

# **CardAK**

Card Interchange Files Manager and Helpers

(Card Army Knife)

**DRAFT**

Versión utilizada: 0.8.x (y anteriores)

DRAFT VERSION

## Tabla de contenidos

Introducción.....	4
Como utilizar este manual.....	4
Características y funcionalidades.....	4
Casos de uso.....	5
Identificar y obtener información de los archivos.....	5
Validar los archivos en busca de errores.....	9
Corregir errores automáticamente.....	11
Cambiar el formato de los archivos.....	12
Búsqueda de datos en archivos IPM.....	13
Visualizar el contenido en algún programa externo.....	19
Manejar archivos lógicos.....	20
Eliminar registros de un archivo.....	20
Eliminar campos en algunos registros.....	22
Pasar registros de un archivo a otro.....	22
Comandos.....	24
CHOP.....	25
CONVERT.....	26
DELETE.....	29
DESCRIBE.....	32
DISTRIBUTE.....	34
DUPLICATES.....	39
EXPORT.....	40
FILTER.....	41
FIX.....	45
GREP.....	46
IDENTIFY.....	47
IMPORT.....	48
JOIN.....	49
OPEN.....	50
PRINT.....	51
SPLIT.....	53
VALIDATE.....	54
Flags y Filtros.....	55
TUI (Text User Interface).....	57
Navegación.....	58
Búsqueda y Filtro.....	60
Filtrado de presentación.....	64
Marcar registros.....	65
Borrar registros.....	67
Recuperación de registros borrados.....	68
Modificación de registros.....	68
Reemplazar valores.....	72
Función de auto guardado.....	74
Guardado manual.....	74
Exportar registros.....	75
Importación de registros.....	75
Mark slots.....	76



# Introducción

Es una herramienta creada para el manejo de archivos de intercambio de la marca **MasterCard**

Nació de la necesidad de facilitar el manejo de dichos archivos, que son difíciles de visualizar y manipular debido a su formato.

Por el momento maneja en la mayoría de sus funciones, archivos **IPM** de **MasterCard** (acciones limitadas sobre archivos **MPE**)

## Como utilizar este manual

Debido a la complejidad de la herramienta, y al uso de conceptos que no necesariamente son de uso frecuente, se ha decidido separar este documento en tres grandes áreas. La primera, con ejemplos de uso práctico y como utilizar la herramienta para solucionarlos. La segunda, es una explicación detallada de cada uno de los comandos, pensada tanto para el uso de un operador como en la automatización de procesos.

Por ultimo, en una tercera parte, se describe y se muestra la utilización de la opción grafica (a nivel de textos) de la misma, que si bien incluye la mayoría de las funcionalidades de los comandos, requiere del uso de combinaciones de teclas para poder interactuar con ella.

Se sugiere repasar la sección [Casos de uso](#) tanto para entender su aplicación en situaciones típicas como para referencia ante casos similares. Para profundizar mas en las características y opciones de cada comando, utilizar la sección correspondiente a los [Comandos](#)

## Características y funcionalidades

Sus características principales son:

- Multiplataforma (existe una versión para Linux y para Windows)
- No requiere de una instalación especial, se trata de un solo binario ejecutable y basta con copiarlo en un directorio que se encuentre en el **PATH**
- Es una aplicación de linea de comandos, por lo que se utiliza desde una consola o terminal. Esto permite utilizarla fácilmente en servidores remotos
- Se puede utilizar tanto para procesos automatizados, como en forma interactiva por un usuario

Entre sus funcionalidades se destacan:

- Identificar los tipos de archivo por su contenido y no por su nombre
- Convertir entre formatos (**ASCII/EBCDIC**, con/sin bloques de 1014, etc.)
- Validar los archivos ante errores de formato y algunos errores de datos
- Posibilidad de corrección de algunos errores en forma automática
- Soporta archivos lógicos en archivos físicos (trasnsmissions), permitiendo separarlos en archivos físicos y viceversa
- Permite exportar sus datos en formatos "legibles", como ser archivos de texto planos, en formato **CSV** (que pueden abrirse en Excel), o como valores en hexadecimal para poder ser importados en otro archivo
- Importar en un archivo datos previamente exportados desde otro
- Detección de registros duplicados dentro de un mismo archivo
- Fraccionamiento de archivos grandes en archivos mas chicos para facilitar su manejo
- Búsqueda de datos mediante condiciones flexibles en uno o varios archivos
- Distribución de registros de un archivo en otros, de acuerdo a criterios definidos por el usuario
- Eliminación de registros de un archivo
- Abrir los archivos en forma interactiva, donde se pueden realizar acciones tales como:
  - Visualizar el contenido de los registros.
  - Eliminar registros y/o campos de un registro
  - Editar registros
  - Realizar búsquedas y filtrado
  - Exportar los registros deseados

- Importar registros exportados desde otro archivo
- Agregar registros y/o campos

Ademas, hay otras funcionalidades en desarrollo.

## Casos de uso

A continuación mostraremos algunas situaciones a las que nos enfrentamos al manejar estos archivos, y como la herramienta nos puede ayudar a solucionarlas. Son ejemplos prácticos que permiten entender el funcionamiento y la utilidad de la herramienta en forma practica y en algunos casos utilizan mas de un comando.

Para conocer mas acerca de los diferentes comandos expuestos, podemos visitar la sección correspondiente a cada uno de los comandos utilizados aquí, y presentados mas adelante en este mismo documento en la sección [Comandos](#).

## Identificar y obtener información de los archivos

Muchas veces nos encontramos con una lista de archivos de diferentes tipos, y por la nomenclatura nos es difícil determinar el contenido. Simplemente listando el directorio no podemos saber el tipo de información que contienen, y menos aun, el formato del archivo.

A modo de ejemplo, veamos el contenido de un directorio con archivos para explicar esta situación con un ejemplo:

```
$ ls -l originals/*
-rw-rw-r-- 1 eduardo eduardo 2421783 feb  7 15:13 originals/002-211104-00000555666-04241.ipm
-rw-rw-r-- 1 eduardo eduardo 4137828 feb  7 15:13 originals/002-211104-00000777111-04241.ipm
-rw-rw-r-- 1 eduardo eduardo 23322 abr 19 2022 originals/11000000292_01082020.txt
-rw-rw-r-- 1 eduardo eduardo 5600 abr 19 2022 originals/11000000292_01212020d
-rw-rw-r-- 1 eduardo eduardo 435066 abr 19 2022 originals/11000000292_12262019.txt
-rw-rw-r-- 1 eduardo eduardo 2358 abr 19 2022 originals/CLEARING_FILE_07-06-2018-050002.blk_LIF_28078.ipm
-rw-rw-r-- 1 eduardo eduardo 4056 set 26 2018 originals/CLEARING_FILE_16-11-2017-124522.blk
-rw-rw-r-- 1 eduardo eduardo 30420 set 26 2018 originals/CLEARING_FILE_24-11-2017-014549.blk
-rw-rw-r-- 1 eduardo eduardo 870012 set 26 2018 originals/CLEARING_FILE_26-09-2018-103539.blk
-rw-rw-r-- 1 eduardo eduardo 5616 abr 19 2022 originals/FeeCollection_EBCDIC_RDW_1
-rw-rw-r-- 1 eduardo eduardo 6559425 nov  4 2021 originals/R1110787730851.IPM
-rw-rw-r-- 1 eduardo eduardo 23278 abr 19 2022 originals/R1110787732851.IPM
-rw-rw-r-- 1 eduardo eduardo 21254 abr 19 2022 originals/R119_27-01-2020-050643.ipm
-rw-rw-r-- 1 eduardo eduardo 20526 abr 19 2022 originals/R119_27-01-2020-Test.ipm
-rw-rw-r-- 1 eduardo eduardo 12146 abr 19 2022 originals/R119_29-01-2020-045758.ipm
-rw-rw-r-- 1 eduardo eduardo 11218 abr 19 2022 originals/Report_900000_20200207.txt
-rw-rw-r-- 1 eduardo eduardo 3012 abr 19 2022 originals/SETTLEMENT_PRM_20201120172857.out
-rw-rw-r-- 1 eduardo eduardo 3466866 abr 19 2022 originals/T067_200106
-rw-rw-r-- 1 eduardo eduardo 61786 abr 19 2022 originals/T1400787732901
-rw-rw-r-- 1 eduardo eduardo 8119617 may  27 2020 originals/T4580798714305
-rw-rw-r-- 1 eduardo eduardo 2529 abr 19 2022 originals/Test27012020.IPM
-rw-rw-r-- 1 eduardo eduardo 34476 nov  4 2021 originals/YTF.AR.T120.C.E0084568.D211012.T210547.A002.ipm
$ █
```

Podemos ver una serie de archivos, algunos nombres nos pueden resultar familiares, pero las extensiones son variadas y no todos cumplen con una nomenclatura en particular. Ademas, tampoco sabemos la codificación utilizada (si están en **ASCII** o en **EBCDIC**), si son archivos con registros de longitud fija, variable, o delimitados, o incluso si están en bloques o no (**MasterCard** suele enviar los archivos en bloques).

Veamos ahora como utilizaría la herramienta para determinar esto. Cuando la invocamos sin especificar ningún comando, se asume que estamos usando el comando [IDENTIFY](#)

```

$ cardak originals/
=====
Card Army Knife version 0.7.2 built on 2023-07-20T22:54:57Z main 1565cba
=====
002-211104-00000555666-04241.ipm      2,421,783 bytes EBCDIC   RDW      NORMAL    MC IPM file 2021-11-04
002-211104-00000777111-04241.ipm      4,137,828 bytes EBCDIC   RDW      NORMAL    MC IPM file 2021-11-04
11000000292_01082020.txt              23,322 bytes EBCDIC   RDW      BLOCK 1014 AMEX Interchange
11000000292_01212020d                5,600 bytes EBCDIC   FRL (1400)  NORMAL    AMEX Interchange
11000000292_12262019.txt              435,006 bytes EBCDIC   RDW      BLOCK 1014 AMEX Interchange
CLEARING_FILE_07-06-2018-050002.blk_LIF_28078.ipm 2,358 bytes EBCDIC   RDW      NORMAL    MC IPM file 2018-03-22
CLEARING_FILE_16-11-2017-124522.blk        4,056 bytes EBCDIC   RDW      BLOCK 1014 MC IPM file 2017-11-16
CLEARING_FILE_24-11-2017-014549.blk        30,420 bytes ASCII    RDW      BLOCK 1014 MC IPM file 2017-11-24
CLEARING_FILE_26-09-2018-103539.blk        870,012 bytes ASCII    RDW      BLOCK 1014 MC IPM file 2018-09-26
FeeCollection_EBCDIC_RDW_1                 5,616 bytes EBCDIC   RDW      NORMAL    AMEX Interchange
R1110787730851.IPM                      6,559,425 bytes EBCDIC   RDW      NORMAL    MC IPM file 2021-11-04
R1110787732851.IPM                      23,278 bytes EBCDIC   RDW      NORMAL    MC IPM file 2020-11-23
R119_27-01-2020-050643.ipm               21,254 bytes EBCDIC   RDW      NORMAL    MC IPM file 2020-01-27
R119_27-01-2020-Test.ipm                20,526 bytes EBCDIC   RDW      NORMAL    MC IPM file 2020-01-27
R119_29-01-2020-045758.ipm               12,146 bytes EBCDIC   RDW      NORMAL    MC IPM file 2020-01-29
Report_900000_20200207.txt              11,218 bytes ASCII    DEL (UNIX)  NORMAL    AMEX Report Demographics
SETTLEMENT_PRM_20201120172857.out       3,012 bytes ASCII    DEL (UNIX)  NORMAL
T067_200106                                3,466,866 bytes EBCDIC   RDW      BLOCK 1014 MC MPE UPDATE 2020-01-06
T1400787732901                            61,706 bytes ASCII    DEL (UNIX)  NORMAL
T4580798714305                            8,119,617 bytes ASCII    DEL (UNIX)  NORMAL    FIT file
Test27012020.IPM                          2,529 bytes EBCDIC   RDW      NORMAL    MC IPM file 2018-03-22
YTF.AR.T120.C.E0084568.D211012.T210547.A002.ipm 34,476 bytes EBCDIC   RDW      BLOCK 1014 MC IPM file 2021-10-13

22 files using 25 MB

Job ended
Elapsed time : 13.891067ms
$ █

```

La información que vemos es la siguiente: primero, el nombre del archivo, luego su tamaño en bytes, la codificación utilizada (**ASCII** o **EBCDIC**), el tipo de registros (**RDW** – Registros de largo variable, **FRL** – Registros de longitud fija y su largo, **DEL** – Delimitados, y si se usan los delimitadores de Unix o de Windows), y si el archivo esta en bloques o no.

Ademas se muestra, en caso de ser posible, de que tipo de archivo se trata. Vemos que en su mayoría son archivos **IPM** de **MasterCard**, pero tenemos también archivos de **MPE**, de **AMEX** e incluso de **VISA**

Para los archivos **IPM** y **MPE** de **MasterCard**, se nos muestra la fecha que viene indicada en el Header del archivo.

Si bien en estos momentos la herramienta esta desarrollada para trabajar principalmente con archivos de **IPM** de **MasterCard**, hay algunas operaciones que funcionan con otros tipos de archivos, como por ejemplo lo que veremos mas adelante que es el cambio de formato.

A continuación, y a efectos de mostrar ejemplos concretos, vamos a trabajar con los siguientes archivos:

```

$ ls -Agoh b* f*
-rw-rw-r-- 1 32M jul 12 18:32 big_logic
-rw-rw-r-- 1 32M jul 13 14:24 big_one
-rw-rw-r-- 1 21K mar 20 11:18 file10
-rw-rw-r-- 1 34K jul 10 18:07 file14
-rw-rw-r-- 1 2,4K jul 4 14:21 file5
-rw-r--r-- 1 850K jul 15 01:16 file8
-rw-r--r-- 1 6,3M mar 20 10:05 file9
$ █

```

Veamos de que se tratan estos archivos:

```

$ cardak *
=====
Card Army Knife version 0.7.2 built on 2023-07-20T22:54:57Z main 1565cba
=====
big_logic 32,797,125 bytes EBCDIC   RDW      NORMAL    MC IPM file 2021-11-04
big_one   32,796,381 bytes EBCDIC   RDW      NORMAL    MC IPM file 2021-11-04
file10     21,254 bytes EBCDIC   RDW      NORMAL    MC IPM file 2020-01-27
file14     34,316 bytes EBCDIC   RDW      NORMAL    MC IPM file 2021-10-13
file5      2,358 bytes EBCDIC   RDW      NORMAL    MC IPM file 2018-03-22
file8      870,012 bytes ASCII    RDW      BLOCK 1014 MC IPM file 2018-09-26
file9      6,559,425 bytes EBCDIC   RDW      NORMAL    MC IPM file 2021-11-04

7 files using 70 MB

Job ended
Elapsed time : 3.535968ms
$ █

```

Son todos archivos **IPM** de **MasterCard**, y hay archivos en **ASCII**, en **EBCDIC**, con y sin bloques. Todos los archivos **IPM** contienen registros de longitud variable (o **RDW**)

Esta vista es compacta, con los datos de cada archivo en una sola linea. Podemos tener una vista diferente si especificamos solo un archivo, como por ejemplo *file8*, especificando el nombre del archivo como parámetro, de esta forma:

```
$ cardak file8
=====
Card Army Knife version 0.7.2 built on 2023-07-20T22:54:57Z main 1565cba
=====

File Name      : file8
File Size     : 870,012
File Encoding : ASCII
Record Format : RDW (Variable Record Length)
File Format   : BLOCK 1014 ( 858 blocks of 1014 bytes )
File Usage    : MC IPM file
                2018-09-26

Job ended
Elapsed time  : 2.893387ms
$
```

Esta información se obtiene a partir de una muestra de los primeros 10Kb del archivo (o el archivo completo si su tamaño es menor que eso), por lo que el proceso es sumamente rápido.

Si quisieramos tener mas información, podemos especificar el flag --analyze (-a), que leerá el contenido total de los archivos y nos podrá brindar otros datos. Por ejemplo:

```
$ cardak * --analyze
=====
Card Army Knife version 0.7.2 built on 2023-07-20T22:54:57Z main 1565cba
=====

IPM FILES -----
big_logic 32,797,125 bytes EBCDIC   RDW      NORMAL   MC IPM file 2021-11-04
          Records: 54,400 Logical files: 5 (10880,10880,10880,10880,10880)
big_one   32,796,381 bytes EBCDIC   RDW      NORMAL   MC IPM file 2021-11-04
          Records: 54,392 Logical files: 1 (54392)
file10    21,254 bytes EBCDIC   RDW      NORMAL   MC IPM file 2020-01-27
          * Records: 43 Logical files: 1 (43)
file14    34,316 bytes EBCDIC   RDW      NORMAL   MC IPM file 2021-10-13
          Records: 61 Logical files: 1 (61)
file5     2,358 bytes EBCDIC   RDW      NORMAL   MC IPM file 2018-03-22
          Records: 6 Logical files: 1 (6)
file8     870,012 bytes ASCII    RDW      BLOCK 1014 MC IPM file 2018-09-26
          * Records: 2,049 Logical files: 1 (2049)
file9     6,559,425 bytes EBCDIC   RDW      NORMAL   MC IPM file 2021-11-04
          * Records: 10,880 Logical files: 1 (10880)

7 files using 70 MB

Job ended
Elapsed time  : 17.881571459s
$
```

En este caso, se agrega una linea donde podemos ver la cantidad de registros que contiene cada archivo, la cantidad de archivos lógicos, y la cantidad de registros por cada archivo lógico que contenga. En el caso del primer archivo, vemos que contiene 5 archivos lógicos, cada uno con 10880 registros (todos tienen la misma cantidad porque el archivo fue creado con fines demostrativos, uniendo 5 veces el mismo archivo)

Vemos que algunos contiene la palabra “Records” en color rojo y con un asterisco delante. Esto significa que se encontraron algunos errores al procesar el archivo.

Para investigar un poco mas, indicaremos como parámetro solamente el nombre del archivo a verificar, lo que nos mostrara información mas detallada. Veamos que problemas contiene el archivo *file10*

```

$ cardak file10 -a
=====
Card Army Knife version 0.7.2 built on 2023-07-20T22:54:57Z main 1565cba
=====

IPM FILES ----

File Name      : file10
File Size     : 21,254
File Encoding : EBCDIC
Record Format : RDW (Variable Record Length)
File Format   : NORMAL
File Usage    : MC IPM file
                2020-01-27

FILE HAS ERRORS:
    LF: 1 PRN:      43 DE71 in Trailer is not greater than the last one, current: [00000043], previous: [00000209]
    LF: 1 PRN:      43 PDS 0301 (checksum) in Trailer is not correct, value: [0000000000000000], expected: [00000000043
79378]

Total records : 43
Logical files : 1

Records by MTI      : 33 - 1240 Presentment
                           8 - 1740 Fee Collection
                           2 - 1644 Administrative

Records by Tran type: 33 - First Presentment
                           8 - Fee Collection (Customer-generated)
                           1 - File Trailer
                           1 - File Header

Job ended
Elapsed time : 20.892598ms
$
```

Vemos que el archivo contiene errores en el archivo lógico numero 1 (LF: 1), y en particular en el registro físico numero 43 (PRN: 43) que es el trailer del archivo. Se nos indica que el campo **DE71** (indicador de numero de linea dentro del archivo) no cumple con el requisito que sea mayor al anterior. También se nos indica que el campo PDS0301, que contiene el checksum los los importes, no es correcto.

Ademas se nos brinda información estadística, por ejemplo, la cantidad de registros por cada **MTI**, o la cantidad de registros por cada tipo de transacción.

Si ademas agregáramos el flag --verbose (-v), se nos mostraría la cantidad de registros encontrados por

```

$ cardak file10 -a -v
=====
Card Army Knife version 0.7.2 built on 2023-07-20T22:54:57Z main 1565cba
=====

IPM FILES ----

File Name      : file10
File Size     : 21,254
File Encoding : EBCDIC
Record Format : RDW (Variable Record Length)
File Format   : NORMAL
File Usage    : MC IPM file
                2020-01-27

FILE HAS ERRORS:
    LF: 1 PRN:      43 DE71 in Trailer is not greater than the last one, current: [00000043], previous: [00000209]
    LF: 1 PRN:      43 PDS 0301 (checksum) in Trailer is not correct, value: [0000000000000000], expected: [00000000043
79378]

Total records : 43
Logical files : 1

Records by MTI      : 2 - 1644 Administrative
                           33 - 1240 Presentment
                           8 - 1740 Fee Collection

Records by Tran type: 1 - File Header
                           33 - First Presentment
                           8 - Fee Collection (Customer-generated)
                           1 - File Trailer

Records by MCC       : 11 - 5251 HARDWARE STORES
                           6 - 5541 SERVICE STATIONS WITH OR WITHOUT ANCILLARY SERVICE
                           9 - 5732 ELECTRONIC SALES
                           7 - 5812 EATING PLACES, RESTAURANTS

Job ended
Elapsed time : 31.550791ms
$
```

cada valor de **MCC**

## Validar los archivos en busca de errores

Esto es útil para ser utilizado en un proceso automatizado, donde es ventajoso tomar acciones antes de procesar un archivo si detectamos que este contiene errores. Se pueden validar luego de ser generados y antes de ser transferidos a fin de evitar rechazos de la marca ante errores detectables, ya que después de determinada cantidad de rechazos, la marca puede aplicar multas económicas.

La validación puede hacerse de un archivo en particular o de una lista de archivos. La herramienta devuelve un código de error al sistema operativo en caso de detectar algún error, o un valor cero en caso que todos los archivos pasen la validación, lo que permite automatizar fácilmente el chequeo y posterior acción.

Veamos como funciona. Primero, validaremos los archivos cuyo nombre son file\*

```
$ cardak validate file*
=====
Card Army Knife version 0.7.2 built on 2023-07-20T22:54:57Z  main 1565cba
=====
FAIL file10      1 rec  with error  and is Fixable
PASS file14
PASS file5
FAIL file8    2,050 recs with errors and cannot be fixed
FAIL file9      1 rec  with error  and is Fixable

Job ended
Elapsed time : 1.700836565s
$ echo $?
1
$ █
```

Vemos que hay dos archivos que pasan las validaciones, y tres que fallan. Vemos la cantidad de registros que contienen errores, y un mensaje que nos indica si los tipos de errores encontrados pueden ser corregidos automáticamente o no. Vemos que el valor devuelto al sistema operativo es 1, lo que indica que se encontraron errores.

Una salida alternativa es agregar el flag --silent (-z) para analizar la salida y tomar acciones en forma automática.

```
$ cardak validate file* --silent
PASS:file14:::
PASS:file5:::
FAIL:file10:1:F-Fixable
FAIL:file8:2050:X-Cannot fix
FAIL:file9:1:F-Fixable
$ █
```

El flag --silent omite toda salida por consola, excepto los resultados. Como vemos, la salida es una linea por archivo, que contiene 4 secciones separadas por el carácter ":". La primera indica si la validación ha fallado o no, la segunda es el nombre del archivo, la tercera es la cantidad de registros con errores (en caso de haberlos), y la cuarta indica si los errores encontrados pueden ser corregidos automáticamente o no.

Continuando con su utilidad para las automatizaciones, podemos especificar si queremos obtener solamente la lista de los archivos sin errores o aquellos que contienen errores. Por ejemplo:

```
$ cardak validate file* --pass
file14
file5
$
$ cardak validate file* --fail
file10
file8
file9
$ █
```

De esta forma, podemos analizar la salida y mover solamente aquellos archivos que sean correctos o que contengan errores.

Volviendo a la validación, podemos obtener información mas detallada de los errores encontrados, agregando el flag --verbose (-v) de esta forma:

```

$ cardak validate file* --verbose
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====
FAIL file10      1 rec with error and is Fixable
PRN: 43 Record number is not greater than the previous one, current: [43], previous: [209]
      PDS0301 (Checksum Amount) in Trailer is not correct, value: [0000000000000000], expected: [0000000004379378]

PASS file14
PASS file5
FAIL file8      2,050 recs with errors and cannot be fixed
Offset: 236261 (39AE5) Block: 233 Block filler value is 0x00 instead of 0x40
Offset: 236262 (39AE6) Block: 233 Block filler value is 0x00 instead of 0x40
PRN: 2 Missing mandatory field DE094 for MTI 1240 and DE24 200
PRN: 3 Missing mandatory field DE094 for MTI 1240 and DE24 200
PRN: 4 Missing mandatory field DE094 for MTI 1240 and DE24 200
PRN: 5 Missing mandatory field DE094 for MTI 1240 and DE24 200
PRN: 6 Missing mandatory field DE094 for MTI 1240 and DE24 200
PRN: 7 Missing mandatory field DE094 for MTI 1240 and DE24 200
PRN: 8 Missing mandatory field DE094 for MTI 1240 and DE24 200
PRN: 9 Missing mandatory field DE094 for MTI 1240 and DE24 200
PRN: 10 Missing mandatory field DE094 for MTI 1240 and DE24 200
PRN: 11 Missing mandatory field DE094 for MTI 1240 and DE24 200
...
Showing 10 errors out of 2,048
FAIL file9      1 rec with error and is Fixable
PRN: 10,880 PDS0301 (Checksum Amount) in Trailer is not correct, value: [0000000000000000], expected: [0000000700378029]

Job ended
Elapsed time : 1.696968582s
$ 

```

Vemos que el archivo file10 contiene el error en el trailer (que puede ser corregido), y que file9 contiene un error también en el trailer ya que el checksum no contiene el valor esperado. Este error también se puede corregir automáticamente.

Sin embargo, vemos que file8 contiene varios errores de diferente tipo. Los dos primeros errores son de formato del archivo, y nos esta indicando que en las posiciones 236261 y 236262, se encontraron bytes con el valor **0x00** cuando lo correcto seria que contuvieran el valor **0x40**. Estos dos errores pueden ser corregidos, pero vemos que ademas hay otros errores (se muestran solo los primeros 10 errores), 2048 en total, donde vemos que para esa combinación de **MTI** (1240) y Function Code **DE24** (200), el campo **DE094** (Transaction Originator Institution ID Code) es mandatorio, pero no esta presente. Este tipo de errores no se pueden corregir automáticamente ya que no podemos saber que valor deberíamos poner en este campo sin un análisis de estos casos en particular.

## Corregir errores automáticamente

Algunos errores pueden ser corregidos automáticamente. Por ejemplo, los números de registro dentro del archivo (deben tener valores ascendentes y cada header de un archivo lógico debe comenzar con el valor 1), o el checksum del trailer, o que los caracteres de relleno de los bloques (el estándar especifica el valor **0x40** pero puede llegar a venir otro valor, como por ejemplo **0x00**), son todos errores que pueden ser corregidos recalcando los valores.

Otro tipo de errores, como ser campos que faltan, o errores físicos del archivo (por ejemplo un largo que este incorrecto, o caracteres no autorizados en campos), seguramente no puedan ser corregidos en forma automática, pero la herramienta ofrece otras formas de corregirlos en forma manual (Ver la sección [TUI - Text User Interface](#) )

Habíamos visto previamente que el archivo file10 contenía dos errores y que era posible corregirlos automáticamente. Para eso utilizaremos el comando **FIX**

```
$ cardak validate file10
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====
FAIL file10      1 rec with error and is Fixable

Job ended
Elapsed time : 13.594638ms
$ cardak fix file10
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====
Source file   : file10      as EBCDIC, RDW, NORMAL
Destination file : file10-FIX.ipm as EBCDIC, RDW, EBCDIC
Loaded 20,530 and 43 records from file10

The following errors have been fixed:

Generated file: file10-FIX.ipm
Bytes read    : 20,530
Records Read : 43
Bytes written : 20,526

Job ended
Elapsed time : 23.018388ms
$ cardak validate file10*
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====
FAIL file10      1 rec with error and is Fixable
PASS file10-FIX.ipm

Job ended
Elapsed time : 39.480493ms
$ █
```

El archivo corregido mantiene el nombre del archivo original, pero le agrega “-FIX.ipm” al final. Se agrega esa extensión porque el comando **FIX** solo está disponible para archivos de tipo **IPM**

Podemos aplicar este comando a un archivo a la vez, o a una lista de archivos, pero solo se corregirán los archivos detectados como **MasterCard IPM**

## Cambiar el formato de los archivos

Algunas veces, y especialmente durante los periodos de certificación, el formato de los archivos generados no coinciden con el formato esperado por la marca y viceversa. En algunos sistemas es sencillo (aunque inconveniente) cambiar la configuración para poder procesarlos correctamente.

En estos casos, la herramienta nos permite, fácilmente, cambiar el formato de un archivo a otro deseado.

Por ejemplo, veamos el formato del archivo file14:

```
$ cardak file14
=====
Card Army Knife version 0.7.2 built on 2023-07-20T22:54:57Z main 1565cba
=====

File Name   : file14
File Size   : 34,316
File Encoding : EBCDIC
Record Format : RDW (Variable Record Length)
File Format  : NORMAL
File Usage    : MC IPM file
                2021-10-13

Job ended
Elapsed time : 4.071181ms
$
```

Es un archivo codificado en **EBCDIC**, y no esta en bloques de 1014. Supongamos que queremos convertirlo en un archivo codificado en **ASCII** y en bloques de 1014. Esto lo podemos hacer fácilmente de esta forma:

```
$ cardak convert ARB file14
=====
Card Army Knife version 0.7.2 built on 2023-07-20T22:54:57Z main 1565cba
=====

Source file   : file14      as EBCDIC, RDW, NORMAL
Destination file : file14.ARBCVT as ASCII, RDW, BLOCK 1014

Generated file: file14.ARBCVT
Bytes read   : 34,316
Records Read : 61
Bytes written : 34,476

Job ended
Elapsed time : 29.76651ms
$
```

Utilizamos el comando CONVERT, especificando el formato (son tres letras que identifican codificación, tipo de registro y formato de archivo) – para mas detalles ver la descripción del comando [CONVERT](#) en la sección correspondiente.

El archivo generado convertido queda con el mismo nombre del original, pero agregándole al final las tres letras utilizadas para el formato de conversión, y la extensión .cvt

Este comando funciona para varios tipos de archivos, no solo los **IPM** de **MasterCard**. Podemos por ejemplo, convertir los archivos de **MPE** y algunos otros tipos de archivo.

Este comando se puede utilizar con un archivo a la vez, o una lista de archivos a los cuales se les aplicara la conversión solicitada de una sola vez.

# Búsqueda de datos en archivos IPM

En Linux (y Unix en general), solemos disponer del comando grep para realizar búsquedas en archivos. Esto funciona muy bien cuando los archivos que contienen los datos son de texto plano, lo cual no es el caso para archivos **IPM**.

