



Первая программа

Занятие 2



Дмитрий Демидов
Разработчик Яндекс



Дмитрий Демидов

Разработчик в Яндекс



[Дмитрий Демидов](#)

Вопросы по домашнему заданию



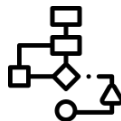
Andrea Picquadio, Pexels

План занятия

1. Алгоритмы
2. Условные конструкции
3. Логический тип данных
4. Первая версия программы
5. Циклы
6. Домашнее задание

Алгоритмы

1



Алгоритмы

Программа = Данные + Алгоритмы

Простой алгоритм:

1. Пойти в магазин.
2. Купить десяток яиц.
3. Купить колбасу.
4. Вернуться домой.

Усложняем алгоритмы

Жена посылает программиста в магазин:

- Дорогой, купи, пожалуйста, палку колбасы, и если будут яйца, то купи десяток.

Через полчаса программист возвращается с десятью палками колбасы.

Жена:

- Что это?! Зачем ты купил столько колбасы?

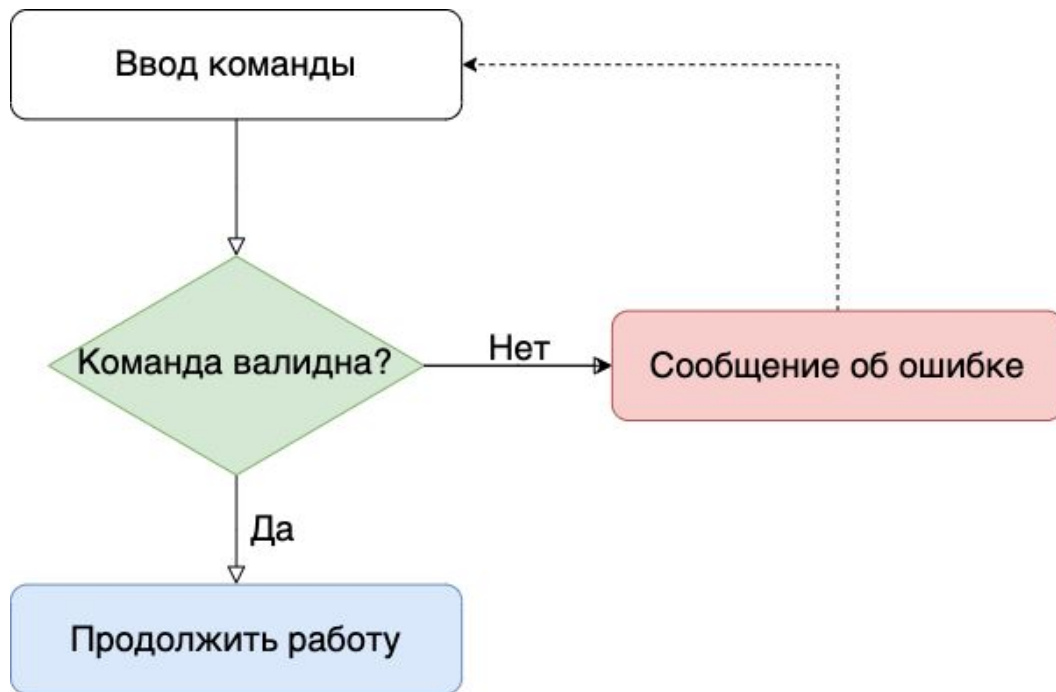
Программист:

- Ну так яйца-то были...

Алгоритм

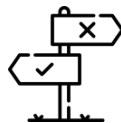


Практический пример



Условные конструкции

2



Условные конструкции в Python

```
if <логическое выражение>:  
    <блок кода 1>
```

Если <логическое выражение> принимает значение True, выполнить <блок кода 1>.

А если False? Ничего не делать.

Условные конструкции в Python

```
if <логическое выражение>:  
    <блок кода 1>  
else:  
    <блок кода 2>
```

- Если <логическое выражение> принимает значение True, выполнить <блок кода 1>.
- Иначе выполнить <блок кода 2>.

Условные конструкции в Python

Что если нужно задать несколько разных условий в одном месте?

У меня нет куриных яиц для пирога, но есть перепелиные.

```
if <логическое выражение>:  
    <блок кода 1>  
elif <логическое выражение 2>:  
    <блок кода 2>  
elif <логическое выражение 3>:  
    <блок кода 3>
```

- Если <логическое выражение> принимает значение True, выполнить <блок кода 1>.
- Иначе если <логическое выражение 2> принимает значение True, выполнить <блок кода 2> и т.д.

Логический тип данных

3



Логический тип данных

Принимает одно из двух значений: **True/False**

Пример: Результаты сравнений (==, <, >, >=, <=, !=)

Основные операции с логическим типом данных:

- and: Логическое И
- or: Логическое ИЛИ
- not: Отрицание

Как вы думаете, зачем нужны логические операции?

Логическое И

x	y	x and y
False	False	False
False	True	False
True	False	False
True	True	True

Логическое ИЛИ

x	y	x or y
False	False	False
False	True	True
True	False	True
True	True	True

Логическое отрицание

x	not x
False	True
True	False

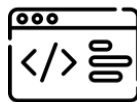
Важно запомнить!

- Код внутри блока выделяется отступами.
- Оператор `if` может быть без `else` и `elif`, но не наоборот.
- Операторы `elif`, относящиеся к одному блоку `if` - взаимоисключающие.

Продemonстрируем каждый из пунктов в коде.

Первая версия программы

4



Что будет делать наша программа

Функциональность

- Запрашивать у пользователя команду.
- В зависимости от введенной команды выполнять действие.

Команды

- **help** - напечатать справку по программе.
- **add** - добавить задачу в список (название задачи запрашиваем у пользователя).
- **print** - напечатать все добавленные задачи.

Циклы

5



Все хорошо, но...

Мы разработали основную функциональность программы

Но она более или менее бесполезна, так как весь код выполнится только один раз.

Что же делать?

Используем **циклы!**

Что такое циклы?

Способ организовать многократное выполнение кода.

Цикл while

```
while <логическое выражение>:
```

```
    <блок кода>
```

```
    break
```

- Пока истинно (равно True) <логическое выражение>, выполняй <блок кода>.
- Если встретишь break, прервись (необязательное условие).

Домашнее задание

Давайте посмотрим ваше домашнее задание.

6



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



Дмитрий Демидов