

## Tabla de contenido

Usar entrada/salida de archivos .....	1
Usar Entrada/Salida de archivos: Introducción .....	1
Comprender los conceptos básicos de E/S de archivo.....	2
Usar el cuadro de diálogo E/S de archivo.....	4
Abrir un archivo para lectura o escritura .....	4
Muestra de código de Abrir archivo.....	5
Cerrar un archivo después de leer o escribir datos .....	6
Muestra de código de Cerrar archivo .....	7
Leer un carácter en un archivo .....	8
Muestra de código de Leer carácter.....	9
Leer una línea en un archivo .....	11
Muestra de código de Leer línea.....	12
Leer un bloque de texto en un archivo.....	18
Muestra de código de Leer bloque.....	19
Leer texto hasta un delimitador.....	22
Muestra de código de Leer hasta.....	23
Escribir un carácter en un archivo .....	25
Muestra de código de Escribir carácter.....	26
Escribir una línea en un archivo.....	28
Muestra de código de Escribir línea .....	29
Escribir un bloque de texto en un archivo.....	30
Muestra de código de Escribir bloque .....	32

Colocar un puntero al principio de un archivo.....	33
Muestra de código de Regresar al inicio .....	34
Guardar la posición actual del puntero de archivo.....	35
Muestra de código de Guardar posición de archivo .....	36
Recuperar la posición del puntero de archivo guardado.....	38
Muestra de código de Recuperar posición del archivo.....	39
Copiar un archivo.....	40
Muestra de código de Copiar archivo.....	41
Mover un archivo .....	43
Muestra de código de Mover archivo .....	44
Suprimir un archivo.....	46
Muestra de código de Suprimir archivo.....	47
Comprobar la existencia de un archivo.....	48
Muestra de código de la opción El archivo ya existe.....	49
Mostrar un cuadro de diálogo Archivo .....	49
Muestra de código de un cuadro de diálogo Archivo .....	51
Comprobar si se ha llegado al final de un archivo o de una línea .....	52
Muestra de código de EOF y EOL .....	52

# Usar entrada/salida de archivos

---

## Usar Entrada/Salida de archivos: Introducción

En este capítulo se explica cómo efectuar la entrada y la salida de información a y desde las rutinas de medición. Las opciones de menú disponibles permiten abrir archivos en modo de lectura o escritura. Estos archivos se pueden utilizar luego para escribir o leer datos. Los comandos de E/S de archivos permiten leer los datos de archivos externos que se utilizarán en una rutina de medición. También se puede escribir información sobre las mediciones y tolerancias en los archivos con estos comandos. Con estos comandos también puede llevar a cabo otras operaciones con archivos.

En este capítulo se explica con detalle estas operaciones de E/S de archivos y se incluyen ejemplos funcionales para cada una de las diversas operaciones. En estos ejemplos se utilizan elementos tratados en los capítulos "Ramificación mediante control de flujo" y "Usar expresiones y variables".

En este tema se tratan los siguientes temas principales:

- Comprender los conceptos básicos de E/S de archivo
- Usar el cuadro de diálogo E/S de archivo
- Abrir un archivo para lectura o escritura
- Cerrar un archivo después de leer o escribir datos
- Leer un carácter en un archivo
- Leer una línea en un archivo
- Leer un bloque de texto en un archivo
- Leer texto hasta un delimitador
- Escribir un carácter en un archivo
- Escribir una línea en un archivo
- Escribir un bloque de texto en un archivo
- Colocar un puntero al principio de un archivo
- Guardar la posición actual del puntero de archivo
- Recuperar la posición del puntero de archivo guardado

- Copiar un archivo
- Mover un archivo
- Suprimir un archivo
- Comprobar la existencia de un archivo
- Mostrar un cuadro de diálogo Archivo
- Comprobar si se ha llegado al final de un archivo o de una línea

### Comandos en modo Comando después de los comentarios

Dado que muchos de los ejemplos de código de este capítulo utilizan comandos `COMENTARIO`, tenga en cuenta lo siguiente:



Después de insertar un comentario de PC-DMIS, para introducir más comandos de PC-DMIS en modo Comando primero debe pulsar Intro *dos veces* después del comando `COMENTARIO`. Con ello se indica a PC-DMIS que ya no desea añadir texto al comentario, pero que está listo para añadir un nuevo comando.

---

## Comprender los conceptos básicos de E/S de archivo

### Comprobar la existencia de un archivo:

Para todas las operaciones de E/S de archivos, probablemente resultará conveniente comprobar primero si los archivos existen. Esto debería ponerse en un bucle SI / ENTONCES, de modo que si la comprobación falla, se lo pueda notificar al usuario. Al escribir en un archivo, debe crear primero el archivo en el entorno de Windows.

Consulte el tema "Comprobar la existencia de un archivo".

### Abrir y cerrar archivos:

Para realizar operaciones que lean o escriban en los archivos, deberá abrirlos primero en los procesos de su sistema. Esto se hace asignando el archivo a una variable llamada puntero de archivo. Al abrir un archivo puede especificar si se abre para lectura, escritura (sobrescritura) o anexas. Una vez abierto, puede leer el archivo o escribir en él. Cuando haya acabado de trabajar con un archivo, deberá cerrar el puntero de archivo; así se cierra el archivo y otros procesos del sistema pueden acceder a él. No es posible abrir archivos que ya ha abierto otro proceso.

## Usar entrada y salida de archivos

Consulte los temas "Abrir un archivo para lectura o escritura" y "Cerrar un archivo después de leer o escribir datos".

### Punteros de archivo y posiciones:

Los punteros de archivo son variables que apuntan a un archivo. Almacenan el nombre y la posición de un archivo abierto, y posteriormente se utilizan para leer ese archivo o escribir en él. Una vez que se ha abierto un archivo y se ha asignado a un puntero de archivo, el puntero actúa igual que el cursor en un procesador de textos. Indican dónde se está leyendo o escribiendo dentro del archivo.

- Si va a realizar un anexo a un archivo, el puntero de archivo suele encontrarse al final del archivo.
- Si va a leer o sobrescribir un archivo, el puntero de archivo normalmente deberá situarse al principio del archivo.

### Uso de delimitadores al escribir o leer datos

*Cuando vaya a escribir datos*, considere la posibilidad de utilizar delimitadores para separar componentes de datos. Así se facilita el proceso de lectura de los datos por parte de la rutina de medición. Un delimitador puede ser cualquier carácter o cadena de caracteres. Por ejemplo, suponga que tiene un punto denominado PNT1 con los valores medidos X, Y y Z de 2.5,4.3,6.1. Estos valores pueden escribirse fácilmente separándolos con un delimitador de comas en un archivo de datos con un código semejante al siguiente:



```
ARCHIVO/ESCRIBIR LÍNEA,FPTR,PNT1.X + "," + PNT1.Y +  
"," + PNT1.Z
```

*Cuando vaya a leer datos*, puede separar los datos entrantes basados en un delimitador especificado y colocar los datos en variables para manipularlos posteriormente. Por ejemplo, suponga que desea leer los mismos valores X, Y y Z indicados anteriormente. Los valores deberán aparecer en una sola línea de texto como la siguiente: 2.5,4.3,6.1. Puede separar el texto en la coma y colocar esos valores en las variables correspondientes mediante una línea de código parecida a la siguiente:



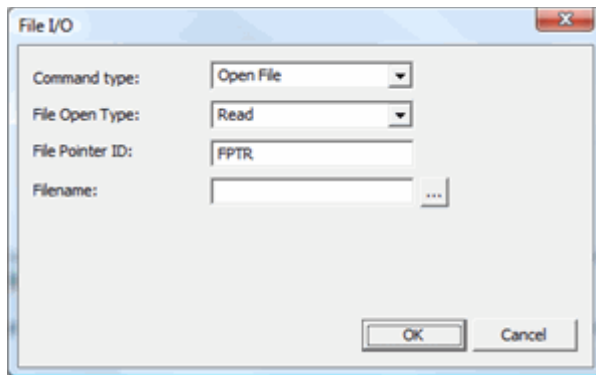
```
V1=ARCHIVO/LEER  
LÍNEA,FPTR,{ValX}+","+"{ValY}+","+"{ValZ}
```

Entonces puede utilizar `ValX`, `ValY` y `ValZ` como variables normales en su rutina de medición. El resultado de ello es: `ValX = 2,5`, `ValY = 4,3` y `ValZ = 6,1`.

---

## Usar el cuadro de diálogo E/S de archivo

Todos los comandos de E/S de archivo se insertan inicialmente en la rutina de medición seleccionando la opción de menú de E/S de archivo correspondiente (seleccione **Insertar | Comando E/S de archivos** en el menú). Una vez que haya un comando en la ventana de edición, puede pulsar F9 en el comando para acceder a su correspondiente cuadro de diálogo **E/S de archivo**.



*Cuadro de diálogo E/S de archivo*

Este cuadro de diálogo proporciona una manera visual de editar el comando de E/S de archivo *actual*. También puede modificar un comando en la ventana de edición utilizando los métodos descritos en el capítulo "Usar la ventana de edición".

No debe utilizar este cuadro de diálogo para insertar comandos de E/S de archivo *nuevos*. Para ello debe seleccionar la opción de menú correspondiente o escribir los comandos directamente en la ventana de edición.

---

## Abrir un archivo para lectura o escritura

La opción de menú **Insertar | Comando E/S de archivos | Abrir archivo** permite introducir un comando en la ventana de edición para abrir un archivo de su equipo durante la ejecución de la rutina de medición.

Es posible abrir archivos para añadir y guardar información, o simplemente para ver su contenido.

Este comando requiere la siguiente sintaxis en la ventana de edición:



```
<puntero_archivo>  
=Archivo/Abrir,<nombre_archivo>,<modo_apertura>
```

A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

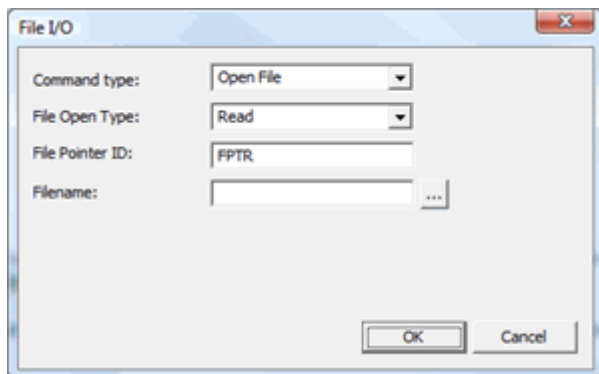
**<puntero\_archivo>**: ID elegida por el usuario para el puntero de archivo que permite acceder al archivo abierto. Esta ID se emplea para hacer referencia al archivo abierto en otros comandos de E/S de archivos.

**<nombre\_archivo>**: Nombre del archivo que se desea abrir.

**<modo\_apertura>**: Este es el modo en el que debe abrirse el archivo. Los archivos se pueden abrir en los modos siguientes: lectura, escritura o anexar.

