**ПРАВИЛА И РЕКОМЕНДАЦИИ ПО КОДИРОВАНИЮ НА PYTHON**

Все описанные ниже правила разделены на три категории: запреты, требования, рекомендации.

**Запреты** описывают положения, которые нельзя нарушать ни при каких условиях.

**Требования** описывают то, что должно выполняться всегда и в любом проекте. В каждом конкретном проекте лидер проекта может принять решение, что определённые требования могут не соблюдаться. При этом должны быть проделаны следующие шаги: следует доказать необходимость исключений, следует описать, где и по какой причине были сделаны исключения.

**Рекомендации** описывают то, что может существенно облегчить жизнь разработчикам. Пренебрежение рекомендациями не приведёт к провалу проекта, но может осложнить его сопровождение или взаимодействие между разработчиками.

1. Запреты

* Рекомендуется использовать 4 пробела на каждый уровень отступа. Python 3 запрещает смешивание табуляции и пробелов в отступах. Код, в котором используются и те, и другие типы отступов, должен быть исправлен так, чтобы отступы в нем были расставлены только с помощью пробелов.

Хорошо

def•no\_tab\_using():

••••no\_tab•=•'Using•4•spaces'

Плохо

def•use\_tab():

→| one\_tab\_using•=•'Ugly'

* Когда вы генерируете исключение, пишите raise ValueError('message') вместо старого синтаксиса raise ValueError, message.

Старая форма записи запрещена в python 3.

Такое использование предпочтительнее, потому что из-за скобок не нужно использовать символы для продолжения перенесенных строк, если эти строки длинные или если используется форматирование.

* Двойные подчеркивания (в начале и конце имен) зарезервированы для языка.

Хорошо

my\_variable = 'Variable'

Плохо

\_\_myvariable\_\_ = 'Variable'

1. Требования

* Ограничьте длину строки максимум 79 символами.

Стандартная библиотека Python консервативна и требует ограничения длины строки в 79 символов (а строк документации/комментариев в 72).

Предпочтительный способ переноса длинных строк является использование подразумеваемых продолжений строк Python внутри круглых, квадратных и фигурных скобок. Длинные строки могут быть разбиты на несколько строк, обернутые в скобки. Это предпочтительнее использования обратной косой черты для продолжения строки.

* Комментарии, противоречащие коду, хуже, чем отсутствие комментариев. Всегда исправляйте комментарии, если меняете код!

Комментарии должны являться законченными предложениями. Если комментарий — фраза или предложение, первое слово должно быть написано с большой буквы, если только это не имя переменной, которая начинается с маленькой буквы (никогда не изменяйте регистр переменной!).

Если комментарий короткий, можно опустить точку в конце предложения. Блок комментариев обычно состоит из одного или более абзацев, составленных из полноценных предложений, поэтому каждое предложение должно оканчиваться точкой.

Ставьте два пробела после точки в конце предложения.

Программисты, которые не говорят на английском языке, пожалуйста, пишите комментарии на английском, если только вы не уверены на 120%, что ваш код никогда не будут читать люди, не знающие вашего родного языка.

1. Рекомендации

* Не разделяйте ваши строки с помощью точек с запятой и не используйте точки с запятой для разделения команд, находящихся на одной строке.

Хорошо

a = 'String'

b = 15

c = 7.2

Плохо

a = 'String';

b = 15; c = 7.2;

* Не ставьте пробелы внутри каких-либо скобок (обычных, фигурных и квадратных).

Хорошо

pineapple(pine[1], {apple: 2})

Плохо

pineapple( pine[ 1 ], { apple: 2 } )

* Если есть необходимость пройти в цикле по ряду чисел, то метод range будет намного приемлемее, как минимум потому, что этот метод потребляет намного меньше памяти, чем вариант в блоке "Плохо". А представьте, что у вас ряд из трёх миллиардов последовательных чисел!

Хорошо

for i in range(6):

print(i\*\*2)

Плохо

for i in [0, 1, 2, 3, 4, 5]:

print(i\*\*2)

**Чек-лист для проверки корректности кода**

**Appearance**

1. Обновляется история изменений (версия, данные, автор и комментарии к изменениям) и информация об авторских правах.
2. Все комментарии написаны на английском языке.
3. Размещение скобок, отступы, интервалы, соглашение об именовании и длина строки соответствуют стандарту кодирования.

**Miscellaneous**

1. Каждая строка кода содержит не более одного оператора.

**Constants**

1. Все константы записаны заглавными буквами, а слова разделены символами подчёркивания.