

# MS-DOS avanzado

**Autor: okoto nipon**

[\[Ver curso online\]](#)

## Presentación del curso

Microsoft Windows nunca fue realmente un Sistema Operativo con verdadero entorno gráfico hasta Windows95.

Este curso es la segunda parte del exitoso curso de introducción a MS-DOS. Reforzarás los conocimientos de este sistema operativo y conocerás nuevas formas de implementarlo en función con Windows.

Visita más cursos como este en mailxmail:

[<http://www.mailxmail.com/cursos-informatica>]

[<http://www.mailxmail.com/cursos-programacion>]



¡Tu opinión cuenta! Lee todas las opiniones de este curso y déjanos la tuya:

[<http://www.mailxmail.com/curso-ms-dos-avanzado/opiniones>]

### Cursos similares

Cursos	Valoración	Alumnos	Vídeo
<b>Photoshop. Dibujar un Código de Barras</b> Dibujar en Photoshop un Código de Barras es lo que nuestro curso te ofrece en esta oportunidad. Mediante unos sencillos pasos y el uso eficaz de las herramientas de Photo... [27/10/08]	●●●●●	662	
<b>PHP y MySQL. Aplicaciones Web: HTML I (segunda parte)</b> Programación de aplicaciones Web con PHP y MySQL. Ahora estudiaremos las páginas Web HTML, la estructura de los documentos HTML y los tipos de documentos... [02/12/08]	●●●●●	1.570	
<b>Selección de objetos con la herramienta pluma en Photoshop</b> En este curso vamos a aprender, mediante el programa Photoshop, a seleccionar cualquier tipo de objeto que uno pueda encontrar en una fotografía o en cualquier otro tipo ... [04/07/08]	●●●●●	1.401	
<b>Introducción al lenguaje Pascal</b> Pascal es un lenguaje de alto nivel y de propósito general (es aplicable a un gran número de aplicaciones diversas) desarrollado por el profesor suizo Niklaus Wirth como ... [01/03/06]	●●●●●	19.284	
<b>Holística informática</b> Aprende con nuestro curso de holística informática, sobre los avances informáticos que te ayudará a actualizarte conociendo más acer... [28/04/09]	●●●●●	2.219	

# 1. Introducción.

[<http://www.mailxmail.com/curso-ms-dos-avanzado/introduccion>]

Por fin ha llegado la tan esperada segunda parte del curso de ms-dos, Esta vez lo he elaborado basándome en todos sus comentarios, así que he incluido nuevas mejoras:

Menos imágenes

Ejemplos blanco y negro para facilitar su impresion.(invertidos)

Temas más avanzados.

Opciones y alternativas.

Lista de comandos más completa.

Este curso abarcará los comandos no cubiertos por el curso 1, así como usos de msdos en el Internet y la programación, además de alternativas y usos extras de Dos. ¡¡Espero les guste!!!

## 2. Personalizar MSDos

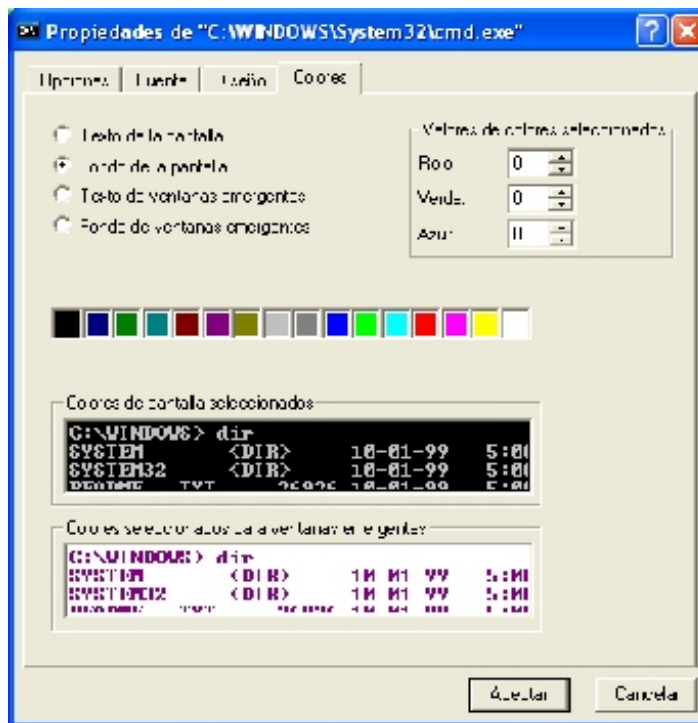
[<http://www.mailxmail.com/curso-ms-dos-avanzado/personalizar-msdos>]

Para empezar, veremos como modificar la apariencia de una ventana de msdos a una interface más personalizada y amigable.

Pues bien, cuando estamos trabajando con msdos es común ver una ventana a 2 colores (gris y negro).

Esto representa un problema para algunos usuarios. (en mi caso, para incluir imágenes para este curso que puedan imprimirse).

Si hacemos click con el botón derecho del mouse, en la parte del menú ventana, y seleccionamos "propiedades", veremos que podemos cambiar varias configuraciones de msdos, incluyendo la combinación de colores tanto para la ventana actual, como para las ventanas emergentes.



Ventanas emergentes.

Esto se refiere a las ventanas que puedan salir de el uso de los comandos, esto es muy importante, por que debemos aprender a trabajar en dos modos: en modo ventana y en modo sistema.

El modo sistema se refiere a cuando solamente estamos utilizando msdos, como sistema operativo, es decir, no estamos corriendo Windows al mismo tiempo. Un ejemplo de esto, es cuando reiniciamos el sistema en modo msdos.

Cualquier comando que tecleemos, nos devolverá un resultado dentro de msdos, en cambio, si iniciamos Windows y luego ejecutamos msdos, nos será devuelto en ventanas.

Por ejemplo: al teclear en modo sistema "scanreg" o "regedit" la computadora abrirá un cuadro en msdos, desde el cual podremos editar el

registro de Windows.

Pero si estamos usando Windows, nos abrirá una ventana del registro de Windows.

Otro ejemplo de esto es scandisk (solo disponible para versiones WinMe y anteriores), si lo tecleamos en modo sistema (la pc reiniciada en modo msdos), nos abrirá un cuadro azul, más en cambio en Windows, nos abrirá una ventana.

Esto no ocurre con todos los comandos, son solo para los externos y en su mayoría, a los que hacen referencia a alguna aplicación de Windows.



### 3. Comandos Pendientes

[<http://www.mailxmail.com/curso-ms-dos-avanzado/comandos-pendientes>]

#### Copy

##### Sintaxis:

**C:\>copy origen:nombre de archivo.extension destino:nombrede archivo2.extension**

Por ejemplo, podemos copiar un archivo **EH1.doc** desde un **diskete** a **C:** sería:

**C:\>a: (cambiamos de unidad)**

**A:\>**

**A:\>copy EH1.doc C:/ (ejecutamos el comando)**  
**1 archivos copiados.**

**Nota:** en los comandos que sea necesario especificar una ruta o destino, de unidades distintas, es incorrecto escribir: **A:/** o **C:/**, basta con poner la letra de la unidad, dos puntos y el archivo, así podemos trabajar con un comando indirecto, para el ejemplo anterior sería:

**C:/>copy A:EH1.doc C:Eh2.doc**

Este comando es poco usado y no muy recomendable, ya que contiene muy pocos modificadores (disponibles con /? ) y solo copia 1 archivo a la vez.

Aquí además hemos cambiado el nombre por Eh2.doc, es también posible renombrar archivos de esta manera, simplemente cambiando al extensión del destino.

**C:\>copy unidad:nombrearchivo1 destino:nombrearchivo2**

En este caso, solo incluimos la unidad C en el destino, como no hemos especificado un nuevo nombre para el archivo, sera el original EH1.doc.

## 4. Comandos pendientes 2.

[<http://www.mailxmail.com/curso-ms-dos-avanzado/comandos-pendientes-2>]

### Format.

Formatear una unidad

Este comando sirve para dar formato a unidades de disco, tanto discos duros locales como diskettes. **En windows xp este comando suele no estar disponible** ya que windows xp utiliza el sistema ntfs y el comando format esta diseñado para funcionar con sistemas de archivos del tipo Fat32.(File allocation table) como son todos los windows anteriores.

En windows Me por ejemplo, este comando sirve muy bien para formatear una pc. Si bien es cierto que format contiene varios modificadores, tambien es cierto que no es necesario escribirlos, ya que durante el formateo de una unidad nos seran solicitados, tales atributos son:

**El nombre de la etiqueta, /V:**

**El tipo de formato /FS: (fat o ntfs)**

**El tamaño /F:**

Por ejemplo, podríamos formatear un diskette para que sea ntfs (algo muy poco usual), sería:

**C:\>format A: /FS:NTFS**

No es comun que esto funcione.

**Nota:** recuerda que al formatear estas borrando todo el contenido de la unidad, sin posibilidad de recuperarlo(a menos que hasyas hecho previamente un respaldo).

Para formatear el disco duro en fat32 , basta con teclear:

**C:\>format C: /all**

(/all es opcional y puede no estar disponible en todas las versiones de msdos)

## 5. Comandos pendientes 3.

[<http://www.mailxmail.com/curso-ms-dos-avanzado/comandos-pendientes-3>]

### Diskcopy

Otro comando olvidado es diskcopy, este comando sirve para copiar fielmente de un diskette a otro.

La sintaxis es:

diskcopy unidad1 unidad 2

Lo más común sería:

**C:\>diskcopy a: b:**

Este proceso es muy tardado y hay que estar metiendo y sacando los diskettes, es por eso que se recomienda hacerlo desde Windows.

