



Урок 2. Python: Документирование и читаемость кода

Часть 1. Материалы ученика

Программирование на Python и C++ · 8 класс

Урок 2 из 33 · 45 минут · Модуль «Документирование программы. Процедуры с параметрами»

Уровень: начинающий программист; Python 3.

Программа: <https://schoolotzyv.ru/blog/python-cpp-8-klass-programma>

Код $2*a+2*b-4$ работает, но непонятно, что он считает. Отредактируем программу рамки так, чтобы другой человек смог проверить смысл и не посчитать углы дважды.

Результат урока: Улучшить имена, структуру и документацию без изменения результата; доказать сохранение поведения тестами.

01. Проблемная ситуация

0–5 минут. Прямоугольная сетка имеет 5 столбцов и 4 строки. Посчитайте граничные клетки рисунком в тетради. Сравните формулы $2*5+2*4$ и $2*5+2*4-4$. Объясните, что означают лишние четыре клетки и какие имена нужны вместо a, b .

02. Понятия и разобранный пример

5–14 минут: 4 минуты на понятия, 3 на пример, 2 на объяснение способа партнёру.

Имена объясняют предмет, комментарий — причину

Рефакторинг меняет устройство кода, сохраняя его наблюдаемое поведение на согласованных входах. До правки фиксируют контракт и тесты. Переименование переменной должно охватить все её употребления: частичная замена может дать ошибку или скрыто изменить вычисление.

`width` и `height` показывают смысл параметров. Docstring описывает результат и допустимые размеры. Комментарий «умножаем на два» повторяет код и почти не помогает; «боковые стороны считаем без уже учтённых углов» объясняет причину `height-2`. Для размеров меньше 2 наша формула и контракт не предназначены.

Короткая функция решает одну задачу. Ввод и печать оставляем снаружи, чтобы её можно было проверять отдельно. Читаемость — не максимальное число комментариев: устаревшее описание опаснее точного краткого. В этом уроке не добавляем новый алгоритм под видом переименования.

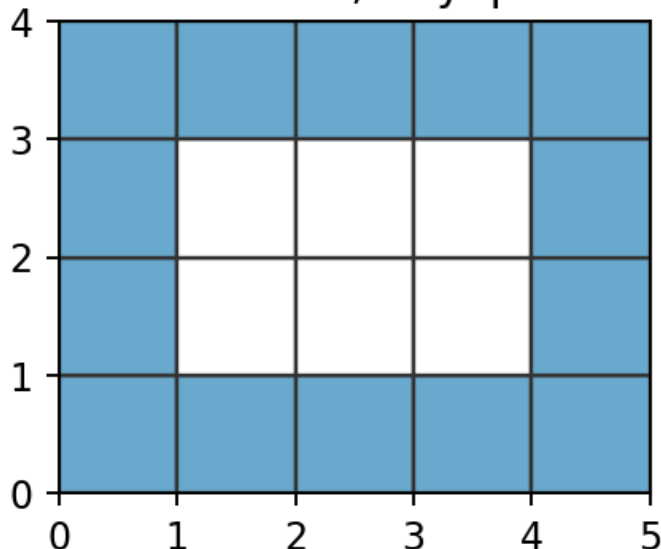
Отступы набирайте обычными пробелами: 4 на уровень. При копировании с веб-страницы замените неразрывные пробелы обычными; полный код с обычными пробелами есть в PDF.

Для 5×4 верхняя и нижняя стороны дают 10 клеток, боковые без углов — $2 \cdot (4-2)=4$. Всего 14. Для 2×2 получаем 4, внутренних клеток нет. В старой записи `def f(a,b): return 2*a+2*b-4` то же поведение на целых $2 \dots 100$.

```
def border_cells(width, height):  
    """Вернуть число клеток рамки; целые размеры 2...100."""  
    return 2 * width + 2 * (height - 2)  
  
print(border_cells(5, 4))
```



Рамка: 14 клеток; внутренность: 6



Рамка 5×4: две горизонтальные стороны по 5 и две боковые части по 2; всего 14 клеток.

03. Практика 1. Проверяем материал

Рефакторинг с контрольными примерами

14–25 минут. Отрефакторьте старую функцию $f(a,b)$ из примера. Используйте осмысленные имена, краткий docstring и один комментарий о двойном счёте углов.

Размеры	Рамка / внутренние клетки
2×2	4 / 0
3×3	8 / 1
5×4	14 / 6
2×9	18 / 0
100×100	396 / 9604

2 минуты — нарисуйте контрольный прямоугольник; 4 — новая версия; 3 — сравните обе версии на всех пяти тестах; 2 — партнёр объясняет код без подсказки автора. Продукт: две версии и журнал совпадений. Критерии: неизменные ответы, смысл имён, границы, полезный комментарий.

