp As Integer
Dim Fic As String
Dim E1 As String
Dim E2 As String
Dim E3 As String
Dim E4 As String
Dim E5 As String
Dim sp As String
Dim dm As String

dm=| |
sp=|; |
Fp=Freefile()
Fic="C:\ALPHA\FICLI4.TXT"
Open Fic For Output As Fp
E1= dm & "CLI01" & dm & sp & dm & "MARCHAND" & dm
E2= dm & "CLI02" & dm & sp & dm & "GOURMET" & dm
E3= dm & "CLI03" & dm & sp & dm & "ATTCHOUM" & dm
E4= dm & "CLI04" & dm & sp & dm & "PETIT" & dm
E5= dm & "CLI05" & dm & sp & dm & "LEGRAND" & dm
Print #Fp, E1
Print #Fp, E2
Print #Fp, E3
Print #Fp, E4
Print #Fp, E5
Close Fp1
```

#### RESULTAT OBTENU DANS LE FICHIER

```
"CLI01";"MARCHAND"
"CLI02";"GOURMET"
"CLI03";"ATTCHOUM"
"CLI04";"PETIT"
"CLI05";"LEGRAND"
```

#### 👉 Remarque

Ce procédé certes un peu long permet de générer un enregistrement avec des séparateurs voulus. Ici le séparateur de champ est le point-virgule. Il faut cette fois préciser le délimiteur de champ, ici les guillemets.

## LE TRAITEMENT DES FICHIERS EN LOTUSSCRIPT

### NE PAS OUBLIER :

La fin d'un enregistrement est marquée par la combinaison de caractères **Retour chariot+Ligne suivante**, soit en hexadécimal **x0D et x0A ; Chr\$(13)+Chr\$(10)**.

Cette combinaison de caractère est exploitée pour la lecture des fichiers de type texte.

### 1.4.2 ECRITURE EN MODE AJOUT

👉 Rappel du mode Ouverture

#### Open FileName For Append As FileName

#### FICHER FICLI4 AVANT ECRITURE

```
"CLI01"; "MARCHAND"  
"CLI02"; "GOURMET"  
"CLI03"; "ATTCHOUM"  
"CLI04"; "PETIT"  
"CLI05"; "LEGRAND"
```

#### CODE

```
Dim Fp As Integer  
Dim Fic As String  
Dim E6 As String  
Dim sp As String  
Dim dm As String  
dm="|"  
sp=";|"  
Fp=Freefile()  
Fic="C:\ALPHA\FICLI4.TXT"  
Open Fic For Append As Fp  
E6= dm & "CLI06" & dm & sp & dm & "ARTS ET DECO" & dm  
Print #Fp, E6  
Close Fp1
```

#### FICHER FICLI4 APRES AJOUT

```
"CLI01"; "MARCHAND"  
"CLI02"; "GOURMET"  
"CLI03"; "ATTCHOUM"  
"CLI04"; "PETIT"  
"CLI05"; "LEGRAND"  
"CLI06"; "ARTS ET DECO"
```

## 1.5 LECTURE D'UN FICHIER

La lecture d'un fichier peut être réalisée via deux instructions :

- **Input #FileNumber, Chaine1\$, Chaine2\$, ..., Chaine\$**
- **Line Input #FileNumber, Chaine\$**

La lecture d'un fichier texte est effectuée de manière séquentielle tant que la fin de fichier (**Eof**) n'est pas atteinte.

### 1.5.1 DETECTION DE LA FIN DE FICHIER

La détection de la fin de fichier est fournie par la fonction **Eof(FileNumber)**. Cette fonction renvoie **True** quand celle-ci est atteinte. Le fichier doit être au préalable ouvert en mode **Lecture**.

#### Cas général

```
While Not Eof(1)
. . . instruction de lecture
. . . traitements
Wend
```

#### Remarque

**Eof** est à **True** dès que le dernier enregistrement est lu. Dans certains fichiers texte, le marque de fichier de fichier est codifié **x1A**.

### 1.5.2 LECTURE PAR L'INSTRUCTION INPUT

**Input #FileNumber, Chaine1\$, Chaine2\$,..., Chaine\$**

#### 1.5.2.1 LECTURE D'UN FICHIER AVEC SEPARATEUR VIGURLE

On se propose de relire le fichier suivant

