

export

为shell变量或函数设置导出属性。

概要

```
export [-fn] [name[=word]]...  
export -p
```

主要用途

- 定义一到多个变量并设置导出属性。
- 修改一到多个变量的值并设置导出属性。
- 删除一到多个变量的导出属性。
- 显示全部拥有导出属性的变量。
- 为一到多个已定义函数新增导出属性。
- 删除一到多个函数的导出属性。
- 显示全部拥有导出属性的函数。

选项

```
-f[]指向函数。  
-n[]删除变量的导出属性。  
-p[]显示全部拥有导出属性的变量。  
-pf[]显示全部拥有导出属性的函数。  
-nf[]删除函数的导出属性。  
-- : 在它之后的选项无效。
```

参数

name[]可选) : 变量名或已定义函数名。

value[]可选) : 变量的值。

返回值

export返回true除非你提供了非法选项或非法名称。

例子

```
# 显示全部拥有导出属性的变量。  
# export -p  
# export  
# 显示全部拥有导出属性的函数。  
# export -pf
```

```
# 首先删除要演示的变量名  
#unset a b  
# 定义变量的同时增加导出属性  
export a b=3
```

```
# 当然也可以先定义后增加导出属性
b=3
export b

# 修改拥有导出属性的变量的值
export a=5 b=7
# 当然也可以直接赋值修改
a=5;b=7

# 删除变量的导出属性
export -n a b
```

```
# 首先删除要演示的函数名
unset func_1 func_2
# 创建函数
function func_1(){ echo '123'; }
function func_2(){ echo '890'; }

# 为已定义函数增加导出属性
export -f func_1 func_2

# 删除函数的导出属性
export -fn a b
```

```
# 添加环境变量JAVA到~/.bashrc`
PATH=/usr/local/jdk1.7.0/bin:$PATH
# 添加当前位置到动态库环境变量
export LD_LIBRARY_PATH=$(pwd):${LD_LIBRARY_PATH}
```

错误用法

- 对未定义的函数添加导出属性。
- 对没有导出属性的函数/变量执行删除导出属性操作。
- 在 -- 后使用选项。

Q&A

Q对变量或函数设置导出属性有什么用？

A它们会成为环境变量，可以在脚本中访问它们，尤其是脚本中调用的子进程需要时。（[参考链接4](#)）

Q如果我编写的脚本修改了已有的环境变量的值，那么执行它会在当前终端生效吗？会影响之前以及之后打开的终端吗？

A只有通过source方式调用的脚本会生效，您可以查看source命令获得更多信息；其他方式只是在子shell中执行。之前的不会影响，之后的除非是修改了~/.bashrc这种启动终端时加载的脚本。（[参考链接1](#)）

Q我脚本文件中调用中定义的函数和变量。为什么在新打开的终端中通过方式调用该脚本或直接运行

这个当前用户有执行权限的脚本却不能使用这些函数和变量？

A请在~/.bashrc文件中增加export它们的语句。另请参阅[知识点](#)段落。

Q[]数组和关联数组也可以设置导出属性吗？

A[]是可以的（如果你的bash支持它们），不过有些问题（[参考链接2](#) []）

Q[]为什么我在查看变量或函数导出属性的时候显示的开头是？

A[]因为declare也能够设置变量或函数的导出属性，详见declare命令。

注意

1. 该命令是bash内建命令，相关的帮助信息请查看help命令。

知识点

在info bash或[bash在线文档](#)的3.7.3节提到了shell执行环境，其中涉及变量和函数的内容如下

- shell parameters that are set by variable assignment or with set or inherited from the shell's parent in the environment
- shell functions defined during execution or inherited from the shell's parent in the environment

那么第一句话中的参数又和变量有什么关系呢？在3.4节第一段中提到：

A variable is a parameter denoted by a name.

变量是有名字的参数。

那么子shell确实继承了父shell中带有导出属性的变量或函数。

可参考链接：[执行脚本方式的区别](#)

参考链接

1. [关于bashrc profile文件的讨论](#)
2. [关于export数组的讨论](#)
3. [export -pf用法](#)
4. [环境变量和shell变量的区别](#)

扩展阅读

一般来说，配置交叉编译工具链的时候需要指定编译工具的路径，此时就需要设置环境变量。查看已经存在的环境变量：

```
[root@localhost ~]# export
declare -x G_BROKEN_FILENAMES="1"
declare -x HISTSIZE="1000"
declare -x HOME="/root"
declare -x hostname="localhost"
declare -x INPUTRC="/etc/inputrc"
declare -x LANG="zh_CN.UTF-8"
declare -x LESSOPEN="|/usr/bin/lesspipe.sh %s"
```

```
declare -x logname="root"
declare -x
LS_COLORS="no=00:fi=00:di=01;34:ln=01;36:pi=40;33:so=01;35:bd=40;33;01:cd=40
;33;01:or=01;05;37;41:mi=01;05;37;41:ex=01;32:*.cmd=01;32:*.exe=01;32:*.com=
01;32:*.btm=01;32:*.bat=01;32:*.sh=01;32:*.csh=01;32:*.tar=01;31:*.tgz=01;31
:*.arj=01;31:*.taz=01;31:*.lzh=01;31:*.zip=01;31:*.z=01;31:*.Z=01;31:*.gz=01
;31:*.bz2=01;31:*.bz=01;31:*.tz=01;31:*.rpm=01;31:*.cpio=01;31:*.jpg=01;35:*
.gif=01;35:*.bmp=01;35:*.xbm=01;35:*.xpm=01;35:*.png=01;35:*.tif=01;35:"
declare -x mail="/var/spool/mail/root"
declare -x OLDPWD
declare -x
PATH="/usr/kerberos/sbin:/usr/kerberos/bin:/usr/local/sbin:/usr/local/bin:/s
bin:/bin:/usr/sbin:/usr/bin:/root/bin"
declare -x pwd="/root"
declare -x SHELL="/bin/bash"
declare -x SHLVL="1"
declare -x SSH_CLIENT="192.168.2.111 2705 22"
declare -x SSH_CONNECTION="192.168.2.111 2705 192.168.2.2 22"
declare -x SSH_TTY="/dev/pts/0"
declare -x TERM="linux"
declare -x USER="root"
```

From: <https://rd.irust.top/> - 学习笔记

Permanent link: <https://rd.irust.top/doku.php?id=command:export>

Last update: **2021/10/15 14:58**

