

Bash-скрипты

Кузнецов Антон Михайлович

Санкт-Петербургский Академический Университет

29 сентября 2011 г.

- Любой bash-скрипт должен начинаться со строки:
`#!/bin/bash`
(Последовательность `#!` называется Sha-Bang - указание интерпретатора)
- Запуск интерпретатора:
`bash scriptname [arguments]`
- Комментарии начинаются с символа `#`

- Сделать исполняемым:
`chmod 555 scriptname`
а затем просто запустить:
`./scriptname`
- Если поместить сценарий в каталог `/usr/local/bin`, то вызвать его можно просто набрав название файла в командной строке

Ключ `bash -x`

```
bash -x script.sh
+ echo "Hello, World!"
Hello, World!
+ who
antonk  tty1  2011-09-28 16:01
```

`set -x` – включить отладку с этого момента
`set +x` – выключить отладку

Могут быть строками, целыми числами и массивами.
Можно объявлять константы.

- `export foo=93` – экспорт во все дочерние оболочки.
- `foo=93` – установка для текущей оболочки
* вокруг знака `=` пробелов быть не должно!
- `printenv` – вывести глобальные переменные.
Пример: `HOME`, `MAIL`, `PATH`, `LC_ALL`, `PWD`, `UID`.
- `set` – вывести все переменные и функции.

Пример с переменными

```
a=1
b=$a
echo $b    # 1
hello="A B C"
echo hello  # hello
echo $hello # A B C
echo "$hello" # A B C
echo '$hello' # $hello
```

- `$@` – параметры скрипта (все!)
- `$0` – имя скрипта
- `$1, $2, $3, ...` – параметры скрипта по одному
- `$#` – количество переданных аргументов

- `a=1`
- `let a=16+5 # let` – арифметические действия
- `for a in 7 8 9 11`
 - do ...
 - done
- `read a`
- `a='echo Hello!'`
- `echo $a # Hello!`
- `a='ls -l'`
- `a=$(ls) #` то же самое

- `if EXPR; then команды; fi`
- `if EXPR; then команды; else другое; fi`
- `[! EXPR]` – отрицание
- `[(EXPR)]` – скобки
- `[EXPR1 -a EXPR2]` – И
- `[EXPR1 && EXPR2]` – И
- `[EXPR1 -o EXPR2]` – ИЛИ
- `[EXPR1 || EXPR2]` – ИЛИ
- `((арифим. выражение))`
- **man test**

- [*-e FILE*] - файл существует
- [*-d FILE*] - это директория
- [*-f FILE*] - это обычный файл
- [*-s FILE*] - размер ненулевой
- [*-r FILE*] - доступен для чтения
- [*-w FILE*] - доступен для записи
- [*-x FILE*] - исполняемый

```
if ! grep -q $USER /etc/passwd; then  
echo "not a local account"  
fi
```

```
grep -q $USER /etc/passwd if [ $? -ne 0 ]; then  
echo "not a local account"  
fi
```

```
if [ "$(whoami)" != root ]; then echo "Oh"; fi
```

```
echo "1 Запуск программы nano"  
echo "2 Запуск программы vi"  
echo "3 Выход"  
read doing  
case $doing in  
1) /usr/bin/nano ;;  
2) /usr/bin/vi ;;  
3) exit 0;;  
*) echo "Введено неправильное действие"  
esac
```

```
for arg in list
do
command(s)
done
```

```
for arg in $( seq n ) # 1 2 3 ... n
for arg in 1..n # 1 2 3 ... n
```

```
for arg in "$@"
```

```
function function_name { command... }
```

```
function_name () { command... }
```

НЕ СТОИТ СМЕШИВАТЬ ОПРЕДЕЛЕНИЯ! Функции могут принимать входные аргументы и возвращать код завершения.

return – завершает исполнение функции.

Может возвращать "код завершения"(int), который записывается в переменную \$?.

Если "код завершения" не указан – возвращается код последней команды в функции

```
OPERATION=docToPdf
SUFFIX=pdf
directory=$PWD
for file in $directory/*
do
filename=$file
$OPERATION $file > "$filename.$SUFFIX"
rm -f $file
done
```