```
"CLI01", "MARCHAND", "LILLE", 10, "ROUTE"
"CLI02", "GOURMET", "PARIS", 50, "MER"
"CLI03", "ATTCHOUM", "MARSEILLE", 140, "AIR"
"CLI04", "PETIT", "RENNES", 130, "MER"
"CLI05", "LEGRAND", "TROYES", 146, "ROUTE"
```

La relecture des champs doit respecter l'ordre d'écriture de ceux-ci afin de ne pas avoir de décalage dans les rubriques lues.

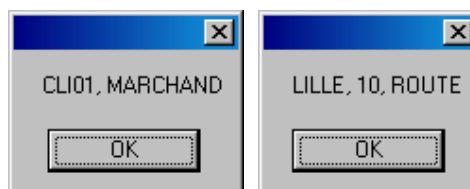
## LE TRAITEMENT DES FICHIERS EN LOTUSSCRIPT

```
Dim Fp As Integer
Dim Fic As String
Dim Code As String
Dim Nom As String
Dim Ville As String
Dim coef As Integer
Dim Transport As String
Fp=Freefile()
Fic="C:\ALPHA\FICLI.TXT"
Open Fic For Input As Fp
While Not Eof(Fp)
    Input #Fp, Code, Nom, Ville, Coef, Transport
    MsgBox Code & ", " & Nom & ", " & Ville & ",
    " & Coef & ", " & Transport
Wend
Close Fp1
```



### 1.5.2.1.1 VARIANTE 1

```
While Not Eof(Fp)
    Input #Fp, Code, Nom
    Input #Fp, Ville, Coef, Transport
    MsgBox Code & ", " & Nom
    MsgBox Ville & ", " & Coef & ", " & Transport
Wend
```



## LE TRAITEMENT DES FICHIERS EN LOTUSSCRIPT

### Mise en garde

- La relecture se fait dans l'ordre des champs
- La fin d'enregistrement est détectée par la séquence de caractères x0D et x0A

### 1.5.2.1.2 VARIANTE 2

```
Dim rub as string
While Not Eof(Fp)
    Input #Fp, Rub
    MsgBox Rub
Wend
```

Cet exemple afficherait :

- CLI01
- MARCHAND
- LILLE
- 10
- ROUTE
- Et ainsi de suite jusqu'à la fin de fichier

Avec cette instruction il convient donc d'être très prudent sur l'ordre de relecture des zones qui composent l'enregistrement.

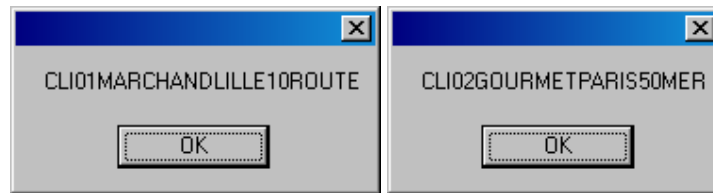
### 1.5.2.2 LECTURE D'UN FICHIER SANS SEPARATEUR

Exemple de fichier :

```
CLI01MARCHANDLILLE10ROUTE
CLI02GOURMETPARIS50MER
CLI03ATTCHOUMMARSEILLE140AIR
CLI04PETITRENNES130MER
CLI05LEGRANDTROYES146ROUTE
```

```
Dim Fp As Integer
Dim Fic As String
Dim rub As String
Fp=Freefile()
Fic="C:\ALPHA\FICLI3.TXT"
Open Fic For Input As Fp
While Not Eof(Fp)
    Input #Fp, Rub
    MsgBox Rub
Wend
Close Fp1
```

## LE TRAITEMENT DES FICHIERS EN LOTUSSCRIPT



Et ainsi de suite.

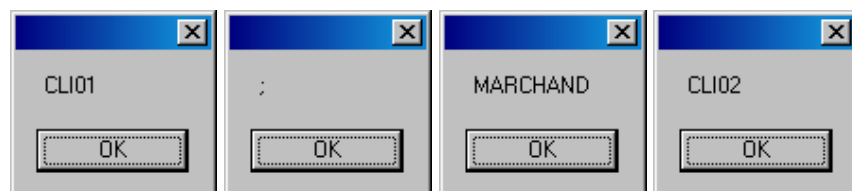
Dans cet exemple c'est au programmeur de découper l'enregistrement à condition d'en connaître sa structure. **Ce dernier est ici inexploitable car les rubriques ne sont pas délimitées.**

### 1.5.2.3 LECTURE D'UN FICHIER AVEC SEPARATEUR POINT VIRGULE

Exemple de fichier

