

Prompt

ChatGPT для программиста

50 готовых промптов

для отладки, ревью
и тестов

ChatGPT

YandexGPT

GigaChat

Стас Ростов

Стас Ростов

**ChatGPT для программиста:
50 готовых промптов для
отладки, код-ревью и тестов**

«Автор»

2026

Ростов С.

ChatGPT для программиста: 50 готовых промптов для отладки, код-ревью и тестов / С. Ростов — «Автор», 2026

50 готовых промптов для программистов: поиск и исправление ошибок, разбор чужого кода, код-ревью, рефакторинг, тесты, документация, подготовка к собеседованию и план роста. Примеры в книге на Python, но промпты подходят для любого языка: укажите свой в скобках. Что внутри: Отладка и разбор чужого кода; Код-ревью и рефакторинг; Тесты и документация; Обучение, собеседования и рост; 5 примеров готовых ответов нейросети: код работает не так, как ожидаю, код-ревью, тесты для функции, docstring и другие; шпаргалка из 13 фраз, чтобы быстро поправить ответ. Каждый промпт выделен рамкой: скопируйте, замените на свои и отправьте. Работает в ChatGPT, YandexGPT, GigaChat, Claude и других нейросетях.

© Ростов С., 2026

© Автор, 2026

Содержание

Как пользоваться набором	6
Содержание	7
Отладка и разбор чужого кода (1–12)	8
Конец ознакомительного фрагмента.	10

Стас Ростов

ChatGPT для программиста: 50 готовых промптов для отладки, код-ревью и тестов

ChatGPT для программиста: 50 готовых промптов для отладки, код-ревью и тестов

Как пользоваться набором

50 готовых промптов для программистов: поиск и исправление ошибок, разбор чужого кода, код-ревью, рефакторинг, тесты, документация, подготовка к собеседованию и план роста. Примеры в книге на Python, но промпты подходят для любого языка: укажите свой в скобках. Промпты работают в ChatGPT, YandexGPT, GigaChat и Claude.

1. Начните с промпта №39: он узнает ваш уровень и задачи, составит план и короткое описание вас как программиста. Вставляйте это описание перед другими промптами, чтобы ответы были под ваш язык, стек и опыт.

2. Замените всё в [квадратных скобках] на свои данные: язык и его версию, фрагмент кода, полный текст ошибки, что ожидали получить и что получили.

3. Правьте ответ диалогом: «короче», «объясни проще», «покажи без сторонних библиотек».

Важно. Не вставляйте в нейросеть код работодателя или заказчика без их разрешения, а также пароли, токены, ключи API, строки подключения к базам данных и персональные данные пользователей. Заменяйте их заглушками вроде `API_KEY = "..."` или воспроизводите проблему на учебном примере. Код от нейросети запускайте и проверяйте тестами до отправки в репозиторий: она может ошибиться или сослаться на несуществующую функцию. Сверяйтесь с официальной документацией.

Содержание

Отладка и разбор чужого кода (1–12)

1. Объясни ошибку · 2. Код работает не так, как ожидаю · 3. План отладки · 4. Логирование вместо `print` · 5. Работа с отладчиком · 6. Минимальный пример · 7. Вопрос на форум · 8. Разбор чужого кода · 9. Карта проекта · 10. Регулярное выражение · 11. Медленный код · 12. Ошибка установки или импорта

Код-ревью и рефакторинг (13–26)

13. Код-ревью · 14. Граничные случаи · 15. Типичные уязвимости · 16. Длинная функция · 17. Повторяющийся код · 18. Понятные имена · 19. Упрощение условий · 20. Стилль кода · 21. Аннотации типов · 22. Обработка исключений · 23. Ответ на замечания ревьюера · 24. Ревью коллеге · 25. Перенос на другой язык · 26. Описание pull request

Тесты и документация (27–38)