Para acceder al cuadro de diálogo asociado con este comando de E/S de archivos:

1. Abra la ventana de edición.
2. Coloque el cursor en el comando Abrir archivo
3. Pulse F9.



## Muestra de código de Abrir archivo



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

Este código abre un archivo llamado TEST.TXT para leer, escribir o anexar. Almacena el nombre de archivo en un puntero de archivo denominado `FPTR`.



```
FPTR=ARCHIVO/ABRIR,C:\PCDMIWS\TEST.TXT,LEER
FPTR=ARCHIVO/ABRIR,C:\PCDMIWS\TEST.TXT,ESCRIBIR
FPTR=ARCHIVO/ABRIR,C:\PCDMIWS\TEST.TXT,ANEXAR
```

Tenga en cuenta que puede utilizar un comentario de entrada para tomar la ruta de acceso completa como entrada y utilizarla en un comando `ARCHIVO/ABRIR`. También puede hacer lo mismo utilizando el comando `ARCHIVO/DIÁLOGO`. Observe los siguientes ejemplos:



```
C1=COMENTARIO/ENTRADA, escriba la ruta de acceso
completa y el nombre del archivo.
V1=ARCHIVO/DIALOGO ELIJA UN ARCHIVO PARA ABRIR
FPTR=ARCHIVO/ABRIR,C1.INPUT,LEER
FPTR=ARCHIVO/ABRIR,V1,LEER
```

Consulte el tema "Mostrar un cuadro de diálogo Archivo".

---

## Cerrar un archivo después de leer o escribir datos

La opción de menú **Insertar | Comando E/S de archivos | Cerrar archivo** permite introducir un comando en la ventana de edición para cerrar un archivo abierto durante la ejecución de la rutina de medición. El cierre de archivos libera los recursos que estos emplean cuando están abiertos y registra en disco los cambios efectuados.

Este comando requiere la siguiente sintaxis en la ventana de edición:

```
Archivo/Cerrar, <puntero_archivo>,<modo_cerrar>
```

A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

### <puntero\_archivo>

ID utilizada para identificar el archivo, creada cuando éste se abre.

### <modo\_cerrar>

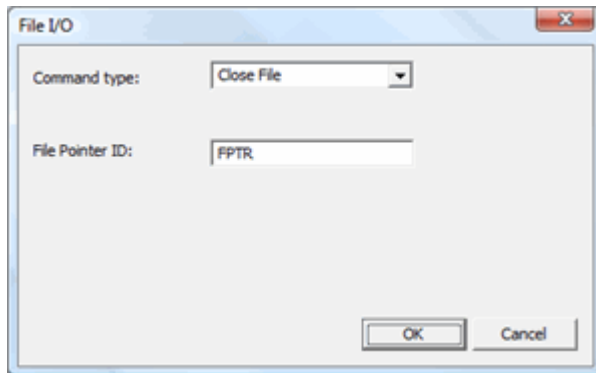
Este parámetro tiene dos opciones: MANTENER y SUPRIMIR. Con MANTENER, PC-DMIS sólo cierra el archivo indicado en el puntero de archivo. Con SUPRIMIR, PC-DMIS cierra el archivo y lo suprime.



## Usar entrada y salida de archivos

Para acceder al cuadro de diálogo asociado con este comando de E/S de archivos:

1. Abra la ventana de edición.
2. Coloque el cursor en el comando Cerrar archivo.
3. Pulse F9.



## Muestra de código de Cerrar archivo



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

Este código cierra el archivo asignado al puntero de archivo `FPTR`:



```
ARCHIVO/CERRAR, FPTR, MANTENER
```

Este código, que utiliza el parámetro `SUPRIMIR`, cierra y suprime el archivo asignado a `FPTR`:



```
ARCHIVO/CERRAR, FPTR, SUPRIMIR
```

## Leer un carácter en un archivo

La opción de menú **Insertar | Comando E/S de archivos | Comandos de lectura | Leer carácter** introduce un comando en la ventana de edición para leer un solo carácter del archivo especificado en el campo de puntero al archivo (fíjese en la sintaxis del ejemplo que sigue). Asigna dicho carácter a la variable especificada en el campo de nombre de variable.

Este comando requiere la siguiente sintaxis en la ventana de edición:



```
<nombre_variable> = Archivo/Leer  
carácter,<puntero_archivo>
```

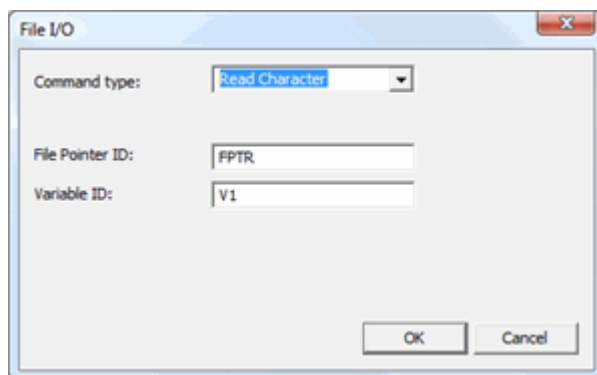
A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

**<puntero\_archivo>**: ID utilizada para abrir el archivo.

**<nombre\_variable>**: Nombre de la variable donde se almacenará el carácter.

Para acceder al cuadro de diálogo asociado con este comando de E/S de archivos:

1. Abra la ventana de edición.
2. Coloque el cursor en el comando Leer carácter.
3. Pulse F9.



## Muestra de código de Leer carácter



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

Observe el ejemplo siguiente que lee una línea de un archivo de datos carácter por carácter hasta que encuentra un espacio.



```
V1=ARCHIVO/EXISTE,test.txt
SI/V1<>0
    COMENTARIO/OPER,Se ha podido leer en un archivo
    de datos. Haga clic en Aceptar para continuar.
    ASIGN/V3=""
    FPTR=ARCHIVO/ABRIR,D:\Archivos de
    programa\pcdmis35\test.txt,LEER
    HACER/
        V2=ARCHIVO/LEER CARÁCTER,FPTR
        ASIGN/V3=V3+V2
    HASTA/V2==" "
    ARCHIVO/CERRAR,FPTR
    COMENTARIO/OPER,"La primera palabra de una línea
    de texto del archivo es: "+V3
TERMINAR SI/
O BIEN/
    COMENTARIO/OPER,No se ha podido leer en el
    archivo de datos. La rutina se cerrará.
    IR_A/FIN
TERMINAR O BIEN/
FIN=ETIQUETA/
ROUTINA/FIN
```

### Explicación de los códigos

#### **V1=ARCHIVO/EXISTE**

Esta línea comprueba si existe el archivo especificado. Para que este código funcione, el archivo debe estar colocado en el directorio donde reside PC-DMIS; de lo contrario, la línea que contiene el archivo también debe contener la ruta de acceso completa del archivo. **V1** recibe el resultado de la comprobación del archivo. Si existe, es un valor distinto de cero; de lo contrario, es 0.

#### **SI/V1<>0**

SI/V1<>0 Esta línea toma el valor de `V1` y comprueba si da como resultado un valor distinto de cero. Si es así, aparece un comentario que indica que está listo para comenzar el proceso de lectura. Si es igual a cero, la rutina de medición finaliza.

**ASIGN/V3=""**

Esta línea crea una cadena vacía y la asigna a `V3`. El código utiliza esta variable para crear una cadena a partir de los caracteres individuales leídos. Si no se crea la cadena vacía, `V3` tiene su valor por omisión, que es 0.

**FPTR=ARCHIVO/ABRIR**

Esta línea abre el archivo especificado para lectura y se lo asigna al puntero de archivo por omisión `FPTR`.

**HACER**

Esta línea inicia un bucle `HACER/HASTA`. Delimita el código `ARCHIVO/LEER CARÁCTER` de modo que la lectura se realice de forma continuada carácter por carácter. El bucle sale siempre que lee un carácter de espacio.

**V2=ARCHIVO/LEER CARÁCTER, FPTR**

Esta línea lee un carácter del archivo abierto vinculado al puntero de archivo, `FPTR`. El carácter se almacena en la variable `V2`.

**ASIGN/V3=V3+V2**

Esta línea utiliza la variable vacía `V3`, concatena la cadena `V3` con `V2` y luego reasigna el valor a `V3`. De este modo, con las ejecuciones siguientes del bucle `HACER/HASTA`, se añade un carácter más a `V3`.

**HASTA/V2==" "**

Esta línea finaliza el bucle `HACER/HASTA` cuando el código `ARCHIVO/LEER CARÁCTER` encuentra un carácter de espacio en el archivo abierto.

**ARCHIVO/CERRAR, FPTR**

Esta línea finaliza el archivo de datos abierto; así pueden acceder a él otros procesos del sistema. El resto del código termina de ejecutarse y muestra la primera palabra del archivo de datos en un comentario de operador.

## Leer una línea en un archivo

La opción de menú **Insertar | Comando E/S de archivos | Comandos de lectura | Leer línea** introduce un comando en la ventana de edición para leer, durante la ejecución, una línea del archivo especificado. Este comando establece la variable especificada por la ID en 1 (verdadero) o 0 (falso) para indicar, respectivamente, el éxito o el fracaso de la llamada. La expresión requerida por este comando puede utilizarse para delimitar la línea leída y para poblar automáticamente las variables y referencias con datos leídos del archivo. La información del archivo de entrada se lee hasta el siguiente carácter de retorno de carro.

Este comando requiere la siguiente sintaxis en la ventana de edición:



```
<nombre_variable> = Archivo/Leer  
línea,<puntero_archivo>,<expresión>
```

A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

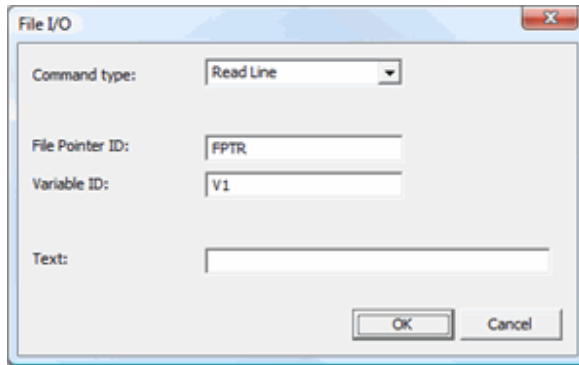
**<nombre\_variable>**: Nombre de la variable donde se almacenará el resultado que indica el éxito o el fracaso del comando Leer línea. Devuelve "OK" o "EOF".

**<puntero\_archivo>**: Nombre especificado para el puntero al abrirse el archivo.

**<expresión>**: Variables de destino para los datos de entrada. Los datos de entrada pueden delimitarse con texto para facilitar el análisis sintáctico de las líneas de datos entrantes. Las variables y referencias a elementos deben encerrarse entre llaves.