### Sys.

Anteriormente era comun encende runa pc con un disco que contuviera msdos como SO, por lo cual, un diskette de arranque de este tipo era indispensable, para lo cual esta el comando:

### Sys

#### Sintaxis: sys unidad

También es posible hacer un diskette de inicio desde el comando format, con el parámetro /s. el cual agrega al diskette, los archivos necesarios para arrancar el sistema.

La razón por la que se ha omitido su uso es por que resulta más cómodo hacerlo desde windows, además de que en sistemas operativos más complejos, como windows xp, los archivos de arranque suelen variar mucho de las versiones anteriores.

### CHKDSK

Este comando de antaño, era de gran utilidad para buscar y corregir problemas en discos. Desde la llegada de scandisk, este comando ha quedado en desuso, la sintaxis es.

**chkdsk unidad:**

**C:\>chkdsk A: /f (para revisar y reparar un diskette)**

/f sirve para corregir automáticamente los errores encontrados.

/v despliega información detallada acerca del disco y la spistas y o sectores dañados.

### Tree

Este comando sigue vigente hoy en día, pero es poco usado, sirve para mostrar de una manera gráfica (de msdos, no se emocionen) las ubicaciones y rutas de archivos y directorios.

La sintaxis es:

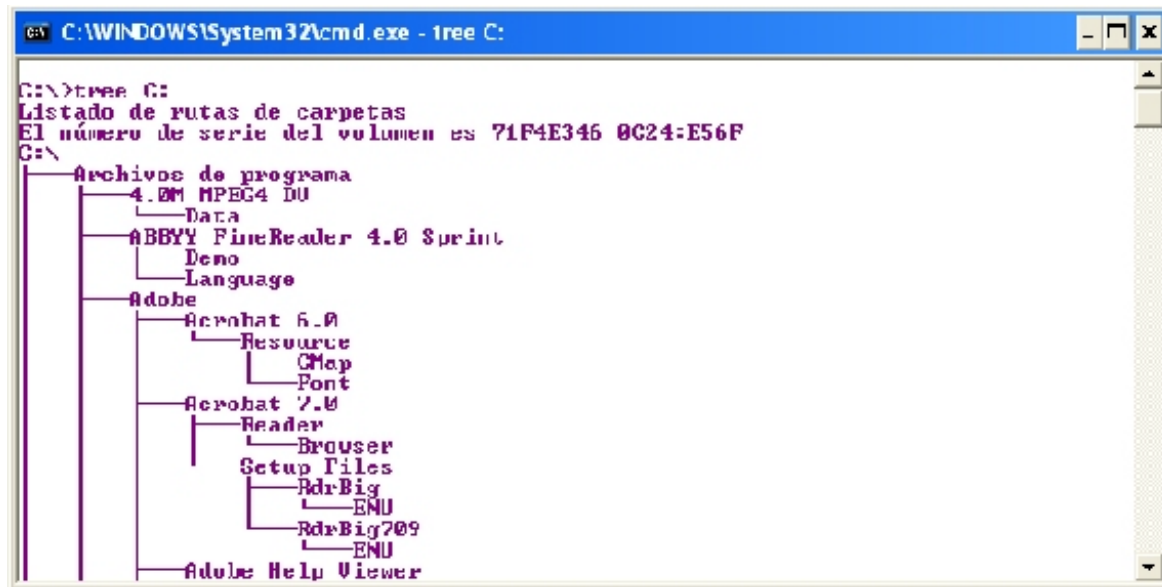
tree unidad:

**C:\>tree C:**



Si se le agrega el parametro /f , mostrara los archivos que hay en cada carpeta.

La razón por la que no se utiliza es por que pese a que presenta lo que pudiera ser una interface gráfica, esta es muy grande y resulta incomodo a la vista, además de que es difícil localizar nuestros archivos.



## Path

Este comando establece una ruta para un programa, por ejemplo si queremos establecer la ubicación de un programa en **C:\dell/program.exe**

**C:\>path C:dell**

Esto hace que cuando queramos ejecutar el programa, la pc lo busque automaticamente en C:/dell.

Es como una especie de "Favoritos" pero no es muy útil.

## 6. Presentación y salida.

[<http://www.mailxmail.com/curso-ms-dos-avanzado/presentacion-salida>]

En msdos, es común que después de teclear un comando, el sistema nos devuelva algún tipo de información, en la misma ventana de msdos o en una emergente.

En msdos (en las últimas versiones), hay una utilidad que nos permite personalizar la salida de la información devuelta por un comando, poniéndola toda dentro de un archivo.

Ejemplo:

Si tecleamos el comando ver, obtenemos lo siguiente:

```
C:\>Ver  
Microsoft Windows XP [Version 5.1.26000]
```

Si deseamos que esta información nos sea devuelta en un archivo de texto, basta con usar el indicador > seguido del nombre del archivo.

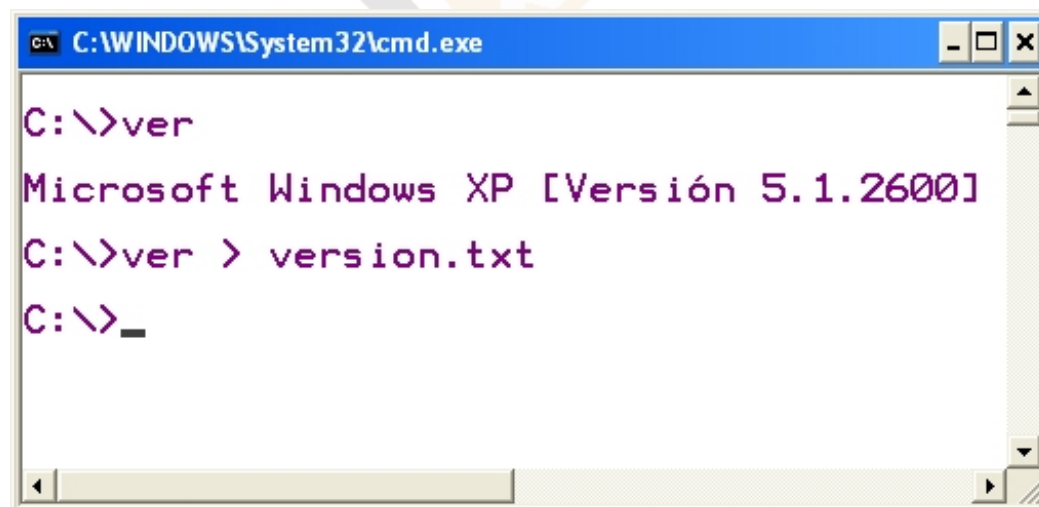
Por ejemplo:

Queremos que la información devuelta por el comando ver, nos sea devuelta en el archivo "version.txt", entonces la sintaxis sería la siguiente:

```
C:\>ver > version.txt
```

¡Listo!

Aquí la comparación:



```
C:\WINDOWS\System32\cmd.exe  
  
C:\>ver  
Microsoft Windows XP [Versión 5.1.2600]  
C:\>ver > version.txt  
C:\>_
```

Si queremos comprobar que ha funcionado, basta con revisar el archivo, que en este caso ( y por default), ha quedado en la ubicación de nuestro prompt.

```
C:\>type version.txt  
Microsoft Windows Xp [Version 5.1.26000]  
C:\>
```

El indicador > funciona con casi todos los comandos, sobretodo los que devuelven información en la misma ventana.

**C:\>comando > ruta/nombre del archivo.extension**

## 7. Attrib

[<http://www.mailxmail.com/curso-ms-dos-avanzado/attrib>]

Este comando nos permite ver y modificar las propiedades de un archivo.

Hay varias maneras de usarlo:

Si se teclea attrib sin ningún parámetro ni modificador, mostrará todos los archivos y sus propiedades en la unidad del prompt.

Podemos especificar una unidad:

### **attrib A:**

Aquí mostrar todos los archivos y propiedades de un diskette. Podemos hacer que nos muestre solo los archivos que cumplen con ciertas propiedades:

**+R** Este parámetro indica Solo lectura.

**-R** Este parámetro desactiva el parámetro anterior e indica que el archivo es editable.

**+A** Este parámetro indica que el archivo es de almacenamiento (un respaldo).

**-A** Este parámetro desactiva el anterior e indica que el archivo no es de almacenamiento.

**+H** Este parámetro indica que el archivo esta oculto.

**-H** Este parámetro desactiva el anterior e indica que el archivo es visible.

Por ejemplo si queremos ver todos los archivos ocultos dentro de C:

**C:\>attrib +H**

o si queremos ver los de solo lectura (que suelen ser los mismos que los ocultos):

**C:\>attrib +R**

De la misma manera podemos cambiar las propiedades de un archivo:

por ejemplo, si queremos proteger un archivo, ocultándolo, digamos el archivo, **example.log** que está dentro de C:

**C:\>attrib example.log +H**

o si queremos que un archivo de solo lectura sea editable:

**C:\>attrib example.txt -R**

Existen así mismo modificadores:

**/s** permite que la propiedad se aplique a la carpeta actual y sus subcarpetas.

**/d** (no siempre esta disponible), procesa una carpeta.

**Nota:** recuerda que en casi todos los comandos es posible usar comodines y trabajar de manera indirecta o directa.

## 8. XCopy.

[<http://www.mailxmail.com/curso-ms-dos-avanzado/xcopy>]

Este comando es mucho más poderoso que copy, ya que sirve para copiar varios archivos a la vez, además de que incluye muchos modificadores:

Copia árboles de archivos y directorios.

