

# UCHI.RU

**Подходы к  
обучению программированию  
на языке Python  
в современной школе**

Спикер:  
**Сергей Шашков**



## Этапы развития самостоятельности ученика





### **Зависимый**




«Я сделаю, а ты посмотришь»

Лекции, демонстрации, задания с немедленной обратной связью





## Этапы развития самостоятельности ученика

-  **Зависимый**  
«Я сделаю, а ты посмотришь»  
Лекции, демонстрации, задания с немедленной обратной связью
-  **Заинтересованный**  
«Мы сделаем вместе»  
Более сложные задания, обсуждения, вовлечение





## Этапы развития самостоятельности ученика

-  **Зависимый**  
«Я сделаю, а ты посмотришь»  
Лекции, демонстрации, задания с немедленной обратной связью
-  **Заинтересованный**  
«Мы сделаем вместе»  
Более сложные задания, обсуждения, вовлечение
-  **Вовлечённый**  
«Ты сделаешь, а я поддержу»  
Семинары, групповые проекты





## Этапы развития самостоятельности ученика

-  **Зависимый**  
«Я сделаю, а ты посмотришь»  
Лекции, демонстрации, задания с немедленной обратной связью
-  **Заинтересованный**  
«Мы сделаем вместе»  
Более сложные задания, обсуждения, вовлечение
-  **Вовлечённый**  
«Ты сделаешь, а я поддержу»  
Семинары, групповые проекты
-  **Самостоятельный**  
«Ты сделаешь самостоятельно»  
Самостоятельная учебная работа, стажировки

## Осмысленная проектная деятельность школьников начинается здесь:

- |   |   |
|---|---|
|  <b>Зависимый</b><br>«Я сделаю, а ты посмотришь»        | Лекции, демонстрации, задания с немедленной обратной связью |
|  <b>Заинтересованный</b><br>«Мы сделаем вместе»         | Более сложные задания, обсуждения, вовлечение               |
|  <b>Вовлечённый</b><br>«Ты сделаешь, а я поддержу»      | Семинары, групповые проекты                                 |
|  <b>Самостоятельный</b><br>«Ты сделаешь самостоятельно» | Самостоятельная учебная работа, стажировки                  |

# Чтобы успешно перейти к проектам, нужно сначала заинтересовать, вовлечь, не напугать

- |   |   |
|---|---|
|  <b>Зависимый</b><br>«Я сделаю, а ты посмотришь»        | Лекции, демонстрации, задания с немедленной обратной связью |
|  <b>Заинтересованный</b><br>«Мы сделаем вместе»         | Более сложные задания, обсуждения, вовлечение               |
|  <b>Вовлечённый</b><br>«Ты сделаешь, а я поддержу»      | Семинары, групповые проекты                                 |
|  <b>Самостоятельный</b><br>«Ты сделаешь самостоятельно» | Самостоятельная учебная работа, стажировки                  |