Para acceder al cuadro de diálogo asociado con este comando de E/S de archivos:

1. Abra la ventana de edición.
2. Coloque el cursor en el comando Leer línea.
3. Pulse F9.



## Muestra de código de Leer línea



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

Observe el ejemplo siguiente que lee una línea en un archivo de datos, línea a línea, hasta que el comando **ARCHIVO/LEER LÍNEA** encuentra una línea vacía. La rutina de medición mostrará entonces el bloque de texto resultante y se cerrará.

```
.
.
V1      =ARCHIVO/EXISTE,D:\HEXAGON\PCDMIS
FILES\BASIC_SCRIPTS\TEST.TXT
        SI/V1<>0
            COMENTARIO/OPER,NO,PANTALLA
COMPLETA=NO,CONTINUAR AUTOMÁTICAMENTE=NO,OVC=NO,
        Se ha podido leer en un archivo de datos. Haga
clic en Aceptar para continuar.
        ASIGN/V3=""
FPTR    =ARCHIVO/ABRIR,D:\HEXAGON\PCDMIS
FILES\BASIC_SCRIPTS\TEST.TXT,READ
        HACER/
V2      =ARCHIVO/LEER LÍNEA,FPTR,{LÍNEA}
        ASIGN/V3=V3+LÍNEA
        COMENTARIO/OPER,NO,PANTALLA
COMPLETA=NO,CONTINUAR AUTO=NO,OVC=NO,
        "El valor actual de la variable V3 es: "+V3
        HASTA/V2=="EOF"
        ARCHIVO/CERRAR,FPTR,SUPRIMIR
        COMENTARIO/OPER,NO,PANTALLA
COMPLETA=NO,CONTINUAR AUTOMÁTICAMENTE=NO,OVC=NO,
        "El bloque de texto es el siguiente: "+V3
        TERMINAR SI/
        OBIEN/
            COMENTARIO/OPER,NO,PANTALLA
COMPLETA=NO,CONTINUAR AUTOMÁTICAMENTE=NO,OVC=NO,
        No se ha podido leer en el archivo de datos. La
rutina se cerrará.
        IR_A/FIN
        TERMINAR O BIEN/
FIN      =ETIQUETA/
.
.
.
```

### Explicación de los códigos

Una gran parte de este código es similar al que se ha explicado en "Muestra de código de Leer carácter". A continuación se describe únicamente un código especial de este ejemplo.

#### HACER

Esta línea inicia un bucle HACER/HASTA. Delimita el código ARCHIVO/LEER LÍNEA de modo que la lectura se realice de forma continuada línea a línea. El bucle sale cuando llega al final del archivo.

**V2=ARCHIVO/LEER LÍNEA,FPTR,{LÍNEA}**

Esta línea lee todo el texto hasta que encuentra un retorno de carro. Sin embargo, en lugar de almacenar el texto en V2, como haría ARCHIVO/LEER CARÁCTER, este código actúa de forma diferente.

- En este caso V2 devuelve dos valores: "OK" o "EOF". "OK" si todavía queda una línea por leer. "EOF" si se ha llegado al final del archivo.
- El código {LÍNEA} es una variable introducida por el usuario que almacena el texto real. Se escribe entre llaves para indicar a PC-DMIS que se trata de una variable y no de una parte de cualquier texto delimitador. Sin las llaves, PC-DMIS buscaría una cadena de caracteres en el archivo llamada "LÍNEA" y devolvería sólo el texto posterior a "LÍNEA" y anterior al retorno de carro.

**ASIGN/V3=V3+LÍNEA**

Esta línea utiliza la variable vacía V3, concatena la cadena V3 con LÍNEA y luego reasigna el valor concatenado a V3. De este modo, con las ejecuciones siguientes del bucle HACER/HASTA, se añade una línea más a V3.

**HASTA/V2=="EOF"**

Esta línea comprueba la condición del bucle HACER/HASTA. Cuando el código ARCHIVO/LEER LÍNEA encuentra el final del archivo, el bucle sale. Cuando el flujo de la rutina sale del bucle, el resto del código termina de ejecutarse y muestra todo el bloque de código que se encuentra dentro de un comentario de operador.





`Resultado=Archivo/Leer línea,F1, "ID_pieza :"+{V1}`: Esto hace que el texto presente en la línea leída que sigue al texto "ID de pieza:" se asigne a V1. La línea se lee desde el archivo abierto utilizando F1 como el nombre del puntero a archivo. El resultado de la lectura (éxito o fracaso) se almacenaría en la variable Resultado.

```
Archivo/Leer
línea,F1,"Ubicación:"+{VARX}+", "+{VARY}+", "+{VARZ}+", "+{VARI
}+", "+{VARJ}+", "+{VARK}

ASIGN/CIR1.XYZ=MPUNTO (VARX,VARY,VARZ)

ASIGN/CIR1.IJK=MPUNTO (VARI,VARJ,VARK)
```

Las tres líneas de comando anteriores leen el texto delimitado por comas después de la cadena "Ubicación" y almacenan los valores en los valores X, Y, Z e I, J, K de CIR1.

`Archivo/Leer línea,F1,"Valor N.º"+var_bucle+" : "+{var2}`: Esto hace que var2 almacene el texto que aparece después de los dos puntos. La variable var\_bucle en este ejemplo no aparece entre llaves, y por consiguiente contribuye al texto delimitador.

### Muestra de código relativa a números que contienen ceros a la izquierda

Si el archivo que se está leyendo contiene líneas de números, observará que PC-DMIS no tiene en cuenta los caracteres de ceros a la izquierda. Por ejemplo, si la línea contiene el valor 005450, leería este valor estrictamente como un número y devolvería el valor 5450, haciendo caso omiso de los dos ceros iniciales. Este comportamiento puede ser o no ser el deseado.

Supongamos que tiene un archivo de texto creado por un software lector de código de barras externo y que ese archivo contiene estas dos líneas de datos:

290291143;582750;0010

291143;5827;0010

Puede utilizar un código sencillo como este para obtener los valores numéricos entre los caracteres de punto y coma:



```

      ASIGN/PRIMER_VALOR=0
      ASIGN/SEGUNDO_VALOR=0
      ASIGN/TERCER_VALOR=0
      ASIGN/NUMLIN=1
      FPTR=ARCHIVO/ABRIR,D:\TEMP\CODES.TXT,LEER
      HACER/
      EN LÍNEA=ARCHIVO/LEER
      LÍNEA,FPTR,{PRIMER_VALOR}+";"+{SEGUNDO_VALOR}+";"+{TE
      RCER_VALOR}
      COMENTARIO/OPER,NO,"NÚMERO DE LÍNEA: "+NUMLIN
      ,"Primer valor: "+PRIMER_VALOR
      ,"Segundo valor: "+SEGUNDO_VALOR
      ,"Tercer valor: "+TERCER_VALOR
      HASTA/EN LÍNEA=="EOF"
      ARCHIVO/CERRAR,FPTR,MANTENER

```

Si bien eso analizará correctamente las líneas de texto y devolverá los valores numéricos, también eliminará cualquier cero inicial en los valores que devuelva. Por lo tanto, la variable TERCER\_VALOR contendría el valor 10 en lugar de 0010.

Para conservar los ceros a la izquierda tendría que tratar toda la línea como cadena y utilizar en su lugar las funciones de cadena ÍNDICE, IZQUIERDA y MEDIO para hallar las posiciones de los puntos y coma en una línea de texto y obtener los valores numéricos:



```
FPTR=ARCHIVO/ABRIR,D:\TEMP\CODES.TXT,LEER
  ASIGN/NUMLIN=1
  HACER/
ESTADO LÍNEA=ARCHIVO/LEER LÍNEA,FPTR,{CADLIN}
  ASIGN/CADLIN=STR(CADLIN)
  ASIGN/PRIMER_ÍNDICE=INDEX(CADLIN,";")
  ASIGN/PRIMER_VALOR=STR(LEFT(CADLIN,PRIMER_ÍNDICE-1))
  ASIGN/CADREST=STR(MID(CADLIN,(PRIMER_ÍNDICE)))
  ASIGN/SEGUNDO_ÍNDICE=INDEX(CADREST,";")
  ASIGN/SEGUNDO_VALOR=STR(LEFT(CADREST,SEGUNDO_ÍNDICE-1))
  ASIGN/TERCER_VALOR=STR(MID(CADREST,SEGUNDO_ÍNDICE))
  COMENTARIO/OPER,NO,"NÚMERO DE LÍNEA: "+NUMLIN
    ,"Primer valor: "+PRIMER_VALOR
    ,"Segundo valor: "+SEGUNDO_VALOR
    ,"Tercer valor: "+TERCER_VALOR
  ASIGN/NUMLIN=NUMLIN+1
  HASTA/ESTADO LÍNEA=="EOF"
ARCHIVO/CERRAR,FPTR,MANTENER
```

### Explicación del código

Una gran parte de este código es similar al que se ha explicado anteriormente. Aquí se detallan únicamente los códigos únicos para las funciones de cadena mencionadas.

**ASIGN/PRIMER\_ÍNDICE=INDEX(CADLIN,";")**

Esta línea halla la posición del primer punto y coma de la línea y la asigna a la variable `PRIMER_ÍNDICE`.

**ASIGN/PRIMER\_VALOR=STR(LEFT(CADLIN,PRIMER\_ÍNDICE-1))**

Esta línea asigna a la variable `PRIMER_VALOR` la cadena de caracteres hasta el primer punto y coma (no incluido) de la variable `CADLIN`. `CADLIN` contiene la línea de texto completa.

**ASIGN/CADREST=STR(MID(CADLIN,(PRIMER\_ÍNDICE)))**

Esta línea asigna a la variable `CADREST` (que significa "cadena restante") la cadena de caracteres que quedan, empezando en la posición `PRIMER_ÍNDICE` (la posición del primer punto y coma) y hasta el final de la línea.

**ASIGN/SEGUNDO\_ÍNDICE=INDEX(CADREST,";")**

Busca dentro de la variable CADREST otro punto y coma (el segundo punto y coma de la línea) y asigna la posición a la variable `SEGUNDO_ÍNDICE`.

```
ASIGN/SEGUNDO_VALOR=STR(LEFT(CADREST,SEGUNDO_ÍNDICE-1))
```

Esta línea asigna a la variable `SEGUNDO_VALOR` la cadena de caracteres que va hasta el primer punto y coma (no incluido) de la variable `CADREST` (el segundo punto y coma de la línea completa).

```
ASIGN/TERCER_VALOR=STR(MID(CADREST,SEGUNDO_ÍNDICE))
```

Esta línea asigna a la variable `TERCER_VALOR` la cadena de caracteres que empieza en la posición `SEGUNDO_ÍNDICE` y termina al final de la línea.

## Leer un bloque de texto en un archivo

La opción de menú **Insertar | Comando E/S de archivos | Comandos de lectura | Leer bloque** introduce un comando en la ventana de edición para leer, durante la ejecución, un bloque de caracteres del archivo abierto. La cantidad de caracteres leídos depende del parámetro Tamaño.

Este comando requiere la siguiente sintaxis en la ventana de edición:



```
<nombre_variable>=Archivo/Leer_bloque,<puntero_archivo>,<tamaño>
```

A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

**<nombre\_variable>**: ID de la variable que recibe el valor que indica el éxito o el fracaso de la operación de lectura del bloque.