**XCOPY** origen [destino] [/A | /M] [/D:fecha] [/P] [/S [/E]] [/V] [/W]  
 [/C] [/I] [/Q] [/F] [/L] [/G] [/H] [/R] [/T] [/U] [/K]  
 [/N] [/O] [/X] [/Y] [/Y] [/Z]  
 [/EXCLUDE:archivo1[+archivo2][+archivo3]...]

**origen** Especifica el archivo(s) que hay que copiar.

**destino** Especifica la ubicación y/o nombres de archivos nuevos.

**/A** Copia sólo archivos con el atributo del archivo establecido, no cambia el atributo.

**/M** Copia sólo archivos con el atributo del archivo establecido, quita los atributos del archivo.

**/D:m-d-y** Copia archivos cambiados durante o después de la fecha. Si no se da una fecha, copia sólo aquellos archivos cuya fecha de origen es m s nueva que la fecha de origen.

**/EXCLUDE:archivo1[+archivo2][+archivo3]...**

Especifica una lista de archivos que contienen cadenas. Los archivos deben tener una cadena por línea.

Cuando cualquiera de las cadenas que coincidan con cualquier parte de la ruta absoluta del archivo que se debe copiar, ese archivo se excluye de ser copiado. Por ejemplo, el especificar una cadena como \obj\ o .obj excluye todos los archivos por debajo del directorio obj o todos los archivos con la extensión obj respectivamente.

**/P** Pregunta antes de crear cada archivo de destino.

**/S** Copia directorios y subdirectorios, exceptuando los vacíos.

**/E** Copia directorios y subdirectorios, incluyendo los vacíos. Igual que /S /E. Puede ser usado para modificar /T.

**/V** Verifica cada archivo nuevo.

**/W** Pide que se presione una tecla antes de copiar.

**/C** Continúa copiando incluso si ocurren errores.

**/I** Si el destino no existe y hay que copiar más de un archivo, se asume que el destino debe ser un directorio.

**/Q** No muestra nombres de archivos mientras copia.

**/F** Muestra todo el origen y el destino de los nombres de archivo mientras se copian.

**/L** Muestra los archivos que se han copiado.

**/G** Permite la copia de archivos destino cifrados para el destino que no es compatible con el cifrado.

**/H** Copia archivos ocultos y también archivos del sistema.

**/R** Sobrescribe archivos de sólo-lectura.

**/T** Crea una estructura de directorio, pero no copia archivos. No incluye directorios o subdirectorios vacíos. /T /E incluye directorios y subdirectorios vacíos.

- /U** Copia sólo archivos que ya existen en el destino.
- /K** Copia atributos. Xcopy normal, restablecer atributos de sólo lectura.
- /N** Copia utilizando los nombres cortos generados.
- /O** Copia archivos de propiedad e información ACL.
- /X** Copia archivos de configuración de auditoría (implica /O).
- /Y** Suprime la pregunta de confirmación para suprimir un archivo de destino existente.
- /-Y** Pregunta por la confirmación de sobrescritura de un archivo de destino existente.
- /Z** Copia archivos de red en modo reinicializable.

El modificador /Y debe ser preestablecido en la variable de entorno COPYCMD. Esto se puede sobrescribir con /-Y en la línea del comando.

### ¿Complicado?

No tanto, veamos un ejemplo práctico:  
copiemos todos los archivos que hay dentro de C, a una subcarpeta:

Los archivos serán **file1.log** y **file2.log** la subcarpeta estará en **C:/dell** sería:  
**C:\>xcopy C: C:dell**

Y esto nos devuelve:

**C:file1.log**  
**C:file2.log**  
**2 archivos copiados.**



```
C:\WINDOWS\System32\cmd.exe
C:\>xcopy C: C:dell
C:file1.log
C:file2.log
2 archivos copiados
```

Este comando permite copiar el contenido de varios archivos dentro de uno mismo, aclaro, no los comprime, solo los mezcla.

**C:\>xcopy C: C:dell\file3.log**

Te preguntara si file3 es un archivo, teclea f, despues te preguntará si deseas sobrescribir, di que si, y listo!, ya tienes file1.log y file2.log fusionados dentro de file3.log (¿para qué sirve esto? ¡ni idea!).

Si quieres dominar este comando, te recomiendo practicarlo mucho y experimentar las combinaciones de parámetros y modificadores, este comando puede ser muy útil, pero sigue siendo mejor hacer una copia de archivos desde otro programa.

## 9. Los 5 errores más comunes de msdos.

[<http://www.mailxmail.com/curso-ms-dos-avanzado/5-errores-mas-comunes-msdos>]

### Errores más comunes de msdos:

En msdos, es común que un comando nos devuelva un error, esto no amerita que nos espantemos, basta con saber identificar.

#### 1.- Anular, Reintentar, Error? (teclas A ,R,E)

Este es quizás el error más común de msdos, puede deberse a diferentes causas:

La unidad sobre la que actúa el comando no está disponible (por ejemplo, no hemos insertado el diskette).

El archivo que estamos usando está dañado o no existe.

Hemos retirado el diskette o disco en el que estábamos trabajando antes de que msdos terminara de ejecutar el comando.

El archivo está protegido.

El disco o diskette está dañado.

Este error hace referencia a un destino u origen defectuoso o inexistente.

#### ¿Qué hacer?

Lo mejor es seleccionar anular, ya que por lo general, el reintentar no funciona, error es lo mismo que anular, solo que registra el error. (pare ello escribe A)

Verifica que todo esté como debiera y revisa la lista de causas posibles.

Vuelve a teclear el comando.

Cambia de unidad por una que no esté dañada.

#### 2.-Sintaxis incorrecta/formato del parámetro incorrecto.

Este error hace referencia a un comando mal tecleado, basta con revisar la sintaxis del comando, recuerda que si tienes duda sobre cómo utilizar un comando, puedes usar/?.

#### 3.- Acceso denegado

Esto ocurre cuando un archivo está siendo usado por sistema o tiene una cadena en el registro de windows.

Lo mejor es volverlo a intentar. Puedes cambiar de archivo o sus propiedades, si no funciona, borra la cadena del archivo desde el registro de windows. (Esto solo si es absolutamente indispensable, ya que puede dañar tu pc gravemente). .

También revisa que el archivo no tenga propiedad de oculto o de solo lectura si piensas editarlo.

#### 4.-No se encuentra el archivo.

Este error hace referencia a algún parámetro del comando (origen o destino).

Puede deberse a que:

Hemos escrito mal la ruta del archivo.

El archivo no existe.

El archivo tiene un nombre demasiado largo. (esto se corrige usando ~)

El nombre del archivo incluye caracteres no permitidos (!,?,",',,,(,),[,],\*,.)

También puede ocurrir cuando tecleamos la una ruta de unidad:

por ejemplo, para referirse a un archivo dentro de C, usamos C:/el nombre del archivo.

Esto es incorrecto, se debe referirse de la siguiente manera:

C:\el nombre del archivo

o cuando utilizamos las diagonales para indicar un subdirectorío, en ocasiones el sistema no reconoce /, es mejor utilizar la diagonal inversa \.

### **5.- Programa no ejecutable en modo dos.**

Este error hace referencia a programas que se ejecutan en ventanas emergentes o desde windows, suele ocurrir si reiniciamos el sistema en modo msdos, lo mejor es ejecutar dicho programa desde windows.



## 10. Dudas

[<http://www.mailxmail.com/curso-ms-dos-avanzado/dudas>]

En los capítulos anteriores hemos mencionado los comandos más comunes y hemos explicado el porqué en su mayoría son poco útiles o más trabajosos que ejecutar windows.

El usuario podría preguntarse:

¿Entonces para qué me estan enseñando msdos? o ¿Si msdos no sirve de nada para qué haces un curso?

Pues bien, he aquí la explicación:

Los comandos listados en los capítulos anteriores los he puesto solo por que me he recibido comentarios acerca de que el curso parecía incompleto y que era inútil mencionar algunos comandos si no los pensaba incluir.

Lo cierto es que msdos no es nada inútil, créanme que no por nada se incluyó en windows xp.

A partir de este capítulo abordaremos solo las aplicaciones más útiles de msdos, así como su incursión en la programación y en el internet.

## 11. Anexo IP

[<http://www.mailxmail.com/curso-ms-dos-avanzado/anexo-ip>]

IP

Internet protocol.

Una dirección IP es la asignación numérica de una dirección, esto funciona así:

Supongamos que tenemos la dirección IP:

148.199.24.3

El primer número hace referencia a la localización mundial, es decir, el país del sitio o la computadora.

El segundo hace referencia a una región local.

El tercer número hace referencia a la zona. (También se refiere a una red.)

El último número hace referencia al número de computadora.

Al usar internet, todos tenemos una dirección IP asignada, esto es parte del DNS (domain name system).

Nuestra Ip, sirve para localizar nuestra computadora desde cualquier parte del mundo.

También los sitios web tienen IP.

A través de una IP podemos conectarnos a un sitio o a una pc remota.

Si deseas más información específica y detallada acerca de las IP y su protocolo, te dejo un enlace:

<http://es.wikipedia.org/wiki/IP>

## 12. MSdos e internet.

[<http://www.mailxmail.com/curso-ms-dos-avanzado/msdos-internet>]

### MSdos e internet.

Hay quienes piensan que msdos es totalmente incompatible con internet, esto no es así, hay una serie de comandos que demuestran lo contrario.

