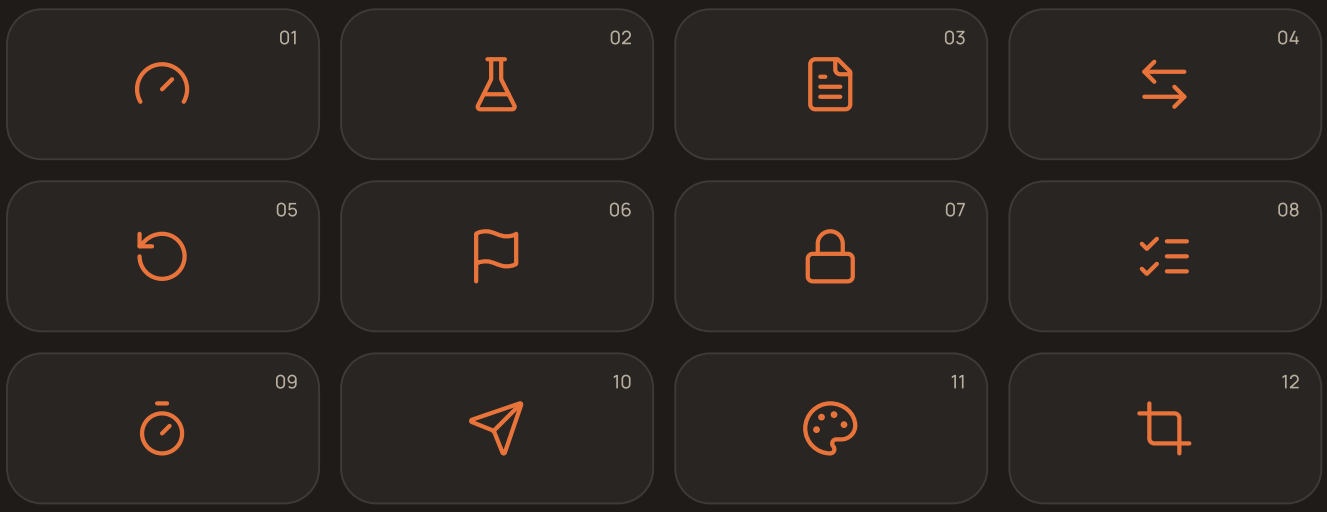
Иван · ИИ на практике

Гайд к ролику · 30.09.2026



Официальный гайд Anthropic на русском.

Opus 5.5: 12 правил промптинга

Что поменять в промптах и настройках, чтобы Opus 5.5 работал быстрее и дешевле. К каждому правилу: где нажать и готовый промпт на русском.

---

12 правил
из документации Anthropic

medium
новый уровень по умолчанию

Sonnet 5.5
вдвое дешевле, чем отличается

ПЕРЕД СТАРТОМ

Что это **и для кого.**

Для кого. Для тех, кто работает в Claude по подписке: в чате, в десктопном приложении или в Claude Code. Программировать не нужно. Пара правил касается тех, кто собирает своих ботов через API, они помечены.

Что внутри. 12 правил из официального гайда Anthropic по Opus 5.5. К каждому: что сделать, где нажать и готовый промпт на русском. В конце коротко про Sonnet 5.5 и чем он отличается.

Как пользоваться. Проще всего закинуть этот PDF в Claude и написать: «Какие из этих правил подходят к моей работе? Помоги настроить». Промпты написаны одним абзацем, их можно копировать целиком.

Данные на 30.09.2026. Модели и настройки меняются быстро, лимиты и даты проверяйте у себя.

01 Усилие и инструкции

Правило 1. Начинайте со среднего усилия

Усилие (effort) это настройка, сколько Claude думает перед ответом. На Opus 5.5 по умолчанию теперь medium. У прошлого Opus был high. По тестам Anthropic, Opus 5.5 на medium не хуже, чем Opus 5 на high, а на части задач по коду даже low подходит близко и стоит заметно дешевле.

- В Claude Code наберите `/effort`, выберите уровень ползунком, нажмите Enter. Уровень сохранится для этой модели. Клавиша `s` вместо Enter применит его только к текущей сессии.
- Уберите из своих промптов и сохранённых инструкций «подумай хорошенько», «рассуждай пошагово» и похожие строки. Opus 5.5 думает перед каждым ответом всегда и сам решает, сколько. В тестах Anthropic без такой строки ответы начинали приходить быстрее, качество не падало.
- Поработайте на medium неделю. Не хватает качества, поднимите на одну ступеньку.

