



# Терминал Linux

Начало начал

Для вопросов по курсу: cs@moevm.pro
Иванов Дмитрий Владимирович, dmitry.ivanov@moevm.pro
Бутенко Дмитрий Сергеевич, dmitriy.butenko@moevm.pro
Префикс в теме письма [CS_63XX]



# О чем поговорим сегодня?

- Зачем вообще терминал?
- Что такое shell / terminal / OS-ядро
- Переменные окружения
- Навигация
- Спецсимволы
- Создание файлов
- Чтение файлов
- Автодополнение
- Горячие клавиши
- Потоки
- Инструкции



# Зачем вообще терминал?

1. Graphical user interface (GUI) - надстройка поверх приложения, которая может не покрывать всех сценариев использования приложения;
2. Терминал - прямой канал взаимодействия с приложением. Любое действие можно выразить текстовой командой, а значит, повторить, автоматизировать, передать другому пользователю;
3. На серверах как правило нет GUI.



# Зачем вообще терминал? (примеры)

1. Переименовать  500 файлов, убрав пробелы из имен

```
for f in *\ *; do mv -- "$f" "${f// /}"; done
```

2. Скачивание и запуск скрипта (частный случай установки ПО сегодня)

```
/bin/bash -c "$(curl -fsSL https://domain/install.sh)"
```



# Основные термины

1. Terminal (Терминал) - программа-окно, которая рисует текст (или не только текст) и принимает ввод
2. Shell (Оболочка) - программа интерпретатор команд, принимает ввод, интерпретирует его и отдает вывод (sh, bash, zsh, fish и т.д.)
3. Ядро (Kernel) - компонент ОС, который реально управляет файлами, процессами



# Основные термины (пример)

echo $SHELL

```
$ echo $SHELL
/bin/bash
$
```



# Переменные окружения

SHELL, HOME, PWD, OLDPWD, PATH

USER/LOGNAME, HOSTNAME, LANG, LC_*

TERM, PS1, PS2

?, $, 0, RANDOM, SECONDS



# Навигация

Основной тезис: понимать, в какой директории сейчас сессия пользователя, что вокруг, как переходить в другие директории.

# pwd, ls, cd



# Навигация (pwd)

pwd (print working directory) - вывод полного пути текущей директории

```
$ pwd
/tmp
$
```



# Навигация (ls)

ls (list) - вывод списка файлов в директории (директория это тоже файл), частые опции (-a -l -h)

```
$ ls
1.txt  dir1
$ ls -l
total 0
-rw-r--r-- 1 username1 username1  0 Aug 27 22:06 1.txt
drwxr-xr-x 2 username1 username1 40 Aug 27 22:07 dir1
$ ls -a
.  ..  1.txt  dir1  .hidden_file
$ ls -la
```



# Навигация (cd)

cd (change directory) - смена рабочей (текущей) директории

```
$ pwd
/boot
$ cd /tmp
$ pwd
/tmp
$
```

# Навигация (cd)

~ - домашняя директория пользователя;

Shell

/ - корень файловой системы;

. - текущая директория;

.. - директория уровнем выше.

Kernel

# Навигация (почему ~)





# Создание файлов

mkdir (make directory) - создание директории

touch - создание пустого файла

```
$ ls -la
total 0
drwxr-xr-x  2 username1 username1  40 Aug 27 22:42 .
drwxrwxrwt 30 root      root      880 Aug 27 22:41 ..
$ mkdir new_directory
$ touch new_file
$ ls -la
total 0
drwxr-xr-x  3 username1 username1  80 Aug 27 22:42 .
drwxrwxrwt 30 root      root      880 Aug 27 22:41 ..
drwxr-xr-x  2 username1 username1  40 Aug 27 22:42 new_directory
-rw-r--r--  1 username1 username1   0 Aug 27 22:42 new_file
$
```