Por ejemplo, sabemos que en alguno de estos archivos hay compras en un local llamado “Carpinteria”, pero no sabemos en cual de ellos están esas transacciones.

```
$ grep 'Carpinteria' file*
```

Primero, intentemos utilizar el comando grep del sistema:

Como podemos observar, no obtenemos ningún resultado, y grep nos devuelve el valor 1 indicando que no pudo efectuar búsquedas en archivos “binarios”

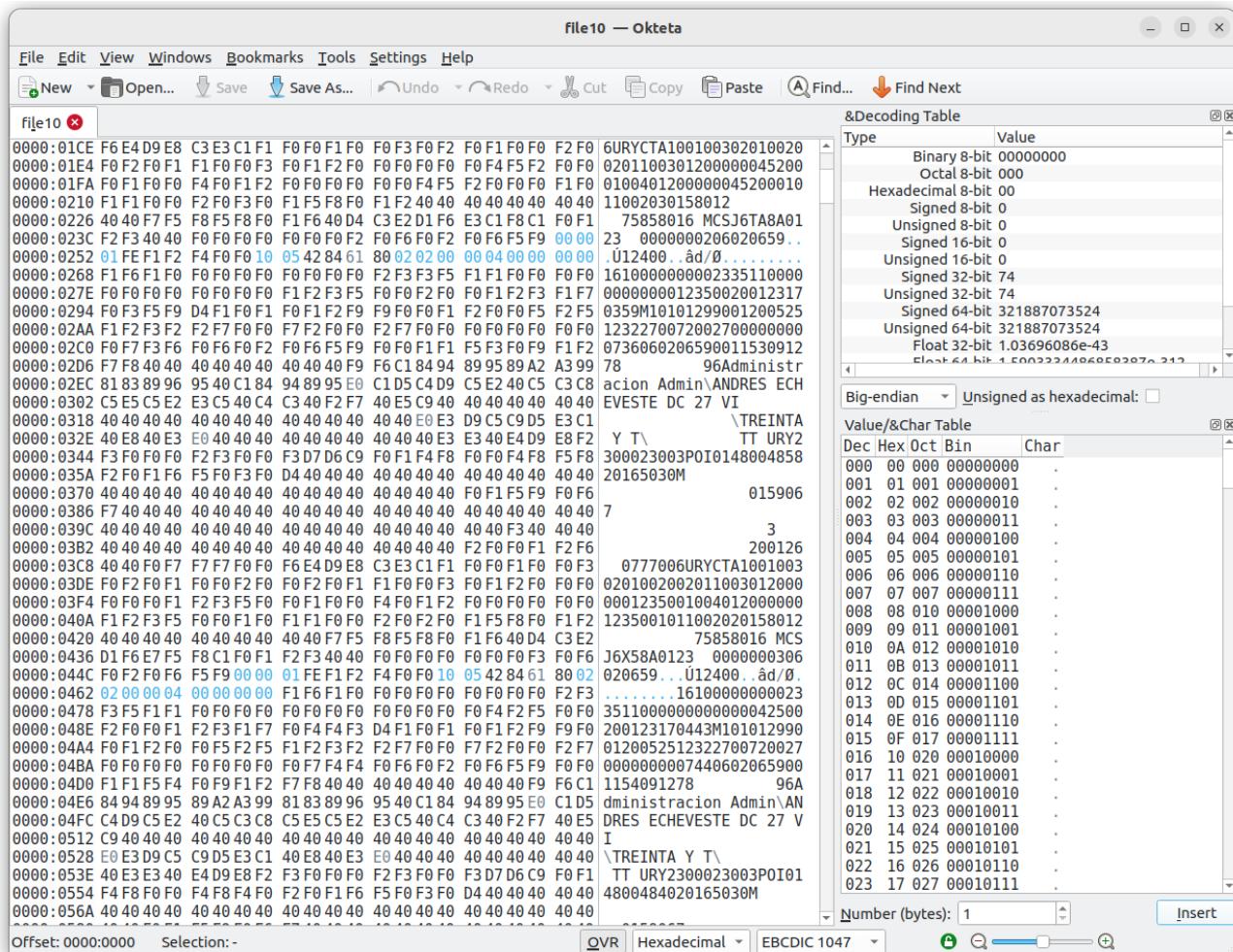
Podemos, por supuesto, abrir el archivo con un editor hexadecimal y realizar ahí la búsqueda. Probemos ver

```
$ hexdump -C file10 | head
00000000  00 00 00 4a f1 f6 f4 f4  80 00 01 00 00 01 00 00 |...J....|
00000010  02 00 00 00 00 00 00 00  f6 f9 f7 f0 f4 f0 f0 f1 |....|
00000020  f0 f5 f0 f2 f5 f0 f2 f2  f2 f0 f0 f1 f2 f7 f0 f0 |....|
00000030  f0 f0 f0 f2 f0 f6 f5 f9  f1 f7 f0 f6 f1 f0 f1 |....|
00000040  f2 f2 f0 f0 f1 e3 f0 f0  f0 f0 f0 f0 f0 f1 00 00 |....|
00000050  01 fe f1 f2 f4 f0 f0 10  05 42 84 61 80 02 02 00 |.....B.a..|
00000060  00 04 00 00 00 00 f1 f6 f1 f0 f0 f0 f0 f0 f0 f0 |....|
00000070  f0 f0 f2 f3 f3 f5 f1 f1 f0 f0 f0 f0 f0 f0 f0 f0 |....|
00000080  f0 f0 f0 f4 f5 f2 f0 f0 f0 f2 f0 f0 f1 f2 f3 f0 f0 |....|
00000090  f1 f7 f0 f3 f0 f9 d4 f1 f0 f1 f0 f1 f2 f9 f9 f0 |....|
$ cardak file10 --compact
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z  main 1565cba
=====
file10      21,254 bytes EBCDIC      RDW      NORMAL      MC IPM file 2020-01-27
Job ended
Elapsed time : 2.840826ms
$
```

si podemos utilizar hexdump desde la linea de comandos:

No parece contener algo que sea legible, pero cual es el problema? Vemos que la codificación es **EBCDIC**, por lo que si bien podemos ver los valores de los bytes, no contienen una representación sencilla en **ASCII**. Claro que podríamos convertirlo primero a **ASCII** y luego usar hexdump, pero cuando tenemos muchos archivos, eso no es una opción muy practica.

Intentemos utilizar otra herramienta, en este caso una herramienta grafica (ya que en Windows también existen editores en hexadecimal gráficos):



Ahora, al cambiar la codificación a **EBCDIC**, comenzamos a ver valores que pueden llegar a identificarse. Sin embargo, esta opción requiere ir abriendo uno por uno los archivos para efectuar las búsquedas, y cuando necesitamos buscar algo mas específico, que no sea solamente una cadena de caracteres, la situación tampoco mejora mucho.

Veamos entonces como podemos utilizar la herramienta para realizar búsquedas en archivos **IPM**.

Como dijimos anteriormente, comenzamos buscando los archivos que contienen transacciones realizadas en un comercio que contiene la palabra “carpinteria” en su nombre. Sabemos que el nombre del comercio viene en el campo **DE043**, pero vamos a comenzar haciendo una búsqueda global

Para ello, invocamos la herramienta con el comando GREP, indicando primero el criterio de búsqueda y luego el o los archivos donde efectuar dicha búsqueda.

Respecto al criterio, esto es muy flexible, y podemos hacer búsquedas globales (en cualquier parte del registro), en campos específicos (incluidos subcampos), o incluso combinaciones lógicas en las condiciones como **OR** y **AND** (para mas información del uso de las condiciones, referirse a la sección correspondiente al comando [GREP](#) )

Por el momento busquemos el texto “carpinteria”. Observemos que la busqueda no es case-sensitive

```

$ k grep 'carpinteria' file*
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z  main 1565cba
=====

-----
file10
TOTAL: 0 matches in file10
-----
file14
TOTAL: 0 matches in file14
-----
file5
TOTAL: 0 matches in file5
-----
file8
TOTAL: 0 matches in file8
-----
file9
Record 763: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\      MO URY]
Record 1083: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\      MO URY]
Record 2339: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\      MO URY]
Record 2681: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\      MO URY]
Record 5805: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\      MO URY]
Record 10638: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\      MO URY]

TOTAL: 6 matches in file9

-----
Bytes read (data) : 7,486,377
Records Read     : 13,039
-----
Job ended
Elapsed time : 921.631435ms
$ █

```

Vemos que en el archivo file9 hay 6 registros que contienen ese texto en el campo **DE043**. Vemos ademas los números de registro donde esta presente, y vemos tambien que los 6 registros corresponden a primeras presentaciones.

A partir de aqui podemos centrarnos en mejorar la busqueda, pero ahora solamente sobre el archivo file9 que es el que contiene nuestra posible transaccion buscada.

Por ejemplo, un siguiente paso podria ser que, ademas de mostrarnos los registros donde esta el texto que buscamos, nos muestre, ademas, el contenido de los campos **DE004** (Importe de la transaccion) y **DE012** (Fecha y hora). Para ello podemos agregar el flag -F donde indicamos los campos que queremos mostrar de los registros encontrados, aunque ellos no contengan el valor buscado.

Para indicar los campos, escribimos la lista de campos separados por comas, y podemos indicar si se trata de un Data Element (**DE**), de un Private Data Subelement (**PDS**), o incluso si queremos especificar un Subfield de cualquiera de estos. Por ejemplo, la fecha y hora que vienen en el DE012, contiene dos subfields que son, la fecha en el subfield 1, y la hora en el subfield 2. Entonces, en lugar de especificar que queremos el **DE12**, podemos indicar que queremos el **DE12 SF 1** y el **DE12 SF 2** por separado.

Para mas informacion de como especificar los campos y tambien de como utilizar listas predefinidas, por favor referirse a la seccion [FILTER](#)

Veamos la salida de este comando:

```

$ k grep 'carpinteria' file9 -F 'D4,D12s1,D12s2'
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====

file9
Record 763: (MTI 1240 - First Presentment)
    DE004 : [000000059017]
    DE012S01: [211029]
    DE012S02: [171126]
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]
Record 1083: (MTI 1240 - First Presentment)
    DE004 : [000000140656]
    DE012S01: [211029]
    DE012S02: [175500]
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]
Record 2339: (MTI 1240 - First Presentment)
    DE004 : [000000197000]
    DE012S01: [211030]
    DE012S02: [171923]
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]
Record 2681: (MTI 1240 - First Presentment)
    DE004 : [000000233115]
    DE012S01: [211030]
    DE012S02: [110446]
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]
Record 5805: (MTI 1240 - First Presentment)
    DE004 : [000000296066]
    DE012S01: [211031]
    DE012S02: [113844]
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]
Record 10638: (MTI 1240 - First Presentment)
    DE004 : [000000204591]
    DE012S01: [211103]
    DE012S02: [105208]
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]

TOTAL:      6 matches in file9

-----
Bytes read (data) : 6,559,425
Records Read     : 10,880

-----
Job ended
Elapsed time   : 812.682041ms
$ 

```

Ahora podemos ver los importes y los horarios en que esas transacciones fueron hechas.

Ahora bien, suponiendo que hay muchos registros que contienen el valor que estamos buscando, pero conocemos algún otro dato de la transacción, podemos ir mejorando el criterio de búsqueda. Por ejemplo, sabemos que la transacción fue hecha en el comercio llamado “carpinteria”, y que el valor de la misma fue de 1970.00

Podemos entonces, buscar ese registro específico de la siguiente forma:

```

$ k grep 'carpinteria,D4:197000' file9
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====

file9
Record 2339: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]
    DE004 : [000000197000]

TOTAL:      1 matches in file9

-----
Bytes read (data) : 6,559,425
Records Read     : 10,880

-----
Job ended
Elapsed time   : 789.731388ms
$ 

```

Ahora obtenemos como resultado un único registro, el 2339, que contiene la transacción que estamos buscando.

En este punto seguramente queramos ver todos los datos de la transacción en cuestión. Si bien podríamos especificar la lista de campos a desplegar, existen otras formas mas eficientes.

Para esto, podemos utilizar el comando PRINT. Este comando permite mostrar, en forma amigable, los contenidos de un archivo o de los registros especificados. Podemos redirigir esa salida a un archivo de texto para poder visualizarla en un editor de texto.

Vamos a ver el contenido del registro encontrado con el comando GREP utilizado anteriormente. Vimos que el registro era el numero 2339, por lo que podemos indicar que queremos ver solamente ese registro

```
$ cardak print file9 -R 2339
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====
file9

2339: MTI 1240 - First Presentment
    DE002   DE003   DE004   DE012   DE022   DE024   DE026   DE031   DE033   DE038   DE040   DE042   DE043   DE049   DE055   DE
063   DE071   DE094
    PDS0023 PDS0148 PDS0158 PDS0159 PDS0165 PDS0208 PDS0777 PDS1001 PDS1002 PDS1003 PDS1004 PDS1011

Bytes read (data) : 6,559,425
Records Read     : 10,880
-----
Job ended
Elapsed time     : 69.054472ms
$
```

Por defecto, el comando PRINT nos muestra, para cada registro de salida, la lista de campos presentes en el registro (primero los DE y luego los PDS presentes).

Si queremos ver los valores de cada campo, agregaremos el flag --detailed (-d)

```
$ cardak print file9 -R 2339 -d
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====
file9

2339: MTI 1240 - First Presentment
    DE002 Primary Account Number (PAN)          : [1000000141003107]
    DE003 Processing Code                      : [000000]
    DE004 Amount, Transaction                 : [000000197000]
    DE012 Date and Time, Local Transaction   : [211030171923]
    DE022 Point of Service Data Code         : [M10101C99001]
    DE024 Function Code                      : [200]
    DE026 Card Acceptor Business Code (MCC)  : [5712]
    DE031 Acquirer Reference Data           : [22700721308000001891412]
    DE033 Forwarding Institution ID Code    : [020659]
    DE038 Approval Code                     : [151049]
    DE040 Service Code                      : [201]
    DE042 Card Acceptor ID Code            : [158084      ]
    DE043 Card Acceptor Name/Location       : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\]      MO URY]
    DE049 Currency Code, Transaction        : [858]
    DE055 Integrated Circuit Card (ICC) System Related Data: [♦♦\0Hn      ***Rm=&*****      *****Eh^*****b*****d*]
*^***]
    DE063 Transaction Life Cycle ID        : [ MCS504W2E1030  ]
    DE071 Message Number                  : [00002339]
    DE094 Transaction Originator Institution ID Code : [020659]
    PDS0023 Terminal Type                : [POI]
    PDS0148 Currency Exponents          : [8582]
    PDS0158 Business Activity          : [      06]
    PDS0159 Settlement Data            : [      ]          3      NO
000000022020200]
    PDS0165 Settlement Indicator        : [M      ]
    PDS0208 Additional Merchant Data   : [00000268588HY000923      ]
    PDS0777 Promotion Code             : [URYCTA]
    PDS1001 Installment Plan Type     : [23]
    PDS1002 Installment Number        : [01]
    PDS1003 Installment Amount        : [000000032833]
    PDS1004 Total Transaction Amount  : [000000197000]
    PDS1011 Total Number of Installments: [06]

Bytes read (data) : 6,559,425
Records Read     : 10,880
-----
Job ended
Elapsed time     : 57.387614ms
$
```

De esta forma veremos la lista de los campos presentes en el registro, su descripción, y el valor correspondiente.

Podemos agregar la opción --subfields (-s) para ver el contenido de cada uno de los subcampos de aquellos que tienen definidos dichos subcampos

```

$ cardak print file9 -R 2339 -d -s
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====
file9
=====
2339: MTI 1240 - First Presentment
    DE002 Primary Account Number (PAN)          : [1000000141003107]
    DE003 Processing Code                      : [000000]
        01 Cardholder Transaction Type          : [00]
        02 Cardholder 'From' Account Type Code : [00]
        03 Cardholder 'To' Account Type Code   : [00]
    DE004 Amount, Transaction                 : [000000197000]
    DE012 Date and Time, Local Transaction    : [211030171923]
        01 Date                                : [211030]
        02 Time                                : [171923]
    DE022 Point of Service Data Code          : [M10101c99601]
        01 Terminal Data: Card Data Input Capability : [M]
        02 Terminal Data: Cardholder Authentication Capability: [1]
        03 Terminal Data: Card Capture Capability   : [0]
        04 Terminal Operating Environment         : [1]
        05 Cardholder Present Data              : [0]
        06 Card Present Data                  : [1]
        07 Card Data: Input Mode               : [C]
        08 Cardholder Authentication Method     : [9]
        09 Cardholder Authentication Entity    : [9]
        10 Card Data Output Capability        : [0]
        11 Terminal Data Output Capability     : [0]
        12 PIN Capture Capability            : [1]
    DE024 Function Code                     : [200]
    DE026 Card Acceptor Business Code (MCC) : [5712]
    DE031 Acquirer Reference Data          : [22700721308000001891412]
        01 Mixed Use; any numeric value       : [2]
        02 Acquirer's BIN                   : [270072]
        03 Julian Processing Date YDDD      : [1308]
        04 Acquirer's Sequence Number       : [00000189141]
        05 Check Digit Numeric; Luhn Formula Modulus-10 : [2]
    DE033 Forwarding Institution ID Code   : [020659]
    DE038 Approval Code                  : [151649]
    DE040 Service Code                  : [201]
    DE042 Card Acceptor ID Code         : [158084]
    DE043 Card Acceptor Name/Location    : [Handy*MH Carpinteria\RIZAL 3555\Maldonado] MO URY]
        01 Card Acceptor Name              : [Handy*MH Carpinteria]
        02 Card Acceptor Street Address    : [RIZAL 3555]
        03 Card Acceptor City             : [Maldonado]
        04 Card Acceptor Postal (ZIP) Code: [ ]
        05 Card Acceptor State, Province, or Region Code : [MO]
        06 Card Acceptor Country Code     : [URY]
    DE049 Currency Code Transaction       : [8581]

```

En caso de querer redirigir esa salida a un archivo para ser visto en un editor de texto, es conveniente agregar los flags --silent (-z) o --mono para que la salida se haga sin los caracteres que indican los colores (salida monocromática).

# Visualizar el contenido en algún programa externo

Quizás queremos ver o manipular los datos que vienen en un archivo **IPM** en algún programa externo, ya sea un editor de textos o una planilla de Excel. Esta herramienta permite, fácilmente, imprimir o exportar todos o algunos registros a un archivo que pueda ser utilizado por esos programas externos.

Comenzaremos viendo como podemos ver el contenido de algunos registros de un archivo **IPM** en un editor de texto

Para eso, utilizaremos el comando **PRINT**. Con este comando podemos mostrar en un formato legible, el contenido de todo un archivo **IPM**, o los registros que indiquemos.

Por defecto, el comando **PRINT** presenta los registros de un archivo, mostrando el numero de registro, el tipo de información que viene en ese registro, y la lista de los campos presentes en el registro.

Si queremos mostrar el contenido de los campos, debemos agregar el flag **--detailed** (-d).

Podemos seleccionar los registros que queremos ver utilizando el flag **-R**, y para mostrar solamente algunos campos, podemos utilizar el flag **-F**

```
$ k print file5 -d -R 1,3
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====
file5

1: MTI 1644 - File Header
    DE024 Function Code : [697]
    DE071 Message Number : [00000001]
    PDS0105 File ID : [0011803220000001849301101]
    PDS0122 Processing Mode : [T]

3: MTI 1442 - First Chargeback (Full)
    DE002 Primary Account Number (PAN) : [5545090000000004]
    DE003 Processing Code : [000000]
    DE004 Amount, Transaction : [000028910000]
    DE012 Date and Time, Local Transaction : [180602081039]
    DE022 Point of Service Data Code : [090010099001]
    DE024 Function Code : [450]
    DE025 Message Reason Code : [4850]
    DE026 Card Acceptor Business Code (MCC) : [5499]
    DE030 Amounts, Original : [000091000000000000000000]
    DE031 Acquirer Reference Data : [22300048158000000000307]
    DE033 Forwarding Institution ID Code : [018493]
    DE038 Approval Code : [403487]
    DE042 Card Acceptor ID Code : [00000000000004]
    DE043 Card Acceptor Name/Location : [Fogo De Ch o Brazili\Av. Moreira Guimar es
o Paulo \3560400 BRA] \S
    DE049 Currency Code, Transaction : [840]
    DE063 Transaction Life Cycle ID : [0607]
    DE071 Message Number : [00000003]
    DE094 Transaction Originator Institution ID Code : [018493]
    DE095 Card Issuer Reference Data : [0000000300]
    PDS0023 Terminal Type : [MAN]
    PDS0146 Amounts, Transaction Fee : [00190198600000000000009860000000000000]
    PDS0148 Currency Exponents : [84029862]
    PDS0149 Currency Codes, Amounts, Original : [986000]
    PDS0158 Business Activity : [MSI FD]
    PDS0159 Settlement Data : [1 F1
803220118032201]
    PDS0165 Settlement Indicator : [M MasterCard Clearing]
    PDS0177 Cross-border : [N ]
    PDS0262 Documentation Indicator : [0]

Bytes read (data) : 2,358
Records Read : 6

Job ended
Elapsed time : 4.255242ms
$
```

Para mas detalles de como utilizar estos flags, ver la sección llamada [FLAGS y FILTROS](#)

Podemos utilizar el flag **--silent** y redireccionar la salida a un archivo, para luego abrirlo con un editor de textos y visualizar los contenidos, hacer búsquedas, imprimir en papel, etc.

Otra opción es utilizar el comando **EXPORT**. Este comando, por defecto, genera un archivo **CSV** el cual puede ser abierto con el programa Excel o similar. Ese archivo contiene una primera linea con los nombres de los campos, y en las filas siguientes los valores correspondientes a cada campo.

Se pueden aplicar los mismos filtros para seleccionar los registros y los campos que deseamos queden en el archivo **CSV**

Hay un flag llamado --console que si esta presente, en lugar de generar un archivo, muestra los contenidos por consola, por lo que podemos redireccionar esa salida a cualquier archivos que deseemos.

## Manejar archivos lógicos

Puede ser que si tenemos un archivo que contiene mas de un archivo lógico (**MasterCard** llama a estos archivos “transmissions”, queramos generar archivos físicos por cada uno de esos archivos lógicos, o viceversa.

Para ello contamos con los comandos SPLIT y JOIN

El comando SPLIT toma el contenido de un archivo, y por cada archivos lógico que se encuentre en el, genera un archivo físico.

El comando JOIN hace lo contrario, toma la lista de archivos físicos que le indiquemos, y genera un nuevo

```
$ cardak join files file10 file14
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====
+      6 records to logical file 1 from file5
+     43 records to logical file 2 from file10
+     61 records to logical file 3 from file14

Generated 20230721233425_J.ipm -> 110 records using 57,200 bytes

Job ended
Elapsed time : 43.55596ms
$ cardak 20230721233425_J.ipm -a --compact
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====

IPM FILES -----
20230721233425_J.ipm      57,200 bytes EBCDIC      RDW      NORMAL      MC IPM file 2018-03-22
                           * Records:    110 Logical files: 3 (6,43,61)

Job ended
Elapsed time : 29.71457ms
$ cardak split 20230721233425_J.ipm
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====
Source file    : 20230721233425_J.ipm      as EBCDIC, RDW, NORMAL

Loaded 57,200 and 110 records from 20230721233425_J.ipm
Analyzing file...
? Please, choose an action: Fix
Fix - Fix the file and proceed

The following errors have been fixed:

   6 records (    2,358 bytes) written to 0011803220000001849301101-230721233456-1.ipm
  43 records (  20,526 bytes) written to 0022001270000002065917061-230721233456-2.ipm
  61 records (  34,316 bytes) written to 0012110130000001444402201-230721233456-3.ipm

Job ended
Elapsed time : 13.152925545s
$
```

archivo físico generando un archivo lógico por cada uno de los archivos físicos indicados.

Vemos esos comandos en acción, Primero, hacemos un JOIN de tres archivos (file5, file10 y file14), y se genera un archivo de nombre 20230721233425\_J.ipm (es la fecha y hora de generación, mas los caracteres “\_J.ipm”

Vemos que efectivamente contiene tres archivos lógicos, conteniendo 6, 43 y 61 registros cada uno

Luego sepáramos esos archivos lógicos usando el comando SPLIT. Al intentar separarlos, detecta que hay errores y nos da la opción de cancelar, ignorar o corregir dichos errores. Seleccionamos por ejemplo Fix, y genera tres nuevos archivos físicos cuyos nombres están compuestos por el campo PDS0105 del header de cada uno, seguidos de un guion y la fecha y hora de creación, mas un numero secuencial que identifica al numero de archivo lógico. También les agrega la extensión “.ipm”

## Eliminar registros de un archivo

Quizás, por alguna razón, luego de generar un archivo necesitamos quitar algún registro antes de enviarlo, ya sea porque no queremos que esa transacción salga en ese envío, o por un rechazo del archivo enviado anteriormente.

A veces es posible volver a generar el archivo, pero no todos los sistemas tienen una forma sencilla de hacerlo.

Lo primero que debemos hacer es identificar el o los registros que queremos quitar del archivo. Para esto podemos realizar una búsqueda tal como se indica en la sección [Búsqueda de datos en archivos IPM](#)

Lo importante aquí es obtener el o los números de registros físicos en el archivo correspondiente.

A modo de ejemplo, utilizaremos el mismo criterio utilizado en el caso tratado anteriormente, de búsqueda de un registro mediante el comando GREP. Vamos a eliminar del archivo file9 el registro correspondiente a la compra en el comercio “Carpinteria” y por un monto de 1970,00

```
$ cardak grep 'carpinteria,04:197000' file9 --matches --silent
file9
2339
$
```

Utilizamos aquí la versión reducida de la búsqueda, ya que sabemos que ese registro está en ese archivo y ya identificamos el registro, por lo que solo necesitamos conocer el número de registro (para ello utilizamos --matches y --silent), obteniendo el número de registro 2339

```
$ cardak delete file9 -R 2339
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====
file9 6,559,425 bytes EBCDIC      RDW      NORMAL      MC IPM file 2021-11-04
Read 6559425 bytes and 10880 records
Deleting records/fields
New file will have 10879 records
Recalculating trailer record
Saving the file: file9-DEL
Generated file: file9-DEL
Bytes read   : 6,559,425
Records Read : 10,880
Bytes written : 6,558,793

Job ended
Elapsed time : 1.788785402s
$ cardak file9* -a -v
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====

IPM FILES -----
file9      6,559,425 bytes EBCDIC      RDW      NORMAL      MC IPM file 2021-11-04
          * Records: 10,880 Logical files: 1 (10880)
          LF: 1 PRN: 10,880 PDS 0301 (checksum) in Trailer is not correct, value: [0000000000000000], expected: [0000000700378029]
file9-DEL  6,558,793 bytes EBCDIC      RDW      NORMAL      MC IPM file 2021-11-04
          Records: 10,879 Logical files: 1 (10879)

2 files using 12 MB

Job ended
Elapsed time : 3.068579703s
$
```

Observamos que se eliminó un registro del archivo, el cual originalmente contenía 10880 registros y ahora contiene 10879 registros. Vemos además, que el archivo original tenía un error en el trailer, pero el nuevo archivo, al tener que recalcular los valores del trailer, ya soluciona ese problema.

También observamos que el archivo original no se modifica, sino que se genera uno nuevo, por defecto con el mismo nombre del original pero agregándole “-DEL” (podemos especificar el nombre que queremos utilizando el flag --out)

Pero qué pasaría si quisieramos eliminar todos los registros correspondientes al comercio “Carpinteria”? Podemos realizar la operación por cada uno de los registros, o podemos anotar los números de registro encontrados, pero eso puede resultar poco práctico si tenemos una cantidad considerable de registros.

Para ello podemos utilizar el flag --last que utiliza el resultado del último comando GREP realizado sobre ese archivo. Veamos cómo se usa:

```

$ cardak grep 'carpinteria' file9
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====

file9
Record 763: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]
Record 1083: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]
Record 2339: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]
Record 2681: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]
Record 5805: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]
Record 10638: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]

TOTAL:      6 matches in file9

=====
Bytes read (data) : 6,559,425
Records Read     : 10,880
=====

Job ended
Elapsed time : 786.173058ms
$ cardak delete file9 --last
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====
file9 6,559,425 bytes EBCDIC      RDW      NORMAL      MC IPM file 2021-11-04
Read 6559425 bytes and 10880 records
Deleting records/fields
New file will have 10874 records
Recalculating trailer record
Saving the file: file9-DEL
Generated file: file9-DEL
Bytes read     : 6,559,425
Records Read   : 10,880
Bytes written  : 6,555,679

Job ended
Elapsed time : 2.015194418s
$ 

```

Realizamos una búsqueda por el valor “carpinteria”, y nos devuelve que hay 6 registros para ese comercio. El próximo comando utilizado es el comando DELETE, pero usando el flag --last, por lo que se utiliza la lista de registros devueltas como resultado del GREP anterior, y vemos que efectivamente se genera un archivo con 10874 registros resultantes de quitar 6 registros de los 10880 originales.

## Eliminar campos en algunos registros

También puede que necesitemos eliminar solo algunos campos de algunos registros, por ejemplo si por error incluimos algún campo que no debería viajar.

En forma similar al caso anterior, simplemente debemos agregar el flag -F con la lista de los campos que deseamos eliminar. Esos campos van a ser eliminados solamente de los registros indicados, o de todo el archivo en caso de no especificar ningún registro en particular.

Hay que tener cuidado ya que no se chequea que estemos borrando algún campo mandatorio.

## Pasar registros de un archivo a otro

Digamos que queremos transferir registros de un archivo a otro. Esto lo podemos conseguir mediante dos pasos, primero, quitando los registros del primer archivo y luego agregándolos al segundo archivo.

El primer paso se consigue con el comando EXPORT, y el segundo con el comando IMPORT

Para el comando EXPORT tenemos dos opciones, exportar a un archivo **CSV** o una archivo **HEX**. Ambos formatos se pueden usar para la importación, aunque el formato **HEX** se considera una forma mas directa de pasaje.

Primero, vamos a seleccionar los registros que queremos exportar. Utilizaremos nuevamente el ejemplo anterior de los registros correspondientes al comercio “Carpinteria”:

```

$ cardak grep 'carpinteria' file9
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====

-----
file9
Record 763: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]
Record 1083: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]
Record 2339: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]
Record 2681: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]
Record 5805: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]
Record 10638: (MTI 1240 - First Presentment)
    DE043 : [Handy*MH Carpinteria\RIZAL 3555\Maldonado\] MO URY]

TOTAL:      6 matches in file9

-----
Bytes read (data) : 6,559,425
Records Read     : 10,880
-----

Job ended
Elapsed time : 801.227122ms
$ 
```

A continuación, procederemos a exportar esos registros a un archivo en formato HEX:

```

$ cardak export file9 --last --hex
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====

6      lines written to file9-EXP.ckh

Job ended
Elapsed time : 56.223089ms
$ 
```

Luego, importaremos esos registros al archivo file14 que como vemos, contiene 61 registros:

```

$ cardak file14 -a --compact
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====

IPM FILES -----
file14      34,316 bytes EBCDIC      RDW      NORMAL      MC IPM file 2021-10-13
          Records:   61 Logical files: 1 (61)

Job ended
Elapsed time : 41.590677ms
$ cardak import file14 file9-EXP.ckh
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====
file14      34,316 bytes EBCDIC      RDW      NORMAL      MC IPM file 2021-10-13
Loaded 61 records from file14

Import from file9-EXP.ckh
    6 records added

Generated file: file14-IMP.ipm
Records Added : 6
Total records : 67
Bytes written : 38,062

Job ended
Elapsed time : 17.870269ms
$ cardak file14* -a
=====
Card Army Knife version 0.7.2 built on 2023-07-21T22:47:38Z main 1565cba
=====

IPM FILES -----
file14      34,316 bytes EBCDIC      RDW      NORMAL      MC IPM file 2021-10-13
          Records:   61 Logical files: 1 (61)
file14-IMP.ipm  38,062 bytes EBCDIC      RDW      NORMAL      MC IPM file 2021-10-13
          Records:   67 Logical files: 1 (67)

2 files using 71 kB

Job ended
Elapsed time : 60.233875ms
$ 
```

Observamos que se genera un nuevo archivo llamado file14-IMP.ipm que ahora contiene 67 registros, que son los 61 originales mas los 6 importados.

