

# Python, ЗНАКОМСТВО С КОНСОЛЬЮ



Евгений  
Шмаргунов



**Евгений Шмаргунов**

Ведущий специалист

по автоматизированному тестированию в Мединдекс

 [eshmargunov@gmail.com](mailto:eshmargunov@gmail.com)

|  [evgeny.shmargunov](https://www.facebook.com/evgeny.shmargunov)



# План занятия

1. [Почему Python?](#)
2. [Онлайн-интерпретатор repl.it](#)
3. [Числа и арифметические операции](#)
4. [Переменные](#)
5. [Строки](#)
6. [Комментарии](#)



# Почему Python?



# Python 3.x

Python – интерпретируемый язык с динамической типизацией.

Плюсы:

- Прекрасно подходит для новичков
- Широкая область применения
- Богатое и дружелюбное сообщество разработчиков
- Востребованность

Сферы применения:

- Веб-разработка (Google, Yandex, Instagram, Youtube..)
- Анализ данных и машинное обучение (Google, Microsoft)
- Игры (Wargaming, Eve)
- Прототипирование
- Тестирование

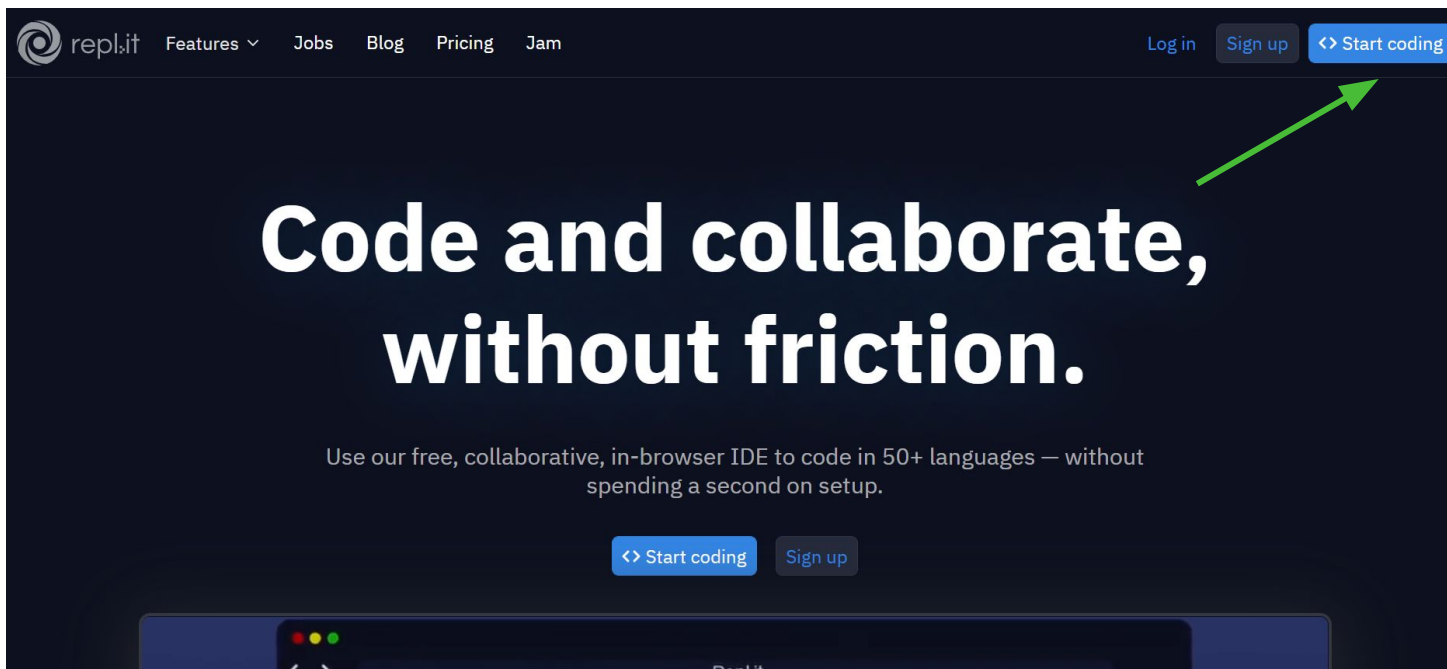


# Онлайн-интерпретатор repl.it

# Создание программы в Repl.it

## Шаг 1

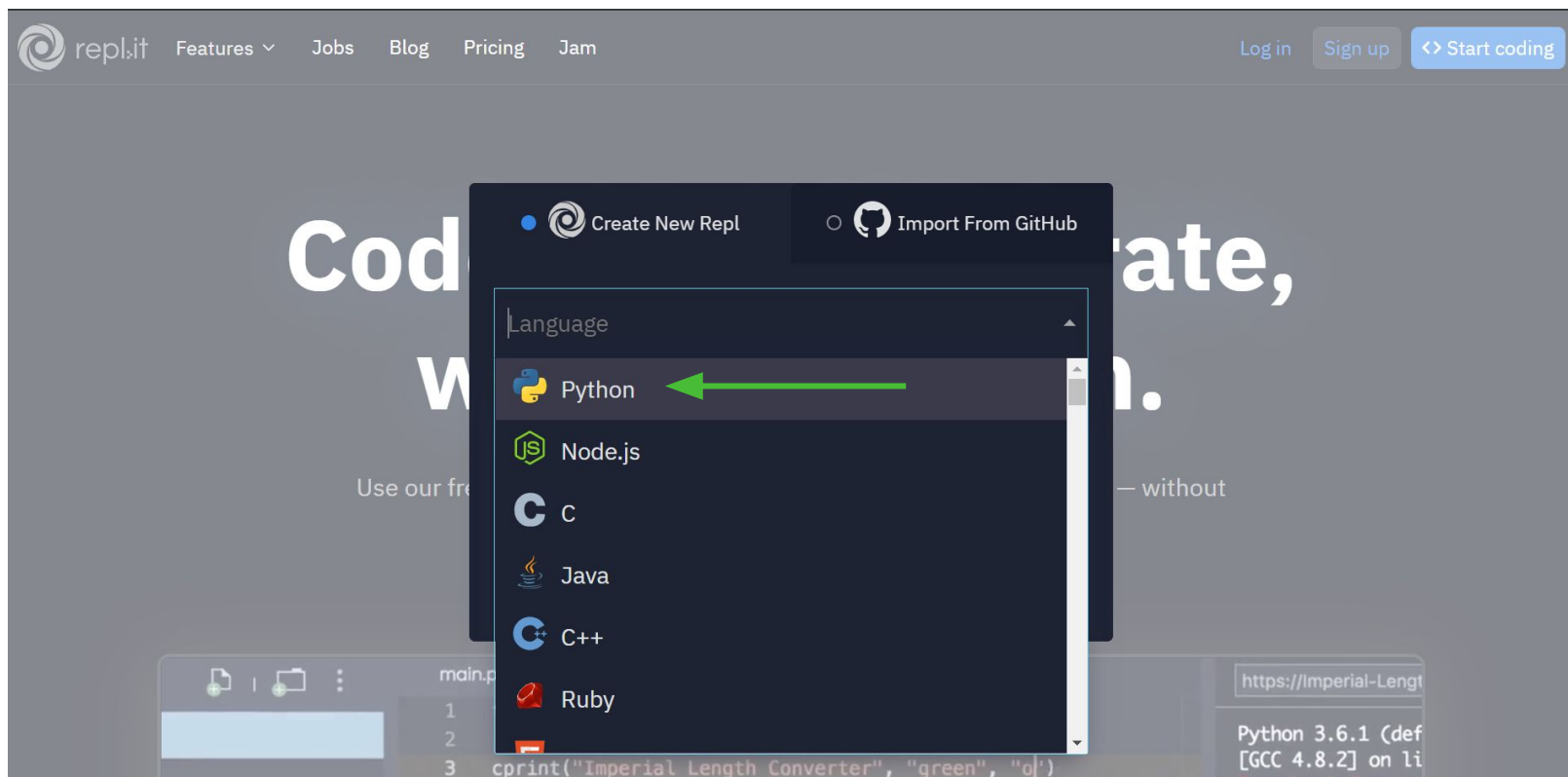
Открываем в браузере сайт <https://repl.it/> и нажимаем кнопку **Start coding** (правый верхний угол)