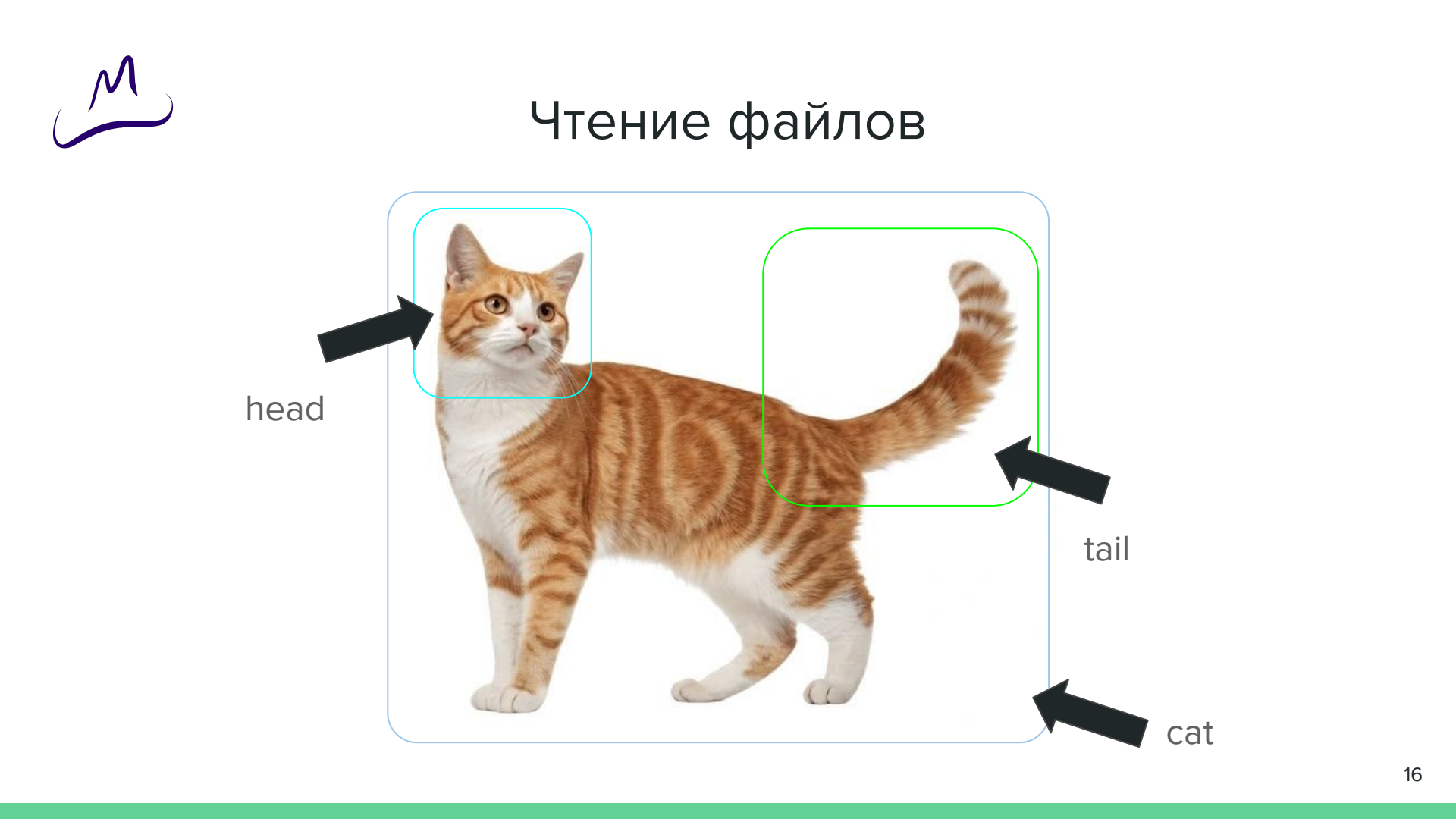


# Чтение файлов

cat (concatenate) - вывод объединения файла/файлов

head/tail - чтение начала/конца файла. -n, tail -f

more/less - чтение файла постранично. g/G /substr



# Чтение файлов



# Автодополнение

[Tab] - shell умеет достраивать ввод на основе текущего

```
$ cd /tmp
$ cd
example/
.font-unix/
hsperfdata_username1/
.ICE-unix/
.mount_nvtopljcLpo/
.org.chromium.Chromium.FqDxN8/
.org.chromium.Chromium.gTFPZ5/
.org.chromium.Chromium.mC8rtt/
.org.chromium.Chromium.RF5q7b/
.org.chromium.Chromium.yocsAE/
```



# Автодополнение (как работает)

Поведение по умолчанию: поиск среди встроенных команд, алиасов, функций и исполняемых файлов в каталогах $PATH

Если не первое слово - по умолчанию делает дополнение по именам файлов в текущей директории:

cd Doc<Tab> -> Documents/



# Горячие клавиши

- Стрелки (вверх, вниз) - перемещение по истории
- Стрелки (влево, вправо) - перемещение курсора
- Ctrl+c - завершение выполняемого процесса
- Ctrl+l - очистка экрана (аналог команды clear)
- Ctrl+d - завершение сессии



# Потоки

вывод - stderr/stdout

ввод - stdin

> >> - вывод в файл (создание нового или добавление)

< - ввод для команды

<< - многострочный ввод

потоки можно комбинировать



# Инструкции

man (manual) - вывод документации по запрашиваемой сущности (команда, функция)

pydoc - вывод документации по python (синтаксис, модули, классы)

# Источники

- POSIX-спецификация Shell Command Language

# Вопросы по курсу задавайте по почте:

Иванов Дмитрий Владимирович
dmitry.ivanov@moevm.pro

Бутенко Дмитрий Сергеевич
dmitriy.butenko@moevm.pro

Правила коммуникации по электронной почте:
https://se.moevm.pro/doku.php/courses:communication_rules