# Comandos

A continuación se detalla el uso y parámetros de cada uno de los comandos

Para ver los comandos disponibles se puede utilizar el comando HELP que nos muestra lo siguiente:

```
$ cardak help
usage: cardak [<flags>] <command> [<args> ...]

A command-line tool for analyzing files.

Flags:
  --help      Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose Add more information displayed on some commands.
  --mono      Suppress color on output.
  --ignore    Try to ignore some errors and continue processing the file
  -z, --silent Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE command piped to
               a search program like fzf

Commands:
  help [<command>...]
  version
  chop [<flags>] <file>
  convert [<flags>] <format> <files>...
  delete [<flags>] <file>
  describe [<flags>] <field name> [<search pattern>]
  distribute [<flags>] <files>...
  duplicates [<flags>] <files>...
  export [<flags>] <files>...
  filter
    list [<flags>] [<file>]
    delete [<flags>] <file>
    rename [<flags>] <old> <new>
    copy [<flags>] <source> <destination>
    add [<flags>] <file> <fields>
    remove [<flags>] <file> <fields>
  fix [<flags>] <files>...
  grep [<flags>] <criteria> <files>...
  identify* [<flags>] <files>...
  import <file> <source>...
  join [<flags>] <files>...
  open [<file>]
  print [<flags>] <file>...
  split <files>...
  validate [<flags>] <files>...

$ █
```

Podemos obtener mas información sobre cada comando, utilizando la forma cardak help <comando>

## CHOP

---

```
$ cardak help chop
usage: cardak chop [<flags>] <file>

Create smaller phisical files from an IPM file

Flags:
  --help      Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose   Add more information displayed on some commands.
  --mono       Supress color on output.
  --ignore     Try to ignore some errors and continue processing the file
  -z, --silent   Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE command piped
                 to a search prgram like fzf
  -m, --max=100000 Maximum number of records for each generated file

Args:
  <file>  File name to chop

$
```

Este comando sirve para fraccionar archivos grandes en archivos más pequeños, lo que facilita su uso, particularmente con comandos como OPEN, donde el consumo de memoria se incrementa de acuerdo al número de registros, y podemos quedarnos sin recursos al intentar trabajar con archivos grandes.

Como ejemplo, un archivo con 50.000 registros consume un poco más de 1 Gb de memoria, y uno de 100.000 registros consume unos 2.2 Gb de memoria, por lo que trabajar con archivos con más cantidad de registros se puede volver poco práctico. Con este comando podemos fraccionar un archivo con gran cantidad de registros en archivos más chicos y manejables.

Si bien la mayoría de los comandos utilizan un método que consiste en ir procesando los registros a medida que son leídos, por lo que el consumo de memoria es acotado e independiente de la cantidad de registros, hay algunos en los que es necesario cargar el contenido total del archivo en memoria (por ejemplo con el comando OPEN, o el comando EXPORT con formato **CSV**, ya que para obtener la lista de campos presentes en el archivo, debemos haber leído el contenido total antes de proceder a la generación del archivo exportado).

Este comando toma el contenido del archivo de entrada, hace un estimado rápido de la cantidad de registros presentes en el archivo (es un estimado ya que para saber el número exacto, habría que procesar todos los registros ya que son de longitud variable) y procede a ir leyendo los registros de entrada. Cuando encuentra un trailer (fin de un archivo lógico) o se llega a la cantidad de registros especificadas para los archivos de salida, se procede a generar un archivo parcial. Cada uno de los archivos generados tiene como nombre el del archivo original, más un número secuencial.

El parámetro que se puede incluir es el flag **--max (-m)** para indicar la cantidad de registros que queremos contenga cada archivo generado, ya que si no se indica ningún valor, se asume que cada uno va a contener 100 mil registros (que consumirán aproximadamente 2Gb, lo que consideramos es aceptable, pero se puede utilizar un valor menor en caso de utilizarse en una máquina con pocos recursos de memoria).

## CONVERT

```
$ cardak help convert
usage: cardak convert [<flags>] <format> <files>...

Convert files between formats.

Files have three characteristics:

  Encoding      : Which characters each byte value represents (ASCII or EBCDIC are supported)
  Record definition: Records can be of fixed length, delimited, or determined by a length
  File packing   : Then can have no packing (NORMAL) or can be packed in blocks (typically 1014)

This tool allow to convert files changing those formats

Flags:
  --help           Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose    Add more information displayed on some commands.
  --mono           Suppress color on output.
  --ignore         Try to ignore some errors and continue processing the file
  -z, --silent     Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE
  command piped to a search prgram like fzf
  -T, --file-type=FILE-TYPE Filter by file type when supplying several files. File types are represented by a single letter as:
                        I-IPM files, M-MPE files

Args:
  <format>  New file format The format consists of three consecutive characters that indicate the Encoding, the record format,
            and the file format. Valid values are:
            Encoding      : [E] - EBCDIC, [A] - ASCII
            Record format : [R] - RDW, [F] - Fixed 1400, [1] - Fixed 1400, [6] - Fixed 600, [D] - Delimited
            File format   : [B] - Block 1014, [N] - Nomrmal
  <files>    List of files. This can be a single file or you can use wildcards

$
```

Este comando permite convertir el formato de archivos.

Los archivos y en particular los que utiliza **MasterCard**, tienen tres propiedades que son la codificación, el formato del registro, y el formato del archivo.

Las codificaciones utilizadas son **ASCII** o **EBCDIC**

El formato del registro puede ser de longitud variable, de longitud fija, o delimitado.

Los archivos **IPM** y **MPE** manejados por **MasterCard** contienen registros de longitud variable. Archivos manejados por otras marcas, suelen ser de longitud fija (típicamente de 600 o 1400 caracteres), y archivos de texto son delimitados (en Windows por los bytes **0x0A** y **0x0D**) y en Unix y variantes, por el carácter **0xD**

Por ultimo, **MasterCard** suele marcar el contenido de sus archivos de intercambios en bloques de 1014 bytes, es decir, los archivos contienen bloques de 1014 bytes (por lo que su tamaño es siempre múltiplo de ese valor), y en cada bloque, el contenido del archivo esta en los primeros 1012 bytes, agregando dos bytes con el valor **0x40** cada uno para completar el bloque. A su vez, el ultimo bloque es es completado con el valor **0x40** hasta llegar a 1014 bytes si los bytes del dato no llegan a los 1012 bytes.

Generalmente, los sistemas que generan o procesan archivos de intercambio con **MasterCard** suelen estar configurados para procesar solamente una combinación de estos valores, por lo que es importante estipular con la marca cual sera el formato a utilizar. En etapas tempranas, especialmente durante el desarrollo, test y certificación, suele suceder que los archivos no están en el formato correcto, lo que hace necesario cambiar la configuración del sistema en cada caso.

Esta herramienta permite convertir el formato de un archivo, independientemente de cual sea ese formato, en un archivo en el formato deseado.

El formato lo especificamos utilizando tres letras (consecutivas, sin espacios), donde la primera corresponde a la codificación, la segunda al tipo de registro, y la tercera al formato del archivo.

Para la codificación, utilizaremos la letra A para indicar **ASCII**, o la letra E para indicar **EBCDIC** (las dos opciones utilizadas en los archivos de intercambio)

Para el tipo de registro utilizaremos R para registro de longitud variable (o **RDW**), F o 1 para registros de longitud fija de 1400 caracteres, 6 para registros de longitud fija de 600 caracteres, o D para archivos delimitados (el delimitador va a depender de si estamos utilizando la herramienta bajo Windows o Linux)

Por ultimo, para la tercer opción, utilizaremos la letra B si queremos que el archivo quede en bloques de 1014, o N si queremos que el archivo quede en formato normal (sin bloques)

El primer parámetro debe ser el formato, y a continuación viene uno o mas nombres de archivos para convertir.

El archivo convertido se nombra a partir del nombre del archivo original, un punto y las tres letras correspondientes al formato, mas los caracteres ".cvt"

Este comando se puede utilizar con cualquier tipo de archivo, pero los archivos **IPM** son tratados en forma diferente, ya que pueden contener datos que son binarios y no debemos intentar convertir su formato ya que estaríamos cambiando el valor y su significado. Por eso, la herramienta, al detectar que se trata de un archivo **IPM**, interpreta los registros y solamente cambia la codificación de aquellos campos que no son binarios.

Este comando esta optimizado y procesa los registros uno por uno, por lo que es capaz de trabajar con archivos grandes sin requerir un uso excesivo de memoria.

Podemos utilizar el flag -T para indicar el tipo de archivo que queremos procesar de la lista de archivos indicados. Esto nos permite utilizar comodines pero solamente aplicarlo a aquellos archivos del tipo especificado. Actualmente podemos indicar el tipo utilizando la letra I para archivos **IPM**, y la letra M para archivos de **MPE**

## CREATE

---

```
$ cardak help create
usage: cardak create [<flags>] <file>

Create a new IPM file

Flags:
  --help           Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose    Add more information displayed on some commands.
  --mono          Suppress color on output.
  --ignore         Try to ignore some errors and continue processing the file
  -W, --width      Ignore small terminal width check and force execution
  -z, --silent     Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE
                   command piped to a search utility like fzf
  --import=IMPORT   File containing the records to be imported
  --pds105=PDS105   String representing the value of PDS 0105
  --file-type=FILE-TYPE File Type to be used in the header
  --file-date=FILE-DATE File date, defaults to today
  --processor-id=PROCESSOR-ID
                   The processor ID
  --sequence-number=SEQUENCE-NUMBER
                   The sequence number
  --processing-mode=PROCESSING-MODE
                   The processing mode, T or P
  --encoding=ENCODING Encoding for new file, default ASCII
  --block           If present, the file will be in block 1014

Args:
  <file>  IPM file name to create
$
```

Con este comando podemos crear un nuevo archivo IPM y opcionalmente cargar registros exportados previamente mediante el comando EXPORT

Para la creación del nuevo archivo, necesitamos indicar el valor del File ID que forma parte del cabezal del archivo, y el File Processing Mode. Podemos indicar el contenido completo del PDS 0105 (los 25 caracteres del File ID), o indicar los subcampos de dicho PDS.

Si decidimos indicar los subcampos, es necesario al menos indicar el SF 01(File Type) y el SF 03 (Processor ID). Si no indicamos el SF 02 (File Reference Date) se va a tomar por defecto la fecha actual, y para el SF 04 (File Sequence Number) se toma el valor 1000

Si no indicamos el Processing Mode, se va a tomar por defecto el valor “T” (Test)

Otros parámetros a indicar son la codificación (si no indicamos nada, se asume ASCII) y si el archivo va a estar o no en Block 1014 (por defecto no)

Si no indicamos un archivo que contenga registros previamente exportados, se va a generar un archivo “vacio”, es decir, conteniendo solamente un Header y un Trailer

En caso de indicar un archivo con registros exportados previamente, el mismo puede ser un export en formato CSV completo, o un export en formato HEX (extension .ckh)

## DELETE

```
$ cardak help delete
usage: cardak delete [<flags>] <file>

Delete records and/or fields from the IPM file

Flags:
  --help           Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose    Add more information displayed on some commands.
  --mono          Suppress color on output.
  --ignore         Try to ignore some errors and continue processing the file
  -W, --width      Ignore small terminal width check and force execution
  -z, --silent     Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE
  command piped to a search utility like fzf
  -R, --records=RECORDS List of record numbers to be considered for deletion. Values are separated by comma (,) and ranges are
                        indicated by the starting and ending record separated by a hyphen (-)
  -F, --fields=FIELDS List of IPM fields to be deleted (can use a filter name)
  -l, --last        Use the record numbers returned on the last GREP command
  -O, --omitdel    Do not create a file with deleted records
  -x, --export      Export the deleted records to a .ckx file (in Hex format)

Args:
  <file>  IPM file name to delete the records from

$
```

Este comando permite borrar registros y/o campos de un archivo **IPM**.

El archivo original no es modificado, y en su lugar se generan por defecto dos archivos con el nombre del original, pero agregándole “**\_DISCARD**” al archivo que contiene los registros eliminados, y “**\_KEEP**” al archivo resultante que no contiene los registros eliminados.

Si lo que se solicita es eliminar campos, el nombre del archivo resultante contendrá “**\_DISCARDEDFIELDS**” en su nombre

En todos los casos, se agrega la extensión “.ipm” al nombre

El parámetro requerido es el nombre del archivo que contiene los registros que queremos borrar, y para indicar los registros a eliminar o modificar (quitando los campos indicados) utilizaremos el flag -R (referirse a la sección [Flags y Filtros](#) para ver en detalle como funciona ese flag).

Como ejemplo, eliminaremos los registros numero 5, y los registros 10 al 15 del archivo file10:

```
$ cardak delete file10 -R 5,10-15
=====
Card Army Knife version 0.9.9413 built on 2023-08-30T18:23:12Z  delete 4c92352
=====
file10  21,254 bytes EBCDIC   RDW      NORMAL   MC IPM file 2020-01-27
Loaded 20,530 bytes and 43 records from file10

Generated files:
  9 records using      3,442 bytes to file10_DISCARD.ipm
  36 records using     17,286 bytes to file10_KEEP.ipm

Job ended
Elapsed time : 18.618694ms
$
```

De los 43 registros del archivo original, se genera un archivo con 9 registros (los 7 seleccionados mas el header y trailer), y otro archivo con 36 registros (los 34 restantes mas el header y trailer)

Para que los archivos puedan ser procesados correctamente por otros sistemas, el valor del PDS0105 SF04 (File Sequence Number) del header del **\_KEEP** contendrá el valor siguiente al del archivo original, y el del **\_DISCARD** dos valores mas.

Si no nos interesa que se genere el archivo **\_DISCARD**, podemos utilizar el flag --omitdel (-O), y utilizando el flag --export (-x), se va a generar un archivo de extension .ckx contenido esos registros eliminados en formato **HEX** (que pueden ser importados en otro archivo en forma posterior)

Si quisieramos eliminar algunos campos pero no registros enteros, podemos utilizar el flag -F para indicar la lista de campos a eliminar. Si no especificamos ningun registro o rango de registros (mediante el flag -R), los campos se eliminaran de todos los registros del archivo, de lo contrario, solamente se van a eliminar de los registros indicados.

Por ejemplo, vemos los primeros 5 registros:

```

$ k print file10 -R 1-5
=====
Card Army Knife version 0.9.9413 built on 2023-08-30T18:23:12Z delete 4c92352
=====
file10

1: MTI 1644 - File Header
    DE024    DE071
    PDS0105 PDS0122

2: MTI 1240 - First Presentment
    DE002    DE003    DE004    DE012    DE022    DE024    DE026    DE031    DE033    DE038    DE042    DE043    DE049    DE063
DE071    DE094
    PDS0023 PDS0148 PDS0158 PDS0159 PDS0165 PDS0777 PDS1001 PDS1002 PDS1003 PDS1004 PDS1011

3: MTI 1240 - First Presentment
    DE002    DE003    DE004    DE012    DE022    DE024    DE026    DE031    DE033    DE038    DE042    DE043    DE049    DE063
DE071    DE094
    PDS0023 PDS0148 PDS0158 PDS0159 PDS0165 PDS0777 PDS1001 PDS1002 PDS1003 PDS1004 PDS1011

4: MTI 1240 - First Presentment
    DE002    DE003    DE004    DE012    DE022    DE024    DE026    DE031    DE033    DE038    DE042    DE043    DE049    DE063
DE071    DE094
    PDS0023 PDS0148 PDS0158 PDS0159 PDS0165 PDS0777 PDS1001 PDS1002 PDS1003 PDS1004 PDS1011

5: MTI 1240 - First Presentment
    DE002    DE003    DE004    DE012    DE022    DE024    DE026    DE031    DE033    DE038    DE042    DE043    DE049    DE063
DE071    DE094
    PDS0023 PDS0148 PDS0158 PDS0159 PDS0165 PDS0777 PDS1001 PDS1002 PDS1003 PDS1004 PDS1011

Bytes read (data) : 20,530
Records Read     : 43

Job ended
Elapsed time   : 6.589193ms
$ 

```

Supongamos que queremos eliminar los campos PDS0023 y PDS1011 de los registros 3 y 4, haríamos lo siguiente:

```

$ cardak delete file10 -R 3,4 -F P23,P1011
=====
Card Army Knife version 0.9.9413 built on 2023-08-30T19:17:43Z delete 4c92352
=====
file10      21,254 bytes EBCDIC      RDW      NORMAL      MC IPM file 2020-01-27
Loaded 20,530 bytes and 43 records from file10

Generated file:
    43 records using      20,496 bytes to file10_DISCARDFIELDS.ipm

Job ended
Elapsed time   : 8.499751ms
$ 

```

Veremos ahora como han quedado los 5 primeros registros del archivo generado.  
Podemos observar que los registros 3 y 4 ahora no contienen los campos eliminados.

Es importante aclarar que no hay un chequeo en la eliminación de campos, por lo que podemos estar eliminando campos que son mandatorios y el resultado puede ser un archivo incorrecto.

```

$ cardak print file10_DISCARDFIELDS.ipm -R 1-5
=====
Card Army Knife version 0.9.9413 built on 2023-08-30T19:17:43Z delete 4c92352
=====
file10_DISCARDFIELDS.ipm

 1: MTI 1644 - File Header
     DE024   DE071
     PDS0105 PDS0122

 2: MTI 1240 - First Presentment
     DE002   DE003   DE004   DE012   DE022   DE024   DE026   DE031   DE033   DE038   DE042   DE043   DE049   DE
063   DE071   DE094
     PDS0023 PDS0148 PDS0158 PDS0159 PDS0165 PDS0777 PDS1001 PDS1002 PDS1003 PDS1004 PDS1011

 3: MTI 1240 - First Presentment
     DE002   DE003   DE004   DE012   DE022   DE024   DE026   DE031   DE033   DE038   DE042   DE043   DE049   DE
063   DE071   DE094
     PDS0148 PDS0158 PDS0159 PDS0165 PDS0777 PDS1001 PDS1002 PDS1003 PDS1004

 4: MTI 1240 - First Presentment
     DE002   DE003   DE004   DE012   DE022   DE024   DE026   DE031   DE033   DE038   DE042   DE043   DE049   DE
063   DE071   DE094
     PDS0148 PDS0158 PDS0159 PDS0165 PDS0777 PDS1001 PDS1002 PDS1003 PDS1004

 5: MTI 1240 - First Presentment
     DE002   DE003   DE004   DE012   DE022   DE024   DE026   DE031   DE033   DE038   DE042   DE043   DE049   DE
063   DE071   DE094
     PDS0023 PDS0148 PDS0158 PDS0159 PDS0165 PDS0777 PDS1001 PDS1002 PDS1003 PDS1004 PDS1011

Bytes read (data) : 20,496
Records Read      : 43

Job ended
Elapsed time : 7.968503ms
$ 
```

Podemos realizar una validación del archivo generado:

```

$ cardak validate file10_DISCARDFIELDS.ipm -v
=====
Card Army Knife version 0.9.9413 built on 2023-08-30T19:17:43Z delete 4c92352
=====
PASS file10_DISCARDFIELDS.ipm

Job ended
Elapsed time : 10.40563ms
$ 
```

Podemos también generar un archivo **HEXA** con los registros eliminados:

```

$ cardak delete file10 -R 2-10 -x
=====
Card Army Knife version 0.9.9413 built on 2023-08-30T21:43:04Z delete 4c92352
=====
file10      21,254 bytes EBCDIC      RDW      NORMAL      MC IPM file 2020-01-27
Loaded 20,530 bytes and 43 records from file10

Generated files:
  34 records using      16,423 bytes to file10_KEEP.ipm
  11 records using       4,305 bytes to file10_DISCARD.ipm
    9 records as HEX lines to file10_DISCARD.ckx

Job ended
Elapsed time : 8.846529ms
$ 
```

## DESCRIBE

```
$ cardak help describe
usage: cardak describe [<flags>] <field name> [<search pattern>]

Describe IPM fields and functions.

It accepts the following:

MT[I]          - Show MTI values
TRAN[SACTIONTYPES] - Show valid transaction types
FUNC[TIONCODES] - Show valid function codes
D[E]n          - Data Element n (can use wildcards)
P[DS]n          - PDS element n (can use wildcards)

Flags:
--help          Show context-sensitive help (also try --help-long and --help-man).
-v, --verbose   Add more information displayed on some commands.
--mono          Suppress color on output.
--ignore        Try to ignore some errors and continue processing the file
-z, --silent    Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE command piped
               to a search prgram like fzf
-x, --extended  Display extended information
-s, --subfields Show subfields if they exist

Args:
<field name>      Field name to describe (can use wildcards)
[<search pattern>] Pattern used as filter for results

$
```

Este comando no actúa sobre archivos, es una ayuda rápida para buscar información de campos y valores sin necesidad de recurrir a los manuales.

Nos sirve, por ejemplo, cuando queremos utilizar algún filtro pero no recordamos en que campo viaja el dato requerido.

Es un comando flexible en cuanto a la forma de búsqueda, permitiéndose usar comodines y tambien filtrar por partes de la descripción buscada.

Podemos consultar campos como Data Elements (**DE**), Private Data Subelements (**PDS**), **MTI**, Function Codes y Transaction Codes.

Para el caso de los campos, podemos usar la nomenclatura utilizada en casi todas las partes de la herramienta (para detalles de uso, ver la sección [Flags y Filtros](#)). Podemos, ademas, utilizar comodines para referirnos a mas de un campo, y se puede agregar (opcionalmente) un texto que sera utilizado para filtrar los elementos a mostrar si contienen el texto indicado. Se nos mostrara información similar a la encontrada en los manuales de la marca.

Para los Function y Transaction codes, se nos mostraran los valores y descripciones de las combinaciones de datos correspondientes.

Por defecto se muestra una descripción sumarizada, pero podemos agregar el flag --subfields (-s) para mostrar los subcampos en caso que existan, o el flag --extended (-x) para mostrar información extendida en caso que exista.

Por ejemplo, tomemos el campo **DE43**. Si no estamos seguros que contiene, podemos hacer lo siguiente:

```
$ cardak describe D43
=====
Card Army Knife version 0.7.2 built on 2023-07-22T17:10:47Z main 1565cba
=====
DE
  DE043: Card Acceptor Name/Location
    01 Card Acceptor Name
    02 Card Acceptor Street Address
    03 Card Acceptor City
    04 Card Acceptor Postal (ZIP) Code
    05 Card Acceptor State, Province, or Region Code
    06 Card Acceptor Country Code

  DE 43 (Card Acceptor Name/Location) contains the card acceptor's name and
  location as known to the cardholder

Job ended
Elapsed time : 2.666349ms
$
```

O digamos que queremos buscar un importe pero no recordamos en que campo viene dicho dato. En ese caso podemos pedir todos los **DE** pero filtrando por la descripción:

```
$ cardak describe 'D*' amount
=====
Card Army Knife version 0.7.2 built on 2023-07-22T17:10:47Z main 1565cba
=====
DE -----
DE004: Amount, Transaction
DE005: Amount, Reconciliation
DE006: Amount, Cardholder Billing
DE030: Amounts, Original
DE054: Amounts, Additional
DE111: Amount, Currency Conversion Assessment

Job ended
Elapsed time : 4.701286ms
$
```

Vemos que en este caso pedimos todos los Data Elements indicándolo con 'D\*'. Es necesario usar comillas ya que de lo contrario el shell va a intentar expandir el asterisco.

```
$ cardak describe func
=====
Card Army Knife version 0.7.2 built on 2023-07-22T17:10:47Z main 1565cba
=====

Function Codes -----
MTI 1240 - Presentment
DE24(200) : First Presentment
DE24(205) : Second Presentment (Full)
DE24(282) : Second Presentment (Partial)

MTI 1442 - Chargeback
DE24(450) : First Chargeback (Full)
DE24(451) : Arbitration Chargeback (Full)
DE24(453) : First Chargeback (Partial)
DE24(454) : Arbitration Chargeback (Partial)

MTI 1644 - Administrative
DE24(603) : Retrieval Request
DE24(605) : Retrieval Request Acknowledgement
DE24(640) : Currency Update
DE24(680) : File Currency Summary
DE24(685) : Financial Position Detail
DE24(688) : Settlement Position Detail
DE24(691) : Message Exception
DE24(693) : Text Message
DE24(695) : File Trailer
DE24(696) : Financial Detail Addendum
DE24(697) : File Header
DE24(699) : File Reject

MTI 1740 - Fee Collection
DE24(700) : Fee Collection (Customer-generated)
DE24(780) : Fee Collection Return
DE24(781) : Fee Collection Resubmission
DE24(782) : Fee Collection Arbitration Return
DE24(783) : Fee Collection (Clearing System-generated)
DE24(790) : Fee Collection (Funds Transfer)
DE24(791) : Fee Collection (Funds Transfer Backout)

Job ended
Elapsed time : 9.542244ms
$
```

Por ejemplo, veamos los Function codes:

Se nos muestra por cada valor de **MTI**, los valores que pueden utilizarse en el campo **DE24** y su significado. Esto nos ahorra tener que ir a buscar en los manuales los valores que corresponden a cada tipo de transacción, y es muy útil a la hora de realizar búsquedas en archivos cuando no siempre recordamos los valores que tenemos que usar para los filtros

## DISTRIBUTE

```
$ cardak help distribute
usage: cardak distribute [<flags>] <files>...
Separate one IPM file into several ones by a given criteria.

By defining a criteria configuration file, the contents of an IPM file can be distributed across multiple files where each file contains records for each criteria.

Flags:
  --help           Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose    Add more information displayed on some commands.
  --mono          Suppress color on output.
  --ignore         Try to ignore some errors and continue processing the file.
  -z, --silent     Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE command piped to a search utility like fzf.
  -T, --file-type=FILE-TYPE Filter by file type when supplying several files. File types are represented by a single letter as:
                      I-IPM files, M-MPE files
  -c, --config=CONFIG Configuration file to use, otherwise it uses the default one.
  --dry           Dry-run, simulate the process but do not create any files.
  --keep          Do now delete empty generated files.
  -L, --load=LOAD Load the given configuration for later use. If the configuration already exists, it will be replaced.
  --delete=DELETE Remove the given configuration

Args:
  <files>  List of files. This can be a single file or you can use wildcards
$ █
```

Este comando permite clasificar y distribuir registros de un archivo **IPM** en diferentes archivos de acuerdo a reglas definidas. Esto nos puede servir si, por ejemplo, recibimos un archivo de la marca que contiene transacciones que debemos a su vez enviar a diferentes procesadores, pero esas transacciones vienen todas juntas en un mismo archivo, y queremos que cada procesador reciba solamente las transacciones que les corresponden. O también, si quisiéramos generar archivos que contengan presentaciones, y por otro lado archivos que contengan registros de Fee collection.

Para poder ejecutar este comando, debemos tener archivos de configuración donde indicaremos los criterios a utilizar para esa clasificación de los registros.

Podemos ver la lista de archivos de configuración con el comando “cardak distribute list”

Hay dos ubicaciones posibles para estos archivos, o bien en la carpeta local, o en la carpeta “HOME” de la aplicación. Para copiar el archivo en la carpeta HOME, podemos utilizar el flag --load como veremos mas adelante.

Los archivos de configuración son archivos en formato **JSON** y su “esqueleto” es el siguiente:

```
{
  "<ID>": {
    "FileName": "File1",
    "FilePattern": "$B1.$E",
    "Directory": "/usr/dir/",
    "Description": "",
    "ProcessorID": "123456789",
    "FileSequence": 1,
    "BinRanges": [
      {
        "BinMin": 0,
        "BinMax": 9999
      }
    ],
    "MtiList": {
      "1240": true
      "1442": true
      "1644": true
      "1740": true
    },
    "FieldValues": "DE49:858"
  }
}
```

Veamos en detalle como funciona esto:

En el primer nivel del **JSON**, vamos a colocar entradas, una por cada archivo que vamos a generar. A cada una de estas entradas debemos asignarle un nombre (que vemos como “**<ID>**” pero que puede ser cualquier nombre como veremos en un ejemplo posterior).

Para cada entrada, podemos definir el nombre del archivo ("FileName"), una descripción opcional ("Description"), así como un "ProcessorID" y un "FileSequence" que serán utilizados para crear el cabezal del archivo. Esto porque todos los registros que cumplan las condiciones van a ser guardados en un nuevo archivo, y en su cabezal, en el campo **PDS0105**, necesitamos algunos datos. El **SF01** (File Type) es copiado del archivo original, y lo mismo sucede con el **SF02** (File Reference Date).

Los archivos generados se generarán con un nombre de acuerdo a las siguientes reglas:

Si el campo FilePattern no está presente, los archivos tendrán como nombre el del archivo original (sin la extensión), seguidos por el guion bajo "\_" mas el string indicado en la entrada FileName y la extensión ".ipm"

Si indicamos un FilePattern, que es un string que contiene una combinación opcional de tokens y caracteres que es utilizado como un patrón para armar el nombre del archivo correspondiente.

Los tokens que se pueden utilizar son:

<b>\$B</b>	-> nombre base del archivo de entrada (nombre sin la extensión)
<b>\$E</b>	-> extensión (sin el punto separador)
<b>\$D [YYYYMMDD]</b>	-> fecha/hora usando las siguientes definiciones YYYY, YY, MM, DD, hh, mm, ss
<b>\$F[start:end]</b>	-> partes del nombre base (basados en runas). Valores negativos se cuentan desde el final hacia atrás  start comienza con el valor 0 y end indica el número de carácter con cota superior, pero queda excluido del rango. Se aceptan omitir los extremos start y end
Ejemplos: "ABCDEFGH" no da \$F[2:4] -> "CD" "ABCDEFGH" no da \$F[:4] -> "ABCD" "ABCDEFGH" no da \$F[4:] -> "EFGH"	
<b>\$F[-N]</b>	-> últimos N caracteres del nombre base
<b>\$F[N]</b>	-> primeros N caracteres de la base
<b>\$F[N\$]</b>	-> últimos N caracteres de la base (alias a \$F[-N])
<b>\$\$</b>	-> signo de pesos literal

Ejemplos:

FilePattern: "\$F[:7]\_DATA\_\$D[YYYYMMDD]-\$F[7:].\$E"  
(Siete primeros caracteres) + "\_DATA\_" + (Fecha en formato YYYYMMDD) + "-" + (caracteres desde la posición 8 al final) + "." + (extensión)

Dados estos datos: Nombre de archivo = TT112T0abcdefghijklms.ipm, Fecha=2025-09-27  
→ TT112T0\_DATA\_20250927-abcdefghijklms.dat

Si indicamos un directorio, el archivo será generado en dicha ubicación. En caso de no existir el directorio, la herramienta intentará crearlo, fallando si por ejemplo, el usuario no tiene permisos de creación.

Pero el campo **SF03** (Processor ID) y **SF04** (File Sequence Number) son los indicados en estas entradas. A continuación se especifican las condiciones que debe cumplir un registro para ser incluido en este archivo. Esas condiciones son rango de bytes, **MTI**, o un valor de cualquier campo del registro. Estas condiciones son opcionales, pero debe existir al menos una de ellas.

El rango de bytes consiste en una lista de pares Mínimo y Máximo, y si el **BIN** del campo **DE002** se encuentra en alguno de esos rangos se considera válido (de todos modos se tienen que cumplir TODAS las condiciones especificadas). No se toma ningún largo de **BIN** específico, sino que se considera el mínimo y el máximo para determinar la cantidad de bytes a tomar del **PAN** para la comprobación.

También podemos indicar que valores de **MTI** debemos considerar para que el registro sea incluido en este archivo.

Por último, podemos aplicar la regla "**FieldValues**", donde lo que colocamos es una lista de campos y su valor, con las mismas reglas que utilizamos para las búsquedas del comando **GREP** (o sea, una lista de campo:valor separados por comas o punto y coma)

Para ver las configuraciones existentes, podemos utilizar el parámetro "list" de esta forma:

```
$ cardak distribute list
=====
Card Army Knife version 0.8.3 built on 2023-08-19T14:08:51Z main ad92f3f
=====

Config files in HOME directory

bybin
default

Job ended
Elapsed time : 2.232883ms
$
```

Vemos que tenemos dos configuraciones cargadas. La llamada “default” esta siempre presente y es la que se va a utilizar si no especificamos ninguna.

