

# Разбор домашнего задания

**Роман Булгаков**

Спикер курса

Skillbox

# Переменные

**Роман Булгаков**

Спикер курса

Skillbox

# Вывод одной фразы несколько раз

main.py



saved

```
1 print('Python!')
2 print('Python!')
3 print('Python!')
4 print('Python!')
5 print('Python!')
```

```
Python!
Python!
Python!
Python!
> 
```

# Вывод одной фразы несколько раз

main.py



🔄 saved

```
1 print('Python.')
```

```
2 print('Python.')
```

```
3 print('Python.')
```

```
4 print('Python.')
```

```
5 print('Python.')
```

```
Python.  
Python.  
Python.  
Python.  
Python.  
❏
```

# Вывод одной фразы несколько раз

main.py



saved

```
1 print('Это Python.')
```

```
2 print('Это Python.')
```

```
3 print('Это Python.')
```

```
4 print('Это Python.')
```

```
5 print('Это Python.')
```

```
Это Python.
```

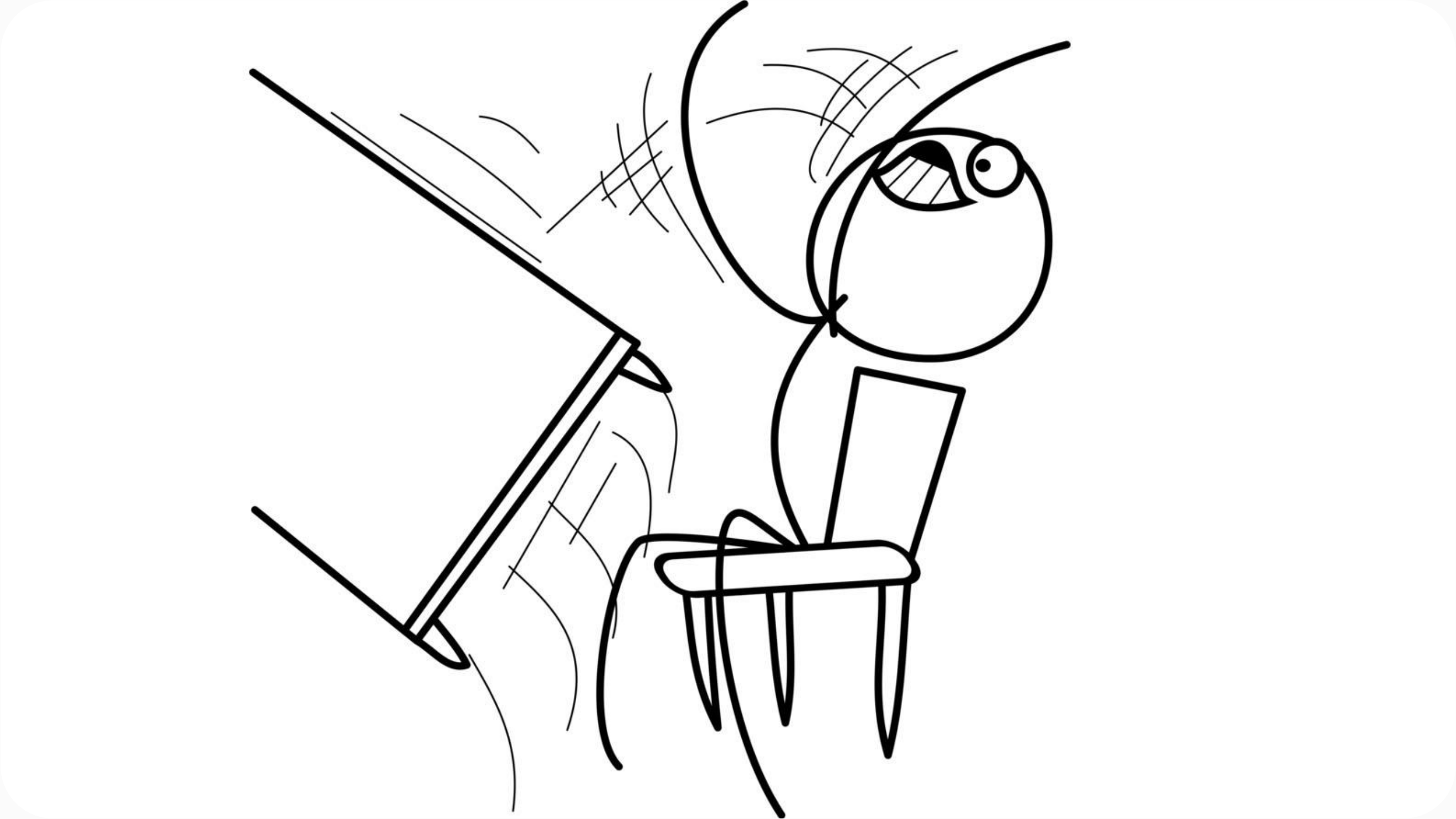
```
Это Python.
```

```
Это Python.
```