```
"CLI01";"MARCHAND"  
"CLI02";"GOURMET"  
"CLI03";"ATTCHOUM"  
"CLI04";"PETIT"  
"CLI05";"LEGRAND"  
"CLI06";"ARTS ET DECO"
```

```
Dim Fp As Integer  
Dim Fic As String  
Dim rub As String  
Fp=Freefile()  
Fic="C:\ALPHA\FICLI4.TXT"  
Open Fic For Input As Fp  
While Not Eof(Fp)  
    Input #Fp, Rub  
    MsgBox Rub  
Wend  
Close Fp1
```



#### **Remarque**

Le point virgule est interprété.

## 1.5.3 LECTURE PAR LINE INPUT

### Line Input #FileNumber, Enregi\$

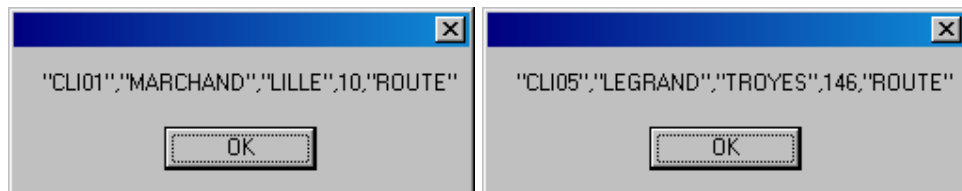
Cette instruction répond à la problématique du traitement des enregistrements non structurés ou avec des séparateurs en dehors du commun.

#### 1.5.3.1 LECTURE D'UN FICHIER AVEC DELIMITEUR ET SEPARATEUR

Traitement du fichier

```
"CLI01", "MARCHAND", "LILLE", 10, "ROUTE"  
"CLI02", "GOURMET", "PARIS", 50, "MER"  
"CLI03", "ATTCHOUM", "MARSEILLE", 140, "AIR"  
"CLI04", "PETIT", "RENNES", 130, "MER"  
"CLI05", "LEGRAND", "TROYES", 146, "ROUTE"
```

```
Dim Fp As Integer  
Dim Fic As String  
Dim rub As String  
Fp=Freefile()  
Fic="C:\ALPHA\FICLI.TXT"  
Open Fic For Input As Fp  
While Not Eof(Fp)  
    Line Input #Fp, Rub  
    Msgbox Rub  
Wend  
Close Fp1
```



#### **Remarque**

Dans cet exemple on obtient bien l'enregistrement complet avec son séparateur et le délimiteur de zone. A partir de là son exploitation est assez aisée.

## 1.5.3.2 LECTURE D'UN FICHER FORMATE

Traitement du fichier

CLI01	MARCHAND	LILLE	10	ROUTE
CLI02	GOURMET	PARIS	50	MER
CLI03	ATTCHOUM	MARSEILLE	140	AIR
CLI04	PETIT	RENNES	130	MER
CLI05	LEGRAND	TROYES	146	ROUTE

```
Dim Fp As Integer
Dim Fic As String
Dim rub As String
Fp=Freefile()
Fic="C:\ALPHA\FICLI2.TXT"
Open Fic For Input As Fp
While Not Eof(Fp)
  Line Input #Fp, Rub
  Msgbox Rub
Wend
Close Fp1
```

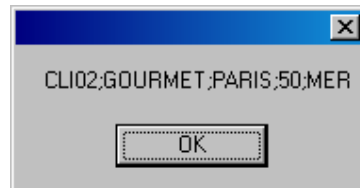


## 1.5.3.3 LECTURE D'UN FICHER AVEC SEPARATEUR SANS DELIMITEUR

```
CLI01;MARCHAND;LILLE;10;ROUTE
CLI02;GOURMET;PARIS;50;MER
CLI03;ATTCHOUM;MARSEILLE;140;AIR
CLI04;PETIT;RENNES;130;MER
CLI05;LEGRAND;TROYES;146;ROUTE
```

## LE TRAITEMENT DES FICHIERS EN LOTUSSCRIPT

```
Dim Fp As Integer
Dim Fic As String
Dim rub As String
Fp=Freefile()
Fic="C:\ALPHA\FICLI3.TXT"
Open Fic For Input As Fp
While Not Eof(Fp)
    Line Input #Fp, Rub
    Msgbox Rub
Wend
Close Fp1
```



## 1.6 SYNTHÈSE – ENREGISTREMENT DE LONGUEUR VARIABLE AVEC SEPARATEUR SANS DELIMITEUR

Comment traiter ce fichier et récupérer les rubriques ?