**<puntero\_archivo>**: Nombre especificado para el puntero al abrirse el archivo.

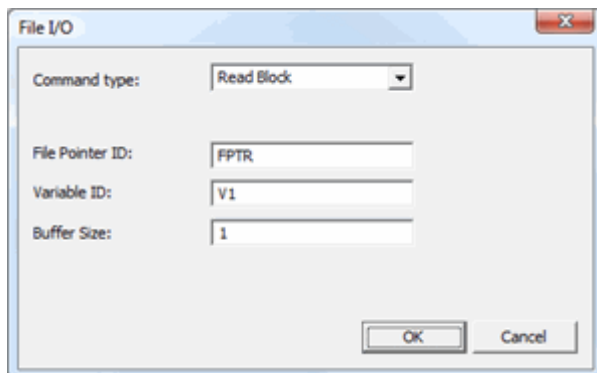
**<tamaño>**: Número de caracteres leídos.

Para acceder al cuadro de diálogo asociado con este comando de E/S de archivos:

1. Abra la ventana de edición.
2. Coloque el cursor en el comando Leer bloque.

## Usar entrada y salida de archivos

3. Pulse F9.



## Muestra de código de Leer bloque



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

Suponga que tiene varios archivos externos de datos que contienen diversos datos de pieza y que los primeros caracteres de cada archivo designan la función del mismo. Puede utilizar el comando [Archivo/Leer bloque](#) para leer sólo esos primeros caracteres antes de decidir leer y procesar todas las líneas. Considere este código:



```

C3=COMENTARIO/ENTRADA, Escriba el nombre del
,código del archivo que se ha de buscar.
ASIGN/TAMAÑO BLOQUE=LEN(C3.INPUT)
ASIGN/CÓDIGO ARCHIVO=C3.INPUT
HACER/
    C1=COMENTARIO/ENTRADA, Escriba la ruta de acceso
    completa,
    , el nombre de archivo y la extensión del
    ,archivo que desea procesar.
    ,escriba [S] para salir.
SI/C1.INPUT=="Q" O C1.INPUT=="q"
    COMENTARIO/OPER, Ha decidido salir. La rutina
    se está finalizando.
    IR_A/FIN
TERMINAR SI/
V1=ARCHIVO/EXISTE, C1.INPUT
SI/V1<>0
    COMENTARIO/OPER, "Archivo de datos
    ["+C1.INPUT+"] ya existe. Haga clic en
    Aceptar para continuar."
    FPTR=ARCHIVO/ABRIR, C1.INPUT, LEER
    V2=ARCHIVO/LEER BLOQUE, FPTR, TAMAÑO BLOQUE
    ARCHIVO/CERRAR, FPTR
    SI/V2<>CÓDIGO ARCHIVO
        COMENTARIO/OPER, "El código de archivo de
        ["+V2+"] no coincide"
        , "el CÓDIGO ARCHIVO de ["+CÓDIGO
        ARCHIVO+"]. "
    TERMINAR SI/
    COMENTARIO/OPER, "El archivo ["+C1.INPUT+"]
    coincide."
    , "El código de archivo de ["+V2+"] coincide"
    , "el CÓDIGO ARCHIVO de ["+CÓDIGO
    ARCHIVO+"]. "
    COMENTARIO/OPER, La rutina procesará entonces
    el archivo.
TERMINAR SI/
O BIEN/
    COMENTARIO/OPER, "Archivo de datos
    ["+C1.INPUT+"] no existe. Vuélvalo a
    intentar utilizando un archivo de datos
    existente."
    IR_A/FIN
TERMINAR O BIEN/
HASTA/V2==CÓDIGO ARCHIVO
FIN=ETIQUETA/

```

RUTINA/FIN

## Explicación de los códigos

Parte de este código es similar al que se describe en "Muestra de código de Leer carácter" y en "Muestra de código de Leer línea".

A continuación se explican únicamente las características especiales de este ejemplo.

**ASIGN/TAMAÑO BLOQUE=LEN(C3.INPUT)**

Esta línea crea una variable definida por el usuario llamada `TAMAÑO BLOQUE` que contiene un entero igual al número de caracteres encontrados en `C3.INPUT`. Esto se utilizará como tamaño del bloque de caracteres que se ha de leer.

**ASIGN/CÓDIGO ARCHIVO=C3.INPUT**

Esta línea crea la variable `CÓDIGO ARCHIVO` y le asigna el valor de `C3.INPUT`.

**C1=COMENTARIO/ENTRADA**

Este comentario almacena la ruta de acceso completa introducida por el usuario en la variable `C1.INPUT`.

**V1=ARCHIVO/EXISTE,C1.INPUT**

Esta línea comprueba si existe el nombre de archivo definido en el comentario C1.

**HACER/**

Esta línea inicia un bucle HACER/HASTA. Delimita el bloque de código que permite al usuario especificar en qué archivo se ha de leer. Continuará los bucles hasta que el texto asignado a la variable `CÓDIGO ARCHIVO` coincida con el texto leído en el archivo.

**V2=ARCHIVO/LEER BLOQUE,FPTR,TAMAÑO BLOQUE**

Esta línea lee una cantidad de caracteres igual al entero que contiene la variable `TAMAÑO BLOQUE`. El texto se almacena entonces en la variable `V2`.

**SI/V2CÓDIGO ARCHIVO**

Esta línea comienza un bloque de código SI/TERMINAR SI que comprueba si el texto de la variable `V2` coincide con el texto almacenado en la variable `CÓDIGO ARCHIVO`. Si coincide, la rutina continuará ejecutándose. De lo contrario, mostrará un mensaje para indicar que los dos códigos no coinciden.

**HASTA/V2==CÓDIGO ARCHIVO**

Esta línea comprueba la condición del bucle HACER/HASTA para ver si el texto de la variable **V2** coincide con el texto de la variable **CÓDIGO ARCHIVO**. Si la instrucción da como resultado FALSO, el bucle **HACER** se volverá a ejecutar y el usuario podrá seleccionar otro nombre de archivo. Si la instrucción da como resultado VERDADERO, el bucle saldrá y la rutina mostrará un mensaje para indicar que coincide. PC-DMIS podría continuar leyendo cada línea de datos en el archivo de datos especificado.

---

## Leer texto hasta un delimitador

La opción de menú **Insertar | Comando E/S de archivos | Comandos de lectura | Leer hasta** introduce un comando en la ventana de edición para leer, durante la ejecución, todo el texto "hasta" uno de los delimitadores del archivo especificado. Cualquier texto leído por este comando se colocará en la variable de destino especificada. El comando detiene la lectura del texto cuando PC-DMIS encuentra lo siguiente:

- Delimitadores definidos
- Retornos de carro
- Caracteres de avance de línea

Si se llega al final del archivo, la variable de destino se establece en "EOF" (fin de archivo).

Este comando requiere la siguiente sintaxis en la ventana de edición:



```
<nombre_variable> =  
ARCHIVO/LEER_HASTA,<puntero_archivo>,<delimitadores>
```

A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

**<nombre\_variable>**: Nombre de la variable de destino.

**<puntero\_archivo>**: Nombre especificado para el puntero al abrirse el archivo.

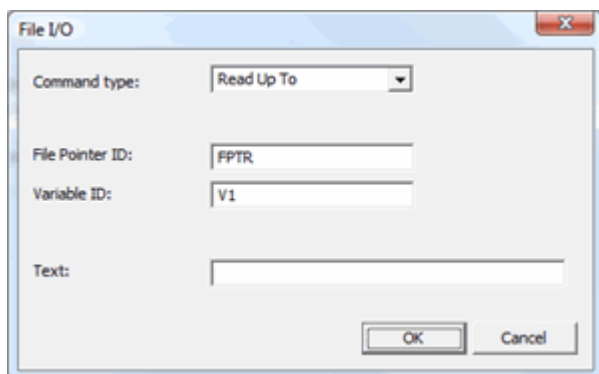
**<delimitadores>**: Cadena que contiene cero o más caracteres delimitadores.

Para acceder al cuadro de diálogo asociado con este comando de E/S de archivos:



## Usar entrada y salida de archivos

1. Seleccione **Ventana de edición** para abrir dicha ventana.
2. Coloque el cursor en el comando [ARCHIVO/LEER](#).
3. Pulse F9. Aparecerá el cuadro de diálogo **E/S de archivo**.



Cuando éste aparezca:

1. Introduzca el nombre de la variable que recibirá la información de lectura, en el cuadro **ID de variable**.
2. Introduzca el nombre del puntero de archivo en el cuadro **ID de puntero de archivo**.
3. Introduzca el delimitador en el cuadro **Texto** (asegúrese de colocar comillas a ambos lados del delimitador elegido).
4. Haga clic en **Aceptar**.

## Muestra de código de Leer hasta



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

Considere este ejemplo en el que tiene un archivo de texto denominado "ejemplo.txt" en el directorio D:\Temp, que contiene la siguiente información en la primera línea.




```
CIR1:2.54:CÍRCULO
```

Para utilizar el comando Leer hasta con este archivo:

1. Introduzca un comando `ARCHIVO/ABRIR` en la ventana de edición.
2. Al asignar un nombre al comando Abrir archivo, utilice el del puntero de archivo que usted elija. En este ejemplo se emplea la palabra "ejemplo" como el nombre del puntero de archivo.

El comando Abrir archivo debería tener entonces este aspecto:



```
EJEMPLO      =ARCHIVO/ABRIR,D:\TEMP\EJEMPLO.TXT,LEER
```

Ahora, utilice los comandos Leer hasta de PC-DMIS para definir algunas variables que invoquen diferentes segmentos de datos. Este ejemplo utiliza las siguientes variables al buscar un signo de dos puntos ":" (sin las comillas) como delimitador.




```
V_LABEL      =ARCHIVO/LEER_HASTA,EJEMPLO, :
V_VALUE      =ARCHIVO/LEER_HASTA,EJEMPLO, :
V_TYPE       =ARCHIVO/LEER_HASTA,EJEMPLO, :
```

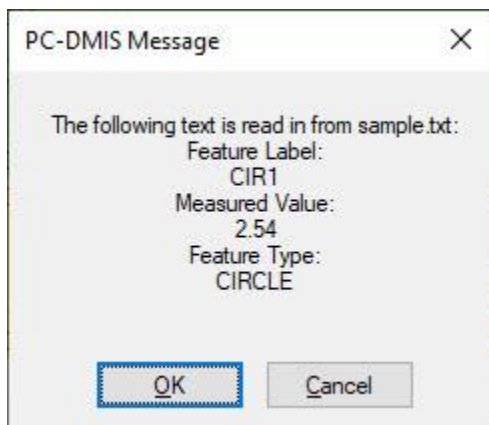
De este modo, cuando PC-DMIS ejecute estas líneas, en estas variables establecerá estos valores:

- V\_LABEL = CIR1
- V\_VALUE = 2.54
- V\_TYPE = CÍRCULO

Para mostrar esta información en pantalla durante la ejecución, puede utilizar un comentario de operador como el que se indica a continuación:



```
COMENTARIO/OPER,NO,PANTALLA COMPLETA=NO,CONTINUAR
AUTOMÁTICAMENTE=NO,OVC=NO,
El siguiente texto se lee del archivo ejemplo.txt
sample.txt:
Etiqueta del elemento:
V_LABEL
Valor medido:
V_VALUE
Tipo de elemento:
V_TYPE
```

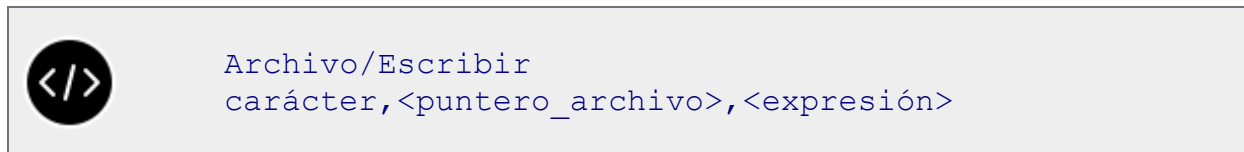


---

## Escribir un carácter en un archivo

La opción de menú **Insertar | Comando E/S de archivos | Comandos de escritura | Escribir carácter** introduce en la ventana de edición un comando para enviar un solo carácter a un archivo de su equipo tras la ejecución.