### ipconfig

Este comando permite obtener la dirección ip de un equipo, así como el nombre del host, máscara de subred y la liberación de ciertos paquetes.

```
C:\>ipconfig
```

Configuración IP de Windows

Adaptador Ethernet Conexión de área local :

```
    sufijo de conexión específica DNS: xxxxxxxxx
    Dirección Ip.....:xxx.xxx.x.xx
    Máscara de subred.....:xxx.xxx.xxx.x
    Puerta de enlace predeterminada :xxx.xxx.x.xxx
```

```
C:\>
```

```
/all
```

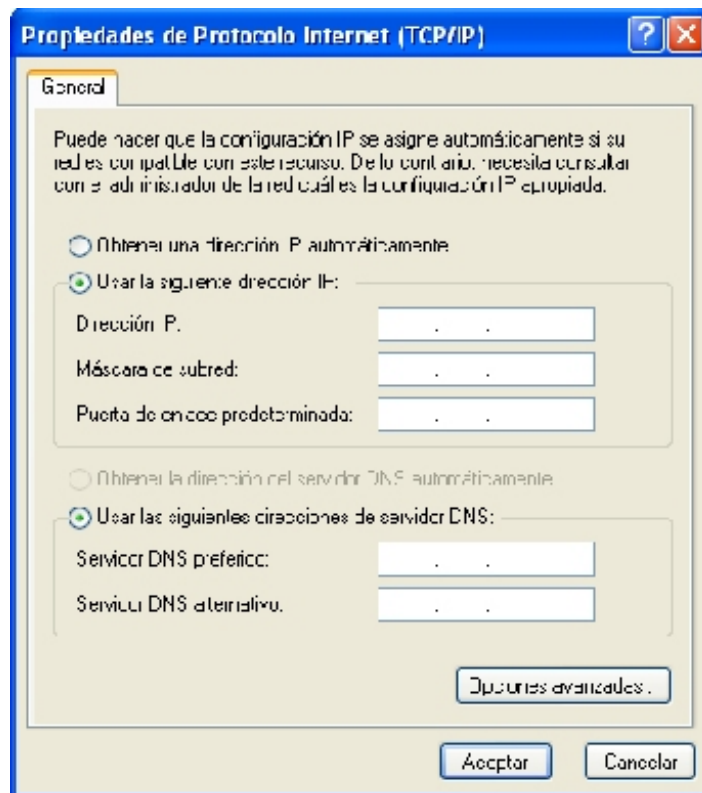
Pues bien, si queremos información más detallada podemos agregar el modificador /all.

Este comando permite comodines y diversas combinaciones.

Un ejemplo útil de este comando es:

Cuando reformateamos o reinstalamos una red:

Hace poco tuve que reformatear una pc y necesitaba guardar los datos de la red de área local , y el internet, esto esta en panel de control, propiedades de internet, tcp/Ip, propiedades:



Aquí están en blanco, pero en esa pc, había las direcciones IP de la red, en lugar de escribirlas en un papel, teclee:

**C:\>ipconfig /all >respaldo.txt**

y listo! ya tenía esa información en un fichero, así, previo a reinstalar windows, solo copie los datos del block de notas y reinstale la red.

**Nota:** este comando debe usarse mientras el equipo esta conectado a internet, ya que de lo contrario, tiende a devolver direcciones ip , dns , puertos , impresoras y máscaras iguales.

Si deseas más detalles de los modificadores de este comando aquí los enlisto:

`/?` muestra la ayuda  
`/all` muestra toda la información de configuración.  
`/release` libera la dirección IP para el adaptador específico.  
`/renew` renueva la dirección IP para el adaptador específico.  
`/flushdns` purga la caché de resolución de DNS.  
`/registerdns` actualiza todas las concesiones y vuelve a registrar los nombres DNS.  
`/displaydns` muestra el contenido de la caché de resolución DNS.  
`/showclassid` muestra todas las id. de clase dhcp permitidas para este adaptador. `/setclassid` modifica la id. de clase dhcp.

De manera predeterminada se muestra solamente la dirección IP, la máscara de subred y la puerta de enlace para cada adaptador enlazado con TCP/IP. Para Release y Renew, si no hay ningún nombre de adaptador especificado, se liberan o renuevan las concesiones de dirección IP enlazadas con TCP/IP. Para Setclassid, si no hay Classid especificada, se quita Classid.

Ejemplos: `ipconfig` ... muestra información  
`ipconfig /all` ... muestra información detallada

```
ipconfig /renew          ... renueva todos los adaptadores    >
ipconfig /renew EL*      ... renueva cualquier conexión cuyo nombre comience con
EL
ipconfig /release *Con*  ... libera todas las conexiones que coincidan por ejemplo:
"Conexión de área local 1" o "Conexión de área local 2"
```

**Nota2:** Los temas referentes a la arquitectura tcp/ip y DNS se consideran fuera del alcance de este curso.

**Nota3:** Este comando podemos encontrarlo también en las versiones antiguas de windows, donde nos desplegara una ventana, solo que en lugar de ipconfig, se debe teclear winipcfg.

## 13. Ping

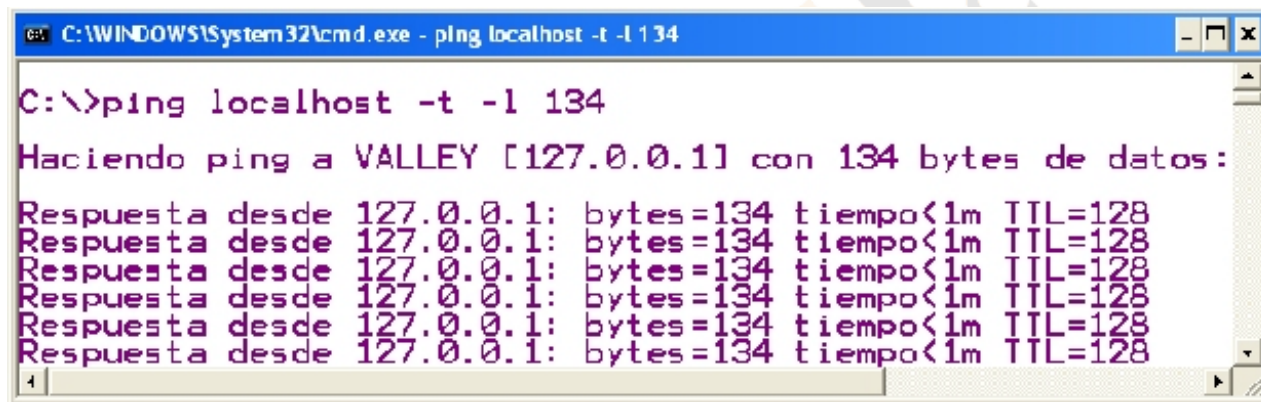
[<http://www.mailxmail.com/curso-ms-dos-avanzado/ping>]

El comando ping es una muestra más de la incursión de msdos en internet. Para empezar, comencemos por definir que es "hacer ping".

Hacer ping a algo es "tocarlo", para ver si responde y así saber que está bien. En el caso de internet, el comando ping sirve para revisar si un servidor esta activo o no, para ver si existe una página web y para comprobar el rendimiento de un server.

Por ejemplo, podemos comprobar si nuestra pc responde a ping (solo es para ejemplificar):

**C:\>ping localhost -t -l 134**



```
C:\WINDOWS\System32\cmd.exe - ping localhost -t -l 134

C:\>ping localhost -t -l 134

Haciendo ping a VALLEY [127.0.0.1] con 134 bytes de datos:

Respuesta desde 127.0.0.1: bytes=134 tiempo<1m TTL=128
Respuesta desde 127.0.0.1: bytes=134 tiempo<1m TTL=128
Respuesta desde 127.0.0.1: bytes=134 tiempo<1m TTL=128
Respuesta desde 127.0.0.1: bytes=134 tiempo<1m TTL=128
Respuesta desde 127.0.0.1: bytes=134 tiempo<1m TTL=128
```

Empecemos por desglosar lo anterior:

ping : esta es la manera de acceder a nuestro comando

localhost : esta es la manera de referirnos a nuestra propia pc, si no queremos usar localhost, podemos obtener el nombre de nuestra pc tacleando el comando

hostname . ejemplo:

C:\>hostname

Mi pc name

-t esto indica que haremos ping hasta que el servidor nos responda o se detenga.

-l (-L) esto indica que enviaremos el tamaño del buffer, es decir, la fuerza con que "tocaremos" al servidor.

134 esta es la "fuerza" que estamos enviando, se envía en paquetes, la misma cantidad de paquetes que enviemos, es la cantidad de paquetes que el servidor nos respondera.

Aquí solo he enviado 134, pero el tamaño del buffer puede variar entre 1 y 65500.

### El ping de la muerte:

Hasta hace unos años existía una vieja técnica para tirar servidores, esto consistía en enviar un paquete de datos a través del ping, lo suficientemente grande para dañar un server (era como inyectarle tráfico).

Esto era más o menos así:

Teníamos la página que queríamos perjudicar.

<http://www.ejemplo.com>

Primero investigabamos el servidor (lo veremos mas adelante), luego que lo obtenemos, para este ejemplo:

<http://www.myserver.com>

Escribíamos lo siguiente:

**C:\>ping myserver.com -t -l 65605**

algunos servidores no resistían mucho esta técnica.

Ahora los servidores tienen siempre respaldos, además de que soportan grandes cantidades de información, sumado a que msdos solo nos permite enviar un paquete de 65500.

A continuación listo datos más precisos sobre como usar ping.