```
CLI01;MARCHAND;LILLE;10;ROUTE  
CLI02;GOURMET;PARIS;50;MER  
CLI03;ATTCHOUM;MARSEILLE;140;AIR  
CLI04;PETIT;RENNES;130;MER  
CLI05;LEGRAND;TROYES;146;ROUTE
```

```
Dim Fp As Integer  
Dim Fic As String  
Dim rub As String  
Dim Enregi As Variant  
Dim Ligne As String  
Fp=Freefile()  
Fic="C:\ALPHA\FICLI3.TXT"  
Open Fic For Input As Fp  
While Not Eof(Fp)  
    Line Input #Fp, Ligne  
    '// Découpage de l'enregistrement  
    Enregi=atExplode(ligne,";")  
    '// Lecture des zones de l'enregistrement  
    '// Celui est contenu dans un tableau  
    Forall r In enregi  
        Msgbox r  
    End Forall  
Wend  
Close Fp1  
  
Resume fin  
fin:  
End Function
```

```
Function atExplode(s As String, div As String) As Variant  
  On Error Goto handleError  
  Redim result(0 To 0) As String  
  Dim i As Integer  
  Dim pos As Long  
  Dim oldpos As Long  
  Dim skip As Long  
  oldpos = 1  
  skip = Len(div)  
  pos = Instr(s, div)  
  Do Until pos = 0  
    Redim Preserve result(0 To i+1)  
    result(i) = Mid$(s, oldpos, pos-oldpos)  
    i = i + 1  
    oldpos = pos + skip  
    pos = Instr(oldpos, s, div)  
  Loop  
  result(i) = Mid$(s, oldpos)  
  atExplode = result  
  Exit Function  
handleError:  
  Print "Erreur atExplode ";Err;" ";Error$;" ligne ";Err  
End Function
```

### 1.7 CONCLUSION

L'instruction **Line Input** est très pratique pour le traitement des fichiers texte dont la longueur d'enregistrement est variable avec des délimiteurs et séparateurs particuliers.

Les séparateur guillemets et point virgule ne sont pas interprétés. La détection de fin d'enregistrement est fondée sur la présence des caractères x0D x0A.

---

## 2 LES FICHIERS A ACCES DIRECT

Ces fichiers ne sont pas directement exploitables avec un éditeur de texte tel que "NotePad".

Ils ne peuvent être relus que par programmes LS.

Ces fichiers ont l'avantage d'offrir la possibilité d'accéder à un enregistrement en fonction de son rang encore appelé Record Range Number (RRN).

### 2.1 OUVERTURE

Open fileName For Random As fileNumber Len = recordLength

FileName et FileNumber répondent aux mêmes règles que les fichiers texte.  
Len : Longueur de l'enregistrement.

### 2.2 DECLARATION DU FORMAT D'ENREGISTREMENT

A l'inverse des fichiers textes le format de l'enregistrement doit être structuré et déclaré par le biais de la clause **Type** et **End Type**.

```
Type DSClient
  Code As String * 5
  Nom As String * 25
  Ville As String * 25
  Distance As Integer
  Transport As String * 10
End Type
```

**DSClient** correspond au nom de la structure d'enregistrement. Celle ne peut pas être utilisée directement. On impérativement passer par un ordre **Dim**.

**Dim Client As DSClient**

L'accès aux rubriques de l'enregistrement se fait par :

**NomDeLaVariableStructure.NomDeZone**

**Client.Code="CLI01"**

**Client.Distance=100**

## 2.3 OUVERTURE DU FICHIER EN FONCTION DE LA STRUCTURE D'ENREGISTREMENT

```
Dim client As DSClient
Dim fileNum As Integer
Dim recNum As Integer
Dim fileName As String
fileNum = Freefile()
fileName = "c:\alpha\randcli.dat"