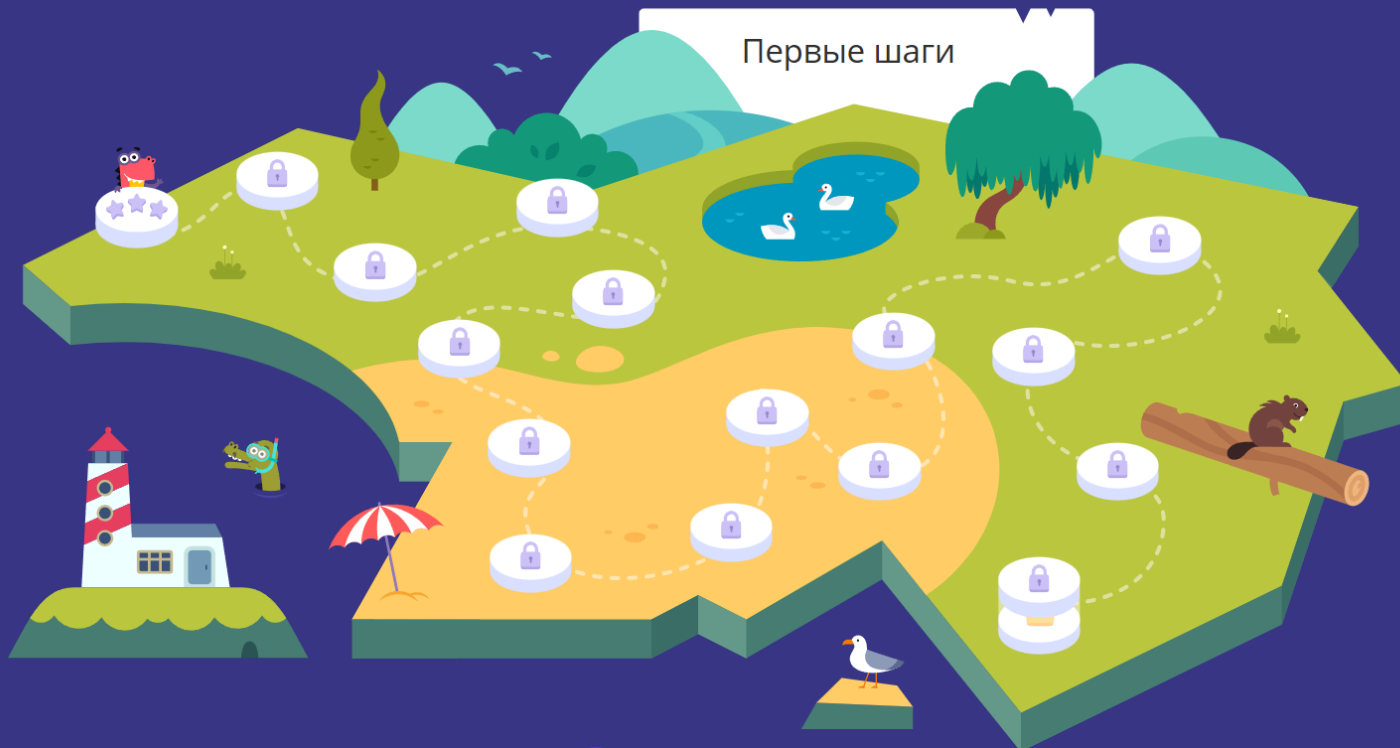
# Игровое обучение

★ 0 Сергей Шашков

Программирование



Первые шаги





## Простая и понятная «теория»



Это новый велосипед `bike` ! Он также может двигаться вправо `right` , вниз `down` , влево `left` и вверх `up` , но ему внутри скобок нужно указать число шагов.

Это называется *параметр*.

```
1 bike.right()
2 bike.right()
3 bike.right()
4 bike.right()
```

→ 1 bike.right(4)

Воспользуйся этим, чтобы побыстрее перебраться через парковку!

[К заданию >](#)

# Немедленная реакция на ошибки и помощь

Напиши две команды `scooter.down()`, чтобы скутер провёз Гришу на две клетки вниз



Нужно проехать ещё одну клетку

garden.py

# scooter

```
1 scooter.down()
```

```
2
```

# Немедленная реакция на ошибки и помощь

🔊 Прокопай ров через все лунки к реке. Используй циклы и функции, чтобы получить короткий код.



desert.py

```
#bulldozer
```

```
1 square()  
2 bulldozer.backward()  
3  
4
```


Допиши действие forward/backward/left/right

```
#functions
```

```
1 def square():  
2     for i in range(4):  
3         bulldozer.forward(4)  
4         bulldozer.right(90)  
5
```



# Очень помогает скорость и отзывчивость интерфейса

 Прокопай ров через все лунки к реке.  
Используй цикл, чтобы получить короткий код.



desert.py





#bulldozer

1

bulldozer 

for 

indent 

delete 



## Доступно через платформу «Цифровой образовательный контент» (ЦОК)

- **Широкая доступность инновационных образовательных продуктов** для жителей РФ, неограниченный бесплатный доступ к верифицированному цифровому образовательному контенту;
- **Рост эффективности образовательного процесса** благодаря созданию индивидуальной траектории обучения каждого ученика;
- **Снижение финансовой нагрузки на родителей;**
- **Развитие инноваций в образовании** благодаря системному партнерству государства и бизнеса

