**Правила написания кода:**

**Рекомендации:**

1. Документируйте ваш код - добавьте комментарии, чтобы другие разработчики могли легко понять вашу логику и функции.
2. Называйте переменные и функции осмысленно и понятно. Не использовать в качестве названия одну букву.
3. Соблюдайте стандарты оформления кода, такие как стиль написания кода, отступы, и т.д. Это сделает ваш код более читаемым и легко поддерживаемым. (в данном случае PEP8)
4. Используйте Git для хранения и управления исходными кодами проекта. Также это поможет откатиться на старую версию, в случае поломки кода.
5. Не делайте изменений на master-ветке без предварительных договоренностей.

**Запреты:**

1. Не используйте общие названия переменных, такие как "temp" или "var". Это затрудняет чтение кода другими разработчиками.
2. Не оставляйте в проекте закомментированный код. Он занимает место и затрудняет чтение кода.
3. Не используйте устаревшие методы и библиотеки, если они не рекомендованы для использования. В дальнейшем это упростит рефакторинг кода.
4. Не вносите изменения в код другого разработчика без его согласия или без документирования причин изменения. Это может привести к неопределенному поведению и багам.
5. Не используйте "магические числа" (константы, заданные прямо в коде), которые могут усложнить понимание кода.

**Требования:**

1. Код должен быть разбит на логические блоки (например, функции, классы) для повышения читабельности и возможности повторного использования. Функции должны быть не длиннее 25 строк. Следует избегать множественного наследования.
2. Код должен быть написан с использованием архитектурных принципов и паттернов, чтобы сделать его более поддерживаемым и расширяемым в будущем.

**ЧЕК ЛИСТ:**

1. Код должен быть оформлен согласно конвенциям и стандартам кодирования на Python.
2. Код должен быть понятен и легко читаем, с объяснением сложных участков кода и использованием комментариев, если необходимо.
3. Код должен быть структурирован в логические блоки, такие как функции, классы и модули, для повторного использования и облегчения поддержки.
4. Код должен быть написан с использованием современных методов и библиотек.
5. Код должен быть безопасным, без уязвимостей или ошибок, которые могут привести к утечкам данных или повреждению системы.
6. Код должен быть документирован, с описанием каждой функции, класса или метода, и задокументированы все зависимости от сторонних библиотек.
7. Код должен быть разделен на отдельные коммиты с понятными сообщениями об изменениях.
8. Код должен быть согласован и соответствовать заданным требованиям и спецификациям.