# Создание программы в Repl.it

## Шаг 2

В открывшемся окне **выбираем язык Python.**

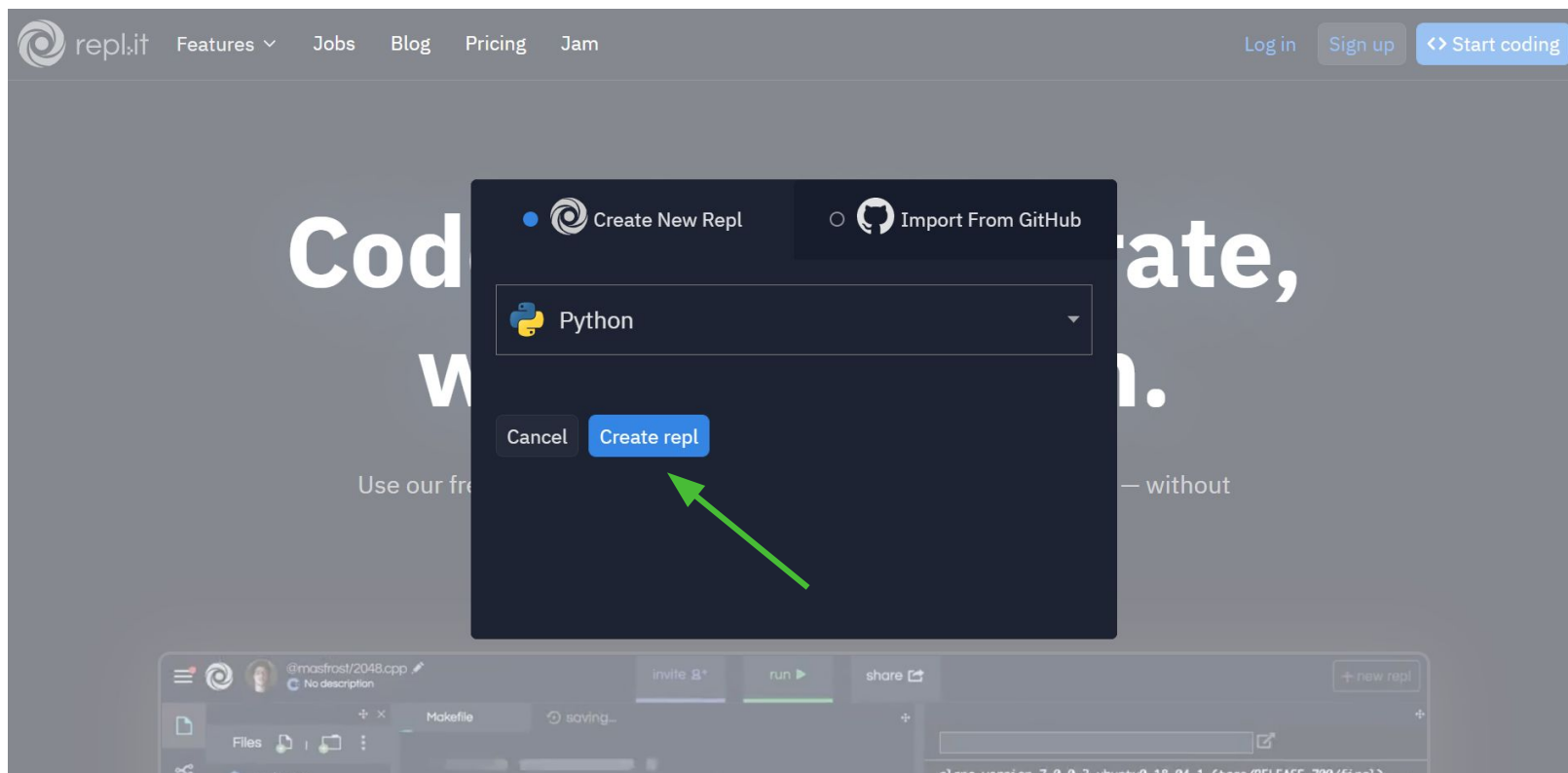




# Создание программы в Repl.it

## Шаг 3

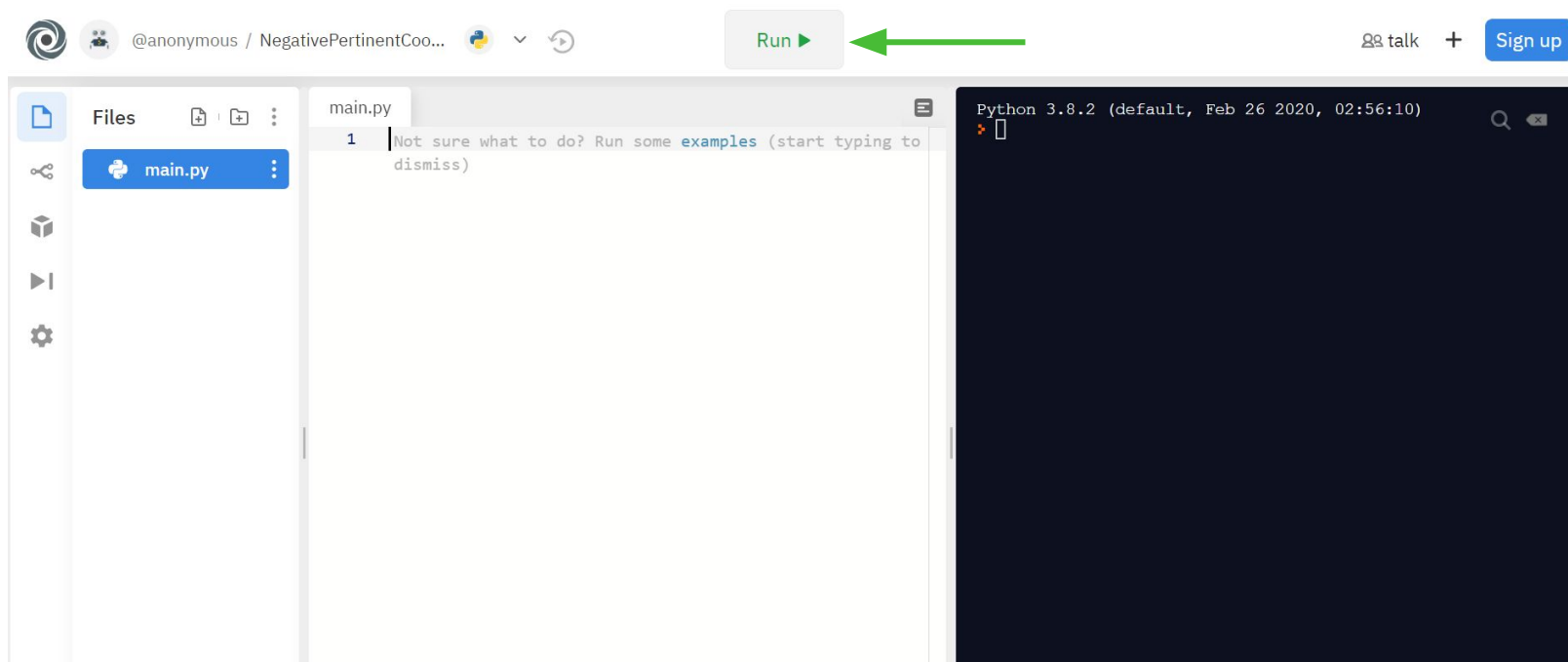
Нажимаем кнопку **Create Repl**.



# Создание программы в Repl.it

## Шаг 4

Для запуска программы нужно нажать кнопку **Run**.





# Числа и арифметические операции

---

# Арифметические операции

- сложение (+);
- вычитание (-);
- умножение (\*);
- деление (/);
- целочисленное деление (//);
- возведение в степень (\*\*);
- взятие остатка от деления (%).

Приоритет операций аналогичен стандартным математическим правилам.

Можно использовать скобки.



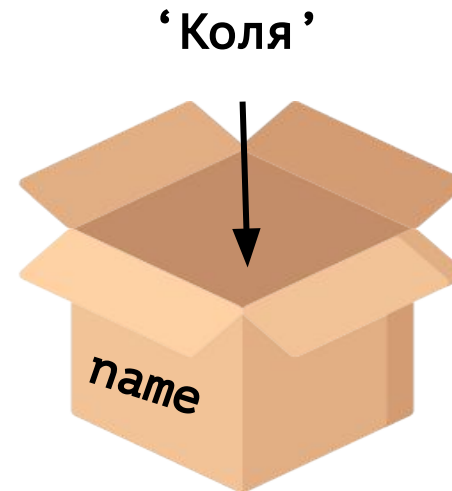
# Переменные

# Переменные

Переменная – это объект, которому дано имя. Необходимо для хранения данных и промежуточных результатов вычислений.

Объект – это:

