

1. Вывести на экран 3 раза имя пользователя, от которого запускается команда.

```
$ cat > user3.sh
#!/bin/bash
for char in {1..3}
do
    echo $USER
done
$ nano user3.sh
$ bash user3.sh
```

2. Вывести с помощью цикла while все четные числа от 0 до 100 включительно.

```
$ cat > print100
#!/bin/bash/
x=0
while [ $x -lt 102 ]
do
    # echo $x # будет выводить каждое число в новой строке
    printf "${x} "
    let x+=2
    # x=$(( $x + 2 )) # то же присвоение, но в другом синтаксисе
done
echo "" # вывод пустой строки
$ bash user3.sh
```

3. Командой 'cut' вывести для текущей папки права доступа файлов (первая колонка вывода команды 'ls'). Отсортировать этот вывод (команда 'sort'). Удалить дубликаты (команда 'uniq'). Использовать для решения конвейер обработки задач (pipeline - вертикальный слэш).

в первой колонке команда **ls** выводит права на файлы

```
ls -l
ll
```

справка по команде **cut**

```
cut --help
```

➤ Обращаем внимание на ключи **-b, -f, -d**

варианты решения с учетом того, что права занимают первые 10 символов в выводе команды **ls -l**

```
11 | cut -b 1-10
11 | cut -b 1-10 | sort
11 | cut -b 1-10 | sort | uniq
11 | cut -b 1-10 | sort | uniq | grep -v итого
11 | cut -b -10 | sort | uniq | grep -v ^и
```

вариант решения с учетом того, что права занимают первую колонку до пробела

```
11 | cut -f 1 -d ' ' | sort | uniq | grep -v ^и
```

вариант решения утилитой **awk**

```
11 | awk '{print $1}' | sort | uniq | grep -v ^и
```

у утилиты **sort** также есть ключик **-u**

```
11 | awk '{print $1}' | sort -u
```