Este comando requiere la siguiente sintaxis en la ventana de edición:



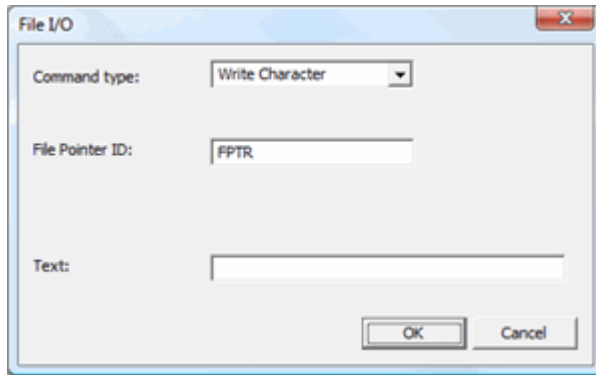
A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

**<puntero\_archivo>**: Nombre del puntero especificado al abrirse el archivo.

**<expresión>**: Carácter que ha de escribirse en archivo. Si al evaluarse la expresión se obtiene más de un carácter, sólo se escribe el primero de ellos.

Para acceder al cuadro de diálogo asociado con este comando de E/S de archivos, siga estos pasos:

1. Abra la ventana de edición.
2. Coloque el cursor en el comando Escribir carácter.
3. Pulse F9.



## Muestra de código de Escribir carácter



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

Observe el código siguiente que escribe, carácter por carácter, una cadena proporcionada por el usuario en un archivo de datos.



```
C1=COMENTARIO/ENTRADA, Escriba el nombre del archivo
en el que se ha de escribir
,(introduzca la ruta completa).
FPTR=ARCHIVO/ABRIR,C1.INPUT, ESCRIBIR
C2=COMENTARIO/ENTRADA, Escriba algo para enviar al
archivo.
,La cadena se enviará un carácter
,a la vez.
ASIGN/RECUENTO=0
ASIGN/LONG=LEN(C2.INPUT)
HACER/
    ASIGN/ESCRIBIR ESTOMID(C2.INPUTRECUENTO,1)
    ARCHIVO/ESCRIBIR_CARÁCTER,FPTR, ESCRIBIR ESTO
    ASIGN/RECUENTO=RECUENTO + 1
HASTA/RECUENTO==LONG
```

### Explicación de los códigos

Parte de este código es similar al que se describe en "Muestra de código de Leer carácter" y en "Muestra de código de Leer línea".

A continuación se explican únicamente las características especiales de este ejemplo.

Usar entrada y salida de archivos

```
FPTR=ARCHIVO/ABRIR,C1.INPUT,ESCRIBIR
```

Esta línea abre el archivo especificado en el comentario C1 para escribir y se lo asigna al puntero de archivo `FPTR`. Todos los datos de este archivo se sobrescribirán siempre que el puntero de archivo empiece al principio del archivo de datos.

```
ASIGN/RECuento=0
```

Esta línea asigna un valor de cero a una variable `RECuento` definida por el usuario. Esto se utiliza para fines relacionados con los bucles para imprimir la cadena carácter por carácter.

```
ASIGN/LONG=LEN(C2.INPUT)
```

Esta línea utiliza la función `LEN( )` para mostrar la longitud de una cadena. Esta función toma un parámetro: la cadena. Cuenta el número de caracteres de la cadena (espacios incluidos) y da como resultado un valor entero de dicha cantidad. En este caso la variable `LONG` definida por el usuario tiene este valor.

```
HACER/
```

Esta línea inicia un bucle `HACER/HASTA`. El código entre las instrucciones `HACER` y `HASTA` se ejecutará hasta que la condición del bucle dé como resultado `VERDADERO`.

```
ASIGN/ESCRIBIR ESTOMID(C2.INPUTRECuento,1)
```

Esta línea crea una variable definida por el usuario denominada `ESCRIBIR ESTO` y utiliza la función `MID( )` para devolver un carácter de subcadena de la cadena `C2.INPUT` y dárselo a `ESCRIBIR ESTO`.

`MID( )` toma tres parámetros.

- Parámetro 1: es la cadena de la que se obtienen valores. En este caso se utiliza `C2.INPUT`.
- Parámetro 2: es la posición de la cadena de la que se toma el carácter. El primer carácter de una cadena sería la posición 0; la segunda posición, 1; la tercera posición, 2, y así sucesivamente. En este caso se utiliza la variable `RECuento`.
- Parámetro 3: cuántos caracteres, contando desde la posición del segundo parámetro, se deben tomar. En este caso se utiliza el valor de 1 (la muestra sólo escribe un carácter cada vez, por lo que no hay motivo para tomar más).

```
ARCHIVO/ESCRIBIR_CARÁCTER,FPTR,ESCRIBIR ESTO
```

Esta línea escribe el carácter almacenado en la variable `ESCRIBIR ESTO` en el archivo especificado por el puntero de archivo `FPTR`.

**ASIGN/RECuento = RECuento+1**

Esta línea toma el valor actual de `RECuento`, lo incrementa en uno, y luego vuelve a colocar el valor nuevo en `RECuento`.

**HASTA/RECuento==LONG**

Esta línea comprueba la condición del bucle `HACER/HASTA`. En este caso el bucle continuará aumentando la variable `RECuento` hasta que tenga el mismo valor que la variable `LONG`. El bucle saldrá entonces, con lo que finalizará la rutina.

## Escribir una línea en un archivo

La opción de menú **Insertar | Comando E/S de archivos | Comandos de escritura | Escribir línea** introduce en la ventana de edición un comando para enviar una línea de texto a un archivo de su equipo tras la ejecución. Utilice la sintaxis de expresión para enviar la información sobre variables y la rutina de medición a un archivo. Se anexa automáticamente un retorno de carro al texto escrito.

Este comando requiere la siguiente sintaxis en la ventana de edición:



`Archivo/Escribir línea,<puntero_archivo>,<expresión>`

A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

**<puntero\_archivo>**: Nombre de la referencia de archivo especificada al abrirse el archivo.

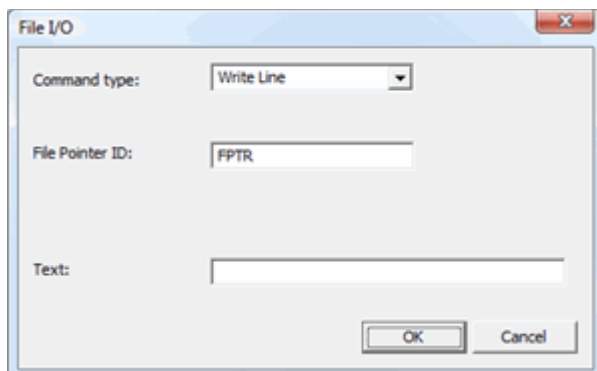
**<expresión>**: Texto que ha de escribirse en archivo. Este campo admite el uso de expresiones.

Para acceder al cuadro de diálogo asociado con este comando de E/S de archivos, siga estos pasos:

1. Abra la ventana de edición.

## Usar entrada y salida de archivos

2. Coloque el cursor en el comando Escribir línea.
3. Pulse F9.



## Muestra de código de Escribir línea



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

Suponga que desea exportar algunos valores XYZ medidos a un archivo de datos. El código siguiente permite introducir una etiqueta de elemento y un archivo de datos, así como enviar los datos X, Y y Z de ese elemento a un archivo de datos.



```
C1=COMENTARIO/ENTRADA,escriba la etiqueta del
elemento.
,que se ha de utilizar.
C2=COMENTARIO/ENTRADA,Escriba el nombre del archivo
en el que se ha de escribir
,(introduzca la ruta completa).
FPTR=ARCHIVO/ABRIR,C2.INPUT,ANEXAR
ASIGN/NOMBRE ELEM=C1.INPUT
ASIGN/TODOS VALS=NOMBRE ELEM.X+", "+NOMBRE
ELEM.Y+", "+NOMBRE ELEM.Z
COMENTARIO/OPER,"El texto que se ha de escribir es:
"+TODOS VALS
ARCHIVO/ESCRIBIR LÍNEA,FPTR,TODOS VALS
ARCHIVO/CERRAR,FPTR
```

## Explicación de los códigos

Parte de este código es similar al que se describe en "Muestra de código de Leer carácter" y en "Muestra de código de Leer línea".

A continuación se explican únicamente las características especiales de este ejemplo.

```
FPTR=ARCHIVO/ABRIR,C2.INPUT,ANEXAR
```

Esta línea abre el archivo especificado en el comentario C2 para anexar y se lo asigna al puntero de archivo `FPTR`. Pero si se cambia de `ANEXAR` a `ESCRIBIR`, el contenido existente del archivo de datos se sobrescribirá.

```
ASIGN/NOMBRE ELEM=C1.INPUT
```

Esta línea asigna la cadena de la etiqueta del elemento de `C1.INPUT` a la variable definida por el usuario `NOMBRE ELEM`.

```
ASIGN/TODOS VALS=NOMBRE ELEM.X+", "+NOMBRE ELEM.Y+", "+NOMBRE ELEM.Z
```

Esta línea asigna a la variable definida por el usuario `TODOS VALS` el valor de `NOMBRE ELEM.X`, `NOMBRE ELEM.Y`, `NOMBRE ELEM.Z`; en otras palabras, ahora contendrá los valores X, Y y Z de la etiqueta de elemento introducida en el comentario de entrada C1.

```
ARCHIVO/ESCRIBIR LÍNEA,FPTR,TODOS VALS
```

Esta línea escribe los valores que contiene `TODOS VALS` en el archivo especificado por el puntero de archivo `FPTR`.

---

## Escribir un bloque de texto en un archivo

La opción de menú **Insertar | Comando E/S de archivos | Comandos de escritura | Escribir bloque** introduce en la ventana de edición un comando para enviar un bloque de texto a un archivo de su equipo tras la ejecución. Utilice la sintaxis de expresión para enviar la información sobre variables y la rutina de medición a un archivo. A diferencia del comando Escribir línea, Escribir bloque *no* añade un retorno de carro al final del bloque.