Para ver el contenido de las definiciones, podemos agregar el flag -v o especificar el nombre de la configuración que queremos ver

```
$ cardak distribute bybin
=====
Card Army Knife version 0.8.3 built on 2023-08-19T14:08:51Z main ad92f3f
=====

bybin

Using configuration file /home/eduardo/.cardak/bybin_dist.json

BANK_1
    Registros que van al Banco 1
    Processor ID : 123456789
    File sequence: 100
    Bin ranges   :          0 to 100000012
                      100000014 to 999999999

BANK_2
    Registros que van al Banco 2
    Processor ID : 112233445
    File sequence: 200
    Bin ranges   : 100000012 to 100000013

BANK_3
    Registros que van al Banco 3
    Processor ID : 987654321
    File sequence: 300
    Bin ranges   : 100000013 to 100000014

-----
Job ended
Elapsed time : 1.929097ms
$
```

Veamos un ejemplo practico y sencillo. Queremos separar las transacciones de un archivo en dos archivos de tal forma que todas las transacciones en Dólares vayan a un archivo y las que sean en Pesos vayan a otro.

```
$ cat money_dist.json | jq
{
  "1-PESOS-858": {
    "FileName": "pesos",
    "Description": "Registros en Pesos Uruguayos",
    "ProcessorID": "123456789",
    "FieldValues": "D49:858"
  },
  "2-DOLARES-840": {
    "FileName": "dolares",
    "Description": "Registros en Dolares",
    "ProcessorID": "987654321",
    "FieldValues": "D49:840"
  }
}
```

Para ello, vamos a crear un archivo llamado “money.json” y le colocaremos este contenido:

Con esto estoy indicando que voy a generar dos archivos, uno cuyo nombre va a ser “pesos” y algo mas, y el otro que sea “dólares” (y algo mas que ya veremos).

Dentro de cada uno, van a tener el ProcessorID indicado, pero no voy a especificar ningn FileSequence por lo que se va a utilizar el mismo del archivo original.

Respecto a las condiciones, no voy a utilizar ni los rangos de bines ni los **MTI**, pero si le indico que para el archivo con transacciones en pesos, el valor del campo **DE049** debe contener el valor **858**, y para el caso de dólares el valor **840**

Veamos las configuraciones disponibles

```
$ cardak distribute list
=====
Card Army Knife version 0.8.3 built on 2023-08-19T14:08:51Z main ad92f3f
=====

Local config files
money
Config files in HOME directory
bybin
default

Job ended
Elapsed time : 2.616722ms
$
```

Ya que no resulta práctico dejar la configuración en el directorio local, vamos a cargarlo en el HOME de la aplicación. Para ello utilizaremos el flag --load

```
$ cardak distribute --load money
=====
Card Army Knife version 0.8.3 built on 2023-08-19T15:59:47Z main ad92f3f
=====

Local config files
money
Config files in HOME directory
bybin
default
money

Job ended
Elapsed time : 1.237577ms
$
```

Y ahora podemos borrar el archivo local “money\_dist.json”, pero la configuración quedará disponible a través de su copia en el directorio HOME. Si la misma configuración está presente en el directorio local y en el HOME, tiene precedencia la local.

Para utilizar este comando, debemos indicar el nombre del archivo a separar y, opcionalmente, el nombre de la configuración que queremos aplicar.

Si no especificamos ningún nombre de configuración, la herramienta va a intentar localizar una configuración llamada “default”. Esto es útil si tenemos una configuración que utilizaremos por defecto a fin de evitar especificarla en cada ejecución.

En este caso tomaremos los contenidos del archivo file10. Primero vamos a realizar una simulación y para ello agregamos el flag --dry (este paso es opcional, lo utilizaremos como demostración y pruebas)

```
$ cardak distribute file10 -c money --dry
=====
Card Army Knife version 0.9.9413 built on 2023-08-30T18:23:12Z delete 4c92352
=====

+-----+
| SIMULATION |
+-----+

Read 20,530 bytes and 43 records from file10
Generated file10_pesos.ipm from 1-PESOS-858
    28 records in          0 bytes written

Generated file10_dolares.ipm from 2-DOLARES-840
    17 records in          0 bytes written

Generated file10_default.ipm from default
    2 records in          0 bytes written

Job ended
Elapsed time : 13.392332ms
$
```

El proceso va a tomar los 43 registros del archivo file10, y va a crear un archivo con los 28 registros en pesos, y otro con los 17 registros en dólares.

Vemos la aparición de un tercer archivo llamado “default”. Este sería el archivo a generar con todos los registros que no cumplan con ninguna de las condiciones. En este caso, como no hay ningún registro que no las cumpla, se estaría generando un archivo llamado “default” que contiene solo un header y un trailer, por lo que en realidad ese archivo no se genera porque estaría vacío, solo se muestra en la simulación. Si quisieramos generarlo de todo modos, bastaría con agregar el flag --keep

Para generar los archivos, omitimos el flag --dry

```
$ ls -l
total 24
-rw-rw-r-- 1 eduardo eduardo 21254 ago 30 15:30 file10
$ cardak distribute file10 -c money
=====
Card Army Knife version 0.9.9413 built on 2023-08-30T18:23:12Z delete 4c92352
=====

Read 20,530 bytes and 43 records from file10

Generated file10_pesos.ipm from 1-PESOS-858
    28 records in      12,862 bytes written

Generated file10_dolares.ipm from 2-DOLARES-840
    17 records in      7,926 bytes written

Job ended
Elapsed time : 8.010495ms
$ ls -l
total 48
-rw-rw-r-- 1 eduardo eduardo 21254 ago 30 15:30 file10
-rw-rw-r-- 1 eduardo eduardo 7926 ago 30 15:31 file10_dolares.ipm
-rw-rw-r-- 1 eduardo eduardo 12802 ago 30 15:31 file10_pesos.ipm
$
```

## DUPLICATES

---

```
$ cardak help duplicates
usage: cardak duplicates [<flags>] <files>...

Find identical records inside the same file

Flags:
  --help                  Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose            Add more information displayed on some commands.
  --mono                  Supress color on output.
  --ignore                Try to ignore some errors and continue processing the file
  -z, --silent              Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE
                           command piped to a search prgram like fzf
  -T, --file-type=FILE-TYPE Filter by file type when supplying several files. File types are represented by a single letter as:
                           I-IPM files, M-MPE files

Args:
  <files>    File names to analyze

$ █
```

Este comando permite buscar registros duplicados en archivos **IPM**

Dos registros se consideran duplicados cuando contienen la misma información en todos los campos excepto en el **DE71** (que es el numero de registro, en caso de duplicados, seguramente este campo pueda tener un valor diferente en cada uno)

## EXPORT

---

```
$ cardak help export
usage: cardak export [<flags>] <files>...

Exports the contents of the file as CSV (comma separated values) or HEX records

It can write the values to a file or display them on the console

Flags:
  --help           Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose    Add more information displayed on some commands.
  --mono          Suppress color on output.
  --ignore         Try to ignore some errors and continue processing the file
  -z, --silent     Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE
                   command piped to a search program like fzf
  -T, --file-type=FILE-TYPE Filter by file type when supplying several files. File types are represented by a single letter as:
                   I-IPM files, M-MPE files
  -R, --records=RECORDS List of record numbers to be exported. Values are separated by comma (,) and ranges are indicated by
                   the starting and ending record separated by a hyphen (-)
  -F, --fields=FIELDS List of IPM fields to be exported (can use a filter name)
  --console        Do now write the file and display output on console
  -C, --code=CODE   Filter by Function Code description
  -x, --hex         Export a .ckh file with records in HEX format
  -l, --last        Use the record numbers returned on the last GREP command

Args:
  <files>  File names to export

$
```

Este comando permite extraer registros completos o parciales de archivos **IPM** y guardarlos en archivos en formato **CSV** o **HEX**, que pueden ser utilizados por programas externos, o como fuente para ser importados en otros archivos **IPM**

En su defecto, cuando no se especifican otras opciones, se genera un archivo CSV con el mismo nombre del original, pero agregándole “-EXP.csv”

Este archivo puede ser abierto en una planilla tipo Excel para manipular sus datos utilizando las herramientas que ofrecen estas planillas.

Los campos binarios se muestran mediante su representación en Hexa

Si quisieramos exportar solamente algunos registros, podemos utilizar el flag -R, y si solamente queremos incluir algunos campos, podemos utilizar el flag -F (por mas información de como utilizar estos filtros, por favor leer la sección [Flags y Filtros](#) )

Podemos utilizar el flag --hex (x) para exportar los registros en un archivo de formato **HEX** y no en **CSV**. En este caso, se ignora el filtro por campo y se exportan los registros completos.

Tenemos la opción de mostrar los registros exportados (en el formato solicitado) por consola sin generar ningún archivo. Para eso debemos utilizar la opción --console

Por ultimo, podemos utilizar la opción --last para exportar los registros devueltos por el ultimo comando GREP realizado sobre el archivo.

## FILTER

```
$ cardak help filter
usage: cardak filter <command> [<args> ...]

Manages filter files

Flags:
  --help      Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose Add more information displayed on some commands.
  --mono      Suppress color on output.
  --ignore    Try to ignore some errors and continue processing the file
  -z, --silent Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE command piped to
               a search program like fzf

Commands:
  filter
    list [<flags>] [<file>]
    delete [<flags>] <file>
    rename [<flags>] <old> <new>
    copy [<flags>] <source> <destination>
    add [<flags>] <file> <fields>
    remove [<flags>] <file> <fields>

$
```

Este comando no actúa sobre archivos, sino que ayuda en el mantenimiento de filtros. Los filtros son configuraciones que nos ayudan a evitar tener que escribir grandes listas de campos para ser usados como filtros. Para ello, este comando maneja (a diferencia de otros comandos), subcomandos.

Para entender mejor el concepto, imaginemos que frecuentemente queremos utilizar los campos DE2, DE3, DE4, DE12, DE24, DE25 y DE26, porque no nos interesa desplegar siempre los valores de todos los campos que vienen en los registros.

Podemos, cada vez que usemos un comando como PRINT, agregar esto:

-R D4,D12,D24,D25,D26

Pero eso es poco práctico. Entonces, lo que podemos hacer es definir esta lista de campos como un filtro, y para ello necesitamos darle un nombre. El nombre es una cadena de caracteres que no debe contener espacios.

Veamos como utilizar esta funcionalidad. Comenzaremos por el subcomando LIST para ver las configuraciones definidas

```
$ cardak filter list
=====
Card Army Knife version 0.7.2 built on 2023-07-22T17:10:47Z main 1565cba
=====

IPM field filter configurations:

WARNING: No configurations have been found

Job ended
Elapsed time : 2.315716ms
$
```

Como podemos ver, no tenemos ningún filtro definido. Procederemos a crear uno llamado "myfilter" y que contenga la lista de campos que mencionamos anteriormente.

```
$ cardak filter add myfilter D4,D12,D24,D25,D26
=====
Card Army Knife version 0.7.2 built on 2023-07-22T17:10:47Z main 1565cba
=====

WARNING: Configuration with name "myfilter" does not exist, a new one will be created

WARNING: Are you sure you want to add 5 fields to "myfilter"? (y/n):
y

Original number of fields : 0
Fields added              : 5
Fields already present   : 0
New number of fields      : 5

New contents of myfilter:

DE004  Amount, Transaction
DE012  Date and Time, Local Transaction
DE024  Function Code
DE025  Message Reason Code
DE026  Card Acceptor Business Code (MCC)

Job ended
Elapsed time : 2.751821294s
$
```

El comando FILTER ADD va a agregar los campos indicados a un archivo de filtro existente, o, en este caso, nos pregunta si queremos crear uno nuevo. Le indicamos que si y nos muestra como quedo configurado este nuevo filtro llamado "myfilter"

A partir de este momento, cuando necesitemos especificar una lista de campos, podemos utilizar el nombre del filtro en su lugar. O sea, en vez de colocar esto:

-R D4,D12,D24,D25,D26

podemos utilizar simplemente:

-R myfilter

Vamos a crear otro filtro mas llamado "otherfilter" pero con otros campos

```
$ cardak filter add otherfilter D2,D12,D24,D4,P5
=====
Card Army Knife version 0.7.2 built on 2023-07-22T17:10:47Z main 1565cba
=====
WARNING: Configuration with name "otherfilter" does not exist, a new one will be created

WARNING: Are you sure you want to add 5 fields to "otherfilter"? (y/n):
y

Original number of fields : 0
Fields added      : 5
Fields already present : 0
New number of fields   : 5

New contents of otherfilter:

  DE002  Primary Account Number (PAN)
  DE004  Amount, Transaction
  DE012  Date and Time, Local Transaction
  DE024  Function Code
PDS0005  Message Error Indicator

Job ended
Elapsed time : 3.082017786s
$
```

Veamos ahora que filtros tenemos definidos utilizando el comando FILTER LIST

```
$ cardak filter list
=====
Card Army Knife version 0.7.2 built on 2023-07-22T17:10:47Z main 1565cba
=====

IPM field filter configurations:

-----
myfilter  :
  DE004  Amount, Transaction
  DE012  Date and Time, Local Transaction
  DE024  Function Code
  DE025  Message Reason Code
  DE026  Card Acceptor Business Code (MCC)

-----
otherfilter :
  DE002  Primary Account Number (PAN)
  DE004  Amount, Transaction
  DE012  Date and Time, Local Transaction
  DE024  Function Code
  PDS0005  Message Error Indicator

Job ended
Elapsed time : 2.127099ms
$
```

ya tenemos dos filtros definidos. Podemos definir todos los que necesitemos, y en caso de tener muchos definidos, podemos ver el contenido en forma mas compacta, usando --summary

```
$ cardak filter list --summary
=====
Card Army Knife version 0.7.2 built on 2023-07-22T17:10:47Z main 1565cba
=====

IPM field filter configurations:

  myfilter    :  DE004      DE012      DE024      DE025      DE026
  otherfilter :  DE002      DE004      DE012      DE024      PDS0005

Job ended
Elapsed time : 2.827634ms
$
```

Si necesitamos ver en detalle uno solo de estos, podemos indicar el nombre

```
$ cardak filter list myfilter
=====
Card Army Knife version 0.7.2 built on 2023-07-22T17:10:47Z main 1565cba
=====

IPM field filter configurations:

myfilter :
    DE004 Amount, Transaction
    DE012 Date and Time, Local Transaction
    DE024 Function Code
    DE025 Message Reason Code
    DE026 Card Acceptor Business Code (MCC)

Job ended
Elapsed time : 2.662939ms
$
```

Ahora, supongamos que no fue una buena elección el nombre de “myfilter” porque no nos da ningún indicio de su uso. Quisiéramos que se llamara, por ejemplo, “basico”. Podemos renombrarlo de la siguiente

```
$ cardak filter rename myfilter basico
=====
Card Army Knife version 0.7.2 built on 2023-07-22T17:10:47Z main 1565cba
=====

WARNING: Are you sure you want to rename "myfilter" to "basico"? (y/n):
y

NOTICE: Configuration file renamed from "myfilter" to "basico"

Job ended
Elapsed time : 2.057105584s
$ cardak filter list --summary
=====
Card Army Knife version 0.7.2 built on 2023-07-22T17:10:47Z main 1565cba
=====

IPM field filter configurations:

    basico : DE004      DE012      DE024      DE025      DE026
    otherfilter : DE002     DE004      DE012      DE024     PDS0005

Job ended
Elapsed time : 3.343463ms
$
```

manera:

Vamos a crear un nuevo filtro a partir de uno ya existente. Tomaremos el filtro “otherfilter” y crearemos uno

```
$ cardak filter copy otherfilter prueba
=====
Card Army Knife version 0.7.2 built on 2023-07-22T17:10:47Z main 1565cba
=====

WARNING: Are you sure you want to copy "otherfilter" into "prueba"? (y/n):
y

NOTICE: Configuration file "otherfilter" copied to "prueba"

Job ended
Elapsed time : 3.04310527s
$ cardak filter list --summary
=====
Card Army Knife version 0.7.2 built on 2023-07-22T17:10:47Z main 1565cba
=====

IPM field filter configurations:

    basico : DE004      DE012      DE024      DE025      DE026
    otherfilter : DE002     DE004      DE012      DE024     PDS0005
    prueba : DE002     DE004      DE012      DE024     PDS0005

Job ended
Elapsed time : 678.737µs
$
```

llamado “prueba”

A continuación vamos a agregar nuevos campos al filtro “prueba”. Para probar, vamos a incluir algunos campos ya existentes y ver que sucede

```
$ cardak filter add prueba D12S2,0005,0149,D22s5,D2,D12S1,D24,P23,p148,d4
=====
Card Army Knife version 0.7.2 built on 2023-07-22T17:10:47Z main 1565cba
=====
WARNING: Are you sure you want to add 10 fields to "prueba"? (y/n):
y

Original number of fields : 5
Fields added : 6
Fields already present : 4
New number of fields : 11

New contents of prueba:

DE002 Primary Account Number (PAN)
DE004 Amount, Transaction
DE012 Date and Time, Local Transaction
DE012S01 Date
DE012S02 Time
DE022S05 Cardholder Present Data
DE024 Function Code
PDS0005 Message Error Indicator
PDS0023 Terminal Type
PDS0148 Currency Exponents
PDS0149 Currency Codes, Amounts, Original

Job ended
Elapsed time : 2.602334706s
$ █
```

Observamos que los campos ya existentes son ignorados, y que podemos agregar directamente subcampos sin tener que incluir el campo genérico (por ejemplo el **DE022S05** esta sin haber incluido el **DE022**).

También podemos quitar campos de un filtro. Por ejemplo, quitemos el **PDS0023**

```
$ cardak filter remove prueba P23
=====
Card Army Knife version 0.7.2 built on 2023-07-22T17:10:47Z main 1565cba
=====
WARNING: Are you sure you want to remove 1 fields from "prueba"? (y/n):
y

Original number of fields : 11
Fields removed : 1
Fields ignored : 0
New number of fields : 10

New contents of prueba:

DE002 Primary Account Number (PAN)
DE004 Amount, Transaction
DE012 Date and Time, Local Transaction
DE012S01 Date
DE012S02 Time
DE022S05 Cardholder Present Data
DE024 Function Code
PDS0005 Message Error Indicator
PDS0148 Currency Exponents
PDS0149 Currency Codes, Amounts, Original

Job ended
Elapsed time : 2.078789417s
$ █
```

Y por ultimo, podemos eliminar un filtro que ya no necesitemos mas.

```
$ cardak filter delete prueba
=====
Card Army Knife version 0.7.2 built on 2023-07-22T17:10:47Z main 1565cba
=====
WARNING: Are you sure you want to delete the file "prueba"? (y/n):
y

NOTICE: Configuration file "prueba" has been deleted

Job ended
Elapsed time : 1.624941732s
$ cardak filter list --summary
=====
Card Army Knife version 0.7.2 built on 2023-07-22T17:10:47Z main 1565cba
=====

IPM field filter configurations:

 basico      : DE004      DE012      DE024      DE025      DE026
 otherfilter : DE002      DE004      DE012      DE024      PDS0005

Job ended
Elapsed time : 2.790468ms
$ █
```

## FIX

---

```
$ cardak help fix
usage: cardak fix [<flags>] <files>...

Fixes an IPM file that has errors

Flags:
  --help            Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose      Add more information displayed on some commands.
  --mono           Suppress color on output.
  --ignore          Try to ignore some errors and continue processing the file
  -z, --silent        Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE command
                      piped to a search program like fzf
  --format=FORMAT    Optional format for the fixed files

Args:
  <files>  File names to fix

$
```

Con este comando podemos intentar corregir errores en archivos.

No todos los tipos de error pueden ser corregidos automáticamente, en particular cuando faltan campos mandatorios o falta algún header, no suele ser posible la corrección automática.

Se genera un archivo con nombre igual al original pero agregándole “-FIX.ipm” al final. El nuevo archivo generado tiene por defecto el mismo formato que el original, pero podemos especificar un formato diferente mediante la opción --format

## GREP

```
$ cardak help grep
usage: cardak grep [<flags>] <criteria> <files>...

Find data in files.

It can search for values regardless of the file format, and it has the ability to understand IPM records to specify in what fields to perform the search

Flags:
  --help           Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose    Add more information displayed on some commands.
  --mono          Suppress color on output.
  --ignore         Try to ignore some errors and continue processing the file
  -z, --silent     Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE command piped to a search prgram like fzf
  -T, --file-type=FILE-TYPE Filter by file type when supplying several files. File types are represented by a single letter as:
                        I-IPM files, M-MPE files
  -R, --records=RECORDS List of record numbers to be printed. Values are separated by comma (,) and ranges are indicated by
                        the starting and ending record separated by a hyphen (-)
  -F, --fields=FIELDS List of IPM fields to list from a matching record, even if these fields don-t have a match (can use a
                        filter name)
  --summary        Only display file names and the matching count for each one.
  --matches       Only display file names and the list of record numbers that match.
  -C, --code=CODE   Filter by Function Code description

Args:
  <criteria>  Search criteria. This is a list of criteria, elements separated by a comma (,) are ANDed together, while elements
              separated by a semi-colon are ORed together. Each element consists of an optional field descriptor followed by a
              colon (:) and the value to search. Field descriptors consist of an optional letter (D-DE fields, P-PDS fields) and
              the corresponding field number. If the field descriptor consist of only numbers, then a three digit number is taken
              as a DE field, and a four digit number is taken as a PDS field. An example could be: 'DE43:Supermarket' to search for
              records having field DE43 that contain the string "Supermarket"
  <files>      List of files. This can be a single file or you can use wildcards
```

Este comando sirve para realizar búsquedas en archivos **IPM**.

El primer parámetro es el criterio de búsqueda, y el resto son nombres de archivos donde realizar la búsqueda.

Ese primer parámetro consiste en una lista de criterios individuales, donde cada uno de ellos es, o bien un valor a buscar en todos los campos del registro, o bien un identificador de campo y el valor a buscar, separados por punto y coma (:). Este identificador de campo obedece a la forma normal de definir identificadores de campos (ver la sección [Flags y Filtros](#) para mas información)

Esta lista de criterios puede estar formada por uno o mas criterios. Los criterios separados por una coma (,) se unen mediante la operación lógica **AND** (o sea, deben cumplirse todos ellos para considerar una coincidencia), y los separados por punto y coma (;) se unen mediante la operación lógica **OR**

Podemos limitar la búsqueda solamente en algunos registros. Para eso utilizamos el flag -R donde especificamos la lista de números de registro o rangos donde realizar la búsqueda.

El flag -F nos permite definir una lista de campos que serán mostrados de los registros que coincidan con el criterio de búsqueda, aunque no formen parte del criterio de la misma. Esto es útil para visualizar valores de registros que buscamos independientemente del criterio de búsqueda.

Si aplicamos el flag --summary, solamente mostraremos los nombres de archivo y la cantidad de coincidencias encontradas en cada uno

Aplicando el flag --matches, mostraremos el nombre del archivo y la lista de registros donde se encontraron coincidencias.

Otro filtro que podemos aplicar para limitar la búsqueda, es mediante el flag --code (-C), donde podemos colocar un texto que forme parte de la descripción del Function Code deseado (por ejemplo “Second Presentment”, o “Partial”)

## IDENTIFY

---

```
$ cardak help identify
usage: cardak identify [<flags>] <files>...

Identify the file type. This is the default command if none is specified

Flags:
  --help      Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose Add more information displayed on some commands.
  --mono      Suppress color on output.
  --ignore    Try to ignore some errors and continue processing the file
  -z, --silent Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE command piped to
              a search program like fzf
  --detailed  Force detailed information regardless of number of files. This option displays more than one line per file
  --compact   Force compact information regardless of number of files. This option displays summary information on just one
              line per file
  -a, --analyze Analyze the contents of the file (for IPM files) and show extended information

Args:
  <files>  List of files to be identified.

$
```

Cuando no especificamos ningún comando, se asume IDENTIFY

Este comando recibe una lista de nombres de archivos y muestra información sobre los mismos, como ser el tamaño (en bytes), la codificación utilizada, el tipo de registros y si el archivo esta en bloques de 1014. Si, solamente indicamos un archivo, se muestra información detallada del mismo, de lo contrario, se muestra en forma compacta (información de un archivo por linea).

De todos modos podemos forzar la salida usando los flags --detailed o --compact

Si agregamos el flag --analyze (-a), se procederá a leer el contenido de los archivos y efectuar un análisis para determinar la cantidad de registros presentes, realizar un chequeo rápido para encontrar errores, y también mostrar algunas estadísticas, como por ejemplo la cantidad de archivos lógicos y cantidad de registros por cada uno, la cantidad de registros por **MTI**, cantidad de registros por Transaction Type, y la cantidad de registros por **MCC** (si agregamos la opción --verbose)

## IMPORT

---

```
$ cardak help import
usage: cardak import <file> <source>...
Import records into the specified file

Flags:
  --help      Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose Add more information displayed on some commands.
  --mono      Supress color on output.
  --ignore    Try to ignore some errors and continue processing the file
  -z, --silent Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE command piped to
               a search prgram like fzf

Args:
  <file>     IPM file name where imported records will be added
  <source>   File that contains the records to be imported

$
```

Este comando efectúa la operación inversa al comando EXPORT

Nos permite agregar a un archivo **IPM**, registros previamente exportados. Se pueden importar archivos HEX o CSV que contengan un export de registros completos.

## JOIN

---

```
$ cardak help join
usage: cardak join [<flags>] <files>...

Join several phisical files as logical files into one file

Flags:
  --help            Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose      Add more information displayed on some commands.
  --mono           Suppress color on output.
  --ignore          Try to ignore some errors and continue processing the file
  -z, --silent        Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE command
                      piped to a search prgram like fzf
  --output=OUTPUT    Name of the generated file
  -f, --fix           Automatically fix the resulting file
  -m, --merge         Merge the file records instead of generating logical files

Args:
  <files>  List of files to join

$
```

Este comando se usa para unir varios archivos **IPM** en otro. Simplemente se indica una lista de archivos y se genera otro contenido los registros de todos esos archivos.

Por defecto, el comando genera un nuevo archivo y coloca los contenidos de los archivos indicados como archivos lógicos, por lo que los registros continúan separados en sus respectivos archivos lógicos.

Podemos cambiar este comportamiento con el flag **--merge** (-m), donde todos los registros de los diferentes archivos quedaran contenidos en un solo archivo lógico. El header de este archivo lógico sera el mismo que el del primer archivo indicado como parámetro en el join.

Tenemos ademas la posibilidad de utilizar el flag **--fix** (-f) para corregir automáticamente errores que puedan encontrarse en los archivos a unir.

## OPEN

---

```
$ cardak help open
usage: cardak open [<file>]

Open the file in a graphical mode

Flags:
  --help      Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose Add more information displayed on some commands.
  --mono      Suppress color on output.
  --ignore    Try to ignore some errors and continue processing the file
  -z, --silent Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE command piped to
              a search program like fzf

Args:
  [<file>]  File name to open

$ █
```

Este comando permite abrir y visualizar los contenidos de un archivo **IPM** en forma interactiva. Para conocer en mas detalle el funcionamiento, continuar leyendo la sección llamada [TUI \(Text User Interface\)](#)

## PRINT

```
$ cardak help print
usage: cardak print [<flags>] <file>...

print the contents of the file record by record

Flags:
  --help                  Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose            Add more information displayed on some commands.
  --mono                  Suppress color on output.
  --ignore                Try to ignore some errors and continue processing the file
  -z, --silent              Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE
                           command piped to a search program like fzf
  -T, --file-type=FILE-TYPE Filter by file type when supplying several files. File types are represented by a single letter as:
                           I-IPM files, M-MPE files
  -R, --records=RECORDS    List of record numbers to be printed. Values are separated by comma (,) and ranges are indicated by
                           the starting and ending record separated by a hyphen (-)
  -F, --fields=FIELDS     List of IPM fields to be listed (can use a filter name)
  -d, --detailed            Print detailed information showing the contents of the fields
  -s, --subfields           Show subfields contents
  -C, --code=CODE            Filter by Function Code DESCRIPTION
  -l, --last                 Use the record numbers returned on the last GREP command

Args:
  <file>  File name to print
$
```

Este comando se utiliza para visualizar los contenidos de archivos **IPM**.

Se recibe como parámetro el nombre del archivo a visualizar, y por defecto nos va a mostrar uno por uno los registros del archivo, indicando el numero de registro, su **MTI** y Function Code, así como una lista de campos **DE** y **PDS** presentes en cada registro.

Esta vista de por si no es de mucha utilidad, por lo que normalmente va en conjunto con el flag **--detailed** (**-d**) que nos muestra los valores de cada uno de los campos.

Como normalmente un archivo contiene muchos registros, la salida suele ser demasiado grande, y si bien es posible redireccionarla a un archivo de texto para luego ver su contenido con un editor de texto, el archivo generado puede llegar a ser muy grande.

Para ser mas prácticos, se suele agregar el flag **--records** (**-R**) donde indicamos los números de registro que deseamos ver, ignorando el resto.

Ademas, para reducir aun mas la salida, podemos especificar que campos queremos visualizar, utilizando el flag **--fields** (**-F**)

Podemos, al igual que con otros comandos, filtrar los registros por Function Code utilizando el flag **--code** (**-C**) y brindando un texto que se usara como filtro en la descripción del Function Code.

Otro flag que podemos utilizar es **--subfields** (**-s**) para mostrar los componentes de los subcampos en aquellos elementos que los tengan definidos.

Por ultimo, podemos utilizar el flag **--last** (**-l**) para mostrar solamente los registros devueltos por el ultimo comando GREP realizados sobre el archivo.

## REPLACE

---

```
$ cardak help replace
usage: cardak replace --search=SEARCH --replace=REPLACE [<flags>] <files>...

Search and replace values in files

Flags:
  --help          Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose   Add more information displayed on some commands.
  --mono          Suppress color on output.
  --ignore         Try to ignore some errors and continue processing the file
  -W, --width      Ignore small terminal width check and force execution
  -z, --silent     Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE
                   command piped to a search utility like fzf
  -s, --search=SEARCH Value of condition to search for
  -r, --replace=REPLACE Value to use for the sustitution
  -R, --records=RECORDS List of record numbers to be Searched. Values are separated by comma (,) and ranges are indicated by
                        the starting and ending record separated by a hyphen (-)
  -F, --fields=FIELDS List of IPM fields to be searched (can use a filter name)
  -l, --last        Use the record numbers returned on the last GREP command

Args:
  <files>  File names to search and replace

$
```

Este comando permite reemplazar valores en un archivo IPM. El valor a buscar se indica con el flag `--search` (`-s`) y el valor a reemplazar se indica con el flag `--replace` (`-r`)

El valor a buscar puede indicarse como un valor constante, o utilizar expresiones regulares.

Si queremos limitar la búsqueda y reemplazo a ciertos registros, podemos utilizar el flag `--records` (`-R`), y si queremos limitarnos a buscar y reemplazar en ciertos campos, podemos utilizar el flag `--fields` (`-F`)

También podemos utilizar el resultado del ultimo GREP realizado utilizando `--last` (`-L`) para hacer el reemplazo solamente en los registros devueltos por el ultimo comando GREP

## SPLIT

---

```
$ cardak help split
usage: cardak split <files>...

Create phisical files from logical files present in an IPM file (a transmission)

Flags:
  --help      Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose Add more information displayed on some commands.
  --mono      Supress color on output.
  --ignore    Try to ignore some errors and continue processing the file
  -z, --silent Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE command piped to
              a search prgram like fzf

Args:
  <files>  File names to split

$ █
```

Este comando es el opuesto al comando JOIN, y permite separar los archivos lógicos contenidos dentro de un archivo **IPM** físico, en tantos archivos físicos como archivos lógicos contenga.

## VALIDATE

---

```
$ cardak help validate
usage: cardak validate [<flags>] <files>...
Validate the IPM files

Flags:
  --help           Show context-sensitive help (also try --help-long and --help-man).
  -v, --verbose    Add more information displayed on some commands.
  --mono          Suppress color on output.
  --ignore         Try to ignore some errors and continue processing the file
  -z, --silent     Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE
                   command piped to a search prgram like fzf
  -T, --file-type=FILE-TYPE Filter by file type when supplying several files. File types are represented by a single letter as:
                        I:IPM files, M:MPE files
  --mcc           Check that MCC values are valid (incomplete list, use with caution)
  --fail          Just display the list of file names that have errors
  --pass          Just display the list of file names that don't have errors

Args:
  <files>  List of files to validate.

$
```

Este comando hace una verificación de la integridad y correctitud de los datos contenidos en un archivo **IPM**. Es una versión mas avanzada al comando IDENTIFY con el parámetro --analyze, y esta pensado para ser utilizado en sistemas de automatización.