Uso: ping [-t] [-a] [-n cuenta] [-l tamaño] [-f] [-i TTL] [-v TOS]  
[-r cuenta] [-s cuenta] [[-j lista-host] | [-k lista-host]] [-w tiempo de espera]  
nombre-destino

Opciones:

-t Ping el host especificado hasta que se pare. Para ver estadísticas y continuar - presionar Control-Inter; Parar - presionar Control-C.  
-a Resolver direcciones en nombres de host.  
-n cuenta Número de peticiones eco para enviar.  
-l tamaño Enviar tamaño del búfer.  
-f Establecer No fragmentar el indicador en paquetes.  
-i TTL Tiempo de vida.  
-v TOS Tipo de servicio.  
-r cuenta Ruta del registro para la cuenta de saltos.  
-s count Sello de hora para la cuenta de saltos.  
-j lista-host Afloja la ruta de origen a lo largo de la lista- host.  
-k lista-host Restringir la ruta de origen a lo largo de la lista- host.  
-w tiempo de espera Tiempo de espera en milisegundos para esperar cada respuesta.

## 14. Netstat.

[<http://www.mailxmail.com/curso-ms-dos-avanzado/netstat>]

Este comando sirve para mostrarnos información detallada acerca del uso del internet, por lo cual solo sirve mientras se esta conectado a internet.

Hay varias maneras de utilizarlo, por ejemplo, si estamos viendo una pagina y tecleamos el comando solo:

**C:\>Netstat**

Proto	Dirección local	Dirección remota	Estado
TCP	Mipcname:####	###.###.###.#	Established

Esto hace referencia a:

**Proto:** protocolo, se refiere al protocolo que se esta usando, (tcp/ip).

**Dirección local:** se refiere a la id o la dirección que se le ha asignado dentro de nuestra pc.

**Dirección remota:** Esto hace referencia a la ubicación desde la cual se esta llamando la página web, la dirección ip.

**Estado:** Esto se refiere a qué acción se esta llevando acabo, por ejemplo: **Established**

**Established**, hace referencia a que ya nos hemos conectado a una página o un servidor. **Listening** se refiere cuando estamos recibiendo o mandando algo a un servidor (subir, bajarb archivos o intercambiar información), **Waiting:** estamos esperando a que la conexion se establezca, es decir, la respuesta de un servidor (sesión, ftp, etc..)

Este comando nos ayuda a obtener el servidor de una página web. Por ejemplo:

Si queremos saber el servidor de : <http://www.mypage.com>, (es solo un ejemplo) mientras estamos viendo esta página tecleamos:

**C:\>Netstat -a**

Listo, ahora entre muchas conexiones y letras, veremos una direccion del tipo: myserver.com seguido de una dirección ip.

Existe un truco, para obtener la ip de alguien a través de este comando.

En msn messenger, cuando platiques con alguien, envíale un archivo muy pesado,cuando acepte, teclea:

**netstat -a** o también **netstat -n**

Para que esto funcione, no debes tener paginas web abiertas o alguna ventana que este utilizando internet.

Listo, entre todo el palabrerío se encuentra (hasta abajo), su dirección ip.

**Nota:** esto también nos da direcciones ip y servidores, solo que es un relajo ubicarlos!

Hay ciertos detalles acerca de este comando que es importante conocer:

- 1.-Solo funciona mientras se esta conectado.
- 2.-Cuando usamos internet, tendemos a abrir varias páginas a la vez, mientras más



ventanas se esten utilizando, más confusa será la información que nos devuelva este comando.

3.-Como al usar internet , se hace un intercambio dinámico de información, este comando cambia siempre, por lo cual es necesario actualizar.

Si usamos el comando nos devolverá cierta información, pero si lo usamos 1 min después nos dará otra. Cada vez que usemos este comando, nos mostrará los procesos de Internet, por lo cual para ver cada nuevo proceso (una página, un protocolo), será necesario volverlo a teclear.

Parámetros y Modificadores:

NETSTAT [-a] [-e] [-n] [-o] [-s] [-p proto] [-r] [intervalo]

- a Muestra todas las conexiones y puertos de escucha.  
(Normalmente, el extremo servidor de las conexiones no se muestra).
- e Muestra estadísticas Ethernet. Se puede combinar con la opción -s.
- n Muestra números de puertos y direcciones en formato numérico.
- o Muestra la Id. de proceso asociado con cada conexión.
- p proto Muestra conexiones del protocolo especificado por proto; que puede ser TCP, UDP, TCPv6 o UDPv6. Si se usa con la opción -s para mostrar estadísticas por protocolo, proto puede ser TCP, UDP, TCPv6 o UDPv6.
- r Muestra el contenido de la tabla de rutas.
- s Muestra estadísticas por protocolo. De forma predeterminada, se muestran para IP, IPv6, ICMP, ICMPv6, TCP, TCPv6, UDP y UDPv; se puede utilizar la opción -p para especificar un subconjunto de los valores predeterminados.
- intervalo Vuelve a mostrar las estadísticas seleccionadas, haciendo pausas en el intervalo de segundos especificado entre cada muestra. Presione Ctrl+C para detener la actualización de estadísticas. Si se omite, netstat imprimirá la información de configuración una vez.

## 15. Nbtstat

[<http://www.mailxmail.com/curso-ms-dos-avanzado/nbtstat>]

### Nbtstat (NetBios Remote Machine Name Table)

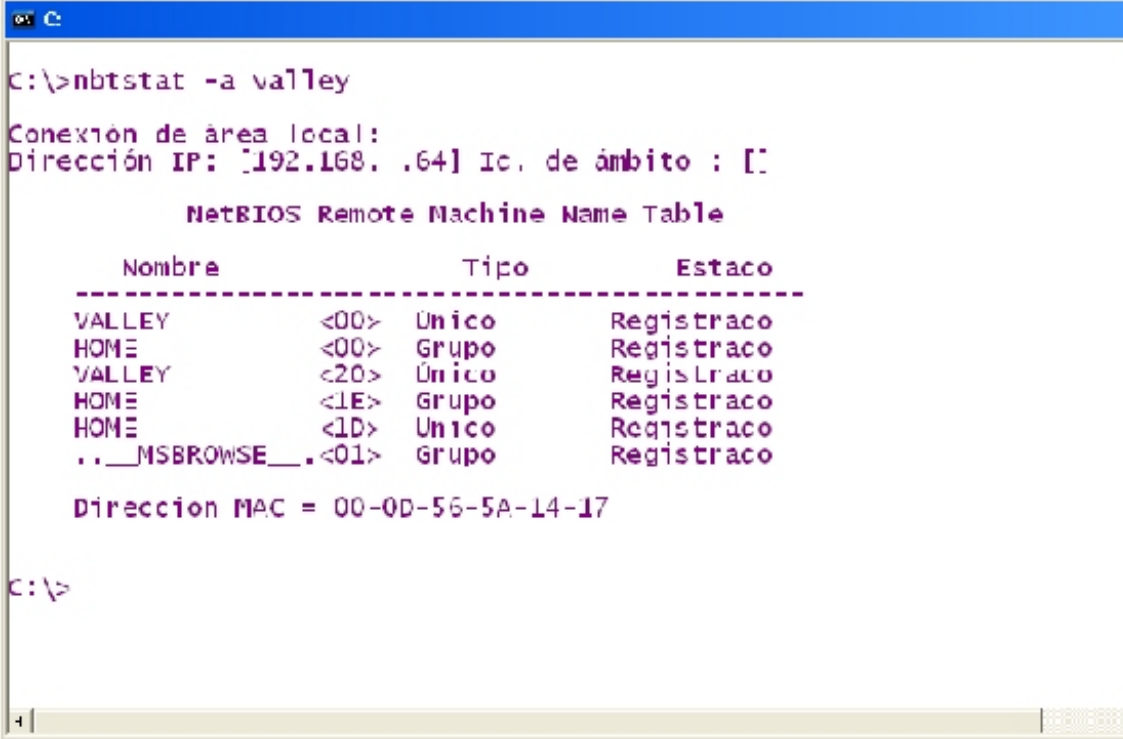
Este comando sirve para obtener información de equipos remotos como:

nombre del host  
IP  
Puertos  
estado

La sintaxis es:

nbtstat -a nombre del host [-modificadores] y también nbtstat -A dirección IP [-modificadores]

En el primer caso usaremos el nombre conocido de algún host (de alguna computadora) generalmente en red, en mi caso mi pc se llama valley, así que podemos usar (solo como ejemplo, pues no nos dira nada útil) nbtstat:



```
C:\>nbtstat -a valley

Conexion de área local:
Direccion IP: [192.168. .64] Id. de ámbito : []

      NetBIOS Remote Machine Name Table

      Nombre                Tipo                Estado
-----
VALLEY      <00>                Unico              Registrado
HOME        <00>                Grupo              Registrado
VALLEY      <20>                Unico              Registrado
HOME        <1E>                Grupo              Registrado
HOME        <1D>                Unico              Registrado
.._MSBROWSE_ <01>                Grupo              Registrado

Direccion MAC = 00-0D-56-5A-14-17

C:\>
```

Antiguamente en el hacking, nbtstat era particularmente útil para introducirse vía internet dentro de otro equipo, veamos:

Hay una linea (en VALLEY) que dice "20", cuando hacíamos nbtstat a una dirección IP y nos devolvía el número 20, sabíamos que podíamos entrar, pues este número indica la posibilidad de enlazarse con el equipo, en este caso, devuelve 20 por que es mi propia PC.

Hacer nbtstat a una dirección ip es difícil, y la información que nos proporciona

suele estar dispersa, solo he incluido este comando como referencia, pero procedo a continuar con otros.

## 16. Editar el Registro de Windows (1)

[<http://www.mailxmail.com/curso-ms-dos-avanzado/editar-registro-windows-1>]

El registro de windows es la parte fundamental del sistema operativo, donde se guardan cadenas y valores, tales como: los programas que se inician al prender tu pc, el tiempo de apagado, los seriales, el tipo de archivos que soporta cada programa, el idioma, las url, licencias, etc.

