

# Создание скриптов

## Инструментарий Современного Программиста

Иван Трепаков

NSU Sys.Pro

# Написание скриптов

## Скрипты

- Последовательность команд в файле
- Расширение файла может быть любое
  - Для shell-скриптов обычно .sh
- Можно сделать файл *исполняемым* с помощью `chmod +x`
- Можно писать на любом интерпретируемом языке

## Исполняемый скрипт на bash

```
$ cat >hello.sh <<EOF
echo "Hello World!"
EOF
$ ls -l hello.sh | sed 's/ .* / /'
-rw-r--r-- hello.sh
$ ./hello.sh
bash: ./hello.sh: Permission denied
$ chmod +x hello.sh
$ ls -l hello.sh | sed 's/ .* / /'
-rwxr-xr-x hello.sh
$ ./hello.sh
Hello World!
```

# Написание скриптов

## Скрипты

- Последовательность команд в файле
- Расширение файла может быть любое
  - Для shell-скриптов обычно .sh
- Можно сделать файл *исполняемым* с помощью `chmod +x`
- Можно писать на любом интерпретируемом языке

## shebang

- `#!/usr/bin/env bash`
- `#!/usr/bin/env python3`
- `#!/usr/bin/env perl`
- ...

## Исполняемый скрипт на python

```
$ cat >hello.py <<EOF
#!/usr/bin/env python3
print("Hello World!")
EOF
$ chmod +x hello.py
$ ./hello.py
Hello World!
```

## Синтаксис

- Аргументы
  - \$1, \$2, ...
  - \$@ — все аргументы

## script.sh

```
echo "$1$2$3"
```

```
echo "$@"
```

## Вывод

```
$ bash script.sh a b c
```

```
abc
```

```
a b c
```

## Синтаксис

- Аргументы
  - \$1, \$2, ...
  - \$@ — все аргументы
- Переменные окружения
  - Однострочные и многострочные

## script.sh

```
foo="some text"
read -r -d '' bar <<EOF
Some multiline
text
EOF
echo "$foo $bar"
```

## Вывод

```
$ bash script.sh
some text Some multiline
text
```

## Синтаксис

- Аргументы
  - \$1, \$2, ...
  - \$@ — все аргументы
- Переменные окружения
  - Однострочные и многострочные
- Условия
  - `test ...` или `[ ... ]` (см. `help test`)
  - `if cond; then ...; else ...; fi`

## script.sh

```
if [ "$1" == "foo" ]; then
    echo "Pass foo"
elif [ "$1" == "bar" ]; then
    echo "Pass bar"
else
    echo "Fail"
fi
```

## Вывод

```
$ bash script.sh foo
Pass foo
$ bash script.sh bar
Pass bar
$ bash script.sh baz
Fail
```

## Синтаксис

- Аргументы
  - \$1, \$2, ...
  - \$@ — все аргументы
- Переменные окружения
  - Однострочные и многострочные
- Условия
  - `test ...` или `[ ... ]` (см. `help test`)
  - `if cond; then ...; else ...; fi`
  - `case x; in a) ...;; *) ...;; esac`

## script.sh

```
case "$1" in
  foo | bar)
    echo "Pass $1"
    ;;
  *)
    echo "Fail"
    ;;
esac
```

## Вывод

```
$ bash script.sh foo
Pass foo
$ bash script.sh bar
Pass bar
$ bash script.sh baz
Fail
```

## Синтаксис

- Аргументы
  - \$1, \$2, ...
  - \$@ — все аргументы
- Переменные окружения
  - Однострочные и многострочные
- Условия
  - `test ...` или `[ ... ]` (см. `help test`)
  - `if cond; then ...; else ...; fi`
  - `case x; in a) ...;; *) ...;; esac`
- Циклы
  - `for x in xs; do ...; done`

## script.sh

```
for arg in "$@"; do  
    echo "$arg"  
done
```

## Вывод

```
$ bash script.sh foo 'Hello World!'  
foo  
Hello World!
```



## Синтаксис

- Аргументы
  - \$1, \$2, ...
  - \$@ — все аргументы
- Переменные окружения
  - Однострочные и многострочные
- Условия
  - `test ...` или `[ ... ]` (см. `help test`)
  - `if cond; then ...; else ...; fi`
  - `case x; in a) ...;; *) ...;; esac`
- Циклы
  - `for x in xs; do ...; done`
  - `while cond; do ...; done`

## script.sh

```
i=$1
factorial=1
while [ $i -gt 0 ]; do
    factorial=$(( $factorial * $i ))
    i=$(( $i - 1 ))
done
echo "$factorial"
```

## Вывод

```
$ bash script.sh 5
120
```

## Синтаксис

- Аргументы
  - \$1, \$2, ...
  - \$@ — все аргументы
- Переменные окружения
  - Однострочные и многострочные
- Условия
  - `test ...` или `[ ... ]` (см. `help test`)
  - `if cond; then ...; else ...; fi`
  - `case x; in a) ...;; *) ...;; esac`
- Циклы
  - `for x in xs; do ...; done`
  - `while cond; do ...; done`
- Функции
  - `function_name () { ...; }`

## script.sh

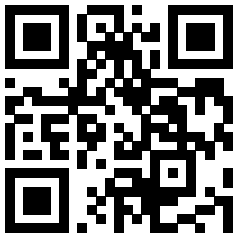
```
hello () {  
    echo "Hello $1!"  
}
```

```
hello "World"  
hello "Sys.Pro"
```

## Вывод

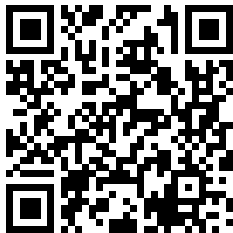
```
$ bash script.sh  
Hello World!  
Hello Sys.Pro!
```

## Bash cheatsheet



<https://devhints.io/bash>

## Bash Reference Manual



[https://www.gnu.org/software/bash/manual/  
bash.html](https://www.gnu.org/software/bash/manual/bash.html)

Q&A