Se analizan los archivos indicados y se presentan los resultados de cada uno de los archivos, indicando el resultado de la validación como PASS o FAIL

Ademas, se devuelve al sistema operativo un código que es cero si todos los archivos pasan la prueba correctamente, o con valor 1 si alguno de los archivos contiene errores.

Se muestra, ademas, si los errores encontrados pueden ser corregidos automáticamente (con el comando FIX por ejemplo) o no, ademas de mostrar la cantidad de registros que contienen errores..

Como variantes, podemos agregar el flag --silent (-z) para tener una salida compacta que puede ser analizada por algún proceso externo.

Otra variante es utilizar el flag --pass para tener la lista de archivos sin errores, o el flag --fail para obtener la lista de archivos con errores. Esto permite procesar automáticamente los archivos en función de si tienen o no errores, como ser moverlos a otras carpetas, enviar alertas, etc.

# Flags y Filtros

Con el fin de limitar la acción de algunos comandos a algunos registros o campos, existen filtros con algunas reglas definidas que se mantienen a través de toda la aplicación. A continuación veremos como utilizarlos.

Los Flags se usan en la linea de comandos y están indicados por dos guiones seguidos de una palabra , o algunas veces de un guion seguido de una letra (esto suele usarse como un “alias” para simplificar).

Algunos flags pueden aceptar valores que son los parámetros del mismo y funcionan como filtro. Otras veces el comportamiento varia simplemente ante la presencia o ausencia del flag.

Hay flags que son globales a la aplicación, otros específicos a un comando, y otros que funcionan de forma similar pero no están presentes en todos los comandos ya que carecen de sentidos.

Veamos los flags globales:

```
Flags:
--help      Show context-sensitive help (also try --help-long and --help-man).
-v, --verbose Add more information displayed on some commands.
--mono      Supress color on output.
--ignore    Try to ignore some errors and continue processing the file
-z, --silent Suppress all output (banner, headers, summary) except the results. Specially useful for DESCRIBE command piped to
a search prgram like fzf
```

**-- help:** Este flag se comporta de forma similar al comando HELP, y se puede agregar a cualquier comando para obtener ayuda específica del mismo.

**--verbose (-v):** En algunos comandos, incrementa la información que se muestra. Usualmente se limita un poco la salida para evitar tener un exceso de información, pero agregando este flag podemos obtener en algunos casos la información completa.

**--mono:** Por defecto la salida utiliza colores para mejorar la presentación. Esto puede ser un problema al redireccionar la salida a un archivo ya que el color se logra insertando caracteres de control que la terminal puede entender para representar los colores. Al redireccionarlo a un archivo y luego abrirlo con un editor de textos, vamos a estar viendo estos caracteres de control que molestan en la visualización. En estos casos es aconsejable agregar este flag para suprimir los colores en la salida y hacer mas amigable la posterior utilización de la salida.

**--ignore:** Este flag casi no se usa, pero en algunos casos puede suceder que al leer un archivo se encuentren errores de formato y se detenga el proceso del archivo. Agregando este flag se intenta ignorar el error y continuar (de ser posible) con la lectura y carga del archivo. Las ultimas versiones de la herramienta intentan corregir automáticamente estas situaciones por lo que son muy pocos los casos en que este flag es necesario.

**--silent (-z):** Normalmente la salida por pantalla de los comandos contiene información extra como ser el banner (mostrando la versión e información del build), así como el tiempo total de ejecución. En otros casos la salida esta en un formato pensado para la visualización en pantalla del operador. Pero hay ocasiones en que necesitamos procesar esa salida mediante alguna herramienta externa. Este flag siempre elimina el color, evita desplegar el banner y la información al pie con el tiempo total de ejecución, y evita mostrar las barras de progreso en archivos grandes. Ademas, dependiendo del comando, modifica el formato de la salida a su mínima expresión para facilitar su procesamiento por herramientas externas, orientado a la automatización de procesos.

Cada comando puede manejar flags particulares cuyo significado esta explicado en la sección correspondiente. Sin embargo, hay algunos que son compartidos por varios comandos por lo que para no repetir su uso en detalle en cada sección, se explican a continuación:

**-T**

Este flag requiere de un valor asociado que es una letra, que puede ser la “I” o la “M”

Se utiliza cuando le pasamos una lista de archivos para procesar (por ejemplo utilizando comodines) pero el comando solamente esta pensado para procesar archivos **IPM** o **MPE**

La presencia de este flag con el valor correspondiente, hace que de esa lista solamente se tomen en cuenta los archivos del tipo indicado (**IPM** o **MPE**) y el resto sea ignorado.

**-C**

Este flag es un filtro que toma en cuenta registros del archivo de acuerdo al Function Code. Recibe un texto y solamente se tomaran en cuenta aquellos registros cuya descripción del Function Code contenga el texto indicado.

Por ejemplo, si queremos tomar en cuenta solamente registros que contengan información de Chargebacks, podemos agregar “-C chargeback” (para ver la lista completa de los Function Codes disponibles, podemos ejecutar “cardak describe func”

#### **-R**

Filtro por numero de registro. Con este flag podemos indicar los registros que queremos considerar utilizando su numero dentro del archivo físico. Recibe una lista separada por comas que contiene numeros específicos de registro o rangos. Para definir un rango, simplemente colocamos el numero de registro inicial y final separados por un guion. Ambos extremos serán incluidos en el rango.

Si, por ejemplo, quisiéramos procesar solamente los registros numero 10, 12, 30 al 39 y 50, tendríamos que poner: “-R 10,12,30-39,50”

Existe una particularidad con el comando GREP, ya que cuando lo utilizamos sobre un solo archivo y se encuentran registros que cumplan con la condición, estos números de registros se guardan automáticamente y pueden ser utilizados por el siguiente comando si aplicamos el flag --last, lo que nos evita tener que escribir la lista de números de registro (que puede ser extensa) para operar sobre los registros encontrados por el ultimo comando GREP. La herramienta chequea que al usar este flag, estemos utilizando el mismo archivo que el utilizado sobre el comando GREP anterior.

#### **-F**

Filtro por campo (fields). En algunos casos, no solo queremos indicar sobre que registros queremos aplicar el comando, sino también limitarnos a determinados campos. Al igual que el flag de filtro por registro, este flag recibe una lista que identifica a los campos, pero con algunas diferencias.

Antes que nada, veamos como identificar un campo. En los archivos **IPM** tenemos dos tipos de elementos, los llamados Data Elements (DE) y los Private Data Subelement (PDS). A su vez, estos pueden estar subdivididos en campos mas pequeños llamados SubFields.

La nomenclatura utilizada en los manuales y en la aplicación es la siguiente. Los DE van seguidos de un numero de entre 1 y 3 cifras que identifican a cada uno de ellos, mientras que los PDS van seguidos de un numero de cuatro cifras. Los SubFields se indican colocando las letras “SF” y un numero de dos cifras a continuación del identificador de campo. Por ejemplo, el campo llamado “Card Acceptor Name” seria el DE043, y estaría dividido en 6 partes, donde cada una seria: DE043SF01, DE043SF02, etc.

Esta herramienta, y a fin de simplificar la forma de identificar los campos, acepta lo siguiente:

Los DE pueden representarse con las letras DE o D, seguidas de uno a tres dígitos, o simplemente con tres dígitos.

Los PDS pueden representarse con las letras PDS, PD o P seguidas de uno a cuatro dígitos, o simplemente usando cuatro dígitos.

Los SubFields se indican agregando las letras SF o S, seguidas de uno o dos dígitos

Teniendo esto en cuenta, el campo DE43 se puede indicar de cualquiera de estas formas:

DE043, DE43, D43, D043, 043

Si quisiéramos indicar un SubField específico, a cualquiera de las opciones anteriores deberíamos agregarle SF o S y el numero de SubField. Por ejemplo, para indicar el SubField 03, podríamos poner:

DE043SF03, D43S3, 043S3, etc.

Para los PDS es algo similar, pero si no agregamos las letras debemos colocar un numero de 4 dígitos. Por ejemplo, para el PDS0021 podemos utilizar:

PDS0021, P21, PD21, P21, 0021, etc.

Para indicar un SubField, se aplica la misma regla que para los DE

# TUI (Text User Interface)

La herramienta tiene una opción para visualizar y realizar operaciones sobre los archivos **IPM** en forma interactiva. Si bien no es una opción gráfica clásica, es un intermedio, donde la apariencia es similar a la de un entorno gráfico, pero confinado a una terminal. Esto ofrece la posibilidad de ser ejecutada en forma remota, por ejemplo mediante una sesión SSH.

A esta forma le llamamos Text User Interface.

Para acceder a esta modalidad, utilizamos el comando **OPEN**, el cual lee el contenido de un archivo y lo presenta en un formato que ocupa la totalidad de la terminal y permite navegar y realizar acciones mediante combinaciones de teclas. En algunas terminales también se acepta el uso del ratón.

Para este ejemplo vamos a utilizar un archivo **file20** que contiene 4 archivos lógicos tal como vemos aquí:

```
$ cardak file20 -a --compact
=====
Card Army Knife version 0.7.2 built on 2023-07-23T19:25:08Z main 1565cba
=====

IPM FILES -----
file20 6,616,625 bytes EBCDIC      RDW      NORMAL      MC IPM file 2020-01-27
Records: 10,990 Logical files: 4 (43,61,6,10880)

Job ended
Elapsed time : 1.867060101s
$
```

Ahora ejecutaremos el comando “**cardak open file20**”

The screenshot shows the Card Army Knife TUI interface. At the top, there's a header with the command used (`file20 -a --compact`), the application version (`Card Army Knife version 0.7.2`), and the build date (`2023-07-23T19:25:08Z`). Below this is the **IPM FILES** section, which provides details about the file: size (6,616,625 bytes), code page (EBCDIC), record type (RDW), and status (NORMAL). It also shows the number of records (10,990) and logical files (4). The data grid below contains 43 records, each with fields like Proc.ID, Date, Type, and various monetary values. The footer contains administrative information and a status bar at the bottom.

Se nos presenta la pantalla principal con el contenido del archivo (**file20** en este caso). La pantalla está dividida en 5 secciones verticales que detallaremos a continuación.

En la parte superior, tenemos una barra que contiene información del archivo. Vemos el tamaño en bytes, la cantidad de registros total, la codificación y si está en formato de bloque. También vemos la fecha incluida en el cabezal del primer header, y luego un área donde vamos a tener la cantidad de registros activos (mas información en secciones posteriores de este manual).

Por último, sobre la derecha, vemos información sobre el uso de memoria (reservadas por la aplicación y total utilizadas en el sistema).

Debajo de esta barra viene el área principal donde vemos los datos principales de los registros incluidos en el archivo. Vemos, por ejemplo, el número de registro dentro del archivo, el **MTI** y el tipo de registro.

Dependiendo del tipo de transacción que viene en el registro, vemos más detalles del contenido. Por

ejemplo, para las compras, vemos en **PAN** (enmascarado), el importe, la moneda y su código, la fecha y hora de la transacción, el **MCC** y el nombre del comercio.

Deabajo de esta area tenemos una mas pequeña donde podemos ver con mas detalle información del registro.

Deabajo tenemos una barra con fondo gris donde se nos muestra un recordatorio/ayuda de las teclas disponibles y sus acciones.

Por ultimo, debajo de todo, tenemos la barra de estado. Sobre la parte izquierda tenemos un indicador de "slots" de memoria (que veremos mas adelante), un área central para mensajes, y a la derecha información sobre el registro sobre el que estamos posicionados. Vemos el numero de registro y el total, luego vemos el numero de archivo lógico (en este caso estamos viendo el primero de 4 archivos lógicos contenidos en el archivo físico), y por ultimo el indicador de "Auto Save" (mas información sobre esto cuando veamos el uso) Presionando la tecla **F1** podemos acceder a una ayuda relacionada a la pantalla en la que estemos posicionados. Por ejemplo, en esta pantalla principal, presionando **F1** vemos lo siguiente:

```
Main View

Keyboard shortcuts ----

APPLICATION:
Ctrl-Q : Exit application
Esc   : Exit application

NAVIGATION:
Ctrl-G  : Goto record number
↑,←   : Move to previous record
↓,→   : Move to next record
Ctrl-← : Move to previous Mark Slot
Ctrl-→ : Move to next Mark Slot
Tab    : Go to next logical file
Shift-Tab : Go to previous logical file
Enter  : Enter record detail view
Space  : Enter record detail view

SEARCH and FILTER:
S      : Search
S      : Clear previous search
→,F3  : Jump to next search result
←,Shift-F3 : Jump to previous search result
f      : Filter records
F      : Clear current filter
Ctrl-F  : Perform a quick filter (by transaction type)
F5     : Select fields to display
Shift-F5 : Reset field display filter

MARKS and SLOTS:
+      : Mark all visible records
-      : Unmark all visible records
*      : Invert marks on visible records
M      : Enter marking state, action depends on next key pressed:
M      : Cancel marking
[0-9]  : Select slot
c      : Clear current slot
a      : Set region start
z      : Set region end and mark all records in the region
s      : Mark all records from the last search result
Ctrl-Space : Toggle marks on current record
F4     : Convert between filters and marks
```

Veremos ahora las acciones que podemos realizar

## Navegación

Para movernos por los registros utilizaremos las teclas de cursores, **PgUp**, **PgDn**, **Home** y **End**.

Podemos ir a un numero de registro específico con la combinación **Ctrl-G**, que nos pedirá un numero de registro, y al presionar **ENTER** nos posicionaremos directamente sobre el, tal como vemos a continuación:

100000-*****-3511	400.00	USD
100000-	00	UYU
100000- Go to record: [ ]	00	USD
100000-	64	UYU
100000-*****-8417	2.901.64	UYU

Con las teclas **Tab** y **BackTab**

(**Shift-Tab**), vamos a ir pasando de un archivo lógico al siguiente/anterior (si hay mas de un archivo lógico presente, de lo contrario esta tecla no tendrá efecto alguno). El archivo lógico sobre el que estaremos trabajando lo veremos reflejado en la parte inferior derecha, y también en la linea superior.

Si queremos ver los detalles de un registro, estando parados sobre el, presionaremos la tecla **SPACE** o **ENTER**, y se nos abrirá una nueva área donde veremos el contenido detallado del registro

```

file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records
MEM: (227 MB/280 MB)

1240 Presentment
└─DE 2 Primary Account Number (PAN) [1000000000240037]
└─DE 3 Processing Code [000000]
   └─SF01 Cardholder Transaction Type [00] Purchase (Goods and Services)
      └─SF02 Cardholder 'From' Account Type Code [00] Default Account
         └─SF03 Cardholder 'To' Account Type Code [00] Default Account
   └─DE 4 Amount, Transaction [000000025100] 251.00
   └─DE 12 Date and Time, Local Transaction [200123] 2020-Jan-23
      └─SF01 Date [165731] 16:57:31
      └─SF02 Time [M10101299001]
   └─DE 22 Point of Service Data Code [M] Contactless EMV/Chip (Proximity Chip)
      └─SF01 Terminal Data: Card Data Input Capability [1] Terminal has PIN entry capability
         └─SF02 Terminal Data: Cardholder Authentication Capability [0] Terminal/operator does not have card capture capability
            └─SF03 Terminal Data: Card Capture Capability [1] On card acceptor premises; attended terminal
            └─SF04 Terminal Operating Environment [0] Cardholder present
            └─SF05 Cardholder Present Data [1] Card present
            └─SF06 Card Present Data [2] Auth. Entry Mode 02
            └─SF07 Card Data: Input Mode [9] Unknown; data unavailable
            └─SF08 Cardholder Authentication Method [9] Unknown; data unavailable
            └─SF09 Cardholder Authentication Entity [0] Unknown; data unavailable
            └─SF10 Card Data Output Capability [0] Unknown; data unavailable
            └─SF11 Terminal Data Output Capability [1] Unknown; data unavailable
            └─SF12 PIN Capture Capability [1] First Presentment
   └─DE 24 Function Code [5251] HARDWARE STORES
   └─DE 26 Card Acceptor Business Code (MCC) [22700720027000000000801]
      └─SF01 Mixed Use; any numeric value [2]
      └─SF02 Acquirer's BIN [270072]
      └─SF03 Julian Processing Date YDDD [0027]
      └─SF04 Acquirer's Sequence Number [000000000080]

1240 1240 Presentment
Purchase
First Presentment
  2020-01-23 16:57:31      251.00 (840 - USD) US Dollar : UNITED STATES OF AMERICA (THE)

Esc Back Del Delete field F2 Hide subfields F5 Filter by fields Ctrl-A Add Ctrl-E Edit F8 Describe
M: 0123456789
R#10 10/43 <1/4> as

```

Esta vista nos muestra los campos presentes con su descripción y sus valores, y por defecto muestra también los subcampos. En algunos casos incluso se muestra el significado del valor de cada campo. Si queremos omitir los buscamos y ver solo los campos principales, podemos cambiar la vista con la tecla **F2**

```

file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records
MEM: (227 MB/280 MB)

1240 Presentment
└─DE 2 Primary Account Number (PAN) [1000000000240037]
└─DE 3 Processing Code [000000]
   └─SF01 Cardholder Transaction Type [000000025100] 251.00
   └─SF02 Cardholder 'From' Account Type Code [200123165731]
   └─SF03 Cardholder 'To' Account Type Code [M10101299001]
   └─DE 4 Amount, Transaction [200] First Presentment
   └─DE 12 Date and Time, Local Transaction [5251] HARDWARE STORES
      └─SF01 Date [22700720027000000000801]
      └─SF02 Time [020659]
      └─SF03 Julian Processing Date YDDD [001145]
      └─SF04 Acquirer's Sequence Number [091278]
   └─DE 22 Point of Service Data Code [Administracion Admin\ANDRES ECHEVESTE DC-27 VI] \TREI
      └─SF01 Terminal Data: Card Data Input Capability [840] (840 - USD) US Dollar : UNITED STATES OF AMERICA (THE)
      └─SF02 Terminal Data: Cardholder Authentication Capability [MCCJ5W28A0123]
      └─SF03 Terminal Data: Card Capture Capability [00000010]
      └─SF04 Terminal Operating Environment [020659]
      └─SF05 Cardholder Present Data [POI] POI terminal
      └─SF06 Card Present Data [8402]
      └─SF07 Card Data: Input Mode [75]
      └─SF08 Cardholder Authentication Method [M]

1240 1240 Presentment
Purchase
First Presentment
  2020-01-23 16:57:31      251.00 (840 - USD) US Dollar : UNITED STATES OF AMERICA (THE)

Esc Back Del Delete field F2 Show subfields F5 Filter by fields Ctrl-A Add Ctrl-E Edit F8 Describe
M: 0123456789
R#10 10/43 <1/4> as

```

Si nos movemos por los diferentes campos, en la parte inferior vamos a ver información mas detallada sobre ese campo. Por ejemplo, si nos paramos sobre el elemento **DE022**, vamos a ver debajo la definición del largo y tipo de datos, así como el valor debajo de una regla que nos facilita encontrar posiciones dentro del valor.

```

file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records
MEM: (227 MB/280 MB)

1240 Presentment
| DE 2 Primary Account Number (PAN) [10000000000246037]
| DE 3 Processing Code [000000]
| | SF01 Cardholder Transaction Type [00] Purchase (Goods and Services)
| | SF02 Cardholder 'From' Account Type Code [00] Default Account
| | SF03 Cardholder 'To' Account Type Code [00] Default Account
| DE 4 Amount, Transaction [000000025100] 251.00
| DE 12 Date and Time, Local Transaction [200123] 2020-Jan-23
| | SF01 Date [165731] 16:57:31
| | SF02 Time [M10101299001]
| DE 22 Point of Service Data Code [M] Contactless EMV/Chip (Proximity Chip)
| | SF01 Terminal Data: Card Data Input Capability [1] Terminal has PIN entry capability
| | SF02 Terminal Data: Cardholder Authentication Capability [0] Terminal/operator does not have card capture capability
| | SF03 Terminal Data: Card Capture Capability [1] On card acceptor premises; attended terminal
| | SF04 Terminal Operating Environment [0] Cardholder present
| | SF05 Cardholder Present Data [1] Card present
| | SF06 Card Present Data [2] Auth. Entry Mode 02
| | SF07 Card Data: Input Mode [9] Unknown; data unavailable
| | SF08 Cardholder Authentication Method [9] Unknown; data unavailable
| | SF09 Cardholder Authentication Entity [0] Unknown; data unavailable
| | SF10 Card Data Output Capability [0] Unknown; data unavailable
| | SF11 Terminal Data Output Capability [0] Unknown; data unavailable
| | SF12 PIN Capture Capability [1] Unknown; data unavailable
| DE 24 Function Code [200] First Presentment
| DE 26 Card Acceptor Business Code (MCC) [5251] HARDWARE STORES
| DE 31 Acquirer Reference Data [2270072002700000000801]
| | SF01 Mixed Use; any numeric value [2]
| | SF02 Acquirer's BIN [270072]
| | SF03 Julian Processing Date YDDD [0027]
| | SF04 Acquirer's Sequence Number [00000000088]

1240 .....|....2....|....3....|....4....|....5....|....6....|....7....|....8....|....9....|....10....|....11....|....12....|....13....|....14
M10101299001

DE 22 Point of Service Data Code (Length 12) (an) Size: 12
Esc Back Del Delete field F2 Hide subfields F5 Filter by fields Ctrl-A Add Ctrl-E Edit F8 Describe
M: 0123456789
R#10 10/43 <1/4> as

```

Presionando la tecla **Esc** volvemos a la vista principal.

## Búsqueda y Filtro

Si bien la opción de visualizar los registros es útil, cuando tenemos miles de registros se vuelve difícil encontrar los datos que necesitamos. Para eso tenemos la herramienta de búsqueda y de filtro. Ambas son muy similares ya que nos permiten realizar búsquedas especificando los criterios, pero se diferencian en que la búsqueda nos marca los registros que cumplen con la condición, mientras que el filtro nos muestra solamente aquellos registros que cumplen con esa condición de búsqueda. De todos modos es muy sencillo pasar de una vista a la otra como veremos mas adelante.

Veamos como funciona. Comenzaremos haciendo una búsqueda global, queremos buscar la palabra "RUBEN" pero no estamos seguros en que campo se encuentra.

Para ingresar el criterio de búsqueda, presionamos la tecla "s" y se nos va a abrir un dialogo donde podemos ingresar lo que queremos buscar

The screenshot shows the main menu of the application. At the top, there is a status bar with file information: file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records | MEM: (227 MB/280 MB). Below the status bar is a menu bar with options like Esc, Back, Del, Delete, field, F2, Hide subfields, F5, Filter by fields, Ctrl-A, Add, Ctrl-E, Edit, F8, Describe. A cursor is positioned over the 'Search for:' input field, which contains the text 'ruben'. The main area of the screen displays a list of transaction records.

Purchase	Amount	Date	Time	Record Length
Purchase	100000-*****-3511	400.00 USD (840)	2020-01-23 17:47:54	[5732]
				49:51 [5732]
				53:06 [5732]
				57:16 [5732]
Purchase	100000-*****-8417	2,901.64 UYU (858)	2020-01-22 10:58:08	[5732]
Purchase	100000-*****-3511	350.00 USD (840)	2020-01-22 16:47:52	[5911]

Al presionar la tecla **ENTER**, se realiza la búsqueda y se nos marcan los registros que contienen ese valor. Para navegar entre los resultados, podemos utilizar los cursores para derecha e izquierda

```

file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records
Record list MEM: (227 MB/280 MB)
1 1644 File Header      PROD 2020-Jan-27 Proc.ID:00000020659 Seq.:17861 Clearing file: Member-generated
2 1240 1st Pres Purchase 100000*****-3511 4,520.00 UYU (858) 2020-01-23 17:03:09 [5251] Administracion Admin
3 1240 1st Pres Purchase 100000*****-3511 1,235.00 UYU (858) 2020-01-23 17:03:59 [5251] Administracion Admin
4 1240 1st Pres Purchase 100000*****-3511 425.00 USD (840) 2020-01-23 17:04:43 [5251] Administracion Admin
5 1240 1st Pres Purchase 100000*****-3511 826.00 USD (840) 2020-01-23 17:05:40 [5251] Administracion Admin
6 1240 1st Pres Purchase 100000*****-3511 1,758.00 UYU (858) 2020-01-23 17:09:25 [5251] Administracion Admin
7 1240 1st Pres Purchase 100000*****-3511 1,350.00 UYU (858) 2020-01-23 17:37:13 [5251] Administracion Admin
8 1240 1st Pres Purchase 100000*****-0037 782.00 UYU (858) 2020-01-23 16:50:49 [5251] Administracion Admin
9 1240 1st Pres Purchase 100000*****-0037 1,248.00 UYU (858) 2020-01-23 16:55:34 [5251] Administracion Admin
10 1240 1st Pres Purchase 100000*****-0037 251.00 USD (840) 2020-01-23 16:57:31 [5251] Administracion Admin
11 1240 1st Pres Purchase 100000*****-8417 1,245.00 UYU (858) 2020-01-23 16:58:42 [5251] Administracion Admin
12 1240 1st Pres Purchase 100000*****-8417 320.00 USD (840) 2020-01-23 16:58:42 [5251] Administracion Admin
13 1240 1st Pres Purchase 100000*****-3511 1,800.00 UYU (858) 2020-01-22 16:21:17 [5732] Sucursal 1 Sucursal 1
14 1240 1st Pres Purchase 100000*****-3511 2,300.00 UYU (858) 2020-01-23 17:45:43 [5732] Sucursal 1 Sucursal 1
15 1240 1st Pres Purchase 100000*****-3511 1,458.00 UYU (858) 2020-01-23 17:46:27 [5732] Sucursal 1 Sucursal 1
16 1240 1st Pres Purchase 100000*****-3511 720.00 USD (840) 2020-01-23 17:47:10 [5732] Sucursal 1 Sucursal 1
17 1240 1st Pres Purchase 100000*****-3511 400.00 USD (840) 2020-01-23 17:47:54 [5732] Sucursal 1 Sucursal 1
18 1240 1st Pres Purchase 100000*****-3511 6,000.00 UYU (858) 2020-01-23 17:49:51 [5732] Sucursal 1 Sucursal 1
19 1240 1st Pres Purchase 100000*****-0037 2,500.00 USD (840) 2020-01-23 17:53:06 [5732] Sucursal 1 Sucursal 1
20 1240 1st Pres Purchase 100000*****-8417 2,901.64 UYU (858) 2020-01-22 10:57:16 [5732] Sucursal 1 Sucursal 1
21 1240 1st Pres Purchase 100000*****-8417 2,901.64 UYU (858) 2020-01-22 10:58:08 [5732] Sucursal 1 Sucursal 1
22 1240 1st Pres Purchase 100000*****-3511 350.00 USD (840) 2020-01-23 16:47:53 [5812] Sucursal 1 Sucursal 1
23 1240 1st Pres Purchase 100000*****-3511 2,580.00 UYU (858) 2020-01-23 16:48:40 [5812] Sucursal 1 Sucursal 1
24 1240 1st Pres Purchase 100000*****-3511 490.00 USD (840) 2020-01-23 17:14:31 [5812] Sucursal 1 Sucursal 1
25 1240 1st Pres Purchase 100000*****-0037 2,200.00 UYU (858) 2020-01-23 16:58:09 [5812] Sucursal 1 Sucursal 1
26 1240 1st Pres Purchase 100000*****-0037 320.00 USD (840) 2020-01-23 16:58:52 [5812] Sucursal 1 Sucursal 1
27 1240 1st Pres Purchase 100000*****-8417 203.12 USD (840) 2020-01-23 17:05:55 [5812] Sucursal 1 Sucursal 1
28 1240 1st Pres Purchase 100000*****-8417 1,257.38 UYU (858) 2020-01-23 17:06:42 [5812] Sucursal 1 Sucursal 1
29 1240 1st Pres Purchase 100000*****-3511 100.00 UYU (858) 2020-01-22 14:02:27 [5541] Sucursal 1 Sucursal 1
30 1240 1st Pres Purchase 100000*****-0037 1,200.00 UYU (858) 2020-01-22 13:10:13 [5541] Sucursal 1 Sucursal 1

1240 Presentment
First Presentment (DE3:000000 DE24:200 DE33:020659)
Purchase 5541 SERVICE STATIONS WITH OR WITHOUT ANCILLARY SERVICE
2020-01-22 14:02:27 100.00 (858 - UYU) Peso Uruguayo : URUGUAY
Sucursal 1 Sucursal 1 (DARIO,RUBEN) MONTEVIDEO, URU

ENTER Detail F4 >>filter F5 Fields F8 Describe f Filter s Search C-f Quick Filter m Mark C-Sp Toggle Del Delete C-u Undelete C-a Add C-s Save C-x Export C
M: 123456789 Search: 6 R#29 29/43 <1/4> as

```

Si entramos en la vista detallada, veremos en qué campo se encontró ese dato

```

file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records
Record list MEM: (227 MB/280 MB)
1240 Presentment
DE 2 Primary Account Number (PAN) [1000000000233511]
DE 3 Processing Code [000000]
DE 4 Amount, Transaction [000000010000] 100.00
DE 12 Date and Time, Local Transaction [200122140227]
DE 22 Point of Service Data Code [M10101299001]
DE 24 Function Code [200 First Presentment
DE 26 Card Acceptor Business Code (MCC) [5541] SERVICE STATIONS WITH OR WITHOUT ANCILLARY SERVICE
DE 31 Acquirer Reference Data [2270072002700000000091]
DE 33 Forwarding Institution ID Code
DE 38 Approval Code [020659]
DE 42 Card Acceptor ID Code [001080]
DE 43 Card Acceptor Name/Location [Sucursal 1 Sucursal 1\DARIO,RUBEN] \MON
DE 49 Currency Code, Transaction [858] (858 - UYU) Peso Uruguayo : URUGUAY
DE 63 Transaction Life Cycle ID [. MCSOM4K7M0122]
DE 71 Message Number [00000629]
DE 94 Transaction Originator Institution ID Code [020659]
PDS 0023 Terminal Type [POI] POI terminal
PDS 0148 Currency Exponents [8582]
PDS 0158 Business Activity [75]
PDS 0165 Settlement Indicator [M]

1240 1240 Presentment
Purchase
First Presentment
2020-01-22 14:02:27 100.00 (858 - UYU) Peso Uruguayo : URUGUAY

Esc Back Del Delete field F2 Show subfields F5 Filter by fields Ctrl-A Add Ctrl-E Edit F8 Describe
M: 123456789 Search: 6 R#29 29/43 <1/4> as

```

Ahora realizaremos la misma búsqueda, pero usando un filtro en lugar de una simple búsqueda. Para eso, en lugar de presionar la tecla "s", presionamos la tecla "f"

```

Purchase 100000*****-3511 720.00 USD (840) 2020-01-23 17:47:10 [5732]
Purchase 100000*****-3511 400.00 USD (840) 2020-01-23 17:47:54 [5732]
Filter records by:ruben
Purchase 100000*****-8417 2,901.64 UYU (858) 2020-01-22 10:58:08 [5732]
Purchase 100000*****-3511 350.00 USD (840) 2020-01-23 16:47:53 [5812]

```

Ahora, solamente vamos a ver los registros que contienen ese dato

```

file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records
Record list
MEM: (227 MB/280 MB)
29 1240 1st Pres Purchase 100000*****-3511 100.00 UYU (858) 2020-01-22 14:02:27 [5541] Sucursal 1 Sucursal 1
30 1240 1st Pres Purchase 100000*****-0037 1,200.00 UYU (858) 2020-01-22 13:16:13 [5541] Sucursal 1 Sucursal 1
31 1240 1st Pres Purchase 100000*****-0037 40.00 UYU (858) 2020-01-22 14:03:01 [5541] Sucursal 1 Sucursal 1
32 1240 1st Pres Purchase 100000*****-0624 40.00 UYU (858) 2020-01-22 11:58:16 [5541] Sucursal 1 Sucursal 1
33 1240 1st Pres Purchase 100000*****-0516 40.00 UYU (858) 2020-01-22 12:04:32 [5541] Sucursal 1 Sucursal 1
34 1240 1st Pres Purchase 100000*****-0329 40.00 UYU (858) 2020-01-22 12:05:31 [5541] Sucursal 1 Sucursal 1

1240 Presentment
First Presentment (DE3:000000 DE24:200 DE33:020659)
Purchase 5541 SERVICE STATIONS WITH OR WITHOUT ANCILLARY SERVICE
2020-01-22 14:02:27 100.00 (858) Peso Uruguayo : URUGUAY
Sucursal 1 Sucursal 1 (DARIO,RUBEN) MONTEVIDEO, URY

ENTER Detail F4 >>search F5 Fields F8 Describe f Filter s Search C-f Quick Filter m Mark C-Sp Toggle Del Delete C-u Undelete C-a Add C-s Save C-x Export C
M: 123456789 Filter: 6 R#29 1/6 <1/4> as

```

Podemos pasar del resultado de filtro al de búsqueda (y viceversa) presionando la tecla **F4**

Pero el motor de búsqueda nos permite mas flexibilidad en las búsquedas. Podemos utilizar los mismos métodos que las condiciones de búsqueda del comando [GREP](#) analizado en otra parte de este documento. Podemos por ejemplo, indicar en que campo realizar la búsqueda. Para eso debemos colocar el identificador de campo, separado del valor a buscar por dos puntos (:)

Por ejemplo, vamos a buscar nuevamente las transacciones que contengan la palabra “RUBEN”, pero que ademas hayan sido efectuadas entre las 14:00 y las 15:00 horas. Como sabemos que el campo hora esta en el **DE12S02**, y como podemos utilizar expresiones regulares, realizaremos la búsqueda de esta forma:

```

Purchase 100000*****-3511 1,458.00 UYU (858) 2020-01-23 17:40:27
Purchase 100000*****-3511 720.00 USD (840) 2020-01-23 17:47:16
Purchase 100000*****-3511 400.00 USD (840) 2020-01-23 17:47:54
49:51
Filter records by:ruben,012502:^14
53:06
Purchase 100000*****-8417 2,901.64 UYU (858) 2020-01-22 10:58:08
57:16
Purchase 100000*****-3511 350.00 USD (840) 2020-01-23 16:47:53
Purchase 100000*****-2511 3,580.00 UYU (858) 2020-01-23 16:48:36

```

El criterio empleado

es, buscar la palabra “ruben” en todos los campos, pero un valor en el campo **DE012SF02** que comience con “14”. Al estar ambas condiciones separadas por una coma, ambas deben cumplirse simultáneamente

```

file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records
Record list
MEM: (227 MB/280 MB)
29 1240 1st Pres Purchase 100000*****-3511 100.00 UYU (858) 2020-01-22 14:02:27 [5541] Sucursal 1 Sucursal 1
31 1240 1st Pres Purchase 100000*****-0037 40.00 UYU (858) 2020-01-22 14:03:01 [5541] Sucursal 1 Sucursal 1

```

Ahora solamente nos muestra dos registros:

Veamos otro ejemplo, supongamos que queremos buscar un importe de 40

Podemos arrancar la búsqueda presionando la tecla “f”, pero supongamos que no recordamos en que campo se guarda el importe de la transacción. En ese caso, cuando estamos ingresando la condicion, podemos presionar la tecla F8 para que nos muestre un listado de todos los campos. Simplemente escribimos la palabra “amount” y se nos van a filtrar las lineas para mostrar solamente aquellas que contengan la palabra “amount” en su descripción.

