

# man pages

`man man`

`man [section] page`

Section	Description
1	General <a href="#">commands</a>
2	<a href="#">System calls</a>
3	<a href="#">Library functions</a> , covering in particular the <a href="#">C standard library</a>
4	<a href="#">Special files</a> (usually devices, those found in <code>/dev</code> ) and <a href="#">drivers</a>
5	<a href="#">File formats</a> and conventions
6	<a href="#">Games</a> and <a href="#">screensavers</a>
7	Miscellanea
8	System administration <a href="#">commands</a> and <a href="#">daemons</a>

# File System API, Unix vfs

- File descriptor o file handle: entero positivo.
- API básico:
  - [creat](#) / [open](#) / [close](#)
  - [read](#) / [write](#)
  - [lseek](#), [stat](#), [unlink](#)
- API avanzado:
  - [dup](#), [link](#), [symlink](#), [umask](#),
  - [chmod](#), [chown](#)
  - [mkdir](#), [getdents](#), [readdir](#) (3)
  - [fcntl](#), [ioctl](#), [pread](#), [pwrite](#)

# select

- Permite monitorear varios file descriptors esperando por que alguno de ellos esté “ready”.
- Ejemplo: select.c
- [select](#)
- [Select tutorial](#)

# File System API, WIN32

- NTFS Streams, los files no son sólo bag-o-bytes.
- [File API Overview](#)
- [CreateFile](#) / [CloseHandle](#)
- [ReadFile](#) / [ReadFileEx](#)
- [WriteFile](#) / [WriteFileEx](#)
- [SetFileAttributes](#)
- [CreateDirectory](#)
- [GetFileInformationByHandleEx](#)
- [SetFileInformationByHandle](#)