27. Тесты для функции · 28. Тест на найденный баг · 29. Тестовые данные · 30. Подмена внешних ресурсов · 31. Чего не хватает в тестах · 32. Падающий тест · 33. Docstring · 34. README проекта · 35. Комментарии в коде · 36. Документация API · 37. Сообщение коммита · 38. Журнал изменений

Обучение, собеседования и рост (39–50)

39. Мой уровень и план · 40. Новый язык · 41. Учебный проект · 42. Задачи для тренировки · 43. Проверь моё решение · 44. Объясни понятие · 45. Подготовка к собеседованию · 46. Пробное собеседование · 47. Решение вслух · 48. Чтение документации · 49. Оценка задачи · 50. Разбор месяца

Примеры готовых ответов · Как получать ответы лучше · Бонус: шпаргалка «как поправить ответ»

Отладка и разбор чужого кода (1–12)

1. Объясни ошибку. Ты опытный разработчик на [язык, версия]. Вот код: [код]. Вот полный текст ошибки: [traceback целиком]. Объясни простыми словами, что означает ошибка, на какой строке она возникла и почему. Предложи исправление и покажи исправленный фрагмент.

2. Код работает не так, как ожидаю. Вот код: [код]. Входные данные: [данные]. Ожидаю: [результат]. Получаю: [результат]. Найди причину, объясни, почему так происходит, и покажи исправление. Добавь короткую проверку, которая покажет, что ошибка ушла.

3. План отладки. Ошибка воспроизводится [всегда / иногда / только на сервере]. Симптомы: [что видно], уже проверил: [список]. Составь план поиска причины из 5–7 шагов: что проверить первым, какие логи добавить, как сузить место ошибки.

4. Логирование вместо print. Вот функция: [код]. Добавь логирование модулем logging: входные данные, промежуточные значения, выбранная ветка условия. Объясни уровни логов и как отключить подробные, не удаляя их из кода.

5. Работа с отладчиком. Объясни, как пройти мой код [код] по шагам в [pdb / отладчик моей среды]: точка останова, значения переменных, шаг внутрь функции. Если не уверен в названиях кнопок, так и скажи.

6. Минимальный пример. Помоги сократить проблему до минимального воспроизводимого примера: [код, симптом]. Что убрать, что оставить, как заменить реальные данные и секреты вымышленными.

7. Вопрос на форум. Составь вопрос для форума разработчиков: [проблема, код, ошибка, что пробовал]. Структура: заголовок, минимальный пример, ожидаемый и фактический результат, версии. Без секретов.

8. Разбор чужого кода. Вот код, который писал не я: [код]. Объясни, что он делает: сначала целиком в 3 предложениях, потом по блокам. Отметь непонятные места, возможные ошибки и вопросы, которые стоит задать автору.

9. Карта проекта. Вот структура папок и главные файлы: [дерево, фрагменты]. Где точка входа, как связаны модули, с чего начать? Составь план знакомства с кодом на [N дней].

10. Регулярное выражение. Разбери по частям регулярное выражение [выражение] для [язык], приведи 3 строки, которые подходят, и 3, которые нет.

11. Медленный код. Код медленно работает на [объём данных]: [код]. Найди узкие места, объясни сложность простыми словами ($O(n)$, $O(n^2)$) и покажи, как замерить время через `timeit` или `cProfile`.

12. Ошибка установки или импорта. Ошибка при установке или импорте библиотеки: [текст], версии: [язык, менеджер пакетов, ОС]. Перечисли причины и шаги проверки. Команды, которые меняют систему, отметь отдельно.

Конец ознакомительного фрагмента.

Текст предоставлен ООО «Литрес».

Прочитайте эту книгу целиком, [купив полную легальную версию](#) на Литрес.

Безопасно оплатить книгу можно банковской картой Visa, MasterCard, Maestro, со счета мобильного телефона, с платежного терминала, в салоне МТС или Связной, через PayPal, WebMoney, Яндекс.Деньги, QIWI Кошелек, бонусными картами или другим удобным Вам способом.