Vemos que el importe de la transacción viene en el campo **DE4**, así que nos paramos sobre el y presionamos la tecla **ENTER**.

```

file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records
Record list
Total: 1,196 Displayed: 240 Marked: 0 Search: Fuzzy
Search amount

DE4 Amount, Transaction
DE5 Amount, Reconciliation
DE6 Amount, Cardholder Billing
DE30 Amounts, Original
DE30 01 Original Amount, Transaction
DE30 02 Original Amount, Reconciliation
DE54 Amounts, Additional
DE54 01 Additional Amount, Account Type
DE54 02 Additional Amount, Amount Type
DE54 03 Additional Amount, Currency Code
DE54 04 Additional Amount, Amount Sign
DE54 05 Additional Amount, Amount
DE111 Amount, Currency Conversion Assessment
PD50080 Amount, Tax
PD50080 02 Amount, Tax
PD50140 Amount, Cardholder Billing Amounts USD
PD50140 01 Amount, Cardholder Billing USD
PD50140 02 Amount, Currency Conversion Assessment USD
PD50141 02 Customer Currency Conversion Revenue Amount
PD50145 Amount, Alternate Transaction Fee
PD50145 02 Amount, Fee
PD50146 Amounts, Transaction Fee
PD50146 05 Amount, Fee
PD50146 07 Amount, Fee, Reconciliation
PD50147 Extended Precision Amounts
PD50147 05 Interchange Amount, Fee

DE4 Amount, Transaction
Attributes: n Size: 12

DE 4 (Amount, Transaction) is the amount of funds the cardholder requested in the currency appearing on the transaction information document (TID), which may be the acquirer's local currency or a currency acceptable to the cardholder and card acceptor that the acquirer supports, exclusive of PDS 0146 (Amounts, Transaction Fee).

If no currency is identified on the TID, the transaction is deemed to have taken place in the currency that is legal tender at the point of interaction.

1644 Administrative
File Header (PROD)
File type: 002 - Clearing file: Member-generated
Processor ID: 00000020659 File Reference Date: 2020-Jan-27 File sequence number: 17061

Esc Back/Cancel ENTER Select letters Inc srch C-z Clear C-r Toggle fuzzy Ins Mark Del Unmark C-Sp Toggle mark + Mark all - Unmark all
M: 0123456789 R#1 1/43 <1/4> as

```

Esto hace que el identificador de campo se coloque automáticamente por nosotros en la entrada del criterio, y también coloca los dos puntos separadores para colocar el valor que queremos buscar.

Purchase	100000*****-3511	720.00 USD (840)	2020-01-23 17:47:10	[573]
Purchase	100000*****-3511	400.00 USD (840)	2020-01-23 17:47:54	[573]
Filter records by: 004:				
Purchase	100000*****-8417	2,901.64 UYU (858)	2020-01-22 10:58:08	[573]
Purchase	100000*****-3511	350.00 USD (840)	2020-01-23 16:47:53	[581]

Colocamos el valor 40 y presionamos **ENTER**

Vemos que nos trae mas registros de los deseados. Eso es porque todos ellos contienen el valor “40” en alguna parte, pero no era lo que teníamos intención de buscar.

```

file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records
Record list
Total: 1,196 Displayed: 240 Marked: 0 Search: Fuzzy
Search amount

9 1240 1st Pres Purchase 100000*****-0037 1,240.00 UYU (858) 2020-01-23 16:55:34 [575] Administracion Admin
17 1240 1st Pres Purchase 100000*****-3511 400.00 USD (840) 2020-01-23 17:47:54 [573] Sucursal 1 Sucursal 1
31 1240 1st Pres Purchase 100000*****-0037 40.00 UYU (858) 2020-01-22 14:03:01 [5541] Sucursal 1 Sucursal 1
32 1240 1st Pres Purchase 100000*****-0624 40.00 UYU (858) 2020-01-22 11:58:16 [5541] Sucursal 1 Sucursal 1
33 1240 1st Pres Purchase 100000*****-0516 40.00 UYU (858) 2020-01-22 12:04:32 [5541] Sucursal 1 Sucursal 1
34 1240 1st Pres Purchase 100000*****-0329 40.00 UYU (858) 2020-01-22 12:05:31 [5541] Sucursal 1 Sucursal 1


```

Sabemos que el importe se guarda con dos posiciones decimales para esta moneda, y que el campo viene con ceros a la izquierda para completar el largo. Entonces, podemos utilizar una expresión regular para forzar esa condición. Nos quedaría de esta forma:



Y ahora si, solo nos muestra los importes de valor 40.00

Si tuviéramos todavía muchos resultados, podríamos seguir afinando la búsqueda, por ejemplo por la hora, por el comercio, por el **PAN**, etc.

Tenemos otra opción de filtrado rápido, a la cual accedemos mediante la combinación **Ctrl-F**

Esto nos muestra una serie de opciones, donde podemos ir seleccionando los tipos de transacción que nos interesa, sin necesidad de buscar los campos y códigos correspondientes.

file20 -- 6,016,625 bytes, 10,990 records   EBCDIC   NORMAL   2020-01-27    43 records							MEM: (227 MB/280 MB)		
Record list									
1 1644 File Header	PROD	2020-Jan-27	Proc.ID:00000020659	Seq.:17001	Clearing file: Member-generated				
2 1240 1st Pres	Purchase			Quick Filter					
3 1240 1st Pres	Purchase				) 2020-01-23 17:03:09 [5251] Administracion Admin				
4 1240 1st Pres	Purchase				) 2020-01-23 17:03:59 [5251] Administracion Admin				
5 1240 1st Pres	Purchase				) 2020-01-23 17:04:43 [5251] Administracion Admin				
6 1240 1st Pres	Purchase				) 2020-01-23 17:05:40 [5251] Administracion Admin				
7 1240 1st Pres	Purchase				) 2020-01-23 17:09:25 [5251] Administracion Admin				
8 1240 1st Pres	Purchase				) 2020-01-23 17:37:13 [5251] Administracion Admin				
9 1240 1st Pres	Purchase				) 2020-01-23 16:55:34 [5251] Administracion Admin				
10 1240 1st Pres	Purchase				) 2020-01-23 16:57:31 [5251] Administracion Admin				
11 1240 1st Pres	Purchase				) 2020-01-23 16:58:49 [5251] Administracion Admin				
12 1240 1st Pres	Purchase				) 2020-01-23 16:51:42 [5251] Administracion Admin				
13 1240 1st Pres	Purchase				) 2020-01-22 16:23:17 [5732] Sucursal 1 Sucursal 1				
14 1240 1st Pres	Purchase				) 2020-01-23 17:49:43 [5732] Sucursal 1 Sucursal 1				
15 1240 1st Pres	Purchase				) 2020-01-23 17:46:27 [5732] Sucursal 1 Sucursal 1				
16 1240 1st Pres	Purchase				) 2020-01-23 17:47:10 [5732] Sucursal 1 Sucursal 1				
17 1240 1st Pres	Purchase				) 2020-01-23 17:47:54 [5732] Sucursal 1 Sucursal 1				
18 1240 1st Pres	Purchase				) 2020-01-23 17:49:51 [5732] Sucursal 1 Sucursal 1				
19 1240 1st Pres	Purchase				) 2020-01-23 17:53:06 [5732] Sucursal 1 Sucursal 1				
20 1240 1st Pres	Purchase				) 2020-01-22 10:57:16 [5732] Sucursal 1 Sucursal 1				
21 1240 1st Pres	Purchase				) 2020-01-22 10:58:08 [5732] Sucursal 1 Sucursal 1				
22 1240 1st Pres	Purchase				) 2020-01-23 16:47:53 [5812] Sucursal 1 Sucursal 1				
23 1240 1st Pres	Purchase				) 2020-01-23 16:48:46 [5812] Sucursal 1 Sucursal 1				
24 1240 1st Pres	Purchase				) 2020-01-23 17:14:31 [5812] Sucursal 1 Sucursal 1				
25 1240 1st Pres	Purchase				) 2020-01-23 16:58:09 [5812] Sucursal 1 Sucursal 1				
26 1240 1st Pres	Purchase				) 2020-01-23 16:58:52 [5812] Sucursal 1 Sucursal 1				
27 1240 1st Pres	Purchase				) 2020-01-23 17:05:55 [5812] Sucursal 1 Sucursal 1				
28 1240 1st Pres	Purchase				) 2020-01-23 17:06:42 [5812] Sucursal 1 Sucursal 1				
29 1240 1st Pres	Purchase				) 2020-01-22 14:02:27 [5541] Sucursal 1 Sucursal 1				
30 1240 1st Pres	Purchase				) 2020-01-22 13:10:13 [5541] Sucursal 1 Sucursal 1				
1644 Administrative									
File Header (PROD)									
File type: 002 - Clearing file: Member-generated									
Processor ID: 00000020659 File Reference Date									
Save Cancel									

Space/ENTER Select Tab navigate]

M: 0123456789

R#1 1/43 <1/4> as

## Filtrado de presentación

Otro tipo de filtrado que podemos realizar, es decidir que campos queremos ver en la vista de detalle. Quizás solo nos interese ver un par de Data Elements y tener a la vista todo nos puede generar confusión. Para eso tenemos la tecla **F5** que nos permite colocar la lista de campos que deseamos ver, y opcionalmente, seleccionar uno de los archivos de filtros tal como vimos en el comando [FILTER](#)

13 1240 1st Pres	Purchase	100000*****-3511	1,800.00	UYU (858)	2020-01-22 16:21:17	[5732] Sucursal 1 Sucursal 1
14 1240 1st Pres	Purchase	100000*****-3511	2,300.00	UYU (858)	2020-01-23 17:45:43	[5732] Sucursal 1 Sucursal 1
15 1240 1st Pres						[5732] Sucursal 1 Sucursal 1
16 1240 1st Pres						[5732] Sucursal 1 Sucursal 1
17 1240 1st Pres						[5732] Sucursal 1 Sucursal 1
18 1240 1st Pres						[5732] Sucursal 1 Sucursal 1
19 1240 1st Pres						[5732] Sucursal 1 Sucursal 1
20 1240 1st Pres						[5732] Sucursal 1 Sucursal 1
21 1240 1st Pres						[5732] Sucursal 1 Sucursal 1
22 1240 1st Pres						[5812] Sucursal 1 Sucursal 1
23 1240 1st Pres						[5812] Sucursal 1 Sucursal 1
24 1240 1st Pres	Purchase	100000*****-3511	490.00	USD (840)	2020-01-23 17:14:31	[5812] Sucursal 1 Sucursal 1
25 1240 1st Pres	Purchase	100000*****-0037	2,200.00	UYU (858)	2020-01-23 16:58:09	[5812] Sucursal 1 Sucursal 1
26 1240 1st Pres	Purchase	100000*****-0037	320.00	USD (840)	2020-01-23 16:58:52	[5812] Sucursal 1 Sucursal 1
27 1240 1st Pres	Purchase	100000*****-8417	203.12	USD (840)	2020-01-23 17:05:55	[5812] Sucursal 1 Sucursal 1

Si nos paramos sobre el área llamada “Defined filter” y presionamos el cursor hacia abajo, se nos muestran los filtros definidos

11 1240 1st Pres	Purchase	100000*****-8417	1,245.00	UYU (858)	2020-01-23 16:50:49	[5251] Administracion Admin
12 1240 1st Pres	Purchase	100000*****-8417	320.00	USD (840)	2020-01-23 16:51:42	[5251] Administracion Admin
13 1240 1st Pres	Purchase	100000*****-3511	1,800.00	UYU (858)	2020-01-22 16:21:17	[5732] Sucursal 1 Sucursal 1
14 1240 1st Pres	Purchase	100000*****-3511	2,300.00	UYU (858)	2020-01-23 17:45:43	[5732] Sucursal 1 Sucursal 1
15 1240 1st Pres	Purchase	100000*****-3511	2,300.00	UYU (858)	2020-01-23 17:45:43	[5732] Sucursal 1 Sucursal 1
16 1240 1st Pres						[5732] Sucursal 1 Sucursal 1
17 1240 1st Pres						[5732] Sucursal 1 Sucursal 1
18 1240 1st Pres						[5732] Sucursal 1 Sucursal 1
19 1240 1st Pres						[5732] Sucursal 1 Sucursal 1
20 1240 1st Pres						[5732] Sucursal 1 Sucursal 1
21 1240 1st Pres						[5732] Sucursal 1 Sucursal 1
22 1240 1st Pres						[5732] Sucursal 1 Sucursal 1
23 1240 1st Pres						[5732] Sucursal 1 Sucursal 1
24 1240 1st Pres	Purchase	100000*****-3511	490.00	USD (840)	2020-01-23 17:14:31	[5812] Sucursal 1 Sucursal 1
25 1240 1st Pres	Purchase	100000*****-0037	2,200.00	UYU (858)	2020-01-23 16:58:09	[5812] Sucursal 1 Sucursal 1
26 1240 1st Pres	Purchase	100000*****-0037	320.00	USD (840)	2020-01-23 16:58:52	[5812] Sucursal 1 Sucursal 1
27 1240 1st Pres	Purchase	100000*****-8417	203.12	USD (840)	2020-01-23 17:05:55	[5812] Sucursal 1 Sucursal 1

Al igual que en todos los lugares donde se nos permite ingresar una lista de campos, podemos presionar la tecla **F8** y obtener la lista de campos posibles. En este caso, marcamos los campos deseados utilizando la combinación **Ctrl-Space**, y cuando tengamos todos los campos marcados, presionando **ENTER** nos devuelve la lista armada

```

file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records
Record list
Total: 491 Displayed: 491 Marked: 6 Search: Fuzzy

Search | DE2 Primary Account Number (PAN)
(*) DE3 Processing Code
(*) DE4 Amount, Transaction
() DE5 Amount, Reconciliation
() DE6 Amount, Cardholder Billing
() DE9 Conversion Rate, Reconciliation
() DE10 Conversion Rate, Cardholder Billing
(*) DE12 Date and Time, Local Transaction
() DE14 Date, Expiration
() DE22 Point of Service Data Code
() DE23 Card Sequence Number
(*) DE24 Function Code
() DE25 Message Reason Code
(*) DE26 Card Acceptor Business Code (MCC)
() DE30 Amounts, Original
() DE31 Acquirer Reference Data
() DE32 Acquiring Institution ID Code
() DE33 Forwarding Institution ID Code
() DE37 Retrieval Reference Number
() DE38 Approval Code
() DE40 Service Code
() DE41 Card Acceptor Terminal ID
() DE42 Card Acceptor ID Code
() DE43 Card Acceptor Name/Location
() DE49 Currency Code, Transaction
() DE50 Currency Code, Reconciliation

DE26 Card Acceptor Business Code (MCC)
Attributes: n Size: 4
DE 26 (Card Acceptor Business Code <MCC>) classifies the type of business applicable to the card acceptor

1644 Administrative
File Header (PROD)
File type: 002 - Clearing file: Member-generated
Processor ID: 00000020659 File Reference Date: 2020-Jan-27 File sequence number: 17061

Esc Back/Cancel ENTER Select letters Inc srch C-z Clear C-r Toggle fuzzy Ins Mark Del Unmark C-Sp Toggle mark + Mark all - Unmark all
M: 0123456789 R#1 1/43 <1/4> as

```

```

13 1240 1st Pres Purchase 100000*****-3511 1,800.00 UYU (858) 2020-01-22 16:21:17 [5732] Sucursal 1 Sucursal 1
14 1240 1st Pres Purchase 100000*****-3511 2,300.00 UYU (858) 2020-01-23 17:45:43 [5732] Sucursal 1 Sucursal 1
15 1240 1st Pres
16 1240 1st Pres
17 1240 1st Pres Select fields to display -
18 1240 1st Pres List of fields: 002,003,004,012,024,026
19 1240 1st Pres
20 1240 1st Pres Defined filter:
21 1240 1st Pres
22 1240 1st Pres
23 1240 1st Pres
24 1240 1st Pres Purchase 100000*****-3511 490.00 USD (840) 2020-01-23 17:14:31 [5812] Sucursal 1 Sucursal 1
25 1240 1st Pres Purchase 100000*****-0037 2,200.00 UYU (858) 2020-01-23 16:58:09 [5812] Sucursal 1 Sucursal 1

```

Y presionando **ENTER**

A continuación, presionamos el botón **Apply**, y cuando entremos a la vista de detalle, solamente veremos los campos seleccionados

```

file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records FIELDS FILTER ON
MEM: (196 MB/338 MB)

1240 Presentment
| DE 2 Primary Account Number (PAN) [10000000000233511]
| DE 3 Processing Code [000000]
|   SF01 Cardholder Transaction Type [00] Purchase (Goods and Services)
|   SF02 Cardholder 'From' Account Type Code [00] Default Account
|   SF03 Cardholder 'To' Account Type Code [00] Default Account
| DE 4 Amount, Transaction [900000452000] 4,520.00
| DE 12 Date and Time, Local Transaction [200123] 2020-Jan-23
|   SF01 Date [170309] 17:03:09
|   SF02 Time [200] First Presentment
| DE 24 Function Code [5251] HARDWARE STORES
| DE 26 Card Acceptor Business Code (MCC)

```

Para quitar el filtro de campos, presionamos **Shift-F5**

## Marcar registros

Podemos marcar registros para luego efectuar operaciones sobre ellos y no sobre el contenido total del archivo. Primero veamos como marcar registros. La primer opción es simplemente posicionarse sobre el registro deseado y presionar las teclas **Ctrl-Space**. En este ejemplo, nos posicionamos sobre el registro numero 6 y presionamos esas teclas. Veremos que el numero de registro queda de otro color, indicando que tiene una marca

file20 -- 6,616,625 bytes, 10,990 records   EBCDIC   NORMAL   2020-01-27    43 records							MEM: (187 MB/338 MB)
Record list							
1	1644	File Header	PROD 2020-Jan-27	Proc.ID:00000020659	Seq.:17061	Clearing file: Member-generated	
2	1240	1st Pres	Purchase	100000*****-3511	4,520.00	UYU (858) 2020-01-23 17:03:09	[5251] Administracion Admin
3	1240	1st Pres	Purchase	100000*****-3511	1,235.00	UYU (858) 2020-01-23 17:03:59	[5251] Administracion Admin
4	1240	1st Pres	Purchase	100000*****-3511	425.00	USD (840) 2020-01-23 17:04:43	[5251] Administracion Admin
5	1240	1st Pres	Purchase	100000*****-3511	826.00	USD (840) 2020-01-23 17:05:40	[5251] Administracion Admin
6	1240	1st Pres	Purchase	100000*****-3511	1,758.00	UYU (858) 2020-01-23 17:09:25	[5251] Administracion Admin
7	1240	1st Pres	Purchase	100000*****-3511	1,350.00	UYU (858) 2020-01-23 17:37:13	[5251] Administracion Admin
8	1240	1st Pres	Purchase	100000*****-0037	782.00	UYU (858) 2020-01-23 16:52:57	[5251] Administracion Admin
9	1240	1st Pres	Purchase	100000*****-0037	1,240.00	UYU (858) 2020-01-23 16:55:34	[5251] Administracion Admin
10	1240	1st Pres	Purchase	100000*****-0037	251.00	USD (840) 2020-01-23 16:57:31	[5251] Administracion Admin
11	1240	1st Pres	Purchase	100000*****-8417	1,245.00	UYU (858) 2020-01-23 16:50:49	[5251] Administracion Admin
12	1240	1st Pres	Purchase	100000*****-8417	320.00	USD (840) 2020-01-23 16:51:42	[5251] Administracion Admin
13	1240	1st Pres	Purchase	100000*****-3511	1,800.00	UYU (858) 2020-01-22 16:21:17	[5732] Sucursal 1 Sucursal 1
14	1240	1st Pres	Purchase	100000*****-3511	2,300.00	UYU (858) 2020-01-23 17:45:43	[5732] Sucursal 1 Sucursal 1
15	1240	1st Pres	Purchase	100000*****-3511	1,458.00	UYU (858) 2020-01-23 17:46:27	[5732] Sucursal 1 Sucursal 1
16	1240	1st Pres	Purchase	100000*****-3511	720.00	USD (840) 2020-01-23 17:47:10	[5732] Sucursal 1 Sucursal 1
17	1240	1st Pres	Purchase	100000*****-3511	400.00	USD (840) 2020-01-23 17:47:54	[5732] Sucursal 1 Sucursal 1
18	1240	1st Pres	Purchase	100000*****-3511	6,000.00	UYU (858) 2020-01-23 17:49:51	[5732] Sucursal 1 Sucursal 1
19	1240	1st Pres	Purchase	100000*****-0037	2,500.00	USD (840) 2020-01-23 17:53:06	[5732] Sucursal 1 Sucursal 1
20	1240	1st Pres	Purchase	100000*****-8417	2,901.64	UYU (858) 2020-01-22 10:57:16	[5732] Sucursal 1 Sucursal 1
21	1240	1st Pres	Purchase	100000*****-8417	2,901.64	UYU (858) 2020-01-22 10:58:08	[5732] Sucursal 1 Sucursal 1
22	1240	1st Pres	Purchase	100000*****-3511	350.00	USD (840) 2020-01-23 16:47:53	[5812] Sucursal 1 Sucursal 1
23	1240	1st Pres	Purchase	100000*****-3511	2,580.00	UYU (858) 2020-01-23 16:48:40	[5812] Sucursal 1 Sucursal 1
24	1240	1st Pres	Purchase	100000*****-3511	490.00	USD (840) 2020-01-23 17:14:31	[5812] Sucursal 1 Sucursal 1
25	1240	1st Pres	Purchase	100000*****-0037	2,200.00	UYU (858) 2020-01-23 16:58:09	[5812] Sucursal 1 Sucursal 1
26	1240	1st Pres	Purchase	100000*****-0037	320.00	USD (840) 2020-01-23 16:58:52	[5812] Sucursal 1 Sucursal 1
27	1240	1st Pres	Purchase	100000*****-8417	203.12	USD (840) 2020-01-23 17:05:55	[5812] Sucursal 1 Sucursal 1
28	1240	1st Pres	Purchase	100000*****-8417	1,257.38	UYU (858) 2020-01-23 17:06:42	[5812] Sucursal 1 Sucursal 1
29	1240	1st Pres	Purchase	100000*****-3511	100.00	UYU (858) 2020-01-22 14:02:27	[5541] Sucursal 1 Sucursal 1
30	1240	1st Pres	Purchase	100000*****-0037	1,200.00	UYU (858) 2020-01-22 13:10:13	[5541] Sucursal 1 Sucursal 1

1240 Presentment  
First Presentment (DE3:000000 DE24:200 DE33:020659)  
Purchase 5251 HARDWARE STORES  
2020-01-23 17:09:25 1,758.00 (858 - UYU) Peso Uruguayo : URUGUAY  
Administracion Admin (ANDRES ECHEVESTE DC 27 VI) TREINTA Y T, URY

ENTER Detail F4 >>filter F5 Fields F8 Describe f Filter s Search C-f Quick Filter m Mark C-Sp Toggle Del Delete C-u Undelete C-a Add C-s Save C-x Export C  
M: 123456789 Marks: 1

Vemos también que en la barra de estado (la inferior) y sobre la izquierda, nos muestra el texto "Marks" y la cantidad de registros actualmente marcados, en este caso 1

Podemos también marcar un rango de registros. Por ejemplo, si queremos marcar los registros 10 al 14, nos paramos sobre el primero del rango (el registro 10) y presionamos la tecla "m" y luego la tecla "a". A continuación, nos movemos al ultimo registro del rango (el registro 14) y presionamos las teclas "m" y luego la "z". Vemos que quedan marcados todos los registros comprendidos en ese rango. Vemos también que el registro marcado anteriormente sigue como marcado, por lo que las acciones de marcar registros son aditivas, o sea, no borran otras marcas. Si queremos borrar las marcas, podemos hacerlo una por una posicionándonos sobre el registro y presionando nuevamente **Ctrl-Space** (es un toggle, o sea, cambia el estado de la marca del registro por el estado opuesto). También podemos presionar la tecla "-" para borrar todas las marcas de una vez

file20 -- 6,616,625 bytes, 10,990 records   EBCDIC   NORMAL   2020-01-27    43 records							MEM: (187 MB/338 MB)
Record list							
1	1644	File Header	PROD 2020-Jan-27	Proc.ID:00000020659	Seq.:17061	Clearing file: Member-generated	
2	1240	1st Pres	Purchase	100000*****-3511	4,520.00	UYU (858) 2020-01-23 17:03:09	[5251] Administracion Admin
3	1240	1st Pres	Purchase	100000*****-3511	1,235.00	UYU (858) 2020-01-23 17:03:59	[5251] Administracion Admin
4	1240	1st Pres	Purchase	100000*****-3511	425.00	USD (840) 2020-01-23 17:04:43	[5251] Administracion Admin
5	1240	1st Pres	Purchase	100000*****-3511	826.00	USD (840) 2020-01-23 17:05:40	[5251] Administracion Admin
6	1240	1st Pres	Purchase	100000*****-3511	1,758.00	UYU (858) 2020-01-23 17:09:25	[5251] Administracion Admin
7	1240	1st Pres	Purchase	100000*****-3511	1,350.00	UYU (858) 2020-01-23 17:37:13	[5251] Administracion Admin
8	1240	1st Pres	Purchase	100000*****-0037	782.00	UYU (858) 2020-01-23 16:52:57	[5251] Administracion Admin
9	1240	1st Pres	Purchase	100000*****-0037	1,240.00	UYU (858) 2020-01-23 16:55:34	[5251] Administracion Admin
10	1240	1st Pres	Purchase	100000*****-0037	251.00	USD (840) 2020-01-23 16:57:31	[5251] Administracion Admin
11	1240	1st Pres	Purchase	100000*****-8417	1,245.00	UYU (858) 2020-01-23 16:50:49	[5251] Administracion Admin
12	1240	1st Pres	Purchase	100000*****-8417	320.00	USD (840) 2020-01-23 16:51:42	[5251] Administracion Admin
13	1240	1st Pres	Purchase	100000*****-3511	1,800.00	UYU (858) 2020-01-22 16:21:17	[5732] Sucursal 1 Sucursal 1
14	1240	1st Pres	Purchase	100000*****-3511	2,300.00	UYU (858) 2020-01-23 17:45:43	[5732] Sucursal 1 Sucursal 1
15	1240	1st Pres	Purchase	100000*****-3511	1,458.00	UYU (858) 2020-01-23 17:46:27	[5732] Sucursal 1 Sucursal 1
16	1240	1st Pres	Purchase	100000*****-3511	720.00	USD (840) 2020-01-23 17:47:10	[5732] Sucursal 1 Sucursal 1
17	1240	1st Pres	Purchase	100000*****-3511	400.00	USD (840) 2020-01-23 17:47:54	[5732] Sucursal 1 Sucursal 1
18	1240	1st Pres	Purchase	100000*****-3511	6,000.00	UYU (858) 2020-01-23 17:49:51	[5732] Sucursal 1 Sucursal 1
19	1240	1st Pres	Purchase	100000*****-0037	2,500.00	USD (840) 2020-01-23 17:53:06	[5732] Sucursal 1 Sucursal 1
20	1240	1st Pres	Purchase	100000*****-8417	2,901.64	UYU (858) 2020-01-22 10:57:16	[5732] Sucursal 1 Sucursal 1
21	1240	1st Pres	Purchase	100000*****-8417	2,901.64	UYU (858) 2020-01-22 10:58:08	[5732] Sucursal 1 Sucursal 1
22	1240	1st Pres	Purchase	100000*****-3511	350.00	USD (840) 2020-01-23 16:47:53	[5812] Sucursal 1 Sucursal 1
23	1240	1st Pres	Purchase	100000*****-3511	2,580.00	UYU (858) 2020-01-23 16:48:40	[5812] Sucursal 1 Sucursal 1
24	1240	1st Pres	Purchase	100000*****-3511	490.00	USD (840) 2020-01-23 17:14:31	[5812] Sucursal 1 Sucursal 1
25	1240	1st Pres	Purchase	100000*****-0037	2,200.00	UYU (858) 2020-01-23 16:58:09	[5812] Sucursal 1 Sucursal 1
26	1240	1st Pres	Purchase	100000*****-0037	320.00	USD (840) 2020-01-23 16:58:52	[5812] Sucursal 1 Sucursal 1
27	1240	1st Pres	Purchase	100000*****-8417	203.12	USD (840) 2020-01-23 17:05:55	[5812] Sucursal 1 Sucursal 1
28	1240	1st Pres	Purchase	100000*****-8417	1,257.38	UYU (858) 2020-01-23 17:06:42	[5812] Sucursal 1 Sucursal 1
29	1240	1st Pres	Purchase	100000*****-3511	100.00	UYU (858) 2020-01-22 14:02:27	[5541] Sucursal 1 Sucursal 1
30	1240	1st Pres	Purchase	100000*****-0037	1,200.00	UYU (858) 2020-01-22 13:10:13	[5541] Sucursal 1 Sucursal 1