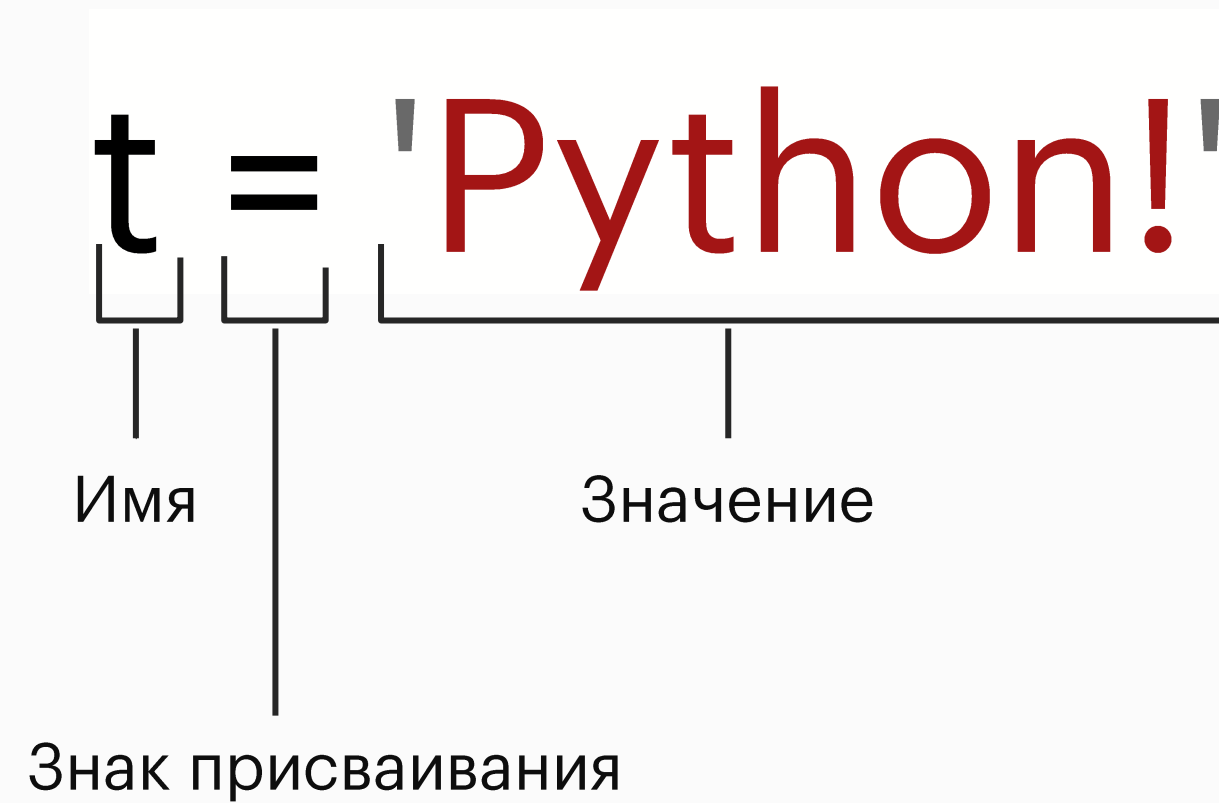
```
Это Python.
```

```
Это Python.
```

```
▮
```



# Инициализация переменной



# Две программы с одним результатом

```
print('Python!')
```

```
Python!
```



```
t = 'Python!'  
print(t)
```

```
Python!
```





# Оператор присваивания

## Пишется

```
t = 'Python!'
```

## Произносится

t присвоить 'Python!'