Когда не сработает. Если хотите, чтобы модель думала меньше, просить об этом словами бесполезно. Снижайте уровень усилия, это надёжнее. Для быстрого ответа на простой вопрос можно написать «Ответь сразу».

Правило 2. Проверьте уровни на своей задаче

Medium хорош в среднем. На ваших задачах лучший уровень может оказаться другим. Самый честный способ узнать: прогнать одну настоящую задачу на всех уровнях и положить результаты рядом.

- Отправьте Opus 5.5 промпт ниже, лучше в Claude Code, где он может запускать прогоны сам.
- Посмотрите результаты глазами и поставьте свой уровень по умолчанию через `/effort`.

```
Помоги подобрать мне уровень усилия. Посмотри, для чего я реально тобой пользуюсь, и возьми одну настоящую задачу, которую я делаю часто и результат которой можно оценить на глаз. Покажи задачу и в двух строках объясни, почему выбрал её. Потом прогони её на каждом уровне: low, medium, high, xhigh, max. Одни и те же инструкции и файлы, каждый раз с чистого листа. Запиши время и примерный расход каждого прогона. Сложи результаты рядом на одной странице, от низшего уровня к высшему, и внизу посоветуй самый низкий уровень, где качество перестаёт расти.
```

Когда не сработает. Пять прогонов на xhigh и max съедают заметную часть лимита. Берите задачу на несколько минут, а не на час.

Правило 3. Один AGENTS.md для всех агентов

Claude Code теперь читает AGENTS.md, это файл инструкций, который используют Codex, Cursor и другие агенты. Если в папке нет CLAUDE.md, Claude Code возьмёт правила из AGENTS.md.

- Обновите Claude Code до свежей версии.
- Если работаете в нескольких агентах, держите правила в одном AGENTS.md и не синхронизируйте два файла руками.
- Если CLAUDE.md уже есть и в нём своё, оставьте в нём одну строку `@AGENTS.md`: Claude Code подтянет общий файл. Так устроен мой рабочий проект.

Когда не сработает. Если в папке лежит и CLAUDE.md, Claude Code читает его, а не AGENTS.md. Проверьте, нет ли старого CLAUDE.md, который перебивает общий файл.

02 Кэш, лимиты и отказы

Правило 4. Меняйте усилие, не сбрасывая кэш

Кэш это короткая память о начале разговора, чтобы модель не перечитывала всё заново и не тратила токены. Если поменять уровень усилия посреди разговора, кэш сбрасывается, и вся переписка читается заново за ваш счёт. Claude Code предупреждает об этом, когда вы трогаете `/effort` посреди работы.

- В Claude Code для одного тяжёлого шага напишите в сообщении слово `ultrathink`. Модель подумает глубже только на этом шаге, уровень усилия сессии не меняется.
- Для своих ботов через API (помечено: API). Вместо смены общего уровня используйте смену усилия для одного сообщения (per-message effort change, бета), она сохраняет кэш. Пример: диалог идёт на low, а сложный вопрос пользователя обрабатывается на high.

Когда не сработает. Смена на одно сообщение пока в бете. Проверьте на своей связке, прежде чем включать для всех пользователей.

Правило 5. Бесплатный сброс лимитов

Сброс сразу возвращает лимит на полный. К выходу Opus 5.5 многим подписчикам дали один бесплатный.

- Откройте claude.ai в браузере или Claude Desktop: Settings, потом Usage. Найдите блок Resets.
- Не жмите сейчас. Держите на момент, когда лимит кончится посреди важной работы. Тогда нажмите «Reset for free» и подтвердите. Та же кнопка появляется в сообщении о том, что лимит кончился.
- Посмотрите дату окончания в том же блоке и поставьте напоминание за пару дней. У акционного сброса к Opus 5.5 по скриншотам пользователей срок 22 октября.

Когда не сработает. В Claude Code и в мобильном приложении кнопки нет, только браузер и Desktop. Сброс сгорает после даты окончания и пропадает, если понизить или отменить подписку. Обычное недельное обновление лимита он не сдвигает.