1240 Presentment  
First Presentment (DE3:000000 DE24:200 DE33:020659)  
Purchase 5732 ELECTRONIC SALES  
2020-01-23 17:45:43 2,300.00 (858 - UYU) Peso Uruguayo : URUGUAY  
Sucursal 1 Sucursal 1 (CALLE ARIEL) MINAS, URY

ENTER Detail F4 >>filter F5 Fields F8 Describe f Filter s Search C-f Quick Filter m Mark C-Sp Toggle Del Delete C-u Undelete C-a Add C-s Save C-x Export C  
M: 123456789 Marks: 5

Otra opción es utilizar el filtro para dejar visibles los registros que deseamos marcar, y presionar la tecla "+" que marcara todos los registros visibles. A continuación presionamos **Shift-F** para limpiar el filtro, pero las marcas permanecerán. Vamos a agregar los registros que habíamos encontrado con el importe 40.00

Primero, aplicamos el filtro y presionamos la tecla “+”, y a continuación Shift-F5

file20 -- 6,616,625 bytes, 10,990 records   EBCDIC   NORMAL   2020-01-27    43 records		Record list		MEM: (187 MB/338 MB)	
6 1240 1st Pres	Purchase	100000-*****-3511	1,758.00	UYU (858)	2020-01-23 17:09:25 [5251] Administracion Admin
7 1240 1st Pres	Purchase	100000-*****-3511	1,350.00	UYU (858)	2020-01-23 17:37:13 [5251] Administracion Admin
8 1240 1st Pres	Purchase	100000-*****-0037	782.00	UYU (858)	2020-01-23 16:52:57 [5251] Administracion Admin
9 1240 1st Pres	Purchase	100000-*****-0037	1,240.00	UYU (858)	2020-01-23 16:55:31 [5251] Administracion Admin
10 1240 1st Pres	Purchase	100000-*****-0037	251.00	USD (840)	2020-01-23 16:57:31 [5251] Administracion Admin
11 1240 1st Pres	Purchase	100000-*****-8417	1,245.00	UYU (858)	2020-01-23 16:50:49 [5251] Administracion Admin
12 1240 1st Pres	Purchase	100000-*****-8417	320.00	USD (840)	2020-01-23 16:51:42 [5251] Administracion Admin
13 1240 1st Pres	Purchase	100000-*****-3511	1,800.00	UYU (858)	2020-01-22 16:21:17 [5732] Sucursal 1 Sucursal 1
14 1240 1st Pres	Purchase	100000-*****-3511	2,300.00	UYU (858)	2020-01-23 17:45:43 [5732] Sucursal 1 Sucursal 1
15 1240 1st Pres	Purchase	100000-*****-3511	1,458.00	UYU (858)	2020-01-23 17:46:27 [5732] Sucursal 1 Sucursal 1
16 1240 1st Pres	Purchase	100000-*****-3511	720.00	USD (840)	2020-01-23 17:47:10 [5732] Sucursal 1 Sucursal 1
17 1240 1st Pres	Purchase	100000-*****-3511	400.00	USD (840)	2020-01-23 17:47:54 [5732] Sucursal 1 Sucursal 1
18 1240 1st Pres	Purchase	100000-*****-3511	6,000.00	UYU (858)	2020-01-23 17:49:51 [5732] Sucursal 1 Sucursal 1
19 1240 1st Pres	Purchase	100000-*****-0037	2,500.00	USD (840)	2020-01-23 17:53:01 [5732] Sucursal 1 Sucursal 1
20 1240 1st Pres	Purchase	100000-*****-8417	2,901.64	UYU (858)	2020-01-22 10:57:16 [5732] Sucursal 1 Sucursal 1
21 1240 1st Pres	Purchase	100000-*****-8417	2,901.64	UYU (858)	2020-01-22 10:58:01 [5732] Sucursal 1 Sucursal 1
22 1240 1st Pres	Purchase	100000-*****-3511	350.00	USD (840)	2020-01-23 16:47:53 [5812] Sucursal 1 Sucursal 1
23 1240 1st Pres	Purchase	100000-*****-3511	2,580.00	UYU (858)	2020-01-23 16:48:40 [5812] Sucursal 1 Sucursal 1
24 1240 1st Pres	Purchase	100000-*****-3511	499.00	USD (840)	2020-01-23 17:14:31 [5812] Sucursal 1 Sucursal 1
25 1240 1st Pres	Purchase	100000-*****-0037	2,200.00	UYU (858)	2020-01-23 16:58:09 [5812] Sucursal 1 Sucursal 1
26 1240 1st Pres	Purchase	100000-*****-0037	320.00	USD (840)	2020-01-23 16:58:52 [5812] Sucursal 1 Sucursal 1
27 1240 1st Pres	Purchase	100000-*****-8417	203.12	USD (840)	2020-01-23 17:05:55 [5812] Sucursal 1 Sucursal 1
28 1240 1st Pres	Purchase	100000-*****-8417	1,257.38	UYU (858)	2020-01-23 17:06:42 [5812] Sucursal 1 Sucursal 1
29 1240 1st Pres	Purchase	100000-*****-3511	100.00	UYU (858)	2020-01-22 14:02:27 [5541] Sucursal 1 Sucursal 1
30 1240 1st Pres	Purchase	100000-*****-0037	1,200.00	UYU (858)	2020-01-22 13:16:13 [5541] Sucursal 1 Sucursal 1
31 1240 1st Pres	Purchase	100000-*****-0037	40.00	UYU (858)	2020-01-22 14:03:01 [5541] Sucursal 1 Sucursal 1
32 1240 1st Pres	Purchase	100000-*****-0624	40.00	UYU (858)	2020-01-22 11:58:16 [5541] Sucursal 1 Sucursal 1
33 1240 1st Pres	Purchase	100000-*****-0516	40.00	UYU (858)	2020-01-22 12:04:32 [5541] Sucursal 1 Sucursal 1
34 1240 1st Pres	Purchase	100000-*****-0329	40.00	UYU (858)	2020-01-22 12:05:31 [5541] Sucursal 1 Sucursal 1
35 1740 Fee (Cust gen)	Fee (Credit to Tr		0.00	UYU (858)	

1644 Administrative	File Header (PROD)
File type: 002 - Clearing file: Member-generated	
Processor ID: 00000020659	File Reference Date: 2020-Jan-27 File sequence number: 17861

```
ENTER Detail F4 >>filter F5 Fields F8 Describe f Filter s Search C-f Quick Filter m Mark C-Sp Toggle Del Delete C-u Undelete C-a Add C-s Save C-x Export C
M: 123456789 Marks: 9
```

## Borrar registros

Presionando la tecla **Del** o **Ctrl-D** podemos eliminar registros del archivo. Si no tenemos ningún registro marcado, borraremos el registro sobre el que estemos posicionados, de lo contrario, se borrarán los registros marcados.

Antes de proceder al borrado, se nos presenta un cuadro de dialogo para confirmar la operación, mostrando ademas la cantidad de registros que serán afectados.

file20 -- 6,616,625 bytes, 10,990 records   EBCDIC   NORMAL   2020-01-27    43 records		Record list		MEM: (187 MB/338 MB)	
6 1240 1st Pres	Purchase	100000-*****-3511	1,758.00	UYU (858)	2020-01-23 17:09:25 [5251] Administracion Admin
7 1240 1st Pres	Purchase	100000-*****-3511	1,350.00	UYU (858)	2020-01-23 17:37:13 [5251] Administracion Admin
8 1240 1st Pres	Purchase	100000-*****-0037	782.00	UYU (858)	2020-01-23 16:52:57 [5251] Administracion Admin
9 1240 1st Pres	Purchase	100000-*****-0037	1,240.00	UYU (858)	2020-01-23 16:55:31 [5251] Administracion Admin
10 1240 1st Pres	Purchase	100000-*****-0037	251.00	USD (840)	2020-01-23 16:57:31 [5251] Administracion Admin
11 1240 1st Pres	Purchase	100000-*****-8417	1,245.00	UYU (858)	2020-01-23 16:50:49 [5251] Administracion Admin
12 1240 1st Pres	Purchase	100000-*****-8417	320.00	USD (840)	2020-01-23 16:51:42 [5251] Administracion Admin
13 1240 1st Pres	Purchase	100000-*****-3511	1,800.00	UYU (858)	2020-01-22 16:21:17 [5732] Sucursal 1 Sucursal 1
14 1240 1st Pres	Purchase	100000-*****-3511	2,300.00	UYU (858)	2020-01-23 17:45:43 [5732] Sucursal 1 Sucursal 1
15 1240 1st Pres	Purchase	100000-*****-3511	1,458.00	UYU (858)	2020-01-23 17:46:27 [5732] Sucursal 1 Sucursal 1
16 1240 1st Pres	Purchase	100000-*****-3511	720.00	USD (840)	2020-01-23 17:47:10 [5732] Sucursal 1 Sucursal 1
17 1240 1st Pres	Purchase	100000-*****-3511	400.00	USD (840)	2020-01-23 17:47:54 [5732] Sucursal 1 Sucursal 1
18 1240 1st Pres	Purchase	100000-*****-3511	6,000.00	UYU (858)	2020-01-23 17:49:51 [5732] Sucursal 1 Sucursal 1
19 1240 1st Pres	Purchase	100000-*****-0037	2,500.00	USD (840)	2020-01-23 17:53:06 [5732] Sucursal 1 Sucursal 1
20 1240 1st Pres	Purchase	100000-*****-8417	2,901.64	UYU (858)	2020-01-22 10:57:16 [5732] Sucursal 1 Sucursal 1
21 1240 1st Pres	Purchase				20-01-22 10:58:08 [5732] Sucursal 1 Sucursal 1
22 1240 1st Pres	Purchase				20-01-23 16:47:53 [5812] Sucursal 1 Sucursal 1
23 1240 1st Pres	Purchase				20-01-23 16:48:40 [5812] Sucursal 1 Sucursal 1
24 1240 1st Pres	Purchase				20-01-23 17:14:31 [5812] Sucursal 1 Sucursal 1
25 1240 1st Pres	Purchase				20-01-23 16:58:09 [5812] Sucursal 1 Sucursal 1
26 1240 1st Pres	Purchase				20-01-23 16:58:52 [5812] Sucursal 1 Sucursal 1
27 1240 1st Pres	Purchase				20-01-23 17:05:55 [5812] Sucursal 1 Sucursal 1
28 1240 1st Pres	Purchase	100000-*****-8417	1,257.38	UYU (858)	2020-01-23 17:06:42 [5812] Sucursal 1 Sucursal 1
29 1240 1st Pres	Purchase	100000-*****-3511	100.00	UYU (858)	2020-01-22 14:02:27 [5541] Sucursal 1 Sucursal 1
30 1240 1st Pres	Purchase	100000-*****-0037	1,200.00	UYU (858)	2020-01-22 13:16:13 [5541] Sucursal 1 Sucursal 1
31 1240 1st Pres	Purchase	100000-*****-0037	40.00	UYU (858)	2020-01-22 14:03:01 [5541] Sucursal 1 Sucursal 1
32 1240 1st Pres	Purchase	100000-*****-0624	40.00	UYU (858)	2020-01-22 11:58:16 [5541] Sucursal 1 Sucursal 1
33 1240 1st Pres	Purchase	100000-*****-0516	40.00	UYU (858)	2020-01-22 12:04:32 [5541] Sucursal 1 Sucursal 1
34 1240 1st Pres	Purchase	100000-*****-0329	40.00	UYU (858)	2020-01-22 12:05:31 [5541] Sucursal 1 Sucursal 1
35 1740 Fee (Cust gen)	Fee (Credit to Tr		0.00	UYU (858)	

Are you sure you want to DELETE 10 records ?

1644 Administrative	File Header (PROD)
File type: 002 - Clearing file: Member-generated	
Processor ID: 00000020659	File Reference Date: 2020-Jan-27 File sequence number: 17861

```
ENTER Detail F4 >>filter F5 Fields F8 Describe f Filter s Search C-f Quick Filter m Mark C-Sp Toggle Del Delete C-u Undelete C-a Add C-s Save C-x Export C
M: 123456789 Marks: 9
```

Si presionamos el botón de confirmación, esos registros serán eliminados y como recordatorio, la barra superior y el borde del área de información quedaran de color rojo, ademas que en la barra superior se nos indicara ahora la cantidad de registros presentes y la cantidad de registros borrados.

```

file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 33 records, 10 deleted
                                         MEM: (186 MB/338 MB)
                                         Record list
 1 1644 File Header      PROD 2020-Jan-27 Proc.ID:00000020659 Seq.:17061 Clearing file: Member-generated
 2 1240 1st Pres Purchase 100000*****-3511   4,520.00 UYU (858) 2020-01-23 17:03:09 [5251] Administracion Admin
 3 1240 1st Pres Purchase 100000*****-3511   1,235.00 UYU (858) 2020-01-23 17:03:59 [5251] Administracion Admin
 4 1240 1st Pres Purchase 100000*****-3511   425.00 USD (840) 2020-01-23 17:04:43 [5251] Administracion Admin
 5 1240 1st Pres Purchase 100000*****-3511   826.00 USD (840) 2020-01-23 17:05:40 [5251] Administracion Admin
 7 1240 1st Pres Purchase 100000*****-3511   1,350.00 UYU (858) 2020-01-23 17:37:13 [5251] Administracion Admin
 8 1240 1st Pres Purchase 100000*****-0037   782.00 UYU (858) 2020-01-23 16:52:57 [5251] Administracion Admin
 9 1240 1st Pres Purchase 100000*****-0037   1,240.00 UYU (858) 2020-01-23 16:55:34 [5251] Administracion Admin
15 1240 1st Pres Purchase 100000*****-3511   1,458.00 UYU (858) 2020-01-23 17:46:27 [5732] Sucursal 1 Sucursal 1
16 1240 1st Pres Purchase 100000*****-3511   720.00 USD (840) 2020-01-23 17:47:10 [5732] Sucursal 1 Sucursal 1
17 1240 1st Pres Purchase 100000*****-3511   400.00 USD (840) 2020-01-23 17:47:54 [5732] Sucursal 1 Sucursal 1
18 1240 1st Pres Purchase 100000*****-3511   6,000.00 UYU (858) 2020-01-23 17:49:51 [5732] Sucursal 1 Sucursal 1
19 1240 1st Pres Purchase 100000*****-0037   2,500.00 USD (840) 2020-01-23 17:53:06 [5732] Sucursal 1 Sucursal 1
20 1240 1st Pres Purchase 100000*****-8417   2,901.64 UYU (858) 2020-01-22 10:57:16 [5732] Sucursal 1 Sucursal 1
21 1240 1st Pres Purchase 100000*****-8417   2,901.64 UYU (858) 2020-01-22 10:58:08 [5732] Sucursal 1 Sucursal 1
22 1240 1st Pres Purchase 100000*****-3511   350.00 USD (840) 2020-01-23 16:47:53 [5812] Sucursal 1 Sucursal 1
23 1240 1st Pres Purchase 100000*****-3511   2,580.00 UYU (858) 2020-01-23 16:48:46 [5812] Sucursal 1 Sucursal 1
24 1240 1st Pres Purchase 100000*****-3511   490.00 USD (840) 2020-01-23 17:14:31 [5812] Sucursal 1 Sucursal 1
25 1240 1st Pres Purchase 100000*****-0037   2,200.00 UYU (858) 2020-01-23 16:58:09 [5812] Sucursal 1 Sucursal 1
26 1240 1st Pres Purchase 100000*****-0037   320.00 USD (840) 2020-01-23 16:58:52 [5812] Sucursal 1 Sucursal 1
27 1240 1st Pres Purchase 100000*****-8417   203.12 USD (840) 2020-01-23 17:05:55 [5812] Sucursal 1 Sucursal 1
28 1240 1st Pres Purchase 100000*****-8417   1,257.38 UYU (858) 2020-01-23 17:06:42 [5812] Sucursal 1 Sucursal 1
29 1240 1st Pres Purchase 100000*****-3511   100.00 UYU (858) 2020-01-22 14:02:27 [5541] Sucursal 1 Sucursal 1
30 1240 1st Pres Purchase 100000*****-0037   1,200.00 UYU (858) 2020-01-22 13:16:13 [5541] Sucursal 1 Sucursal 1
35 1740 Fee (Cust gen) Fee (Credit to Tr    0.00 UYU (858)
36 1740 Fee (Cust gen) Fee (Credit to Tr    0.00 UYU (858)
37 1740 Fee (Cust gen) Fee (Credit to Tr    0.00 UYU (858)
38 1740 Fee (Cust gen) Fee (Credit to Tr    0.00 UYU (858)
39 1740 Fee (Cust gen) Fee (Credit to Tr    0.00 USD (840)
40 1740 Fee (Cust gen) Fee (Credit to Tr    0.00 USD (840)

1644 Administrative
File Header (PROD)
File type: 002 - Clearing file: Member-generated
Processor ID: 00000020659 File Reference Date: 2020-Jan-27 File sequence number: 17061

ENTER Detail F4 >>filter F5 Fields F8 Describe f Filter s Search C-f Quick Filter m Mark C-Sp Toggle Del Delete C-u Undelete C-a Add C-s Save C-x Export C
M: 123456789 Marks: 9
6 fields, R#1 1/33 <1/4> as

```

Debemos tomar en cuenta que todas las acciones no son destructivas, o sea, nunca van a modificar el archivo original. Para guardar los cambios, debemos guardarlos en un nuevo archivo tal como veremos mas adelante.

## Recuperación de registros borrados

A pesar de requerir de una confirmación para eliminar registros, es posible que encontremos que alguno no debía ser borrado. En ese caso, y antes de salir de la aplicación, podemos recuperar los registros marcados como borrados presionando **Ctrl-U**

Eso nos lleva a una pantalla donde vemos todos los registros borrados, y presionando **ENTER** sobre cualquiera, lo desmarcamos y vuelve a estar accesible junto con el resto.

## Modificación de registros

Estando en la vista de detalle (presionando **ENTER** o **SPACE** sobre el registro deseado de la vista general), podemos modificar los registros de varias formas.

Lo primero que podemos hacer es borrar un campo. Tomemos como ejemplo este registro y borremos el campo **PDS1003**. Para ello nos posicionamos sobre este campo y presionamos la tecla **Del**

The screenshot shows a terminal window with a list of fields on the left and their current values on the right. A red rectangular box highlights a confirmation dialog in the center:

```

[DE 38 Approval Code          001160]
[DE 42 Card Acceptor ID Code 091278]
[DE 43 Card Acceptor Name/Location Administracion Admin\ANDRES ECHEVESTE DC 27 VI \TREI
[DE 49 Currency Code, Transaction 858] (858 - UYU) Peso Uruguayo : URUGUAY
[DE 63 Transaction Life Cycle ID .MCSJ7TD8A0123]
[DE 71 Message Number         1
[DE 94 Transaction Originator Institution
PDS 0023 Terminal Type      1
PDS 0148 Currency Exponents  1
PDS 0150 Business Activity   1
PDS 0159 Settlement Data     1
PDS 0165 Settlement Indicator 1
PDS 0777 Promotion Code      1
PDS 1001 Installment Plan Type 1
PDS 1002 Installment Number   01
PDS 1003 Installment Amount   000000175800
PDS 1004 Total Transaction Amount 000000175800
PDS 1011 Total Number of Installments 04

Are you sure you want to DELETE field 1003 ?
Delete Cancel

```

The dialog box contains the message "Are you sure you want to DELETE field 1003?", two buttons ("Delete" and "Cancel"), and the cursor is positioned over the "Delete" button.

Si presionamos el botón **Delete**, el campo sera eliminado del registro. Debemos tener en cuenta que no se nos permitirá eliminar campos que sean mandatorios, por ejemplo, si intentamos eliminar el campo **DE024**, no nos va a dejar.

```
file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records MEM: (187 MB/338 MB)

1240 Presentation
-DE 2 Primary Account Number (PAN) [1000000000233511]
-DE 3 Processing Code [000000]
-DE 4 Amount, Transaction [000000175800] 1,758.00
-DE 12 Date and Time, Local Transaction [200123170925]
-DE 22 Point of Service Data Code [M10101299001]
-DE 24 Function Code [200] First Presentmant
-DE 26 Card Acceptor Business Code (MCC) [5251] HARDWARE STORES
-DE 31 Acquirer Reference Data [2270072002700000000000769]
-DE 33 Forwarding Institution ID Code [020659]
-DE 38 Approval Code [001160]
-DE 42 Card Acceptor ID Code [091278]
-DE 43 Card Acceptor Name/Location [Administracion_Admin\ANDRES_ECHEVESTE_DC_27 VI]
-DE 49 Currency Code, Transaction [858] (858 - UYU) Peso Uruguayo : URUGUAY \TREI
-DE 63 Transaction Life Cycle ID [MCSJ7TD8A0123]
-DE 71 Message Number
-DE 94 Transaction Originator Institution
-PDS 0023 Terminal Type
-PDS 0148 Currency Exponents
-PDS 0158 Business Activity
-PDS 0159 Settlement Data
-PDS 0165 Settlement Indicator
-PDS 0777 Promotion Code
-PDS 1001 Installment Plan Type
-PDS 1002 Installment Number
-PDS 1004 Total Transaction Amount
-PDS 1011 Total Number of Installments

Field 024 cannot be deleted,
It is a mandatory field
OK 3 200126
[020]
[01]
[000000175800]
[04]

1240 .|....1....|....2....|....3....|....4....|....5....|....6....|....7....|....8....|....9....|....10....|....11....|....12....|....13....|....14
200

DE 24 Function Code (Length 3) (n) Size: 3

Esc Back Del Delete field F2 Show subfields F5 Filter by fields Ctrl-A Add Ctrl-E Edit F8 Describe
M: 0123456789 6 fields, R#6 6/43 <1/4> as
```

Otra cosa que podemos hacer es modificar el valor de un campo. Vamos a modificar el importe, de 1758,00 lo vamos a cambiar a 5100,00

Para eso nos posicionamos sobre el campo **DE4** y presionamos **Ctrl-E**

Modificamos el valor y presionamos **ENTER**, con lo que se guardara el cambio en el registro

DE 01 Transaction End Cycle 10  
DE 71 Message Number [0000006]  
DE 94 Transaction Originator Institution ID Code [020659]  
PDS 0023 Terminal Type  
PDS 0148 Currency Exponent Field 004: 00000510000  
PDS 0158 Business Activity  
PDS 0159 Settlement Data [ ] 3 200126 [ ]  
PDS 0165 Settlement Indicator [M]  
PDS 0722 Promotion Code [UDVCTA]

```
file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records MEM: (187 MB/338 MB)

# 1240 Presentment
DE 2 Primary Account Number (PAN) [1000000000233511]
DE 3 Processing Code [000000]
DE 4 Amount Transaction [00000510000] 5,100.00
DE 12 Date and Time, Local Transaction [200123170925]
DE 22 Point of Service Data Code [M10101299001]
DE 24 Function Code [200] First Presentment
DE 26 Card Acceptor Business Code (MCC) [5251] HARDWARE STORES
DE 28 Acceptor Reference Data [3370073002700000000760]
```

Vemos el nuevo valor, y ademas la descripción del campo cambia de color para indicarnos visualmente que el valor de ese campo fue modificado.

Otra cosa que podemos hacer es agregar un campo. Para ello presionamos **Ctrl-A** y se nos abre una lista de campos que podemos agregar (no se muestran los que ya están presentes en el registro).

Seleccionamos por ejemplo el 1003 (que casualmente fue el que habíamos borrado) y presionamos **ENTER**. Entonces, se nos presenta un dialogo donde tenemos la lista de campos a agregar, y a la derecha un lugar donde colocar el valor. Vemos que se nos indica el tipo de dato (en este caso numérico) y el largo del mismo (en este caso, 12).

```
file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records MEM: (187 MB/338 MB)

# 1240 Presentment
DE 2 Primary Account Number (PAN) [1000000000233511]
DE 3 Processing Code [000000]
DE 4 Amount, Transaction [000000510000] 5,100.00
DE 12 Date and Time, Local Transaction [200123170925]
DE 22 Point of Service Data Code [M10101299001]
DE 24 Function Code [200] First Presentment
DE 26 Card Acceptor Business Code (MCC) [5251] HARDWARE STORES
DE 31 Acquirer Reference Data [22700720027000000000769]
DE 33 Forwarding Institution ID Code [020659]
DE 38 Approval Code [001160]
DE 42 Card Acceptor ID Code [091278]
DE 43 Card Acceptor Name/Location [Administracion Admin\ANDRES ECHEVESTE DC 27 VI]
DE 49 Currency Code, Transaction [858] (858 - UYU) Peso Uruguayo : URUGUAY \TREI
DE 63 Transaction Life Cycle ID [MCSJ7TD8A0123]
DE 71 Message N
DE 94 Transacti
  PDS 0023 Termin [P051003 Installment Amount]
  PDS 0148 Curren
  PDS 0150 Busine
  PDS 0159 Settle
  PDS 0165 Settle
  PDS 0777 Promot
  PDS 1001 Instal
  PDS 1002 Installment Number [01]
  PDS 1004 Total Transaction Amount [000000175800]
  PDS 1011 Total Number of Installments [04]

# 1240 1240 Presentment
Purchase
First Presentment
2020-01-23 17:09:25      5,100.00 (858 - UYU) Peso Uruguayo : URUGUAY
```

Ahí colocamos el valor deseado, y cuando el criterio de validación se cumple, el recuadro (originalmente rojo) pasa a color verde, indicando que hemos ingresado un valor válido.

```

file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records           MEM: (187 MB/338 MB)
# 1240 Presentment
  DE 2 Primary Account Number (PAN)          [1000000000233511]
  DE 3 Processing Code                      [000000]
  DE 4 Amount Transaction                  [000000510000] 5,100.00
  DE 12 Date and Time, Local Transaction   [200123170925]
  DE 22 Point of Service Data Code         [M10101299001]
  DE 24 Function Code                     [200] First Presentment
  DE 26 Card Acceptor Business Code (MCC)  [5251] HARDWARE STORES
  DE 31 Acquirer Reference Data          [22700720072000000000769]
  DE 33 Forwarding Institution ID Code    [020659]
  DE 38 Approval Code                   [001160]
  DE 42 Card Acceptor ID Code            [091278]
  DE 43 Card Acceptor Name/Location     [Administracion Admin\ANDRES ECHEVESTE_DC_27_VI]
  DE 49 Currency Code, Transaction      [858] (858 - UYU) Peso Uruguayo : URUGUAY \TREI
  DE 63 Transaction Life Cycle ID       [.MCSJ7TD8A0123...]
  DE 71 Message N
  DE 94 Transaction Originator Institution ID Code
    PDS 0023 Terminal Type              [POI] POI terminal
    PDS 0148 Currency Exponents        [8582]
    PDS 0158 Business Activity         [75]
    PDS 0159 Settlement Data           [3] 200126
    PDS 0165 Settlement Indicator     [M]
    PDS 0777 Promotion code            [URYCTA]
    PDS 1001 Installment Plan Type    [020]
    PDS 1002 Installment Number       [01]
    PDS 1003 Installment Amount        [000000510000]
    PDS 1004 Total Transaction Amount  [000000175800]
    PDS 1011 Total Number of Installments [04]

  Ctrl-S to save
  Value (n/12): 000000510000
  200126

1240 1240 Presentment
Purchase
First Presentment
2020-01-23 17:09:25 5,100.00 (858 - UYU) Peso Uruguayo : URUGUAY

Esc Back Del Delete field F2 Show subfields F5 Filter by fields Ctrl-A Add Ctrl-E Edit F8 Describe
M: 0123456789
6 fields, R#6 6/43 <1/4> as

```

En este momento podemos presionar **Ctrl-S** para agregar el nuevo campo.

Nuevamente vemos que la barra superior queda en color rojo para indicar modificaciones en el archivo, y el recuadro de la vista de detalles queda de color amarillo para indicar modificaciones

```

file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records, 1 modified           MEM: (196 MB/338 MB)
# 1240 Presentment
  DE 2 Primary Account Number (PAN)          [1000000000233511]
  DE 3 Processing Code                      [000000]
  DE 4 Amount Transaction                  [000000510000] 5,100.00
  DE 12 Date and Time, Local Transaction   [200123170925]
  DE 22 Point of Service Data Code         [M10101299001]
  DE 24 Function Code                     [200] First Presentment
  DE 26 Card Acceptor Business Code (MCC)  [5251] HARDWARE STORES
  DE 31 Acquirer Reference Data          [22700720072000000000769]
  DE 33 Forwarding Institution ID Code    [020659]
  DE 38 Approval Code                   [001160]
  DE 42 Card Acceptor ID Code            [091278]
  DE 43 Card Acceptor Name/Location     [Administracion Admin\ANDRES ECHEVESTE_DC_27_VI]
  DE 49 Currency Code, Transaction      [858] (858 - UYU) Peso Uruguayo : URUGUAY \TREI
  DE 63 Transaction Life Cycle ID       [.MCSJ7TD8A0123...]
  DE 71 Message Number
  DE 94 Transaction Originator Institution ID Code
    PDS 0023 Terminal Type              [POI] POI terminal
    PDS 0148 Currency Exponents        [8582]
    PDS 0158 Business Activity         [75]
    PDS 0159 Settlement Data           [3] 200126
    PDS 0165 Settlement Indicator     [M]
    PDS 0777 Promotion code            [URYCTA]
    PDS 1001 Installment Plan Type    [020]
    PDS 1002 Installment Number       [01]
    PDS 1003 Installment Amount        [000000510000]
    PDS 1004 Total Transaction Amount  [000000175800]
    PDS 1011 Total Number of Installments [04]

  Ctrl-S to save
  Value (n/12): 000000510000
  200126

1240 1240 Presentment
Purchase
First Presentment
2020-01-23 17:09:25 5,100.00 (858 - UYU) Peso Uruguayo : URUGUAY

Esc Back Del Delete field F2 Show subfields F5 Filter by fields Ctrl-A Add Ctrl-E Edit F8 Describe
M: 0123456789
6 fields, R#6 6/43 <1/4> as

```

Volviendo a la vista principal, vemos que el registro 6 queda con una marca (el carácter #) para indicar que ese registro ha sido modificado.

file20 -- 6,616,625 bytes, 10,990 records   EBCDIC   NORMAL   2020-01-27    43 records, 1 modified							MEM: (196 MB/338 MB)
Record list							
1	1644	File Header	PROD 2020-Jan-27	Proc.ID:0000020659	Seq.:17061	Clearing file: Member-generated	
2	1240	1st Pres	Purchase	100000*****-3511	4,520.00	UYU (858)	2020-01-23 17:03:09 [5251] Administracion Admin
3	1240	1st Pres	Purchase	100000*****-3511	1,235.00	UYU (858)	2020-01-23 17:03:59 [5251] Administracion Admin
4	1240	1st Pres	Purchase	100000*****-3511	425.00	USD (840)	2020-01-23 17:04:43 [5251] Administracion Admin
5	1240	1st Pres	Purchase	100000*****-3511	826.00	USD (840)	2020-01-23 17:05:40 [5251] Administracion Admin
#	1240	1st Pres	Purchase	100000*****-3511	5,100.00	UYU (858)	2020-01-23 17:09:25 [5251] Administracion Admin
7	1240	1st Pres	Purchase	100000*****-3511	1,350.00	UYU (858)	2020-01-23 17:37:13 [5251] Administracion Admin
8	1240	1st Pres	Purchase	100000*****-0037	782.00	UYU (858)	2020-01-23 16:52:57 [5251] Administracion Admin
9	1240	1st Pres	Purchase	100000*****-0037	1,240.00	UYU (858)	2020-01-23 16:55:34 [5251] Administracion Admin
10	1240	1st Pres	Purchase	100000*****-0037	251.00	USD (840)	2020-01-23 16:57:31 [5251] Administracion Admin
11	1240	1st Pres	Purchase	100000*****-8417	1,245.00	UYU (858)	2020-01-23 16:50:49 [5251] Processor A
12	1240	1st Pres	Purchase	100000*****-8417	320.00	USD (840)	2020-01-23 16:51:42 [5251] Processor A
13	1240	1st Pres	Purchase	100000*****-3511	1,800.00	UYU (858)	2020-01-22 16:21:17 [5732] Alternate processor
14	1240	1st Pres	Purchase	100000*****-3511	2,300.00	UYU (858)	2020-01-23 17:45:43 [5732] Alternate processor
15	1240	1st Pres	Purchase	100000*****-3511	1,458.00	UYU (858)	2020-01-23 17:46:27 [5732] Alternate processor
16	1240	1st Pres	Purchase	100000*****-3511	720.00	USD (840)	2020-01-23 17:47:10 [5732] Alternate processor
17	1240	1st Pres	Purchase	100000*****-3511	400.00	USD (840)	2020-01-23 17:47:54 [5732] Alternate processor
18	1240	1st Pres	Purchase	100000*****-3511	6,000.00	UYU (858)	2020-01-23 17:49:51 [5732] Alternate processor
19	1240	1st Pres	Purchase	100000*****-0037	2,500.00	USD (840)	2020-01-23 17:53:06 [5732] Alternate processor

## Reemplazar valores

Presionando las teclas **Ctrl-R** se nos muestra un cuadro de dialogo donde podemos ingresar un valor a buscar y el valor por el que lo queremos reemplazar. El valor de búsqueda acepta expresiones regulares, y podemos también especificar en qué campos queremos realizar dicha búsqueda.