## ЦОК и Uchi.ru

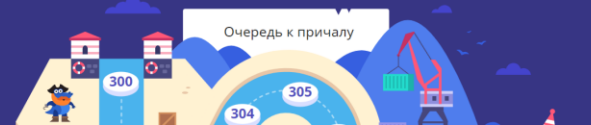
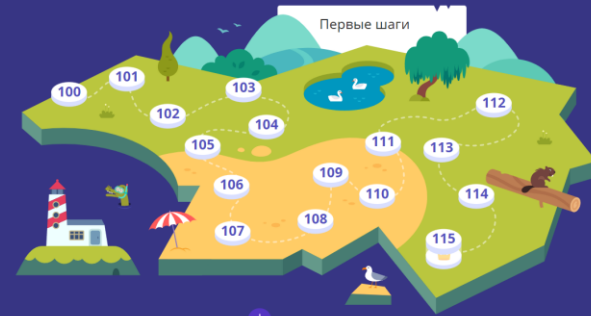
- Уже больше 1,3 млн учеников получили доступ;
- Более 300 тысяч школьников ежемесячно занимаются;
- Более 150 тысяч учителей получили доступ и предлагают своим ученикам возможность заниматься;

# Удобно для учителей

№	Ученик	Прогресс	Посл. активн. ←	ПЕРВЫЕ ШАГИ
1	Максимов Я.	100%	2 нед.	●●●●●●●●●●
2	Фишер М.	78%	2 мес.	●●●●●●●●●●
3	Гутор Е.	50%	2 мес.	●●●●●●●●●●
4	Гизатулина А.	35%	4 дн.	●●●●●●●●●●
5	Воробьёв А.	28%	6 мес.	●●●●●●●●●●
6	Акарачкина А.	25%	1 мес.	●●●●●●●●●●
7	Кириллов А.	8%	5 мес.	●●●●●●●●●●
8	Акентьев М.	5%	7 мес.	●●●●●●●●○
9	Федорова А.	5%	7 мес.	●●●●●●●●○
10	Шиляева С.	5%	7 мес.	●●●●●●●●○
11	Гладков И.	3%	7 мес.	●●●●●●●○○
12	Кочкин А.	3%	1 нед.	●●●●●●○○○○
13	Конищева Е.	2%	4 мес.	●●●●○○○○○○

## Содержание курса

- > Общие понятия  
Про Питон и структуру курса
- > 1. Первые шаги  
Интерфейс, передвижение
- > 2. Полный разворот  
Вперед-назад-поворот  
Информация о теме  
200-204.  
Движение, вид сверху  
205-207.  
Задний ход  
208-210.  
Закрепляем  
211.  
«Экзамен»
- > 3. Очередь к причалу  
Задачи на очереди
- > 4. Если пути нет  
Конструкция «если» if
- > 5. Сокобан  
Движения плюс действия
- > 6. -270 градусов  
Повороты на угол + цикл for
- > 7. Большая сортировка  
Конструкция «если» if
- > 8. Происки в тупиках  
Задачи на стеки
- > 9. Спасательные функции  
Циклы for и функции



# Ещё у нас есть несколько учебных проектов для продолжения



## Определитель животных



### 3. По порядку



### 4. Те же и дельфин

Теперь добавим к нашим двум рыбам дельфина. Тебе пригодится уже написанный в предыдущей задаче код. Придумай вопрос, который "отделит" дельфина от щуки и акулы.

Твой код будет выглядеть примерно так:

```
from ask import *

if ask('КАКОЙ-ТО ВОПРОС') == 'YES':
    if ask('Живёт только в пресной воде?') == 'YES':
        print('щука')
    else:
        print('акула')
else:
    print('дельфин')
```

Текст программы из предыдущей задачи здесь сдвинут внутри условного оператора. Несколько строк кода можно сдвинуть вправо на величину стандартного отступа. Для этого надо выделить нужный блок кода и нажать клавишу Tab.

Получить подсказку

### 5. Двуногие, часть первая

### 6. Двуногие, часть вторая

4. Те же и дельфин

```
1 from ask import *
2
3 if ask('Есть жабры?') == 'YES':
4     if ask('живёт только в пресной воде?') == 'YES':
5         print('щука')
6     else:
7         print('акула')
8 else:
9     print('дельфин')]
```

canvas

Загадай любое из этих животных :

акула, щука, дельфин

Готово

Testing 0/3

#### ✓ 1 Тест 1

✓ 2 Отличный результат!

✓ 3 Вывод

Загаданное животное

акула

акула

#### Тест 2

Отличный результат!

Вывод

Загаданное животное

щука

щука


ЗАПУСТИТЬ



# UCHi.RU

8 800 500 30 72 • [info@uchi.ru](mailto:info@uchi.ru) • [uchi.ru](http://uchi.ru)

**Сергей Шашков**  
руководитель направления  
«Программирование»  
образовательной платформы Учи.ру

 [shashkov@uchi.ru](mailto:shashkov@uchi.ru)