Este tema es bastante avanzado, por lo que debo hacer dos aclaraciones:

- 1.-No se debe editar el registro de windows a menos de que sea estrictamente necesario, cualquier error, conducirá a daños graves en tu computadora.
- 2.-Es recomendable hacerlo desde windows en regedit.exe

Es posible editar el registro de windows a través de msdos a través del comando REG.

### REG

Este comando tienes muchas formas de usarse:

#### reg query

esta forma del comando query nos permite visualizar cadenas y valores del registro, por ejemplo, es bien sabido que **la mayoría de los virus se alojan en:**

**hkeylocal machine\software\microsoft\windows\currentversion\run**

(de esa forma, el virus se ejecuta cada vez que tu pc enciende.) Si queremos revisarlo (solo si se tiene experiencia en esto):

#### reg query cadena /modificador

hay 2 cadenas básicas para reg : **hklm** (local machine) y **hkcu** (current user), pero también puedes usar las demás raíces: (**hkcr** , **hku** y **hkcc**) para acceder, basta con escribir la abreviación:

Entonces, para revisar si tengo virus, tecleo:

**C:\>reg query hklm\software\microsoft\windows\currentversion\run**

```
C:\>reg query hklm\software\microsoft\windows\currentversion\run

! REG.EXE VERSION 3.0

HKEY_LOCAL_MACHINE\software\microsoft\windows\currentversion\run
    IMJPMIG8.1 REG_SZ "C:\WINDOWS\IME\imjp8_1\IMJPMIG.EXE" /Spoil /Rem
Migration32
    PHIME2002ASync REG_SZ C:\WINDOWS\System32\IME\TINTLGNT\TINTSET
YNC
    PHIME2002A REG_SZ C:\WINDOWS\System32\IME\TINTLGNT\TINTSETP.EXE /I
    IgfxTray REG_SZ C:\WINDOWS\System32\igfxtray.exe
    HotKeysCmds REG_SZ C:\WINDOWS\System32\hkcmd.exe
    AVG7_CC REG_SZ C:\ARCHIV~1\Grisoft\AVGFRE~1\avgcc.exe /STARTUP
    QuickTime Task REG_SZ "C:\Archivos de programa\QuickTime\qttas
atboottime
    NeroFilterCheck REG_SZ C:\WINDOWS\System32\NeroCheck.exe

HKEY_LOCAL_MACHINE\software\microsoft\windows\currentversion\run\Optiona
nts

okoto:\>
```

En este caso, veo que no hay virus. (Estos temas se pueden revisar en el curso de virus de mailxmail).

Para estar más seguro, podría haber agregado **/s** al final, para revisar subcadenas y subclaves. También se puede usar **/v** si deso consultar solo una clave específica.

**/ve**, es para evitar ver todas las claves dentro de una cadena.

## 17. Editar el registro de windows (2)

[<http://www.mailxmail.com/curso-ms-dos-avanzado/editar-registro-windows-2>]

En el capítulo anterior vimos como ver los valores de una cadena en el registro de windows, ahora veremos como editarlos:

Supongamos que usando reg query hubiésemos encontrado un virus, en este caso sería **virus.exe** (como ejemplo) y estaría en **System32** y alojado en la cadena **run**.

### Reg delete

Este comando nos permite borrar una clave de una cadena y/o todas sub claves y valores.

reg delete cadena /modificador /clave

En este caso, queremos borrar el archivo **virus.exe**, esto sería:

```
C:\>reg delete
```

```
hklm\software\microsoft\windows\currentversion\run\virus.exe
```

El programa siempre nos preguntará si deseamos continuar (y/n).(yes,si/no.no)

El programa REG (comando externo), nos permite usar las abreviaturas de msdos, veamos un ejemplo:

Si queremos borrar la clave **QuickTime Task** de run:

```
C:\>reg delete
```

```
hklm\software\microsoft\windows\currentversion\run\QuickTime Task
```

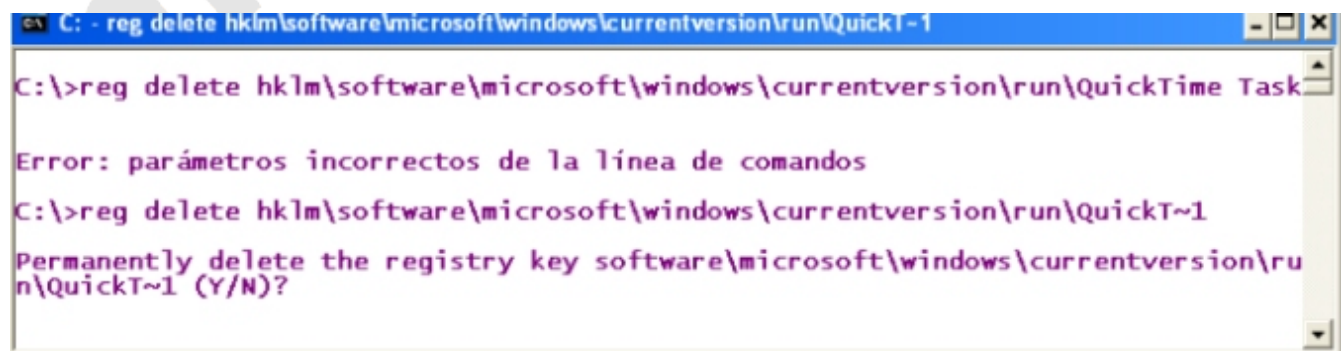
Nos devolverá:

error, parámetros incorrectos, esto se debe al espacio entre QuickTime y Task, para solucionar esto, usaremos un tema de nuestro primer curso: ~1.

Esto sería así:

```
C:\>reg delete
```

```
hklm\software\microsoft\windows\currentversion\run\QuickT~1
```



```
C: - reg delete hklm\software\microsoft\windows\currentversion\run\QuickT~1

C:\>reg delete hklm\software\microsoft\windows\currentversion\run\QuickTime Task

Error: parámetros incorrectos de la línea de comandos

C:\>reg delete hklm\software\microsoft\windows\currentversion\run\QuickT~1

Permanently delete the registry key software\microsoft\windows\currentversion\run\QuickT~1 (Y/N)?
```

**Nota:** Hay varios modificadores disponibles en: reg delete /?

## 18. Editando el registro de windows (3)

[<http://www.mailxmail.com/curso-ms-dos-avanzado/editando-registro-windows-3>]

Una de las funciones más útiles de REG, es exportar en archivos .reg las cadenas y claves, así como sus valores.

Con esto, podremos restaurar el registro de windows en caso de ser necesario (basta con hacer doble click en el archivo reg o usar import). Esto evitará perder información valiosa, en caso de virus podremos dejar el limpio el registro de windows (siempre y cuando la cadena no lo contenga), además, en caso de equivocarnos al editar el registro, no habrá problema si podemos restaurarlo.

### reg export

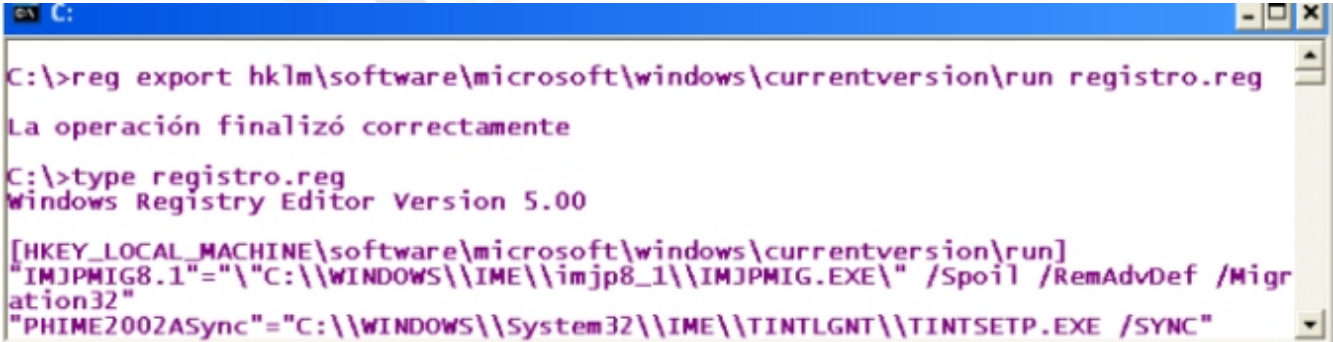
#### (reg export cadena archivo.reg)

Nos permite guardar en nuestro disco duro la información del registro de windows, en este caso, vamos a respaldar una de las cadenas más importantes del registro (si, adivinaron): RUN, en el archivo registro.reg

Esto sería:

```
C:\>reg export hklm\software\microsoft\windows\currentversion\run  
registro.reg
```

Los archivos se guardarán en la ubicación del prompt, en este caso C:\. si queremos revisar como ha quedado, podemos verlo con type:



```
C:\>reg export hklm\software\microsoft\windows\currentversion\run registro.reg  
La operación finalizó correctamente  
C:\>type registro.reg  
Windows Registry Editor Version 5.00  
[HKEY_LOCAL_MACHINE\software\microsoft\windows\currentversion\run]  
"IMJPMIG8.1"="C:\\WINDOWS\\IME\\imjp8_1\\IMJPMIG.EXE" /Spoil /RemAdvDef /Migration32"  
"PHIME2002ASync"="C:\\WINDOWS\\System32\\IME\\TINTLGNT\\TINTSETP.EXE /SYNC"
```

### Reg Import

Una vez respaldado, podemos restaurar el registro con el archivo.reg, mediante reg import, en este caso, si queremos restaurar (no hay para que) el registro de run desde el archivo registro.reg:

```
C:\>reg import registro.reg  
¡Listo!
```

### Reg add

Nos permite agregar una cadena, la sintaxis es:

**reg add clave /v nombre /t tipo /d valor**

por ejemplo: si queremos agregar la clave **example** a la cadena **hklm\Software** con el valor: "hola mundo", y del tipo **SZ** sería:

**C:\>reg add hklm\software /v example /t reg\_sz /d hola mundo**

El tipo default de las claves en el registro es sz (por lo general, el tipo si se escribe en mayúscula) pero podemos especificar mediante /t:

reg\_sz, reg\_multi\_sz, reg\_dword, reg\_none, reg\_binary, reg\_expand\_sz , etc.

si deseamos forzar a una sobreescritura, agregamos al final: /f.