Este comando requiere la siguiente sintaxis en la ventana de edición:





Archivo/Bloque\_escritura,<puntero\_archivo>,<expr>

A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

**<puntero\_archivo>**: Nombre de la referencia de archivo especificada al abrirse el archivo.

**<expresión>**: Texto que ha de escribirse en archivo. Este campo admite el uso de expresiones.



A diferencia del comando Escribir línea, Escribir bloque *no* añade un retorno de carro al final del bloque. No obstante, si necesita colocar texto en una nueva línea dentro del bloque de texto, puede insertar un retorno de carro y un avance de línea manualmente utilizando el código CHR(10) fuera de la cadena entrecomillada, como se muestra en este ejemplo:

```
ARCHIVO/BLOQUE_ESCRITURA,FPTR, "CHR(10) inserta texto... " +  
CHR(10) + " ...en una nueva línea."
```

Esto produciría este resultado dentro del archivo de salida:

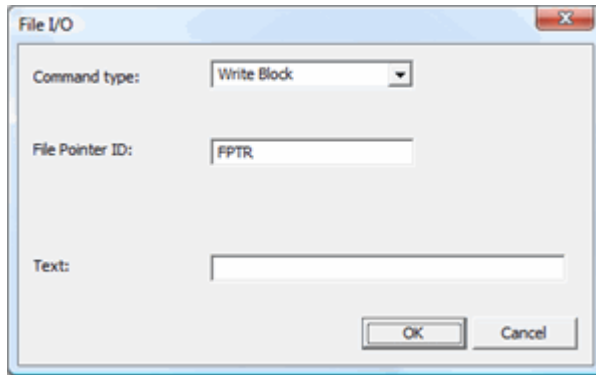
```
CHR(10) inserta texto...
```

```
...en una nueva línea.
```

Observe que si CHR(10) se encuentra dentro de las comillas, se envía al archivo el texto real de CHR(10).

Para acceder al cuadro de diálogo asociado con este comando de E/S de archivos:

1. Abra la ventana de edición.
2. Coloque el cursor en el comando Escribir bloque.
3. Pulse F9.



## Muestra de código de Escribir bloque



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

El código siguiente escribe lo que el usuario introduzca en un comentario de entrada y añade dos puntos que se utilizarán como delimitador.



```
C1=COMENTARIO/ENTRADA, Escriba cualquier cadena. PC-
DMIS añadirá , dos puntos (para delimitar) y
escribirá la cadena en el archivo seleccionado
C2=COMENTARIO/ENTRADA, Escriba el nombre del archivo
en el que se ha de escribir
, (introduzca la ruta completa).
FPTR=ARCHIVO/ABRIR, C2.INPUT, ANEXAR
ASIGN/ESCRIBIR ESTO=C1.INPUT+":"
COMENTARIO/OPER, "El texto que se ha de escribir es:
"+ESCRIBIR ESTO
ARCHIVO/ESCRIBIR LÍNEA, FPTR, ESCRIBIR ESTO
ARCHIVO/CERRAR, FPTR
```

### Explicación de los códigos

Parte de este código es similar al que se describe en "Muestra de código de Leer carácter" y en "Muestra de código de Leer línea".

A continuación se explican únicamente las características especiales de este ejemplo.

**FPTR=ARCHIVO/ABRIR, C2 . INPUT , ANEXAR**

Usar entrada y salida de archivos

Esta línea abre el archivo especificado en el comentario C2 para anexar y se lo asigna al puntero de archivo `FPTR`.

```
ASIGN/ESCRIBIR ESTO=C1.INPUT+" : "
```

Esta línea añade dos puntos al texto que contiene `C1.INPUT` y asigna la cadena nueva a la variable definida por el usuario `ESCRIBIR ESTO`.

```
ARCHIVO/ESCRIBIR LÍNEA,FPTR,ESCRIBIR ESTO
```

Esta línea escribe los valores que contiene `ESCRIBIR ESTO` en el archivo especificado por el puntero de archivo `FPTR`. Posteriormente puede leer texto del archivo utilizando los dos puntos como delimitador.

---

## Colocar un puntero al principio de un archivo

La opción de menú **Insertar | Comando E/S de archivos | Comandos de posición | Regresar al inicio** inserta un comando en la ventana de edición para posicionar el puntero de archivo al principio del flujo de datos.

Este comando requiere la siguiente sintaxis en la ventana de edición:



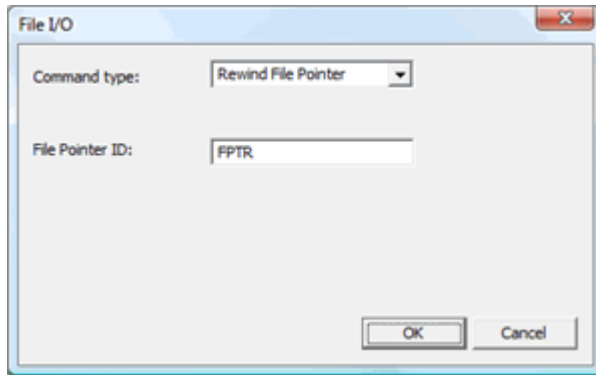
```
Archivo/Retroceder,<puntero_archivo>
```

A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

**<puntero\_archivo>**: Nombre del puntero que se desea volver a colocar al principio del archivo.

Para acceder al cuadro de diálogo asociado con este comando de E/S de archivos:

1. Abra la ventana de edición.
2. Coloque el cursor en el comando Regresar al inicio.
3. Pulse F9.



## Muestra de código de Regresar al inicio



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

Observe el ejemplo siguiente que lee datos en un archivo externo línea por línea. Después de cada línea, tiene la posibilidad de volver a empezar y leer desde el principio del archivo. A continuación se ilustra el empleo del comando ARCHIVO/RETROCEDER.



```
C1=COMENTARIO/ENTRADA, Escriba el nombre del archivo
en el que se ha de leer.
,(introduzca la ruta completa)
V1=ARCHIVO/EXISTE, C1.INPUT
SI/V1<>0
  HACER/
    FPTR=ARCHIVO/ABRIR, C1.INPUT, LEER
    C2=COMENTARIO/SÍNO, ¿Desea leer desde el
    principio?
    SI/C2.INPUT == "SÍ"
      ARCHIVO/RETROCEDER, FPTR
    TERMINAR SI/
    V2=ARCHIVO/LEER LÍNEA, FPTR, {LÍNEA}
    COMENTARIO/OPER, "La línea actual es: " +
    LÍNEA
  HASTA/V2=="EOF"
TERMINAR SI/
ARCHIVO/CERRAR, FPTR
COMENTARIO/OPER, Salir de la rutina sin guardar.
```

## Explicación de los códigos

Parte de este código es similar al que se describe en "Muestra de código de Leer carácter" y en "Muestra de código de Leer línea".

A continuación se explican únicamente las características especiales de este ejemplo.

`C2=COMMENT/YESNO`

Esta línea pregunta si desea empezar a leer el archivo desde el principio. Almacena la respuesta SÍ/NO en la variable `C2.INPUT`.

`SI/C2.INPUT == "SÍ"`

Esta línea inicia un bloque SI/TERMINAR SI. Comprueba la condición de `C2.INPUT` que tiene el valor de SÍ. Si la condición es verdadera, PC-DMIS ejecuta las líneas tras la instrucción `SI`. Si la condición es falsa, PC-DMIS ejecuta el código tras la instrucción `TERMINAR SI`.

`ARCHIVO/RETROCEDER,FPTR`

Esta línea hace retroceder el puntero de archivo hasta el principio del archivo de datos.

`TERMINAR SI/`

Esta línea cierra el bloque de código SI/TERMINAR SI.

---

## Guardar la posición actual del puntero de archivo

La opción de menú **Insertar | Comando E/S de archivos | Comandos de posición | Guardar posición de archivo** inserta un comando en la ventana de edición para guardar la posición actual de un puntero de archivo dentro del flujo de datos. Posteriormente, la posición guardada del puntero en el archivo puede recuperarse utilizando el comando Recuperar posición de archivo.

Este comando requiere la siguiente sintaxis en la ventana de edición:



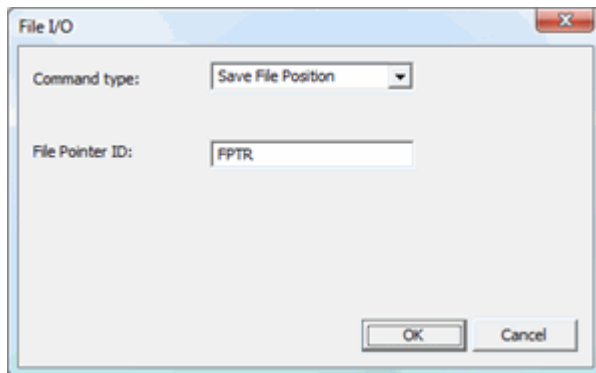
`Archivo/Guardar posición,<puntero_archivo>`

A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

**<puntero\_archivo>**: Nombre del puntero de archivo cuya posición se desea guardar.

Para acceder al cuadro de diálogo asociado con este comando de E/S de archivos:

1. Abra la ventana de edición.
2. Coloque el cursor en el comando Guardar posición de archivo.
3. Pulse F9.



## Muestra de código de Guardar posición de archivo



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

Observe el ejemplo siguiente que lee datos en un archivo externo línea por línea. Después de cada línea tiene la posibilidad de guardar la posición del archivo para poderlo recuperar en futuras ocasiones. A continuación se ilustra el empleo del comando `ARCHIVO/GUARDAR POS.`



```
C1=COMENTARIO/ENTRADA, Escriba el nombre del archivo
en el que se ha de leer.
, (introduzca la ruta completa)
V1=ARCHIVO/EXISTE, C1.INPUT
SI/V1<>0
    HACER/
        FPTR=ARCHIVO/ABRIR, C1.INPUT, LEER
        C2=COMENTARIO/SÍNO, ¿Desea guardar la
        posición del archivo y recuperarla más
        tarde? El bucle se cerrará.
        SI/C2.INPUT == "SÍ"
            ARCHIVO/GUARDAR POS, FPTR
            IR A/CERRAR BUCLE
        TERMINAR SI/
        V2=ARCHIVO/LEER LÍNEA, FPTR, {LÍNEA}
        COMENTARIO/OPER, "La línea actual es: " +
        LÍNEA
        HASTA/V2=="EOF"
    TERMINAR SI/
    ARCHIVO/CERRAR, FPTR
    CERRAR BUCLE=ETIQUETA/
    COMENTARIO/OPER, Ha dejado de leer.
    RUTINA/FIN
```

### Explicación de los códigos

Este código es similar al que se ha explicado en "Muestra de código de Regresar al inicio".

A continuación se explican únicamente las características especiales de este ejemplo.

**C2=COMMENT/YESNO**

Esta línea pregunta si desea almacenar la posición actual del archivo y salir del bucle. Almacena la respuesta SÍ/NO en la variable `C2.INPUT`.

**ARCHIVO/GUARDAR POS, FPTR**

Esta línea almacena la posición del puntero de archivo en el flujo de datos.