## или

t получает значение 'Python!'

# Имя переменной

- ✓ Только английский алфавит
- ✓ Без спецсимволов вроде @#\$%^&\*() и т. д.
- ✓ Без пробелов (исп. нижнее подчёркивание)

~~ИМЯ~~

name

~~@\*ИМЯ\*@~~

name

~~Новое имя~~

new\_name

# Как инициализировать переменную?

- 1 `t`
- 2 `t =`
- 3 `t = 'Моя первая переменная'`

Прежде чем использовать какую-либо переменную в программе, её нужно объявить.

```
main.py  saved
1  t = 'Моя первая переменная'
2  print(t)
```

```
Моя первая переменная
> 
```

# Оператор ввода input

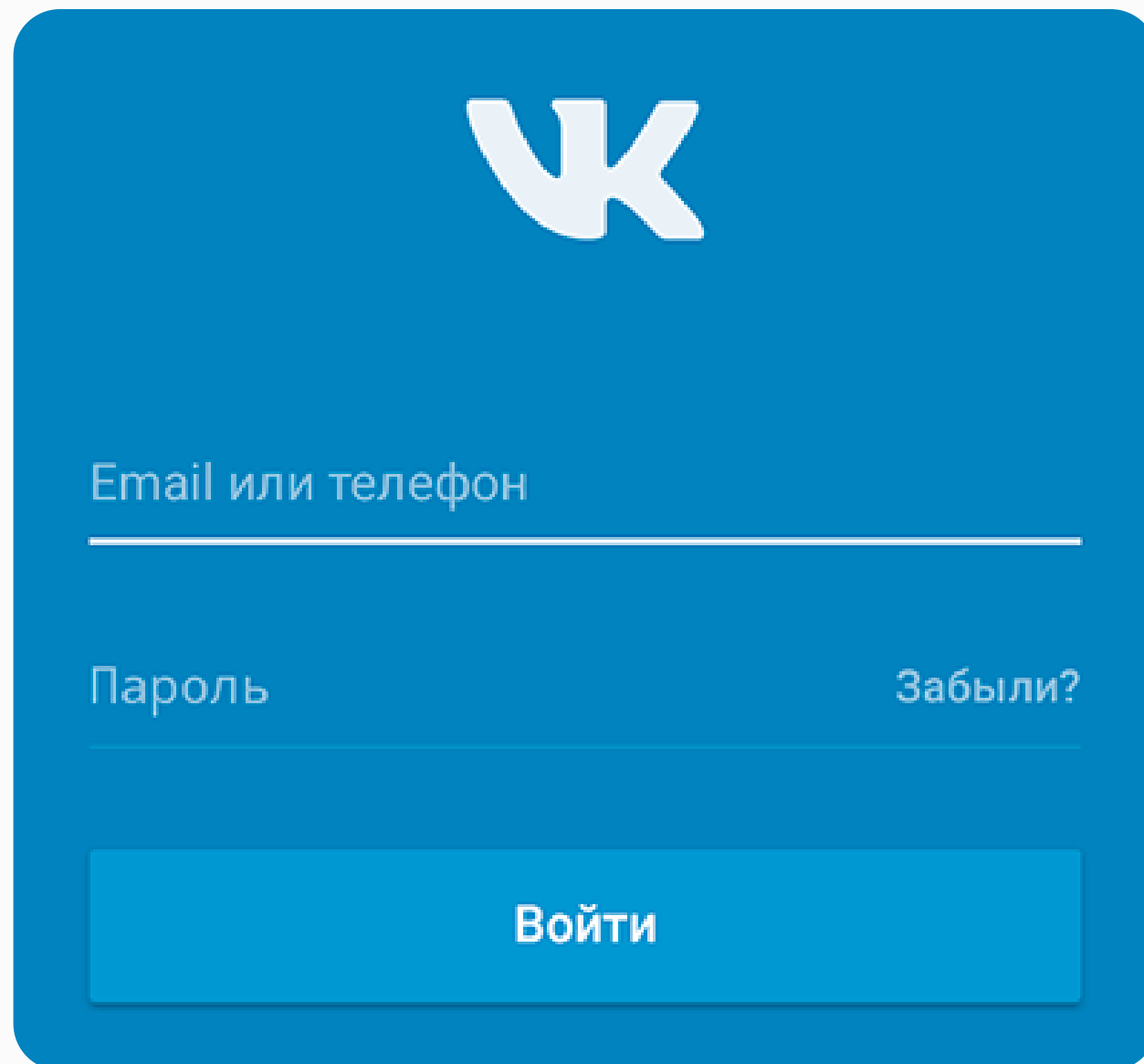
**Роман Булгаков**

Спикер курса

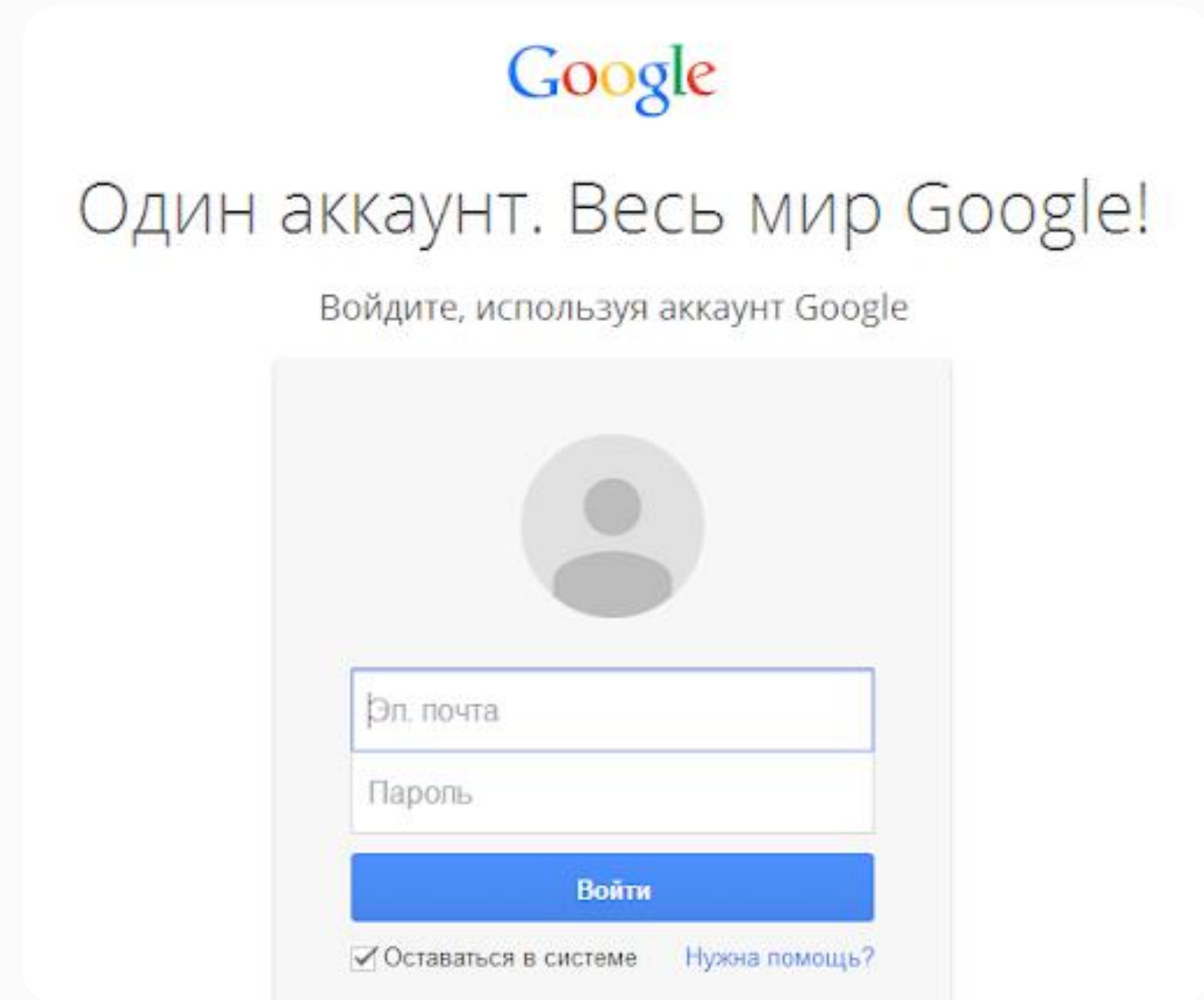
Skillbox

Оператор ввода input

# Примеры интерфейса



The image shows the VK login interface. It features a blue background with the VK logo at the top. Below the logo, there is a text input field labeled "Email или телефон". Underneath that is a password input field labeled "Пароль". To the right of the password field is a link labeled "Забыли?". At the bottom, there is a large blue button labeled "Войти".



