**NAME**

　　　2do – simple 2do list manager

**SYNOPSIS**

　　　2do [ −c ] *file* [ *nr...* | *task...* ]

**DESCRIPTION**

　　　*2do* is a trivial 2do list manager that stores 2do lists in $home/plan9/lib/2do. If no options
　　　are given, *2do* will print all available lists. If just one argument is given, it will be taken as the
　　　name of which 2do list to print, each task will be given an ID number. If you supply 2do with one
　　　or more numbers, separated by space, following the list name, these will be taken as task id's to
　　　remove from that list. The −c flag will clear all contents in the specified 2do list. Finaly, if you add
　　　more than one argument string, the first will be taken as the 2do list name, all the rest will be
　　　added to the 2do list as a new task.

**EXAMPLE**

```
for (thing in eggs milk cheese) 2do buy $thing
2do buy
```
　　　*1: eggs*
　　　*2: milk*
　　　*3: cheese*
```
2do buy 1 3
2do buy
```
　　　*1: milk*

　　　The script is intentionally basic, there is no support for timestamps, tags or priorities. But its easy
　　　enough to add such functionality if you need it:

```
2do call '{/bin/date +%Y−%m−%d}^:10:family:grandma, not on her cell
phone!
2do call | sort −k 2 | awk −F: '$3>5 && $4=/family/{print $1, $5}'
```
　　　*1 grandma, but not on her cell phone!*

　　　The above command prints tasks in the *call* 2do list from oldest to newest, that have priority
　　　greater then *5* and the tag *family*. Of course this is just an example, with careful thought I am sure
　　　you can come up with even more convoluted ways to mismanage your life.

**BUGS**

　　　If the list you specified doesn't exist, it will be created. Whether you view this as a bug or a feature
　　　depends on how many typos you make.

　　　The task id numbers change when you remove tasks from the list.

　　　Permanently removing a list must be done manually (see *rm*(1))

**SEE ALSO**

　　　*calendar*(1), *ckbox*(1), *que*(1), *bm*(1)