Правило 6. Не просите показать ход мыслей

Если в промпте или в инструкциях вашего бота написано «сначала распиши свои рассуждения, потом ответ», Opus 5.5 может отклонить запрос. В ответе придёт отказ с категорией `reasoning_extraction`, «извлечение рассуждений». На Opus 5 такого не было.

- Найдите в своих промптах, проектах и CLAUDE.md просьбы «покажи ход мыслей», «распиши рассуждения по шагам в ответе» и уберите их.
- Просите результат, который вам нужен на самом деле:

```
Объясни в трёх предложениях, почему ты выбрал этот вариант.
```

- Для своих ботов через API (помечено: API): краткую выжимку рассуждений можно получать отдельным блоком, параметр `display: "summarized"`.

Когда не сработает. Такой отказ не переотправляется автоматически на запасную модель, его нужно обработать самому.

03 Длинные разговоры и задачи

Правило 7. Скажите, что прошлые ответы закрыты

В длинных чатах Opus 5.5 иногда возвращается к своим старым ответам и передумывает их, даже на коротком уточняющем вопросе. Отсюда лишние секунды на каждом ответе.

- Допишите два предложения в конец инструкций проекта в Claude или в CLAUDE.md:

```
Если ты уже на что-то ответил, считай этот ответ закрытым. В следующих сообщениях думай о том, что я спрашиваю сейчас, и не возвращайся к прошлому ответу, если я сам о нём не спрошу или не укажу на ошибку.
```

- Если нужно только иногда, сделайте из этого маленький скилл в Claude Code и вызывайте по команде.

Когда не сработает. Не ставьте это туда, где нужна перепроверка: в длинную аналитику и в задачи агента, где поздний шаг может найти ошибку в раннем. С этой строкой модель реже сама замечает ошибку в прошлом ответе.

Правило 8. Чек-лист задачи в файле

На длинных задачах Opus 5.5 пишет отчёты по ходу. Иногда такой отчёт заканчивает его ход, и агент останавливается, хотя сделана только часть.

- Добавьте в задачу:

```
Веди чек-лист задачи в файле TASKS.md. Отмечай пункт, как только он готов, и дописывай новое, что найдёшь по дороге. Не заканчивай работу, пока в списке есть открытые пункты и ничто их не блокирует.
```

- Следите за работой по файлу, а не по переписке. Файл переживает сжатие контекста, а старая переписка нет.
- Если агент всё равно встал с открытыми пунктами, напишите: «В списке остались открытые пункты. Продолжай. Если что-то блокирует, скажи что».

Для тех, кто запускает Claude Code надолго без присмотра, в CLAUDE.md можно добавить правило остановок:

```
Если шаг не требует моего решения, продолжай работу. Короткий статус пиши в том же сообщении, где делаешь следующее действие. Останавливайся и спрашивай, только если без меня дальше нельзя, или перед необратимым: удалить данные, force-push, изменить что-то вне этой папки.
```

Когда не сработает. Правило «продолжай» значит меньше остановок, поэтому подтверждения на опасные команды в Claude Code оставьте включёнными. Для совместной работы в реальном времени, когда вы читаете каждый шаг, это правило лишнее.

Правило 9. Дайте бюджет времени

Opus 5.5 внимательно следит, сколько времени прошло. Когда он видит бюджет, то распределяет работу, чтобы уложиться, и обычно заканчивает раньше. В команде агентов он раздаёт помощникам больше задач одновременно.

- Для своих агентов через API (помечено: API): дописывайте в конец каждого сообщения модели строку со счётчиком, например `elapsed 340s / 1200s` (прошло 340 секунд из 1200).
- Ставьте бюджет немного больше времени, которое реально хотите потратить, и подстройте на нескольких своих задачах.

Когда не сработает. Бюджет это подсказка, а не стоп: на отметке модель не остановится, для жёсткой границы ставьте свой таймер. Под давлением времени модель может чуть меньше искать и перепроверять. В обычном чате счётчика у модели нет, там работает следующее правило.