The image shows the Google login interface. It features a white background with the Google logo at the top. Below the logo, there is a heading "Один аккаунт. Весь мир Google!" and a subheading "Войдите, используя аккаунт Google". In the center, there is a gray circle representing a user profile picture. Below this, there are two text input fields: "Эл. почта" and "Пароль". At the bottom, there is a blue button labeled "Войти". Below the button, there is a checkbox labeled "Оставаться в системе" and a link labeled "Нужна помощь?".

# Как запросить данные у пользователя?

- 1 `name`
- 2 `name = input`
- 3 `name = input()`
- 4 `name = input('Введите имя: ')`

```
Введите имя: 
```

```
Введите имя: Ваня
```

```
Введите имя: Ваня
Привет,
Ваня
> 
```

# Строки и конкатенация

**Роман Булгаков**

Спикер курса

Skillbox

# Склеивание текста

'Торговая' 'площадь'

'Торговая' + 'площадь'

```
print('Торговая' + 'площадь')
```

Конкатенация — операция соединения (склеивания) объектов.

```
main.py  saving...
1  print('Торговая' + 'площадь')
2  |
```



# Что мы узнали?

- ✓ `name = input('Введите имя: ')`  
`print('Привет,', name, '. Я - ПК')`
- ✓ Конкатенация  
`print('Торговая ' + 'площадь')`

```
Введите имя: Вова
Привет, Вова . Я - ПК
> 
```

```
Торговая площадь
> 
```

# Особенности работы с переменными

**Роман Булгаков**

Спикер курса

Skillbox

# Плохой код

```
main.py  saved
1  from urllib2 import urlopen
2
3  u = urlopen("http://python.org")
4  w = {}
5  for l in u:
6      l = l.strip("\n")
7      for ww in l.split(" "):
8          try:
9              w[ww] += 1
10         except KeyError:
11             w[ww] = 1
12
13  p = w.items()
14  p.sort(key=lambda x: x[1], reverse=True)
15
16  for i in p[:10]:
17      print(i[0], i[1])
```

# Хороший код

- name — имя
- surname — фамилия
- first\_number — первое число
- temp (temporary) — временная информация

main.py



saved

```
1 name = 'Роман'
2 surname = 'Булгаков'
3 first_number = '1'
4 temp = 'Временные данные'
```

# Именованные переменные

## Неправильно


```
familiya = 'Петров'  
abc = 'Егор'  
new car = 'Toyota'  
THIRDmonth = 'Март'
```

## Правильно


```
surname = 'Петров'  
name = 'Егор'  
new_car = 'Toyota'  
third_month = 'Март'
```

# Множественное присваивание

Обычный способ

```
main.py  saved
1  a = 'Вова'
2  b = 'Петя'
3  c = 'Лена'
```

Альтернативный способ

```
main.py  saved
1  a, b, c = 'Вова', 'Петя', 'Лена'
```

# Множественное присваивание



```
name1, name2, name3 = 'Вова', 'Петя', 'Лена'
```

The diagram illustrates the correct syntax for multiple assignment. A green bracket above the code groups the variables `name1, name2, name3`. A red bracket below the code groups the values `'Вова', 'Петя', 'Лена'`. A blue bracket below the code groups the entire assignment statement.

**Неправильно:** `name1, name2, name3 = 'Вова', 'Петя', 'Лена'`

# Итоги модуля

- Переменная, её инициализация и вывод
- Ввод данных, команда `input()`
- Перечисление в команде вывода  
`print('Имя', а, 'хорошее')`
- Конкатенация, `'Привет' + 'мир'`
- Синтаксический сахар





# Задача про обмен

