

mapfile

从标准输入读取行并赋值到数组。

概要

```
mapfile [-d delim] [-n count] [-O origin] [-s count] [-t] [-u fd] [-C callback] [-c quantum] [array]
```

主要用途

- 从标准输入或文件描述符读取行并赋值到数组。

选项

-d delim	将delim设为行分隔符，代替默认的换行符。
-n count	从标准输入中获取最多count行，如果count为零那么获取全部。
-O origin	从数组下标为origin的位置开始赋值，默认的下标为0。
-s count	跳过对前count行的读取。
-t	读取时移除行分隔符delim（默认为换行符）。
-u fd	从文件描述符fd中读取。
-C callback	每当读取了quantum行时，调用callback语句。
-c quantum	设定读取的行数为quantum

如果使用-C时没有同时使用-c指定quantum的值，那么quantum默认为5000。

当callback语句执行时，将数组下一个要赋值的下标以及读取的行作为额外的参数传递给callback语句。

如果使用-O时没有提供起始位置，那么mapfile会在实际赋值之前清空该数组。

参数

array（可选）：用于输出的数组名称。如果没有指定数组名称，那么会默认写入到变量名为MAPFILE的数组中。

返回值

返回成功除非使用了非法选项、指定的数组是只读的、指定的数组不是下标数组。

例子

```
# 常见的读取形式。
mapfile < source_file target_array
cat source_file |mapfile target_array
mapfile -u fd target_array

# 只读取前5行。
mapfile < source_file -n 5 target_array

# 跳过前5行。
mapfile < source_file -s 5 target_array
```

```
# 在数组指定的下标开始赋值。
# 请注意：这样做不会清空该数组。
mapfile < source_file -o 2 target_array

# 读取时设定行分隔符为tab
# 注意，第二行的tab在终端需要用ctrl+v tab输入；
mapfile < source_file -d $'\t' target_array
mapfile < source_file -d ' ' target_array

# 读取时移除行分隔符tab
mapfile < source_file -d $'\t' -t target_array
# 读取时移除行分隔符（换行符）。
mapfile < source_file -t target_array

# 每读取2行，执行一次语句（在这里是echo）
mapfile < source_file -C "echo CALLBACK:" -c 2 target_array

# 遍历下标，依次显示数组的元素。
for i in ${!target_array[@]}; do
    printf "%s" ${target_array[i]}
done
```

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。
2. bash内建命令readarray是mapfile的同义词。

From:

<https://rd.irust.top/> - 学习笔记

Permanent link:

<https://rd.irust.top/doku.php?id=command:mapfile>

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