04. Практика 2. Перенос способа

Новая функция для внутренних клеток

25–33 минуты. Теперь задача действительно меняется: нужна `interior_cells(width,height)`, возвращающая число внутренних клеток при тех же ограничениях. За 2 минуты запишите новый контракт, за 4 реализуйте, за 2 проверьте таблицу. Объясните, почему это новая функция, а не только рефакторинг `border_cells`.

05. Обсуждение и границы вывода

33–40 минут. Сравните комментарии «возвращаем результат» и «углы уже учтены». Какой помогает проверить формулу? Можно ли назвать замену рамки на внутренность рефакторингом, если код стал понятнее? Подкрепите ответ парой результатов 5×4.

06. Индивидуальная проверка

40–45 минут. Для сетки 6×3 найдите рамку и внутренность. Предложите имя вместо q для числа внутренних клеток. Напишите одно ограничение контракта. Четыре элемента — четыре балла.



Часть 2. Ответы и материалы учителя

Практика 1: разбор и критерии

Вариант ниже даёт 4,8,14,18,396. Проверка только 3×3 не доказывает правильность на вытянутых прямоугольниках; 2×9 проверяет нулевую длину одной внутренней стороны. Не меняйте −4 на −2 при переименовании.

```
def f(a, b):  
    return 2*a + 2*b - 4  
  
def border_cells(width, height):  
    """Вернуть размер рамки; целые width,height в 2...100."""  
    # Углы уже учтены в верхней и нижней сторонах.  
    return 2 * width + 2 * (height - 2)  
  
for w, h in [(2,2), (3,3), (5,4), (2,9), (100,100)]:  
    print(f(w,h), border_cells(w,h))
```

Практика 2: разбор и критерии

Внутренний прямоугольник имеет стороны width−2 и height−2. Ответы 0,1,6,0,9604. Иное назначение и результаты требуют нового имени и контракта. У границы 2 одна сторона внутреннего прямоугольника равна нулю.

```
def interior_cells(width, height):  
    """Вернуть внутренние клетки; целые размеры 2...100."""  
    return (width - 2) * (height - 2)  
  
for w, h in [(2,2), (3,3), (5,4), (2,9), (100,100)]:  
    print(interior_cells(w, h))
```

Обсуждение: возможный вывод

Комментарий об углах объясняет вычитание; комментарий о return повторяет синтаксис. При 5×4 прежний результат 14, новый 6: изменилось поведение и назначение. Это новая функция. Читаемость оцениваем тем, может ли читатель восстановить смысл и проверить ограничения.

Индивидуальная проверка: ответ и критерии

Рамка 14; внутренность 4; например inner_count; целые размеры 2...100. По баллу за элемент. Сумма 14+4=18 даёт независимую проверку полного числа клеток.

Учителю: подготовка и план на 45 минут

Подготовьте компьютер, редактор, папку с правом записи, распечатку ученической части PDF, бумагу и ручку. Python 3.11+ и IDLE или редактор; запуск python lesson.py (либо python3 lesson.py). Внешние библиотеки не нужны. Установку и проверку запуска учитель выполняет заранее. Все программы, данные и ожидаемые ответы даны; регистрация детей и интернет на занятии не нужны.

Уровень — начинающий программист, 8 класс. Предпосылки: контракт функции, параметры и return. В паре автор прогнозирует и вводит код, проверяющий ведёт журнал, затем роли меняются. Без компьютера выполните трассировку полного кода на бумаге и сопоставьте с готовыми результатами; учитель демонстрирует подготовленный запуск. Бумажный вариант не считается фактическим запуском ученика. Перед практикой учитель выдаёт готовый файл разобранного примера, чтобы дети тратили время на изменение и проверку, а не длинное перепечатывание. Матрица программы предназначена учителю.



Минуты	Работа	Результат
0–5	Проблема	Вопрос и первое решение
5–14	Понятия и пример	Способ проверки
14–25	Практика 1	Готовый продукт и проверка
25–33	Практика 2	Перенос способа
33–40	Обсуждение	Аргумент и ограничение
40–45	Индивидуальная проверка	Самостоятельный ответ

07. Соответствие исходной программе

Пункт программы	Действие ученика	Проверяемый результат
Документирование программы. Процедуры с параметрами	Разбирает пример и создаёт две подпрограммы	Исходники, прогнозы и протокол проверки
Составление и отладка программ	Сравнивает фактический результат с контрактом	Объяснение границ и самостоятельный ответ

Источники и права на изображения

Исходная программа курса

<https://schoolotzyv.ru/blog/python-cpp-8-klass-programma>

Python: функции и управление выполнением

<https://docs.python.org/3/tutorial/controlflow.html>

Тексты, учебные данные, реплики, программы и схемы подготовлены для уроков портала «Школы России». Числовые серии — учебные модели, не результаты проведённых классом опытов. Схемы авторские, сторонние фотографии не используются. Методические ссылки приведены для учителя; ученик получает все материалы здесь. Проверка первичных источников: 18.10.2026.