

УРОК 11. РАБОТА СО СКРИПТАМИ

ПРАКТИЧЕСКАЯ РАБОТА СО СКРИПТАМИ	2
Задача 1	2
Решение 1	2
Задача 2	4
Решение 2	4
Задача 3	5
Решение 3	5
Задача 4	6
Решение 4	6



ПРАКТИЧЕСКАЯ РАБОТА СО СКРИПТАМИ



Скрипт — это небольшая программа, которая содержит последовательность действий, созданных для автоматического выполнения задачи.

Задача 1

1. Написать скрипт `script_ВАШЕИМЯ.sh`, который:
 - Выводит приветствие.
 - Показывает текущий каталог.
 - Выводит список всех процессов.
 - Показывает текущую дату и время.
 - Проверяет наличие файлов, содержащих ключевое слово "error" в каталоге `/var/log/`.
 - Выводит содержимое файла с информацией о системе (`/etc/os-release`).
 - Подсчитывает количество строк в файле с информацией о системе.
 - Выводит последние 5 строк файла с информацией о системе.
 - Выводит информацию о пользователях из файла `/etc/passwd` с использованием `awk`. В виде "Имя пользователя: ", Домашний каталог: "
 - Выводит сообщение о завершении работы скрипта.
2. Дать права на исполнение этого скрипта.
3. Скопировать скрипт в `/tmp/`.

Решение 1

```
#!/bin/bash
```

```
# Выводим приветствие
```

```
echo "Привет! Этот скрипт покажет текущий каталог и список процессов."
```

```
# Выводим текущий каталог
```

```
echo "Текущий каталог:"
```



```
pwd
```

```
# Выводим список всех процессов
```

```
echo "Список всех процессов:"
```

```
ps -ef
```

```
# Выводим текущую дату и время
```

```
echo "Текущая дата и время:"
```

```
date
```

```
# Проверяем, есть ли файлы, содержащие ключевое слово "error"
```

```
echo "Файлы, содержащие ключевое слово 'error':"
```

```
grep -rn "error" /var/log/
```

```
# Выводим содержимое файла с информацией о системе
```

```
echo "Содержимое файла с информацией о системе:"
```

```
cat /etc/os-release
```

```
# Подсчитываем количество строк в файле с информацией о системе
```

```
echo "Количество строк в файле с информацией о системе:"
```

```
wc -l /etc/os-release
```

```
# Выводим последние 5 строк файла с информацией о системе
```

```
echo "Последние 5 строк файла с информацией о системе:"
```

```
tail -n 5 /etc/os-release
```

```
# Выводим информацию о пользователях с использованием awk
```

```
echo "Информация о пользователях:"
```

```
awk -F ':' '{print "Имя пользователя: " $1 ", Домашний каталог: " $6}' /etc/passwd
```

```
echo "Скрипт успешно выполнен!"
```

```
Копируем скрипт в новый каталог
```

```
cp script_ВАШЕИМЯ.sh /tmp/script_ВАШЕИМЯ.sh
```

Задача 2

Разобрать скрипт создания директорий и копирования файлов в них:

```
#!/bin/bash
#
DATE=`date +%d-%m-%y`
#echo $DATE
for i in {1..5}
do
date +%H-%M-%S' > File-$i.txt
sleep 5
done
mkdir -p $DATE
cp File*.txt $DATE
#cd $DATE
for FILE in $DATE/*
do
cat $FILE
done
```

Решение 2

Скрипт создает текстовые файлы, содержащие текущее время, копирует их в каталог с названием, соответствующим текущей дате, а затем выводит содержимое этих файлов на экран.

Устанавливает переменную DATE для текущей даты в формате "день-месяц-год".

В цикле for создает 5 текстовых файлов с названиями вида "File-N.txt", где N - номер файла в диапазоне от 1 до 5. Каждый файл содержит текущее время в формате "час-минута-секунда".

После создания каждого файла скрипт ждет 5 секунд с помощью команды sleep.

Создает каталог с именем, соответствующим текущей дате, используя переменную DATE.

Копирует все файлы с именем "File*.txt" в созданный каталог, используя команду cp.

В цикле for для каждого файла в каталоге с текущей датой считывает содержимое каждого файла и выводит его содержимое на экран с помощью команды cat.

скрипт создает 5 файлов с именами file_1.txt, file_2.txt, и так далее, и выводит их список.



С использованием for:

```
#!/bin/bash
#
for i in {1..5}
do
    echo "Iteration $i"
    touch "file_$i.txt"
    sleep 2
done
echo "All files created:"
ls -l file_*.txt
```

С использованием while:

```
#!/bin/bash
#
COUNT=1
while [ $COUNT -le 5 ]
do
    echo "Iteration $COUNT"
    touch "file_$COUNT.txt"
    sleep 2
    ((COUNT++))
done
echo "All files created:"
ls -l file_*.txt
```

Задача 3

Создать скрипт, который напишет для каждого файла в директории, является ли он директорией или файлом.

Решение 3



```
#!/bin/bash

# Определяем текущий каталог
CURRENT_DIR=$(pwd)

# Выводим приветствие
echo "Привет! Этот скрипт покажет информацию о каждом файле в текущем каталоге."

# Переходим в текущий каталог
cd $CURRENT_DIR

# Запускаем цикл for для итерации по списку файлов
for FILE in *
do
    # Проверяем, является ли файл директорией
    if [ -d "$FILE" ]; then
        echo "$FILE - это директория"
    else
        echo "$FILE - это файл"
    fi
done

echo "Скрипт успешно выполнен!"
```

Задача 4

Написать скрипт, который будет генерировать файлы со случайными именами и расширениями (можно использовать дату как часть имени) и случайный текст в содержимом, а их количество - параметром, который скрипт будут принимать у пользователя.

Решение 4

```
#!/bin/bash

# Проверяем, передано ли количество файлов как аргумент
if [ $# -eq 0 ]; then
```



```
        echo "Ошибка: не указано количество файлов. Используйте $0
<количество_файлов>"
    exit 1
fi

# Переменная с количеством файлов для генерации
NUM_FILES=$1

# Создаем директорию для файлов, если она не существует
mkdir -p generated_files

# Генерируем файлы
for ((i=1; i<=NUM_FILES; i++))
do
    # Генерируем случайное имя файла
    FILE_NAME="generated_files/file_${i}$(date '+%Y-%m-%d').${RANDOM}"
    # Генерируем случайный текст для файла
    RANDOM_TEXT=$(head /dev/urandom | head -c 100)
    # Создаем файл с случайным именем и содержимым
    echo "${RANDOM_TEXT}" > "${FILE_NAME}"
done

echo "Файлы успешно созданы!"
```