**Nota: en los temas del editado del registro de windows, no se han incluido todos los parámetros y modificadores, así también se han omitido algunos ejemplos, la razón es la siguiente:**

Los temas del registro de windows, se encuentran fuera de este curso, además, el usuario que desee editar el registro, debe tener los suficientes conocimientos al respecto, por lo cuál, este tema solo sirve a aquellos usuarios con experiencia en cadenas y claves, siendo así que no es necesario incluir ejemplos para todos y cada uno de los tipos de clave y otros temas.



## 19. Programación mediante msdos

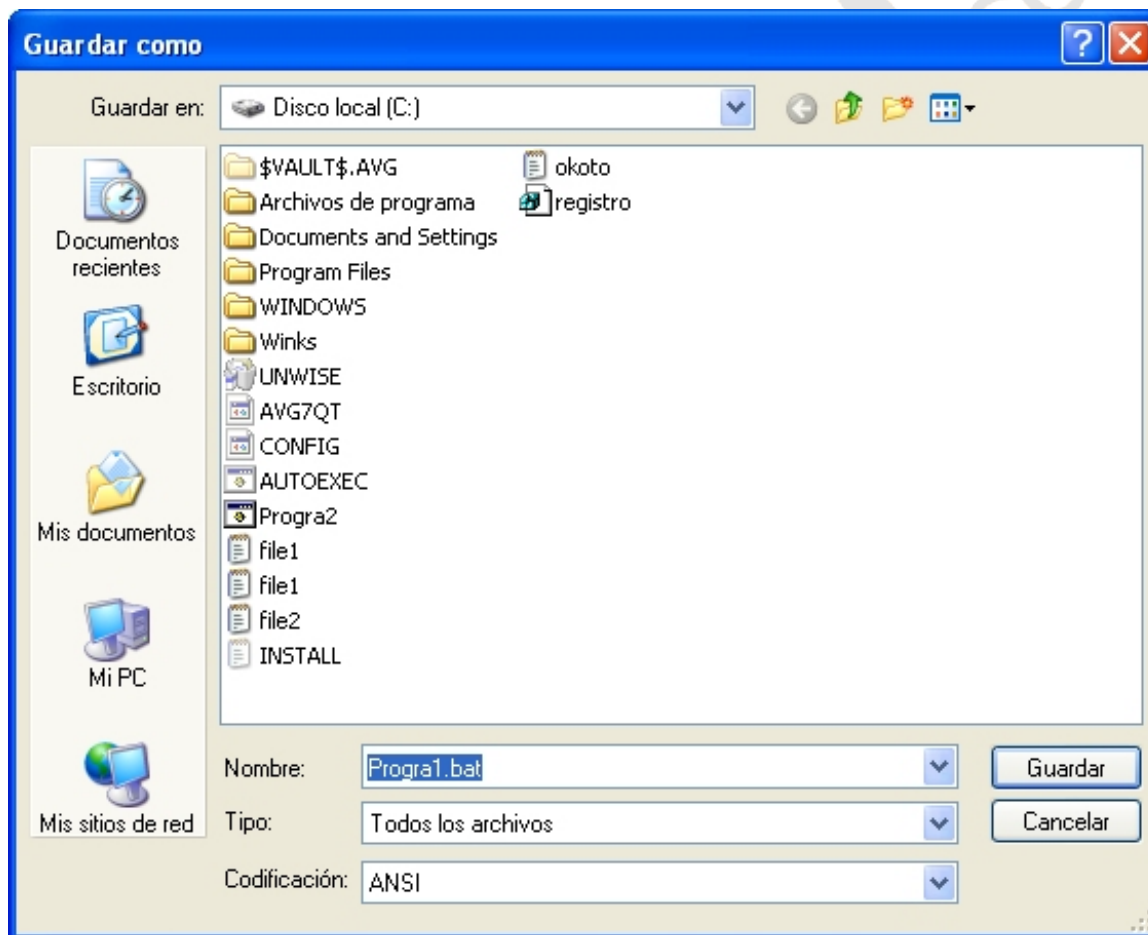
[<http://www.mailxmail.com/curso-ms-dos-avanzado/programacion-msdos>]

Es posible crear en msdos pequeños programas basados en los comandos vistos, estos programas serán del tipo ejecutable en dos y tendrá las extensiones:

.bat .cmd .pif .com(inusual en batch).

Para ello usaremos el lenguaje Batch, uno de los más antiguos y poderosos que hay hoy en día.

Para guardar un programa debemos cambiar su extensión por .bat, basta con hacerlo en un bloc de notas, en guardar como seleccionar "todos los archivos" y escribir el nombre del programa+.bat



En un editor de texto (yo sugiero **bloc de notas**) escribe:

```
@echo off
rem mi primer programa
title programa 1
echo hola mundo
pause
exit /b
```

Ahora veamos:

**Guárdalo como progra1.bat (todo tipo de archivos):****@echo off**

El @ se coloca al principio de un programa en msdos  
**echo off**

Indica que no se mostraran los comandos, solo se mostrara la información que estos devuelva, es decir, no se vera title, ni echo, solo se ejecutara. Si se cambia el valor por on podremos ver todo lo que hemos escrito en nuestro programa.

**rem mi primer programa**

Sirve para escribir comentarios, todo lo que se escriba después de rem no se ejecutará, solo servirá para hacernos recordatorios dentro del código fuente de nuestro programa.

**echo hola mundo**

echo muestra un mensaje en pantalla, en este caso: hola mundo.

**Pause**

Evita que el programa termine, pide que el usuario oprima alguna tecla antes de continuar con al ejecución del programa. Es muy útil.

**exit /b**

/b indica a msdos que finalizó nuestro programa pero que deseamos continuar en msdos, si no se pone /b después de exit, el programa cerrara msdos.

**Ejecutando:**

Para probar nuestro programa basta con hacer doble click en nuestro archivo Progra1.bat o ejecutarlo desde msdos.

Si está bien hecho, mostrara el título "Programa 1", así como "hola mundo" en un mensaje en la pantalla, sin que se vean nuestros comandos.

Nota: recordemos que cuando ejecutamos un programa en msdos, no es necesario escribir la extensión del mismo, en este caso, basta con escribir Progra1.

## 20. Programación mediante msdos (2)

[<http://www.mailxmail.com/curso-ms-dos-avanzado/programacion-msdos-2>]

Probemos con otro programa:

Programa 2 (progra2.bat), escribimos:

```
@echo off
rem mi segundo programa
title Programa 2
echo hola!!! voy a mostrate el archivo file1.txt:
pause
if exist C:/file1.txt type file1.txt
if not exist C:/file1.txt echo error
pause exit /b
```

Veamos:

**pause**

Esto nos permitirá hacer una pequeña pausa donde el usuario deberá presionar una tecla para continuar, similar a lo que hace el botón de pausa en el teclado.

**if exist C:/file1.txt type file1.txt**

Si existe el fichero file1.txt en C, continua con la instrucción type.

**if not exist C:/file1.txt echo error**

Si no existe el fichero file1.txt en C, devuelve un mensaje de error. Este programa muestra, en caso de que exista, el archivo file1.txt en caso, de que éste no exista, devolvera la palabra "error".

Se recomienda colocar el archivo Progra2.bat en la misma ubicación de file1.txt, en este caso, en C.

## 21. Programación mediante msdos 3

[<http://www.mailxmail.com/curso-ms-dos-avanzado/programacion-msdos-3>]

Hora de probar otro programa:

```
@echo off
rem mi tercer programa
title Programa 3
echo mostrar archivo file1.txt
pause
if exist C:\file1.txt goto si
echo error
:si
type C:\file1.txt
pause
exit /b
```

Ahora bien, una vez guardado como .bat en C: (de preferencia), probemos ejecutarlo.

### **goto**

Esta declaración nos permite llamar a una función definida hasta abajo en :funcion . En este caso, si el archivo file1.txt existe, ira directo a "si" , en caso contrario mostrará un mensaje de "error".

### **:si**

Esta es la función, aquí podemos escribir varios comandos, los cuales se ejecutaran desde **goto**, se escriben dos puntos y el nombre de la función (subrutina hasta abajo).

Si el programa funciona bien, nos mostrará el contenido del archivo file1.txt y en caso de que no exista, nos mostrará la palabra error.

**Nota: la mayoría de los errores en batch (este lenguaje de programación en Dos) se deben a un error en la ruta de los archivos, si escribimos:**

type C:/file1.txt , probablemente mande a error, en batch es mejor escribir: type C:\file1.txt (la diagonal inversa).

## 22. Programación mediante msdos (4)

[<http://www.mailxmail.com/curso-ms-dos-avanzado/programacion-msdos-4>]

Una vez más, abrir el bloc de notas o algún editor de texto, para hacer otro archivo .bat :

```
@echo off
rem mi cuarto programa
title Programa 4
echo ¿como te llamas?
set /P nombre=
echo hola %nombre% !!!
pause
exit /b
```

Ahora:

**set**

Set , nos permite definir una variable, es decir, cualquier dato dentro del programa, la sintaxis es: set variable=valor, por ejemplo:

**set total=8**

Lo anterior indica que el valor de "total" es 8.