'// Ouverture du fichier
'// Fichier à accès direct
Open fileName$ For Random As fileNum Len = Len(client)
```

### Remarque

En général l'extension de ce type de fichier est .DAT afin de les différencier avec les fichiers txt. Bien entendu ceci n'est qu'une pure convention. Il convient d'utiliser la fonction **Len** pour calculer la taille de la structure.

## 2.4 ECRITURE D'ENREGISTREMENTS

L'écriture d'un enregistrement s'effectue via l'instruction **Put**.

**Put [#] fileNumber , [ recordNumber ] , variableName**

Le RecordNumber est facultatif. S'il est omis l'enregistrement est ajouté à la fin du fichier.

VariableName correspond à la variable associée à la structure d'enregistrement.

## 2.5 ECRITURE A UN RANG DONNE

**Put #fileNum, 3 , client**

Écriture de l'enregistrement au 3ème rang.

## 2.6 EXEMPLE COMPLET

Cet exemple a pour but d'écrire 5 enregistrements dans un fichier les un à la suite des autres.

### **Type DSClient**

```
Code As String * 5
Nom As String * 25
Ville As String * 25
Distance As Integer
Transport As String * 10
```

End Type

### **Dim client As DSClient**

```
Dim fileNum As Integer
Dim recNum As Integer
Dim fileName As String
```

```
fileNum = Freefile()
fileName = "c:\alpha\randcli.dat"
```

```
'// Ouverture du fichier
'// Fichier à accès direct
```

```
Open fileName For Random As fileNum Len = Len(client)
```

```
'// Alimentation des enregistrement
```

```
Client.Code="CLI01"
```

```
Client.Nom="MARCHAND"
```

```
Client.Ville="LILLE"
```

```
Client.Distance=10
```

```
Client.Transport="ROUTE"
```

```
'// Ecriture de l'enregistrement
```

```
Put #fileNum, , client
```

```
Client.Code="CLI02"
```

```
Client.Nom="GOURMET"
```

```
Client.Ville="PARIS"
```

```
Client.Distance=50
```

```
Client.Transport="MER"
```

```
'// Ecriture au rang suivant dans le fichier
Put #fileNum, , client

Client.Code="CLI03"
Client.Nom="ATTCHOUM"
Client.Ville="MARSEILLE"
Client.Distance=140
Client.Transport="AIR"
'// Ecriture au rang suivant dans le fichier
Put #fileNum, , client

Client.Code="CLI04"
Client.Nom="PETIT"
Client.Ville="RENNES"
Client.Distance=130
Client.Transport="MER"
'// Ecriture au rang suivant dans le fichier
Put #fileNum, , client

Client.Code="CLI05"
Client.Nom="LEGRAND"
Client.Ville="TROYE"
Client.Distance=146
Client.Transport="ROUTE"
'// Ecriture au rang suivant dans le fichier
Put #fileNum, , client

Close fileNum
```

### 2.7 LE POINTEUR D'ENREGISTREMENT

Après chaque écriture le pointeur de rang prend la taille de l'enregistrement +1.

La taille de l'enregistrement se mesure avec la fonction **Len** comme **Len(Client)**.

Le RRN peut être positionné par l'instruction **Seek**.

**Seek [#]fileNumber , position**

**Seek #fileNum, 1**

Place le RRN en début de fichier.

### 2.8 LECTURE D'UN ENREGISTREMENT

La lecture d'un enregistrement s'effectue par l'instruction **Get**.

**Get [#]fileNumber , [ recordNumber ] , variableName**

### 2.9 ACCES DIRECT A UN ENREGISTREMENT

```
Get #fileNum , 2 , client
```

Permet d'accéder à l'enregistrement de rang n° 2.

### 2.10 LECTURE SEQUENTIELLE

```
Dim client As DSClient
Dim fileNum As Integer
Dim recNum As Integer
Dim fileName As String
fileNum = Freefile()
fileName = "c:\alpha\randcli.dat"
'// Ouverture du fichier
'// Fichier à accès direct
Open fileName For Random As fileNum Len = Len(client)
'// Lecture séquentielle
While Not Eof(fileNum)
    Get #fileNum,,client
    MsgBox client.code & " - " & client.nom
Wend
Close fileNum
```

👉 **Remarque**

L'ordre **Seek** est plus souvent exploiter dans le traitement des fichiers binaires.