Tenemos este archivo:

file50 -- 20,383 bytes, 43 records   EBCDIC   NORMAL   2020-01-27    43 records							MEM: (1.8 MB/13 MB)
Record list							
1	1644	File Header	PROD 2020-Jan-27	Proc.ID:00000123670	Seq.:17061	Clearing file: Member-generated	
2	1240	1st Pres	Purchase	100000*****-3511	4,520.00	UYU (858)	2020-01-23 17:03:09 [5251] Processor A
3	1240	1st Pres	Purchase	100000*****-3511	1,235.00	UYU (858)	2020-01-23 17:03:59 [5251] Processor A
4	1240	1st Pres	Purchase	100000*****-3511	425.00	USD (840)	2020-01-23 17:04:43 [5251] Processor A
5	1240	1st Pres	Purchase	100000*****-3511	826.00	USD (840)	2020-01-23 17:05:40 [5251] Processor A
6	1240	1st Pres	Purchase	100000*****-3511	1,758.00	UYU (858)	2020-01-23 17:09:25 [5251] Processor A
7	1240	1st Pres	Purchase	100000*****-3511	1,350.00	UYU (858)	2020-01-23 17:37:13 [5251] Processor A
8	1240	1st Pres	Purchase	100000*****-0037	782.00	UYU (858)	2020-01-23 16:52:57 [5251] Processor A
9	1240	1st Pres	Purchase	100000*****-0037	1,240.00	UYU (858)	2020-01-23 16:55:34 [5251] Processor A
10	1240	1st Pres	Purchase	100000*****-0037	251.00	USD (840)	2020-01-23 16:57:31 [5251] Processor A
11	1240	1st Pres	Purchase	100000*****-8417	1,245.00	UYU (858)	2020-01-23 16:50:49 [5251] Processor A
12	1240	1st Pres	Purchase	100000*****-8417	320.00	USD (840)	2020-01-23 16:51:42 [5251] Processor A
13	1240	1st Pres	Purchase	100000*****-3511	1,800.00	UYU (858)	2020-01-22 16:21:17 [5732] Alternate processor
14	1240	1st Pres	Purchase	100000*****-3511	2,300.00	UYU (858)	2020-01-23 17:45:43 [5732] Alternate processor
15	1240	1st Pres	Purchase	100000*****-3511	1,458.00	UYU (858)	2020-01-23 17:46:27 [5732] Alternate processor
16	1240	1st Pres	Purchase	100000*****-3511	720.00	USD (840)	2020-01-23 17:47:10 [5732] Alternate processor
17	1240	1st Pres	Purchase	100000*****-3511	400.00	USD (840)	2020-01-23 17:47:54 [5732] Alternate processor
18	1240	1st Pres	Purchase	100000*****-3511	6,000.00	UYU (858)	2020-01-23 17:49:51 [5732] Alternate processor
19	1240	1st Pres	Purchase	100000*****-0037	2,500.00	USD (840)	2020-01-23 17:53:06 [5732] Alternate processor

1240 Presentment  
First Presentment (DE3:000000 DE24:200 DE33:020659)  
Purchase 5251 HARDWARE STORES  
2020-01-23 17:03:59 1,235.00 (858 - UYU) Peso Uruguayo : URUGUAY  
Processor A (ANDRES ECHEVESTE DC 27 VI) TREINTA Y T, URY

ENTER Detail F5 Fields F8 Desc f Filter s Srch C-f Quick Filter m Mark C-Sp Toggle Del Del C-u Undel C-a Add C-s Save C-x Export C-l Load C-r Replace  
M: 0123456789

R#3 3/43 <1/1 AS

Y queremos reemplazar el nombre del comercio de “Processor A” al valor “Main processor”. Presionamos **Ctrl-r**

file50 -- 20,383 bytes, 43 records   EBCDIC   NORMAL   2020-01-27    43 records							MEM: (1.8 MB/13 MB)
Record list							
1	1644	File Header	PROD 2020-Jan-27	Proc.ID:00000123670	Seq.:17061	Clearing file: Member-generated	
2	1240	1st Pres	Purchase	100000*****-3511	4,520.00	UYU (858)	2020-01-23 17:03:09 [5251] Processor A
3	1240	1st Pres	Purchase	100000*****-3511	1,235.00	UYU (858)	2020-01-23 17:03:59 [5251] Processor A
4	1240	1st Pres	Purchase	100000*****-3511	425.00	USD (840)	2020-01-23 17:04:43 [5251] Processor A
5	1240	1st Pres	Purchase	100000*****-3511	826.00	USD (840)	2020-01-23 17:05:40 [5251] Processor A
6	1240	1st Pres	Purchase	100000*****-3511	1,758.00	UYU (858)	2020-01-23 17:09:25 [5251] Processor A
7	1240	1st Pres	Purchase	100000*****-3511	1,350.00	UYU (858)	2020-01-23 17:37:13 [5251] Processor A
8	1240	1st Pres	Purchase	100000*****-0037	782.00	UYU (858)	2020-01-23 16:52:57 [5251] Processor A
9	1240	1st Pres	Purchase	100000*****-0037	1,240.00	UYU (858)	2020-01-23 16:55:34 [5251] Processor A
10	1240	1st Pres	Purchase	100000*****-0037	251.00	USD (840)	2020-01-23 16:57:31 [5251] Processor A
11	1240	1st Pres	Purchase	100000*****-8417	1,245.00	UYU (858)	2020-01-23 16:50:49 [5251] Processor A
12	1240	1st Pres	Purchase	100000*****-8417	320.00	USD (840)	2020-01-23 16:51:42 [5251] Processor A
13	1240	1st Pres	Purchase	100000*****-3511	1,800.00	UYU (858)	2020-01-22 16:21:17 [5732] Alternate processor
14	1240	1st Pres	Purchase	100000*****-3511	2,300.00	UYU (858)	2020-01-23 17:45:43 [5732] Alternate processor
15	1240	1st Pres	Purchase	100000*****-3511	1,458.00	UYU (858)	2020-01-23 17:46:27 [5732] Alternate processor
16	1240	1st Pres	Purchase	100000*****-3511	720.00	USD (840)	2020-01-23 17:47:10 [5732] Alternate processor
17	1240	1st Pres	Purchase	100000*****-3511	400.00	USD (840)	2020-01-23 17:47:54 [5732] Alternate processor
18	1240	1st Pres	Purchase	100000*****-3511	6,000.00	UYU (858)	2020-01-23 17:49:51 [5732] Alternate processor
19	1240	1st Pres	Purchase	100000*****-0037	2,500.00	USD (840)	2020-01-23 17:53:06 [5732] Alternate processor

1240 Presentment  
First Presentment (DE3:000000 DE24:200 DE33:020659)  
Purchase 5251 HARDWARE STORES  
2020-01-23 17:03:59 1,235.00 (858 - UYU) Peso Uruguayo : URUGUAY  
Processor A (ANDRES ECHEVESTE DC 27 VI) TREINTA Y T, URY

F8 Open field selector  
M: 0123456789

R#3 3/43 <1/1 AS

Presionando el botón “Start”, nos queda esto:

```

file50 -- 20,383 bytes, 43 records | EBCDIC | NORMAL | 2020-01-27 || 43 records, 11 modified
Record list
# 3 1240 1st Pres Purchase 100000*****-3511 1,235.00 UYU (858) 2020-01-23 17:03:59 [5251] Main processor
# 4 1240 1st Pres Purchase 100000*****-3511 425.00 USD (840) 2020-01-23 17:04:43 [5251] Main processor
# 5 1240 1st Pres Purchase 100000*****-3511 826.00 USD (840) 2020-01-23 17:05:40 [5251] Main processor
# 6 1240 1st Pres Purchase 100000*****-3511 1,758.00 UYU (858) 2020-01-23 17:09:25 [5251] Main processor
# 7 1240 1st Pres Purchase 100000*****-3511 1,350.00 UYU (858) 2020-01-23 17:37:13 [5251] Main processor
# 8 1240 1st Pres Purchase 100000*****-0037 782.00 UYU (858) 2020-01-23 16:52:57 [5251] Main processor
# 9 1240 1st Pres Purchase 100000*****-0037 1,240.00 UYU (858) 2020-01-23 16:55:34 [5251] Main processor
# 10 1240 1st Pres Purchase 100000*****-0037 251.00 USD (840) 2020-01-23 16:57:31 [5251] Main processor
# 11 1240 1st Pres Purchase 100000*****-8417 1,245.00 UYU (858) 2020-01-23 16:50:49 [5251] Main processor
# 12 1240 1st Pres Purchase 100000*****-8417 320.00 USD (840) 2020-01-23 16:51:42 [5251] Main processor
# 13 1240 1st Pres Purchase
# 14 1240 1st Pres Purchase
# 15 1240 1st Pres Purchase
# 16 1240 1st Pres Purchase
# 17 1240 1st Pres Purchase
# 18 1240 1st Pres Purchase
# 19 1240 1st Pres Purchase
# 20 1240 1st Pres Purchase 100000*****-8417 2,901.64 UYU (858) 2020-01-22 17:53:06 [5732] Alternate processor
# 21 1240 1st Pres Purchase 100000*****-8417 2,901.64 UYU (858) 2020-01-22 10:58:08 [5732] Alternate processor

1240 Presentment
First Presentment (DE3:000000 DE24:200 DE33:020659)
Purchase 5251 HARDWARE STORES
2020-01-23 17:03:59 1,235.00 (858 - UYU) Peso Uruguayo : URUGUAY
Main processor (ANDRES ECHEVESTE DC 27 VI) TREINTA Y T, URY

ENTER Detail F5 Fields F8 Desc f Filter s Srch C-f Quick Filter m Mark C-Sp Toggle Del Del C-u Undel C-a Add C-s Save C-x Export C-l Load C-r Replace
M: 0123456789 R#3 3/43 <1/1> A5

```

Vemos que hubieron 11 reemplazos, y la linea superior queda con fondo rojo para indicar que el archivo ha sido modificado. Vemos también sobre la izquierda los registros modificados indicados con el carácter "#" en color celeste.

Debemos recordar que para no perder los cambios, es necesario salvar el archivo.

Si, por ejemplo, queremos hacer el reemplazo solamente en algunos registros, los debemos marcar previamente. En este ejemplo, hemos marcado los registros 6, 8 y 9 (vemos que los numeros quedan en color amarillo) y reemplazamos el valor de "Main processor" por "Secondary processor" en esos registros:

```

file50 -- 20,383 bytes, 43 records | EBCDIC | NORMAL | 2020-01-27 || 43 records, 11 modified
Record list
# 1 1644 File Header PROD 2020-Jan-27 Proc.ID:00000123670 Seq.:17061 Clearing file: Member-generated
# 2 1240 1st Pres Purchase 100000*****-3511 4,520.00 UYU (858) 2020-01-23 17:03:09 [5251] Main processor
# 3 1240 1st Pres Purchase 100000*****-3511 1,235.00 UYU (858) 2020-01-23 17:03:59 [5251] Main processor
# 4 1240 1st Pres Purchase 100000*****-3511 425.00 USD (840) 2020-01-23 17:04:43 [5251] Main processor
# 5 1240 1st Pres Purchase 100000*****-3511 826.00 USD (840) 2020-01-23 17:05:40 [5251] Main processor
# 6 1240 1st Pres Purchase 100000*****-3511 1,758.00 UYU (858) 2020-01-23 17:09:25 [5251] Main processor
# 7 1240 1st Pres Purchase 100000*****-3511 1,350.00 UYU (858) 2020-01-23 17:37:13 [5251] Main processor
# 8 1240 1st Pres Purchase
# 9 1240 1st Pres Purchase
# 10 1240 1st Pres Purchase
# 11 1240 1st Pres Purchase
# 12 1240 1st Pres Purchase
# 13 1240 1st Pres Purchase
# 14 1240 1st Pres Purchase
# 15 1240 1st Pres Purchase
# 16 1240 1st Pres Purchase
# 17 1240 1st Pres Purchase
# 18 1240 1st Pres Purchase
# 19 1240 1st Pres Purchase

1240 Presentment
First Presentment (DE3:000000 DE24:200 DE33:020659)
Purchase 5251 HARDWARE STORES
2020-01-23 16:55:34 1,240.00 (858 - UYU) Peso Uruguayo : URUGUAY
Main processor (ANDRES ECHEVESTE DC 27 VI) TREINTA Y T, URY

F8 Open field selector
M: 0123456789 Marks: 3 R#9 9/43 <1/1> A5

```

Y el resultado es este:

```

file50 -- 20,416 bytes, 43 records | EBCDIC | NORMAL | 2020-01-27 || 43 records, 11 modified
Record list
1 1644 File Header      PROD 2020-Jan-27 Proc.ID:00000123670 Seq.:17061 Clearing file: Member-generated
# 2 1240 1st Pres        Purchase 100000*****-3511 4,520.00 UYU (858) 2020-01-23 17:03:09 [5251] Main processor
# 3 1240 1st Pres        Purchase 100000*****-3511 1,235.00 UYU (858) 2020-01-23 17:03:59 [5251] Main processor
# 4 1240 1st Pres        Purchase 100000*****-3511 425.00 USD (840) 2020-01-23 17:04:43 [5251] Main processor
# 5 1240 1st Pres        Purchase 100000*****-3511 826.00 USD (840) 2020-01-23 17:05:40 [5251] Main processor
# 6 1240 1st Pres        Purchase 100000*****-3511 1,758.00 UYU (858) 2020-01-23 17:09:25 [5251] Secondary processor
# 7 1240 1st Pres        Purchase 100000*****-3511 1,350.00 UYU (858) 2020-01-23 17:37:13 [5251] Main processor
# 8 1240 1st Pres        Purchase 100000*****-0037 782.00 UYU (858) 2020-01-23 16:52:57 [5251] Secondary processor
# 9 1240 1st Pres        Purchase 100000*****-0037 1,240.00 UYU (858) 2020-01-23 16:55:34 [5251] Secondary processor
# 10 1240 1st Pres       Purchase 100000*****-0037 251.00 USD (840) 2020-01-23 16:57:31 [5251] Main processor
# 11 1240 1st Pres       Purchase 100000*****-8417 1,245.00 UYU (858) 2020-01-23 16:58:49 [5251] Main processor
# 12 1240 1st Pres       Purchase 100000*****-8417 320.00 USD (840) 2020-01-23 16:51:42 [5251] Main processor
# 13 1240 1st Pres       Purchase 100000*****-3511 1,800.00 UYU (858) 2020-01-22 16:21:17 [5732] Alternate processor
# 14 1240 1st Pres       Purchase 100000*****-3511 2,300.00 UYU (858) 2020-01-23 17:45:43 [5732] Alternate processor
# 15 1240 1st Pres       Purchase 100000*****-3511 1,458.00 UYU (858) 2020-01-23 17:46:27 [5732] Alternate processor
# 16 1240 1st Pres       Purchase 100000*****-3511 720.00 USD (840) 2020-01-23 17:47:10 [5732] Alternate processor
# 17 1240 1st Pres       Purchase 100000*****-3511 400.00 USD (840) 2020-01-23 17:47:54 [5732] Alternate processor
# 18 1240 1st Pres       Purchase 100000*****-3511 6,000.00 UYU (858) 2020-01-23 17:49:51 [5732] Alternate processor
# 19 1240 1st Pres       Purchase 100000*****-0037 2,500.00 USD (840) 2020-01-23 17:53:06 [5732] Alternate processor

1240 Presentment
First Presentment (DE3:000000 DE24:200 DE33:020659)
Purchase 5251 HARDWARE STORES
2020-01-23 16:55:34 1,240.00 (858 - UYU) Peso Uruguayo : URUGUAY
Secondary processor (ANDRES ECHEVESTE DC 27 VI) TREINTA Y T, URY

ENTER Detail F4 >>filter F5 Fields F8 Desc f Filter s Srch C-f Quick Filter m Mark C-Sp Toggle Del Del C-u Undel C-a Add C-s Save C-x Export C-l Load
M: 0123456789 Marks: 3
R#9 9/43 <1/1> AS

```

## Función de auto guardado

Existe la función de auto guardado, la cual se activa automáticamente cuando el archivo contiene pocos registros (menos de 2000) pero que puede activarse o desactivarse en cualquier momento mediante la tecla **F6**. La razón es que el guardado puede llevar cierto tiempo, y si tenemos muchos registros esta operación puede interferir con el uso normal, por lo que para agilizar la respuesta de la aplicación, solo se aplica si los registros son pocos y el tiempo de guardado es despreciable.

El auto guardado se activa cuando hacemos modificaciones en el archivo, y se guarda automáticamente una copia donde al nombre del archivo original se le agrega “\_AUTOSAVE.ipm”

## Guardado manual

Luego de haber realizado modificaciones al archivo, podemos elegir descartarlas o guardar los cambios en un nuevo archivo.

Para esto, presionamos la función de guardado con **Ctrl-S** donde se nos muestra un dialogo en el cual podemos seleccionar el nombre del archivo a guardar, la codificación, el formato del archivo, y seleccionar si solamente queremos guardar los registros marcados o todos los registros (que no hayan sido borrados)

```

file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records, 1 modified
Record list
1 1644 File Header      PROD 2020-Jan-27 Proc.ID:00000020659 Seq.:17061 Clearing file: Member-generated
# 2 1240 1st Pres        Purchase 100000*****-3511 4,520.00 UYU (858) 2020-01-23 17:03:09 [5251] Administracion Admin
# 3 1240 1st Pres        Purchase 100000*****-3511 1,235.00 UYU (858) 2020-01-23 17:03:59 [5251] Administracion Admin
# 4 1240 1st Pres        Purchase 100000*****-3511 425.00 USD (840) 2020-01-23 17:04:43 [5251] Administracion Admin
# 5 1240 1st Pres        Purchase 100000*****-3511 826.00 USD (840) 2020-01-23 17:05:40 [5251] Administracion Admin
# 6 1240 1st Pres        Purchase 100000*****-3511 5,100.00 UYU (858) 2020-01-23 17:09:25 [5251] Administracion Admin
# 7 1240 1st Pres        Purchase 100000*****-3511 1,350.00 UYU (858) 2020-01-23 17:37:13 [5251] Administracion Admin
# 8 1240 1st Pres        Purchase 100000*****-0037 782.00 UYU (858) 2020-01-23 16:52:57 [5251] Administracion Admin
# 9 1240 1st Pres        Purchase 100000*****-0037 1,240.00 UYU (858) 2020-01-23 16:55:34 [5251] Administracion Admin
# 10 1240 1st Pres       Purchase 100000*****-0037 251.00 USD (840) 2020-01-23 16:57:31 [5251] Administracion Admin
# 11 1240 1st Pres       Purchase 100000*****-8417 1,245.00 UYU (858) 2020-01-23 16:58:49 [5251] Administracion Admin
# 12 1240 1st Pres       Purchase 100000*****-8417 320.00 USD (840) 2020-01-23 16:51:42 [5251] Administracion Admin
# 13 1240 1st Pres       Purchase 100000*****-3511 2,621.17 UYU (858) 2020-01-23 17:09:25 [5251] Administracion Admin
# 14 1240 1st Pres       Purchase 100000*****-3511 3,17:45:43 [5732] Administracion Admin
# 15 1240 1st Pres       Purchase 100000*****-0037 3,17:46:27 [5732] Administracion Admin
# 16 1240 1st Pres       Purchase 100000*****-0037 3,17:47:10 [5732] Administracion Admin
# 17 1240 1st Pres       Purchase 100000*****-0037 3,17:47:54 [5732] Administracion Admin
# 18 1240 1st Pres       Purchase 100000*****-0037 3,17:49:51 [5732] Administracion Admin
# 19 1240 1st Pres       Purchase 100000*****-0037 3,17:53:06 [5732] Administracion Admin
# 20 1240 1st Pres       Purchase 100000*****-0037 2,10:57:16 [5732] Administracion Admin
# 21 1240 1st Pres       Purchase 100000*****-0037 2,10:58:08 [5732] Administracion Admin
# 22 1240 1st Pres       Purchase 100000*****-0037 3,16:47:53 [5812] Administracion Admin
# 23 1240 1st Pres       Purchase 100000*****-0037 3,16:48:40 [5812] Administracion Admin
# 24 1240 1st Pres       Purchase 100000*****-0037 3,17:14:31 [5812] Administracion Admin
# 25 1240 1st Pres       Purchase 100000*****-0037 3,16:58:09 [5812] Administracion Admin
# 26 1240 1st Pres       Purchase 100000*****-0037 3,16:58:09 [5812] Administracion Admin
# 27 1240 1st Pres       Purchase 100000*****-8417 203.12 USD (840) 2020-01-23 17:05:55 [5812] Administracion Admin
# 28 1240 1st Pres       Purchase 100000*****-8417 1,257.38 UYU (858) 2020-01-23 17:06:42 [5812] Administracion Admin
# 29 1240 1st Pres       Purchase 100000*****-3511 100.00 UYU (858) 2020-01-22 14:02:27 [5541] Administracion Admin
# 30 1240 1st Pres       Purchase 100000*****-0037 1,200.00 UYU (858) 2020-01-22 13:16:13 [5541] Administracion Admin

1240 Presentment
First Presentment (DE3:000000 DE24:200 DE33:020659)
Purchase 5251 HARDWARE STORES
2020-01-23 17:09:25 5,100.00 (858 - UYU) Peso Uruguayo : URUGUAY
Administracion Admin (ANDRES ECHEVESTE DC 27 VI) TREINTA Y T, URY

ENTER Detail F4 >>filter F5 Fields F8 Desc f Filter s Srch C-f Quick Filter m Mark C-Sp Toggle Del Del C-u Undel C-a Add C-s Save C-x Export C-l Load
M: 0123456789 Marks: 3
6 fields, R#6 6/43 <1/4> as

```

El nombre por defecto es el nombre del archivo original, agregándole “\_SAVE.ipm”. El formato de codificación y de archivo sugeridos son los mismos que los del archivo original, pero pueden cambiarse, por lo que ademas de guardar las modificaciones, estaríamos efectuando una conversión de formatos.

Aquí ya podemos apreciar una de las utilidades de poder marcar registros.

## Exportar registros

Presionando **Ctrl-X** se nos permite exportar los registros marcados, o todos los registros en caso que no haya ninguno marcado.

Los formatos de exportación son los mismos que los del comando **EXPORT**, o sea, **CSV** o **HEX**. La forma de indicar el tipo de archivo que queremos generar es cambiando la extensión. Si al archivo generado le colocamos la extensión “.csv” entonces su contenido seran lineas en formato **CSV**, si la extensión es “.ckh” entonces el archivo generado sera en formato **HEX**

Como facilidad para cambiar la extensión, podemos presionar la tecla **F3** que ira cambiando entre las dos extensiones.

Si seleccionamos formato **CSV**, hay que tener en cuenta que si existe un filtro de campos seleccionado, solamente se incluirán en el archivo los campos activos

```

file20 -- 6,616,625 bytes, 10,990 records | EBCDIC | NORMAL | 2020-01-27 || 43 records
Record list
MEM: (188 MB/263 MB)
1 1644 File Header      PROD 2020-Jan-27 Proc.ID:00000020659 Seq.:17061 Clearing file: Member-generated
2 1240 1st Pres Purchase 100000*****-3511 4,520.00 UYU (858) 2020-01-23 17:03:09 [5251] Administracion Admin
3 1240 1st Pres Purchase 100000*****-3511 1,235.00 UYU (858) 2020-01-23 17:03:59 [5251] Administracion Admin
4 1240 1st Pres Purchase 100000*****-3511 425.00 USD (840) 2020-01-23 17:04:43 [5251] Administracion Admin
5 1240 1st Pres Purchase 100000*****-3511 826.00 USD (840) 2020-01-23 17:05:40 [5251] Administracion Admin
6 1240 1st Pres Purchase 100000*****-3511 1,758.00 UYU (858) 2020-01-23 17:09:25 [5251] Administracion Admin
7 1240 1st Pres Purchase 100000*****-3511 1,350.00 UYU (858) 2020-01-23 17:37:13 [5251] Administracion Admin
8 1240 1st Pres Purchase 100000*****-0037 782.00 UYU (858) 2020-01-23 16:52:57 [5251] Administracion Admin
9 1240 1st Pres Purchase 100000*****-0037 1,240.00 UYU (858) 2020-01-23 16:55:34 [5251] Administracion Admin
10 1240 1st Pres Purchase 100000*****-0037 251.00 USD (840) 2020-01-23 16:57:31 [5251] Administracion Admin
11 1240 1st Pres Purchase 100000*****-8417 1,245.00 UYU (858) 2020-01-23 16:59:49 [5251] Administracion Admin
12 1240 1st Pres Purchase 100000*****-8417 328.00 USD (840) 2020-01-23 16:51:42 [5251] Administracion Admin
13 1240 1st Pres Purchase 100000*****-3511 1,800.00 UYU (858) 2020-01-22 16:23:17 [5732] Sucursal 1 Sucursal 1
14 1240 1st Pres Purchase 100000*****-3511 2,300.00 UYU (858) 2020-01-23 17:49:43 [5732] Sucursal 1 Sucursal 1
15 1240 1st Pres Purchase 100000*****-3511 1,458.00 UYU (858) 2020-01-23 17:40:27 [5732] Sucursal 1 Sucursal 1
16 1240 1st Pres Purchase 100000*****-3511 720.00 USD (840) 2020-01-23 17:47:10 [5732] Sucursal 1 Sucursal 1
17 1240 1st Pres Purchase 100000*****-3511 400.00 USD (840) 2020-01-23 17:47:54 [5732] Sucursal 1 Sucursal 1
18 1240 1st Pres          Use extension to define format: .csv for CSV or .ckh for HEX [5732] Sucursal 1 Sucursal 1
19 1240 1st Pres          Export file name:file20.ckh [5732] Sucursal 1 Sucursal 1
20 1240 1st Pres          53:06 [5732] Sucursal 1 Sucursal 1
21 1240 1st Pres Purchase 100000*****-8417 57:16 [5732] Sucursal 1 Sucursal 1
22 1240 1st Pres Purchase 100000*****-3511 2,901.64 UYU (858) 2020-01-22 10:58:08 [5732] Sucursal 1 Sucursal 1
23 1240 1st Pres Purchase 100000*****-3511 350.00 USD (840) 2020-01-23 16:47:53 [5812] Sucursal 1 Sucursal 1
24 1240 1st Pres Purchase 100000*****-3511 2,580.00 UYU (858) 2020-01-23 16:48:40 [5812] Sucursal 1 Sucursal 1
25 1240 1st Pres Purchase 100000*****-0037 490.00 USD (840) 2020-01-23 17:14:31 [5812] Sucursal 1 Sucursal 1
26 1240 1st Pres Purchase 100000*****-0037 2,200.00 UYU (858) 2020-01-23 16:58:09 [5812] Sucursal 1 Sucursal 1
27 1240 1st Pres Purchase 100000*****-8417 320.00 USD (840) 2020-01-23 16:58:52 [5812] Sucursal 1 Sucursal 1
28 1240 1st Pres Purchase 100000*****-8417 203.12 USD (840) 2020-01-23 17:05:55 [5812] Sucursal 1 Sucursal 1
29 1240 1st Pres Purchase 100000*****-8417 1,257.38 UYU (858) 2020-01-23 17:00:42 [5812] Sucursal 1 Sucursal 1
30 1240 1st Pres Purchase 100000*****-3511 100.00 UYU (858) 2020-01-22 14:02:27 [5541] Sucursal 1 Sucursal 1
31 1240 1st Pres Purchase 100000*****-0037 1,200.00 UYU (858) 2020-01-22 13:16:13 [5541] Sucursal 1 Sucursal 1

1644 Administrative
File Header (PROD)
File type: 002 - Clearing file: Member-generated
Processor ID: 00000020659 File Reference Date: 2020-Jan-27 File sequence number: 17061

Esc Cancel export ENTER Start export F2 Select CSV format F3 Select HEX format
M: 0123456789
11 fields, R#1 1/43 <1/4> as

```

## Importación de registros

Al igual que con el comando **IMPORT**, podemos importar registros previamente exportados, tanto en formato **CSV** como **HEX**

Los archivos **CSV** que pueden ser importados, deben contener un export de registros enteros, es decir, haber sido exportados con todos su campos ya que un export con campos filtrados no sirve para ser importado.

Para ello, presionamos **Ctrl-L** y se nos mostrara un campo donde seleccionar uno de los archivos que contienen datos exportados, si es que hay alguno.

Al presionar **ENTER**, se agregan los registros desde el archivo y quedan colocados al final. Es necesario salvar el archivo para no perder estos cambios.

## Mark slots

En la barra inferior de estado, sobre la izquierda, podemos observar lo siguiente:

M: 0123456789

Este es un indicador de los diferentes

“Mark slots” y un indicador de su contenido. Disponemos de 10 diferentes “memorias” donde se guardan hasta 10 listas de registros marcados. Cada vez que marcamos registros, se guarda la lista de esos registros en el área activa. Por defecto el área activa es la 0, pero podemos seleccionar cualquiera de ellas.

La utilidad que presenta es tener hasta 10 grupos diferentes de marcas, por ejemplo cada una con resultados de diferentes búsquedas, y al cambiar de área recuperamos la lista de marcas.

Por ejemplo, podemos hacer 4 búsquedas diferentes y el resultado de cada una la almacenamos en un área diferente. Luego, podemos seleccionar las que queremos y exportar o guardar los registros correspondientes.

Para cambiar el área activa, basta con presionar la tecla “m” y a continuación una tecla del “0” al “9