- число
- строка
- практически что угодно в Python

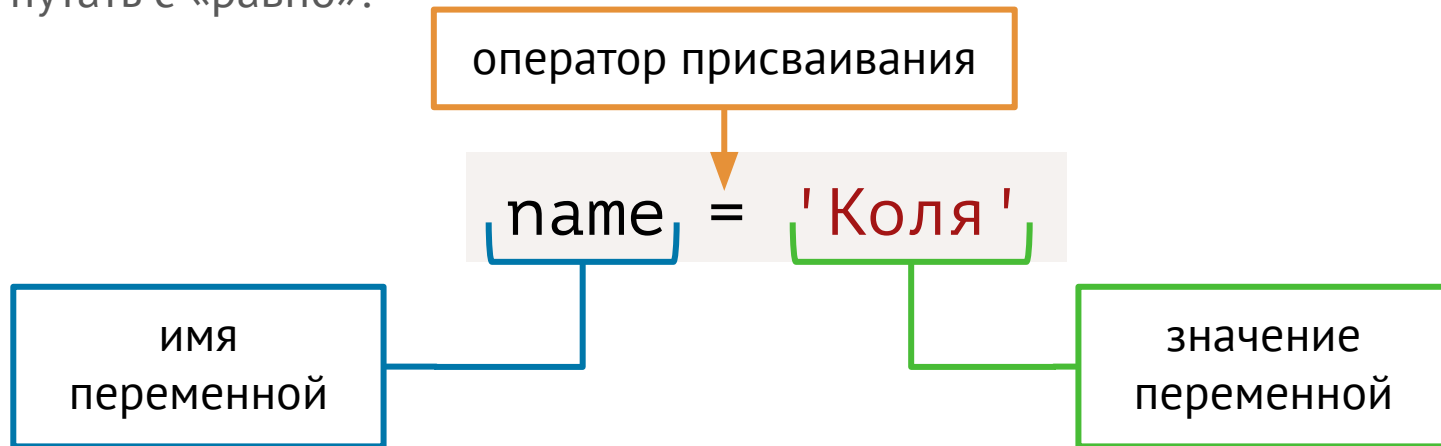


Python – язык с динамической типизацией. Это значит, что он самостоятельно определяет тип объекта, который мы хотим сохранить в переменной.

# Переменные

Чтобы сохранить значение в переменной, используется **оператор присваивания «=»**.

Не путать с «равно»!



Переменной name присвоено значение 'Коля'.

# Именованние переменных

Правила именования:

- Имя переменной может состоять только из цифр, латинских букв и знака подчеркивания.
- Имя переменной **не может** начинаться с цифр.

Рекомендации именования:

- Имя переменной должно описывать её суть.
- Лучше использовать snake\_case (слова с маленькой буквы и разделять подчеркиванием).

Определите, какие имена правильные:

1. zarplata = 1000   2. age = 25   3. 5element = True   4. familia = 45

5. month = 'May'   6. qqg = 'Spartak'   7. is\_number = True



---

# Именованние переменных

Правила именования:

- Имя переменной может состоять только из цифр, латинских букв и знака подчеркивания.
- Имя переменной **не может** начинаться с цифр.

Рекомендации именования:

- Имя переменной должно описывать её суть.
- Лучше использовать snake\_case (слова с маленькой буквы и разделять подчеркиванием).

Проверьте себя:

1. `zarplata = 1000`
2. `age = 25`
3. `Selement = True`
4. `familia = 45`
5. `month = 'May'`
6. `qqq = 'Spartak'`
7. `is_number = True`

# Вывод в консоль

Функция **print()** в Python выводит заданные объекты на экран.

```
print(100)
print('Hello, World!')
print('Привет,', 'мне', 20, 'лет')
```

Результат выполнения:

```
100
Hello, World!
Привет, мне 20 лет
```

Можно задавать разделить слов с помощью параметра **sep**. И  
окончание строки с помощью параметра **end**.

```
print('Hello', 'Python', sep='+')
print(1, 2, end='\t')
```

# input

**input** – функция для ввода данных от пользователя.

```
input('Сколько тебе лет?')
```

Чтобы использовать полученное значение в программе, сохраните его в переменной:

```
age = input('Сколько тебе лет?')
```

Функция **input()** сохраняет **строку**.



# Строки

---

# Строки

Строка – последовательность символов.

Свойства строк:

- Можно применять некоторые арифметические операции (+, \*).
- Поддерживают индексацию.
- У строк есть множество методов (**find**, **split**).

# Преобразование типов

**int()** – преобразование в число.

**str()** – преобразование в строку.

**bool()** – преобразование в булево значение.

Примеры:

```
age = int(input('Сколько тебе лет?')) # вернёт число,  
введенное пользователем
```

```
number = 10  
print(str(number)) # вернёт строку '10'
```



# Комментарии

# Комментарии

Это текст, который присутствует в коде программы, но игнорируется интерпретатором. Используются для того, чтобы добавить объяснение для определенного блока кода.

Написание комментария начинается с символа #.

```
name = input('Enter your name') # сохраняем имя пользователя
```



## Задания с автопроверкой

[hackerrank](https://leetcode.com/)

**HackerRank** 



# Домашнее задание

Давайте посмотрим ваше [домашнее задание](#).

- Вопросы по домашней работе задаём в чате Slack!
- Задачи можно сдавать по частям.
- Зачёт по домашней работе проставляется после того, как приняты **все задачи**.

**Задавайте вопросы и  
пишите отзыв о лекции!**

**Евгений Шмаргунов**



[eshmargunov@gmail.com](mailto:eshmargunov@gmail.com)



[evgeny.shmargunov](https://www.facebook.com/evgeny.shmargunov)