Siempre y cuando abra el mismo archivo con el mismo nombre de puntero de archivo en la misma rutina de medición, podrá recuperar una posición de archivo almacenada y continuar leyendo desde donde lo dejó la última vez. Para continuar este ejemplo, consulte el tema "Muestra de código de Recuperar posición del archivo"

# Recuperar la posición del puntero de archivo guardado

La opción de menú **Insertar | Comando E/S de archivos | Comandos de posición | Recuperar posición de archivo** inserta en la ventana de edición un comando que recupera la posición de un archivo guardado anteriormente. Utilice el comando Guardar posición de archivo para guardar una posición dentro de un archivo abierto.

Este comando requiere la siguiente sintaxis en la ventana de edición:



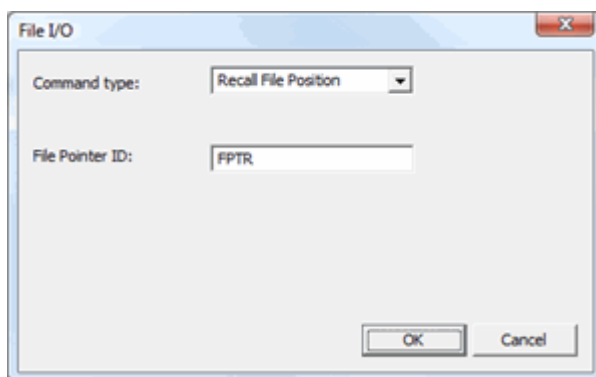
Archivo/Recuperar posición, <puntero\_archivo>

A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

**<puntero\_archivo>**: Nombre del puntero de archivo cuya posición se desea recuperar.

Para acceder al cuadro de diálogo asociado con este comando de E/S de archivos:

1. Abra la ventana de edición.
2. Coloque el cursor en el comando Recuperar posición de archivo.
3. Pulse F9.





## Muestra de código de Recuperar posición del archivo



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

Este ejemplo abre un archivo cerrado anteriormente, utiliza un puntero de archivo anterior y recupera la posición guardada del puntero de archivo almacenado. A continuación lee en los datos de dicha posición. Este código ilustra el empleo del comando `ARCHIVO/RECUPERAR POS`. Es una continuación del ejemplo de código que aparece en el tema "Muestra de código de Guardar posición de archivo".



```
COMENTARIO/OPER,La rutina recuperará ahora la
posición de archivo almacenada.
FPTR=ARCHIVO/ABRIR,C1.INPUT,LEER
ARCHIVO/RETROCEDER,FPTR
COMENTARIO/OPER,Para realizar la comprobación, se ha
retrocedido en el archivo.
,Se leerá la primera línea para comprobar el
retroceso.
V3=ARCHIVO/LEER LÍNEA,FPTR,{LÍNEA}
COMENTARIO/OPER,La primera línea es:
,LÍNEA
ARCHIVO/RETROCEDER,FPTR
ARCHIVO/RECUPERAR POS,FPTR
COMENTARIO/OPER,Se ha recuperado la posición de
archivo almacenada anteriormente.
,A continuación se imprimirán los datos de la línea
en la posición almacenada.
V4=ARCHIVO/LEER LÍNEA,FPTR,{ALMACENADA}
COMENTARIO/OPER,El texto en la posición almacenada
es:
,ALMACENADA
```

### Explicación de los códigos

Este código es similar al que se ha explicado en "Muestra de código de Regresar al inicio".

A continuación se explican únicamente las características especiales de este ejemplo.

`ARCHIVO/RECUPERAR POS,FPTR`

Esta línea recupera la posición del puntero de archivo almacenada en el flujo de datos para el puntero de archivo calificado como `FPTR`.

**`V4=ARCHIVO/LEER LÍNEA, FPTR, {ALMACENADA}`**

Esta línea lee la línea siguiente posterior a la posición del puntero de archivo almacenada, y la asigna a la variable de `ALMACENADA` definida por el usuario. A continuación esta variable se imprime en el siguiente comentario de operador.

---

## Copiar un archivo

La opción de menú **Insertar | Comando E/S de archivos | Copiar archivo** inserta un comando en la ventana de edición que origina una operación de copia de archivo durante la ejecución.

Este comando requiere la siguiente sintaxis en la ventana de edición:



```
Archivo/Copiar,<archivo_origen>,<archivo_destino>,  
<modo_sustitución>
```

A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

**<archivo\_origen>**: Nombre del archivo de origen de la operación de copia.

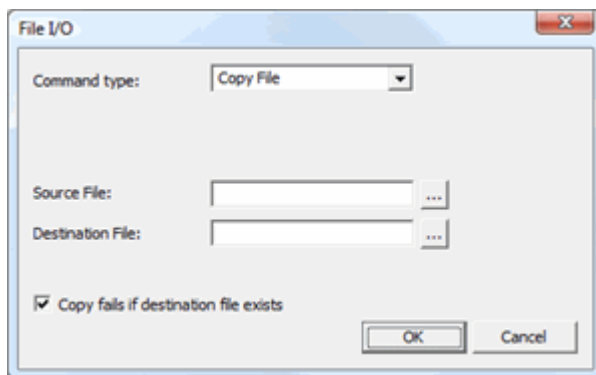
**<archivo\_destino>**: Nombre del archivo de destino de la operación de copia.

**<modo\_sustitución>**: Acción que tiene lugar si el archivo de destino ya existe. Los dos modos posibles son: `SOBRESCRIBIR` y `FALLAR_SI_DEST_EXISTE`.

Para acceder al cuadro de diálogo asociado con este comando de E/S de archivos:

1. Abra la ventana de edición.
2. Coloque el cursor en el comando Copiar archivo.
3. Pulse F9.

Usar entrada y salida de archivos



## Muestra de código de Copiar archivo



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

El siguiente código solicita el nombre de archivo que ha de copiarse, así como el directorio y archivo de destino en los que se va a copiar.



```

C1=COMENTARIO/ENTRADA,Escriba el nombre del archivo
que desea copiar.
,(introduzca la ruta de archivo completa)
C2=COMENTARIO/ENTRADA,Escriba el nombre de un
archivo de destino.
,(introduzca la ruta de archivo completa)
V1=ARCHIVO/EXISTE,C1.INPUT
SI/V1<>0
    COMENTARIO/OPER,El archivo que ha de copiarse
    existe. Iniciar el proceso de copia.
    ARCHIVO/COPIAR,C1.INPUT,C2.INPUT,FALLAR SI DEST
    EXISTE
    V2=ARCHIVO/EXISTE,C2.INPUT
    SI/V2==0
        COMENTARIO/OPER,"El archivo no existe en: "
        + C2.INPUT
        ,Finalizar el proceso de copia.
        RUTINA/FIN
    TERMINAR SI/
O BIEN/
    COMENTARIO/OPER,Proceso de copia del archivo
    realizado con éxito.
    RUTINA/FIN
TERMINAR O BIEN/
TERMINAR SI/
COMENTARIO/OPER,El archivo que ha de copiarse no
existe.

```

## Explicación de los códigos

Una gran parte de este código es similar al que se describe en "Muestra de código de Leer carácter" y en "Muestra de código de Leer línea".

A continuación se explican únicamente las características especiales de este ejemplo.

### **C1=COMENTARIO/ENTRADA**

C1=COMENTARIO/ENTRADA Esta línea toma la ruta completa del archivo que se ha de copiar y la coloca en la variable C1.INPUT.

### **C2=COMENTARIO/INPUT**

Esta línea toma la ruta completa del archivo de destino y la coloca en la variable C2.INPUT.

### **ARCHIVO/COPIAR,C1.INPUT,C2.INPUT,FALLAR SI DEST EXISTE**

Esta línea copia el archivo original en un archivo de destino. Este comando admite tres parámetros.

- El parámetro 1 es `C1.INPUT`. Es la ruta completa del archivo que ha de copiarse.
- El parámetro 2 es `C2.INPUT`, o la ruta completa del archivo de destino.
- El parámetro 3, en este caso, cancela el procedimiento de ARCHIVO/COPIAR si se encuentra con un archivo existente con el mismo nombre de archivo de destino. Esto se puede programar para que se sobrescriban los archivos existentes con el mismo nombre.

## Comandos en modo Comando después de los comentarios

---

# Mover un archivo

La opción de menú **Insertar | Comando E/S de archivos | Mover archivo** inserta un comando en la ventana de edición que origina una operación de movimiento de archivo durante la ejecución.

Este comando requiere la siguiente sintaxis en la ventana de edición:



`Archivo/Mover,<archivo_antiguo>,<archivo_nuevo>`

A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

**<archivo\_antiguo>**: Ubicación y nombre del archivo.

**<archivo\_nuevo>**: Nueva ubicación y nombre del archivo.

Para acceder al cuadro de diálogo asociado con este comando de E/S de archivos:

1. Abra la ventana de edición.
2. Coloque el cursor en el comando Mover archivo.
3. Pulse F9.



## Muestra de código de Mover archivo



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

El siguiente código solicita el nombre de archivo que se ha de mover, así como el directorio de ubicación y el nombre de archivo a los que se ha de mover. A continuación efectúa la operación de mover el archivo.



```
C1=COMENTARIO/ENTRADA,Escriba el nombre del archivo
que desea mover.
,(introduzca la ruta de archivo completa)
C2=COMENTARIO/ENTRADA,Escriba el nombre de un
archivo de destino.
,(introduzca la ruta de archivo completa)
V1=ARCHIVO/EXISTE,C1.INPUT
SI/V1<>0
    COMENTARIO/OPER,El archivo que ha de moverse
    existe. Iniciar el proceso de mover el archivo.
    ARCHIVO/MOVER,C1.INPUT,C2.INPUT
    V2=ARCHIVO/EXISTE,C2.INPUT
    SI/V2==0
        COMENTARIO/OPER,"El archivo no existe en: "
        + C2.INPUT
        ,el proceso de MOVER no ha funcionado
        correctamente.
        RUTINA/FIN
    TERMINAR SI/
O BIEN/
    COMENTARIO/OPER,Proceso de MOVER el archivo
    realizado con éxito.
    RUTINA/FIN
TERMINAR O BIEN/
TERMINAR SI/
COMENTARIO/OPER,El archivo original no existe.
Inténtelo de nuevo.
```

### Explicación de los códigos

Una gran parte de este código es similar al que se ha explicado en "Muestra de código de Mover archivo".

A continuación se explican únicamente las características especiales de este ejemplo.

**ARCHIVO/MOVER,C1.INPUT,C2.INPUT**

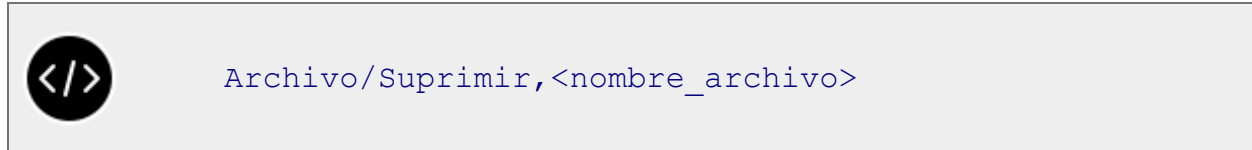
Esta línea copia el archivo original en un archivo de destino. Este comando admite dos parámetros.

