

Спецификация PEP 8

№ урока: 8 **Курс:** Python Starter

Средства обучения: PyCharm

Обзор, цель и назначение урока

Получение знаний спецификации PEP 8 и приобретения навыков писать простой и читаемый код.

Изучив материал данного занятия, учащийся сможет:

- Понимать основные цели и причины создания спецификаций PEP.
- Писать более простой и понятный код, основанный на лучших практиках данной спецификации.

Содержание урока

1. Понятия и определения.
2. Основные цели и причины создания спецификаций PEP.
3. Примеры кода из реальных проектов.
4. Применение и изучение оформления различных конструкций из PEP 8 на практике.

Резюме

За время изучения Python Starter мы неоднократно упоминали термин PEP 8. Разберёмся, что это такое.

PEP 8 - это руководство по написанию кода на Python.

Этот документ описывает соглашение о том, как писать код для языка Python, включая стандартную библиотеку, входящую в состав Python.

Ключевая идея Гвидо Ван Россума (создателя Python) такова: код читается намного больше раз, чем пишется. Рекомендации о стиле написания кода направлены на читаемость кода и его согласованность между большим числом проектов. В идеале, весь код будет написан в едином стиле, и любой сможет легко его прочесть.

Две причины для того, чтобы нарушить данные правила: 1. Когда применение правила сделает код менее читаемым даже для того, кто привык читать код, который следует правилам. 2. Чтобы писать в едином стиле с кодом, который уже есть в проекте и который нарушает правила (возможно, в силу исторических причин) — впрочем, это возможность переписать чужой код.

О качестве работы программиста можно сказать по его коду. А соблюдение соглашений PEP8 - это ключевой показатель качества кода.

Если программист соблюдает правила оформления кода — это значит:

- Код будет легче поддерживать
- Он заботится о времени других программистов, которым необходимо будет работать с его кодом

Полный перечень правил PEP 8:

<https://www.python.org/dev/peps/pep-0008/>

Помимо PEP 8 существует ряд других спецификаций, которые описывают другие аспекты программирования на Python. Например, спецификация PEP 7 описывает правила оформления кода CPython, в случае если вы пишете свои собственные C-расширения для языка Python.

В качестве еще одного примера, существует специальный модуль, который связан с нашим предыдущим уроком «Типизация в Python: модуль typing». Для данного модуля была разработана собственная спецификация 484, в которой можно найти советы по использованию и все возможности данного модуля.

Язык Python не имеет какой-то жесткой системы правил, которые утвердились во всех компаниях и во всем сообществе. Но за долгие годы написания кода на Python возникает необходимость создавать некоторые стандарты, которым бы следовали программисты.

Если взять в пример Java, то у данного языка существуют только лишь один свод общих правил оформления кода, к которому придерживаются все Java-разработчики. Однако, взяв два или три проекта, написанными разными компаниями или разработчиками из разных команд, мы увидим, что кодовая база и её стилизация разительно отличаются. Это и становится причиной того, чтобы придерживаться некоторых рекомендаций по стилизации Python кода, чтобы наш код был более понятен и прост для чтения, ведь, как правило, код больше читают, нежели пишут.

Необходимо понимать, что в написании данных стандартов участвовал и сам создатель языка Python – Гвидо ван Россум, что, как минимум, является показателем целесообразности данных рекомендаций.

Но также, надо учитывать, что PEP 8 не является исчерпывающим руководством по оформлению кода. Зачастую, в практике, вы будете встречать множество неоднозначных ситуаций, в которых сами будете принимать решения. Однако, основываясь вы будете на основных идеях PEP 8.

В Python Starter разбираются простейшие конструкции языка. PEP8 охватывает все возможности Python. В материалах Python Essential и Python Advanced будет уделяться внимание о правильном оформлении кода согласно PEP 8.

Закрепление материала

- Что такое PEP 8 и существуют ли еще спецификации?
- Зачем нужны правила оформления кода?
- Является ли PEP 8 исчерпывающим руководством?

Самостоятельная деятельность учащегося

Задание

Откройте файл fix_me.py из папки homework. Используя обычный текстовый редактор (Notepad), исправьте все ошибки оформления кода согласно PEP 8.

Рекомендуемые ресурсы

Официальный сайт Python – список спецификаций

<https://www.python.org/dev/peps/>

Официальный сайт Python – спецификация PEP 8

<https://www.python.org/dev/peps/pep-0008/>