**set /P variable=**

Esto sirve para que el programa permita al usuario escribir algo, que será luego usado como una variable, se escribe /P después de set y el valor se deja vacío.

**echo %variable%**

Esto sirve para llamar a la variable, siempre se escriben % nombre de la variable%

En este caso, es el nombre que el usuario escribió.

Si el programa funciona, le pregunta al usuario su nombre y luego le devuelve un saludo.

## 23. Programación mediante msdos. Final

[<http://www.mailxmail.com/curso-ms-dos-avanzado/programacion-msdos-final>]

Ahora si, viene un programa que se debe llamar prografinal.bat

```
@echo off
rem programa final
title Programa 5
echo veamos como andas en las matemáticas:
pause
echo dime ¿Cuanto es 12 X 14
set /P respuesta=
if %respuesta%==168 goto fine
echo por favor trata de nuevo
prografinal
:fine
echo has acertado!!! eres un genio!!!
pause
exit /b
```

Ahora bien:

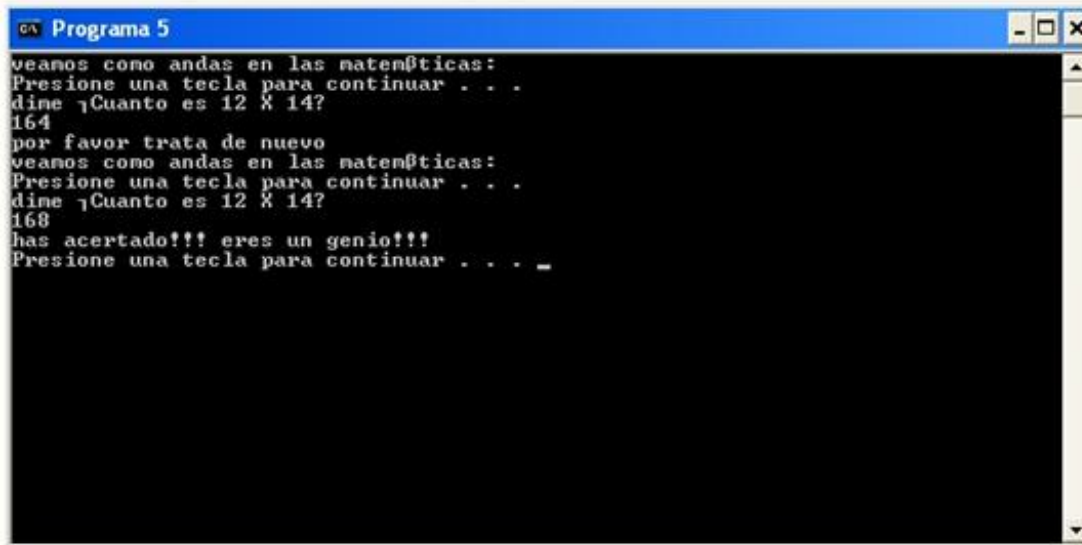
```
if %respuesta%==168 goto fine
```

Una vez definida la variable respuesta, se procede a revisar que esta cumpla con la condición establecida por if, si se cumple, se ira a la subrutina de goto (fine) en caso de no ser así, se procederá a ejecutar los comandos bajo If. En este caso, que respuesta sea igual a 168. la sintaxis para igualdad es:

```
if %variable%==valor goto subrutina (o cualquier comando)
```

**prografinal**

Este es el nombre del programa en fichero y lo he puesto para que en caso de que el usuario no escriba la respuesta correcta, el programa vuelva a ejecutarse, recuerda que para ejecutar un programa en Dos, basta con escribir su nombre sin su extensión.



```
Programa 5
veamos como andas en las matemáticas:
Presione una tecla para continuar . . .
dime ¿Cuanto es 12 X 14?
164
por favor trata de nuevo
veamos como andas en las matemáticas:
Presione una tecla para continuar . . .
dime ¿Cuanto es 12 X 14?
168
has acertado!!! eres un genio!!!
Presione una tecla para continuar . . .
```

En el caso de If, hay varias maneras de usarlo, son a veces válidos comparadores aritméticos, peor para esto, hay funciones previamente definidas:

**equ** igual a  
**neq** no igual a  
**lss** menor que  
**leq** menor o igual a  
**gtr** mayor que  
**geq** mayor o igual a  
eso se usa así:  
por ejemplo:  
**if %respuesta% equ 168 goto fine**

ahora bien, se puede usar **Else** bajo if: (en caso contrario a la condición).  
if condición goto comando **else** goto otra subrutina  
o también  
if condición goto subrutina  
goto otra subrutina  
(funciona igual)

**Nota:** los comandos deberán ir en ( ) :

**if exist file1.txt (del file1.txt) else goto fin.**

En realidad, no es necesario escribir else, basta con If.

Si el programa funciona, preguntará al usuario el resultado de 12X14 , en caso de escribir una respuesta correcta, el programa termina , en caso contrario, el programa se repite.

**Nota:** en ocasiones, para ejecutar un archivo .bat se utiliza el comando **call**, es mejor escribir simplemente el nombre del programa y esta se ejecutará solo; **call**, tiende a trabar la computadora.

## 24. Trucos , secretos y limitaciones:

[<http://www.mailxmail.com/curso-ms-dos-avanzado/trucos-secretos-limitaciones>]

Existen en MSDos comandos ocultos y deshabilitados, veamos:

**choice** en batch, sirve para dar al usuario a escoger entre varias opciones

**ver /R** muestra información del sistema

**telnet** este programa aún se incluye, sirve para conectarse a otra computadora vía internet, se muestra, pero esta deshabilitado.

Existen muchas cosas más ocultas y mucho por hacer en msdos, para ello , existe la versión libre:

<http://www.freedos.org>

de código abierto y extensible. No todo en la vida es microsoft. (compatible con varios os).

En batch, es posible hacer tanto un virus, como un pequeño antivirus, es cuestión de creatividad.

Para ver todos los usos de cada módulo en batch (if, echo,etc.) basta con escribirlos seguidos de /?.

Batch suele usarse para complementar programas, por ejemplo, el compresor de música, **lame**, utiliza varios batch, que puedes editar para obtener mp3.

### Crea tu propio comando:

Para ello has cualquier programa en batch y colocalo en **System32** de la carpeta **windows**, listo, allí es donde se guardan todos los comandos, ya no tienes que saber la ubicación de tu programa, en cualquier ventana de dos, sin importar la ubicación del prompt, escribe el nombre de tu programa y éste se ejecutará como si fuera un comando.

MsDos puede servir para reparar PC's y eliminar virus.



## 25. Para terminar

[<http://www.mailxmail.com/curso-ms-dos-avanzado/terminar>]

MsDos tiene muchos usos y aplicaciones, basta contener la suficiente imaginación.

Aún hoy en día, en muchas empresas se pide que el personal domine msdos. No es necesario que sea de microsoft, como se vio en el capítulo anterior.

He echo este curso con mucho ánimo y esfuerzo y espero les sirva, cualquier duda, sugerencia, o comentario , por favor escribir:  
okoto\_n@hotmail.com

MSDOS 3!!!

Gracias al staff de mailxmail en breve será publicada la continuación de este curso, donde abarcaremos temas como los siguientes:

Creación de un comando

Parámetros y variables para aplicaciones en DOS.

Pequeñas aplicaciones en Batch

Convertir de .bat a .exe

Eliminar virus win 32 desde MSDOS,

Redes y FTP desde DOS

Dos incorporado en el web

Etc...

No te lo pierdas!!!

Dedicado a Sparda y Gustavo.

Este curso cubrió los aspectos más relevantes de dos, si se desea obtener más información, acá les dejo un par de ligas:

<http://www.geocities.com/antonigual/batchtips.html> (batch)

<http://www.freedos.org> (Dos libre)

¡¡¡Adiós!!!!

Visita más cursos como este en mailxmail:

[<http://www.mailxmail.com/cursos-informatica>]

[<http://www.mailxmail.com/cursos-programacion>]



¡Tu opinión cuenta! Lee todas las opiniones de este curso y déjanos la tuya:

[<http://www.mailxmail.com/curso-ms-dos-avanzado/opiniones>]

### Cursos similares

Cursos	Valoración	Alumnos	Vídeo
--------	------------	---------	-------

## Holística informática

Aprende con nuestro curso de holística informática, sobre los avances informáticos que te ayudará a actualizarte conociendo más acer...

[28/04/09]



2.219

## Programación de juegos para móviles con J2ME

Hoy en día se pueden realizar infinidad de cosas con el móvil. Desde la tradicional llamada telefónica hasta conectarse a Internet. Sin embargo, una de las más divertidas...

[07/09/04]



18.328

## Guía para instalar FreeBSD

Guia para instalar FreeBSD como servidor WWW, usando Apache, Mysql, PHP. Configurando FreeBSD, Compilando el Kernel, Activando el Sonido en FreeBSD, Actualizando los port...

[06/04/06]



1.140

## Planificación con MS Project

Introdúcete, paso a paso, en la planificación que necesitarás para conocer todo el programa de MS Project. Te servirán de ayuda las imágenes escogidas y las marcas que la...

[23/06/06]



3.530

## PHP y MySQL. Aplicaciones Web: HTML I (segunda parte)

Programación de aplicaciones Web con PHP y MySQL. Ahora estudiaremos las páginas Web HTML, la estructura de los documentos HTML y los tipos de documentos...

[02/12/08]



1.570