- El parámetro 1 es `C1.INPUT`. Es la ruta completa del archivo que se ha de mover.
- El parámetro 2 es `C2.INPUT`, o la ruta completa del archivo de destino.

# Suprimir un archivo

La opción de menú **Insertar | Comando E/S de archivos | Suprimir archivo** inserta un comando en la ventana de edición que origina una operación de supresión de archivo durante su ejecución.

Este comando requiere la siguiente sintaxis en la ventana de edición:



A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

**<nombre\_archivo>**: Nombre del archivo que se desea suprimir.

Para acceder al cuadro de diálogo asociado con este comando de E/S de archivos:

1. Abra la ventana de edición.
2. Coloque el cursor en el comando Suprimir archivo.
3. Pulse F9.





## Muestra de código de Suprimir archivo



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

El siguiente código solicita un nombre del archivo y, a continuación, suprime dicho archivo.



```
C1=COMENTARIO/ENTRADA, Escriba el nombre del archivo
que desea suprimir.
,(introduzca la ruta de archivo completa)
V1=ARCHIVO/EXISTE, C1.INPUT
SI/V1<>0
    COMENTARIO/OPER, El archivo existe. Preparado
    para suprimirlo.
    ARCHIVO/SUPRIMIR, C1.INPUT
    V2=ARCHIVO/EXISTE,
    SI/V2==0
        COMENTARIO/OPER, El archivo se ha suprimido
        con éxito.
        RUTINA/FIN
    TERMINAR SI/
O BIEN/
    COMENTARIO/OPER, El archivo todavía existe
    RUTINA/FIN
TERMINAR O BIEN/
TERMINAR SI/
COMENTARIO/OPER, El archivo que ha de suprimirse no
existe. Seleccione un archivo que exista.
```

### Explicación de los códigos

Una gran parte de este código es similar al que se ha explicado en "Muestra de código de Mover archivo".

A continuación se explican únicamente las características especiales de este ejemplo.

**ARCHIVO/SUPRIMIR, C1 . INPUT:** Esta línea hace que se suprima el archivo especificado. Este comando admite un parámetro: el nombre del archivo que ha de suprimirse. En este caso, **C1 . INPUT**.

# Comprobar la existencia de un archivo

La opción de menú **Insertar | Comando E/S de archivos | El archivo ya existe** inserta un comando en la ventana de edición que verifica la existencia de un archivo durante su ejecución y asigna el resultado a la variable proporcionada.

Este comando requiere la siguiente sintaxis en la ventana de edición:



```
<nombre_variable> = Archivo/Existe,<nombre_archivo>
```

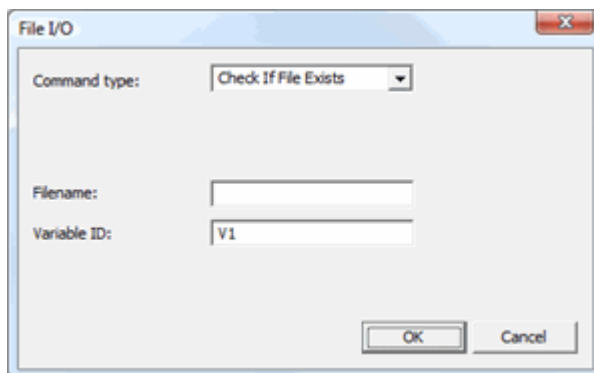
A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

**<nombre\_variable>**: Nombre de la variable a la cual se asigna el resultado de la verificación efectuada. Se asigna el valor de 1 a la variable si el archivo existe, o el 0 en caso contrario.

**<nombre\_archivo>**: Nombre del archivo cuya existencia en el disco se está verificando.

Para acceder al cuadro de diálogo asociado con este comando de E/S de archivos:

1. Abra la ventana de edición.
2. Coloque el cursor en el comando El archivo ya existe.
3. Pulse F9.



## Muestra de código de la opción El archivo ya existe



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

El siguiente código solicita un nombre de archivo y, a continuación, comprueba que existe dicho archivo.



```
C1=COMENTARIO/ENTRADA, Escriba el nombre del archivo
que desea comprobar.
V1=ARCHIVO/EXISTE, C1.INPUT
SI/V1<>0
    COMENTARIO/OPER, El archivo existe.
TERMINAR SI/
O BIEN/
    COMENTARIO/OPER, El archivo no existe.
TERMINAR O BIEN/
```

### Explicación de los códigos

Una gran parte de este código es similar al que se describe en "Muestra de código de Leer carácter" y en "Muestra de código de Leer línea".

A continuación se explican únicamente las características especiales de este ejemplo.

**V1=ARCHIVO/EXISTE, C1.INPUT**

Esta línea comprueba si existe el archivo especificado. Para que este código funcione, el archivo debe estar colocado en el directorio donde reside PC-DMIS; de lo contrario, la línea que contiene el archivo también debe contener la ruta de acceso completa del archivo. **V1** recibe el resultado de la comprobación del archivo. Si existe, es un valor distinto de cero; de lo contrario, es 0.

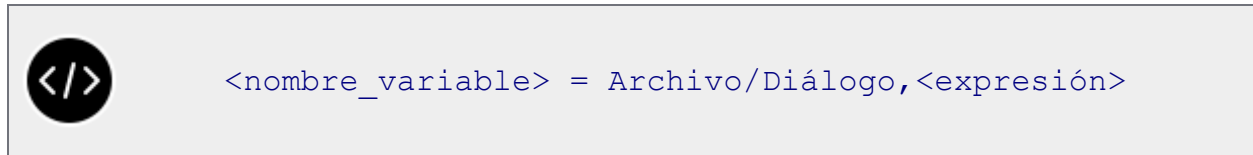
---

## Mostrar un cuadro de diálogo Archivo

La opción de menú **Insertar | Comando E/S de archivos | Cuadro de diálogo Archivo** inserta un comando en la ventana de edición que abre un cuadro de diálogo **Abrir** durante la ejecución. Esto permite al operador elegir un nombre de archivo

durante la ejecución. El nombre del archivo elegido se almacena en la variable especificada.

Este comando requiere la siguiente sintaxis en la ventana de edición:



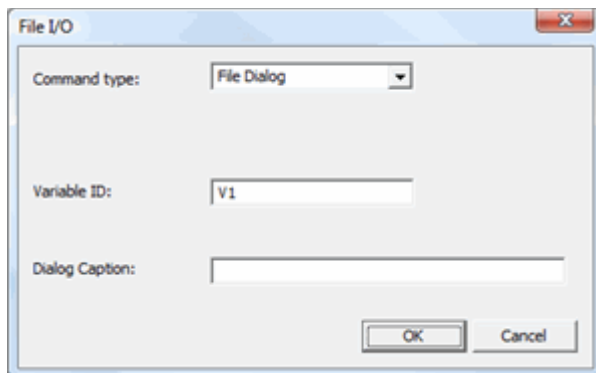
A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

**<nombre\_variable>**: Variable a la cual se asigna el nombre del archivo elegido por el usuario en el cuadro de diálogo.

**<expresión>**: Texto que aparece en la barra de título del cuadro de diálogo Archivo.

Para acceder al cuadro de diálogo asociado con este comando de E/S de archivos:

1. Abra la ventana de edición.
2. Coloque el cursor en el comando Cuadro de diálogo Archivo.
3. Pulse F9.



## Muestra de código de un cuadro de diálogo Archivo



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

El siguiente código presenta un cuadro de diálogo que le permite seleccionar el archivo que desee suprimir.



```
V1=ARCHIVO/DIÁLOGO,Seleccione el archivo que desee
suprimir.
V2=ARCHIVO/EXISTE,V1
SI/V2<>0
    COMENTARIO/OPER,El archivo existe. Preparado
    para suprimirlo.
    ARCHIVO/SUPRIMIR,V1
    V3=ARCHIVO/EXISTE,
    SI/V3==0
        COMENTARIO/OPER,El archivo se ha suprimido
        con éxito.
        RUTINA/FIN
    TERMINAR SI/
O BIEN/
    COMENTARIO/OPER,El archivo todavía existe
    RUTINA/FIN
TERMINAR O BIEN/
TERMINAR SI/
COMENTARIO/OPER,El archivo que ha de suprimirse no
existe. Seleccione un archivo que exista.
```

Una gran parte de este código es similar al que se describe en "Muestra de código de Leer carácter" y en "Muestra de código de Leer línea".

A continuación se explican únicamente las características especiales de este ejemplo.

**V1=ARCHIVO/Diálogo,Seleccione el archivo que desee suprimir.**

Esta línea muestra el cuadro de diálogo con el título "Seleccione el archivo que desee suprimir". Puede navegar hasta el archivo y, al hacer clic en **Abrir**, PC-DMIS dará a V1 la ruta completa al archivo seleccionado. El resto de la rutina suprime el archivo seleccionado.

# Comprobar si se ha llegado al final de un archivo o de una línea

PC-DMIS permite comprobar si se ha llegado al final de un archivo mediante el uso de las funciones `EOF` o `EOL` en una condición.

`EOF` significa FIN DE ARCHIVO (“end of file” en sus siglas en inglés). Esta función toma un puntero de archivo de tipo cadena. Si se inserta correctamente en una declaración condicional, comprueba si el puntero de archivo ha alcanzado el final del archivo especificado. En caso afirmativo, la función devuelve el valor verdadero.

`EOL` significa FIN DE LÍNEA (“end of line” en sus siglas en inglés). Esta función toma un puntero de archivo de tipo cadena. Si se inserta correctamente en una declaración condicional, comprueba si el puntero de archivo ha alcanzado el final de una línea en el archivo especificado. En caso afirmativo, la función devuelve el valor verdadero. El funcionamiento óptimo se da cuando esta función se inserta en un bucle.

Este comando requiere la siguiente sintaxis en la ventana de edición:

`EOF(<punteroarchivo>)` o `EOL(<punteroarchivo>)`

A continuación se proporcionan las descripciones de algunos de los componentes de este comando:

**<punteroarchivo>**: Nombre del puntero de archivo que se está comprobando.

## Muestra de código de EOF y EOL



El ejemplo de código siguiente debe introducirse en la ventana de edición en modo Comando, no en el cuadro de diálogo **E/S de archivo**.

El código siguiente abre el archivo test.txt y lo lee. Mientras no se alcance el fin de archivo (indicado con el código `MIENTRAS/!EOF`), PC-DMIS lee el archivo carácter a carácter y asigna a V1 un carácter.

Si PC-DMIS llega al final de una línea en el archivo, PC-DMIS muestra el último carácter de esa línea.

## Usar entrada y salida de archivos

Esto se repite hasta que PC-DMIS alcanza el final del archivo. A continuación, PC-DMIS muestra el texto "Se ha llegado al final del archivo...".



```
FPTR=ARCHIVO/ABRIR,D:\temp\test.txt,LEER
MIENTRAS/!EOF("FPTR")
V1=ARCHIVO/LEER CARÁCTER,FPTR
SI/EOL("FPTR")
COMENTARIO/OPER,NO,"Se ha llegado al final de la
línea. El último carácter es:"
,V1
TERMINAR SI/
TERMINAR MIENTRAS/
COMENTARIO/OPER,NO,"Se ha llegado al final del
archivo."
```