Правило 10. «Время важно», если бюджета нет

Не знаете, сколько времени нужно, добавьте одну фразу. В тестах Anthropic она заставила команды агентов финишировать раньше.

```
Время важно: не трать время, без которого можно обойтись. Чем раньше будет правильный результат, тем лучше.
```

В оригинале:

```
Time matters here: do not spend time that can be avoided, and the earlier a correct result is obtained, the better.
```

Когда не сработает. Там, где важнее точность, чем скорость, например в проверке договора или расчётах, фразу не добавляйте. Когда время давит, модель может проверять меньше.

04 Дизайн и картинки

Правило 11. У Opus всё ещё есть любимый дизайн

Без указаний по дизайну Opus 5.5 скатывается в свой набор: кремовый фон, курсивные слова в заголовках, нумерация разделов «01 / 02 / 03», моноширинные подписи, кнопки-таблетки. Фраза «сделай не шаблонно» просто меняет один шаблон на другой.

- Перечислите конкретно, чего быть не должно:

```
Сделай личный сайт на HTML и CSS с текстом-заглушкой. Не используй кремовый или молочный фон, курсивные слова в заголовках, нумерацию разделов «01 / 02 / 03», моноширинные подписи и кнопки-таблетки.
```

- Посмотрите, что он выбрал взамен. Не нравится, допишите в список и попросите снова.
- Лучше всего своя дизайн-система: 3–4 цвета, 1–2 шрифта, скругления, пара примеров кнопок и карточек. Дайте её файлом в начале задачи. Вся графика моего канала собирается из одной такой системы.

Когда не сработает. Список запретов убирает шаблон, но не создаёт стиль. Без своей системы каждый новый список даёт новый шаблон.

Правило 12. Дайте ему ножницы и лупу

Opus 5.5 хорошо читает графики, схемы и скриншоты: по тестам Anthropic, даже на самом низком усилии он снимает значения с плотных графиков точнее, чем Opus 5 на самом высоком. На совсем мелком (чертёж, скан договора, таблица на сотню строк) ему помогает возможность приблизить.

- Отправляйте картинку в самом высоком разрешении, какое у вас есть. Не скриншот скриншота.
- В Claude Code дайте задаче инструменты: бесплатные библиотеки Python PIL и OpenCV. Если их нет, Claude Code поставит сам по вашей просьбе.
- Промпт:

```
Вот чертёж в полном разрешении. Если детали мелкие, вырежи и увеличь нужный фрагмент через PIL или OpenCV. Выпиши номера деталей и допуски. Отметь, что не смог разобрать, и где смотрел.
```

- Последнюю фразу добавляйте в любые задачи на поиск и анализ: «Отметь, что не смог подтвердить, и где смотрел». Такое «не нашёл» читать полезнее всего.

Когда не сработает. В обычном чате без Claude Code инструментов кадрирования нет. Тогда поможет только высокое разрешение и, для чертежей, более высокий уровень усилия. Для графиков повышение усилия почти ничего не даёт.

05 Sonnet 5.5 и первый шаг

Чем Sonnet 5.5 отличается

Sonnet 5.5 вышел 28 сентября 2026. По токенам он вдвое дешевле Opus 5.5: $2 против $4 за миллион на входе и $10 против $20 на выходе. На задачу выходит до 30% дешевле прошлого Sonnet.

	Sonnet 5.5	Opus 5.5
Цена за миллион токенов, вход / выход	$2 / $10	$4 / $20
Усилие по умолчанию в Claude Code	medium	medium
Усилие по умолчанию в API	high	medium
Для чего	типовые, хорошо описанные задачи, документы, слайды	самая сложная и длинная работа

Что переносится из этого гайда: правила про уровни усилия, кэш, чек-лист, дизайн и кадрирование.

Что по-другому:

- На low и medium Sonnet 5.5 чаще останавливается спросить «продолжать?» посреди длинной задачи. Длинное давайте ему на high.
- На low он может отчитаться о правке кода, не запустив проверку. Если в ответе нет вывода тестов или сборки, попросите прогнать их явно.
- На плотных графиках инструменты кадрирования помогают ему сильнее, чем повышение усилия.

Моё правило: типовое и хорошо описанное отдаю Sonnet 5.5, самое сложное и длинное Opus 5.5.

Что сделать прямо сейчас

Откройте Claude в браузере, Settings → Usage, и посмотрите, есть ли у вас бесплатный сброс лимита и до какого числа он действует. Это две минуты, и так он точно не сгорит.