

Гайд: 60 вопросов с технических собеседований для frontend-разработчиков

Введение

Этот гайд – результат 80+ проведённых собеседований и разбора реальных вопросов, которые задают frontend-разработчикам. **Я, Тихон Галактионов, собирал их лично, анализируя, что именно спрашивают в разных компаниях – от стартапов до крупных IT-корпораций.**

Почему этот гайд важен?

На собеседованиях **не ждут размышлений вслух**, а хотят услышать **конкретные ответы с ключевыми словами**. Если ты просто рассуждаешь, но не попадаешь в суть, интервьюер ставит галочку "не уверен" и двигается дальше. Поэтому подготовка – это не просто повторение теории, а **умение формулировать ответы так, чтобы тебя поняли и приняли в команду**.

Собрал для тебя **60 ключевых вопросов**, которые нужно знать, если хочешь уверенно пройти техническое интервью.

Вопросы по JavaScript и React

1. Что такое Virtual DOM и как он работает?
2. Как работает алгоритм сравнения (reconciliation) в React?
3. Что такое хуки в React и с какими работал?
4. Зачем нужен key в React?
5. Чем useEffect отличается от useLayoutEffect?
6. Как использовать рефы (refs) в React и для чего они нужны?
7. Чем отличается useMemo от useCallback?
8. Что такое Fragment в React и для чего он используется?
9. Что такое Higher-Order Components (HOC) и для чего они используются?
10. Специфичность CSS.
11. Какие есть варианты для хранения данных на фронтенде? В чем их различия (localStorage, sessionStorage, cookie)?
12. Что такое Web API?
13. Чем отличается использование Promise и async/await?
14. Какие есть способы привязать контекст (bind, call, apply)? В чем разница?
15. Что такое контекст выполнения? Что такое this?
16. Какие ты знаешь методы самого класса Promise?

17. Как работает замыкание?
18. Отличия стрелочной функции от function declaration.
19. Как устроен сборщик мусора в JS? Условие достижимости.
20. Что такое рекурсия и чем она опасна?
21. Что такое всплытие и погружение событий (event propagation)?
22. Что такое вспомогательные типы (Utility Types)? Какие знаете?
23. Что такое обобщения (Generics)?
24. Что такое ограничение обобщений? Как сузить тип?
25. Какие интересные случаи приходилось настраивать в tsconfig?
26. Какие интересные случаи приходилось настраивать в eslint?
27. Из чего состоит HTTP-запрос?
28. Что такое CORS и для чего он нужен?
29. Что такое cookie и для чего они используются? Зачем нужен флаг HttpOnly?
Какие ограничения существуют для cookie?
30. Объясните разницу между HTTP и HTTPS.
31. Какие есть способы клонировать объект в JavaScript?
32. Какой классовый компонент остался в функциональном React?
33. Чем отличается оператор `||` от `???`
34. Как оптимизировать приложение (метрики FCP, LCP, CLS и т.д.)?
35. Какие инструменты для анализа производительности используешь (Lighthouse, Performance, Network и т.д.)?
36. Что такое и для чего нужны WebRTC, WebSocket?
37. Что такое Shadow DOM?
38. Из-за чего может произойти перерендер компонента в React?
39. Что такое Big O и как считается сложность алгоритма?
40. В чем отличие Pick и Omit в TypeScript?
41. Какие хуки появились в React 19?
42. Какие знаешь паттерны проектирования?
43. Какие подходы проектирования знаешь (KISS, DRY, SOLID, YAGNI)?
44. Какие уязвимости знаешь и как их фиксировать?
45. Если у нас небольшой проект и не хочется тащить Redux, какие есть альтернативы?
46. React Query vs Redux: сходства, различия, взаимозаменяемость.
47. Что такое RTK Query?
48. Что такое Redux Saga?
49. Помимо React Hook Form, какие ещё есть варианты работы с формами?
50. Promise и async/await: какая между ними взаимосвязь?
51. Зачем нужны кастомные хуки в React?
52. Что такое SOLID?
53. Работал ли с валидацией данных на фронте?
54. Что такое прототипы в JavaScript и зачем они нужны?

Чтобы получить **бесплатную консультацию по трудоустройству или записаться ко мне на МОК**, напиши в личку 📩 <https://t.me/tikhoncare>

Вопросы по Git

1. Чем отличается `git rebase` от `git merge`?
 2. Чем отличается `git pull` от `git fetch`?
 3. Что такое `git blame`?
 4. Что делает `git cherry-pick`?
 5. Как организовать ветвление в проекте с несколькими окружениями (dev, staging, prod) и регулярными релизами?
 6. Что такое TBD (Trunk Based Development)?
-

Что важно запомнить

Если ты хочешь успешно проходить собеседования, **нужно не просто знать теорию**, а понимать, **как отвечать так, чтобы тебя хотели взять в команду**. Так вышло, что сейчас HR важно слышать ключевые слова, а не размышления.

- **Проанализируй свои пробелы** – посмотри, какие темы тебе нужно подтянуть и отрабатывай их на реальных задачах.
- **Отработай собеседования** – попробуй ответить на вопросы вслух, объясняя так, как будто учишь другого. Или запишись на МОК, будет круче и эффективнее.
- **Попади в сообщество** – это нужно, чтобы оставаться в контексте рынка, общаться с другими разработчиками и иметь возможность получать актуальную информацию с собеседований в разные компании.

Если тебе нужна помощь в подготовке к собеседованию, напиши мне! Я помогу разобрать твои ошибки, подскажу, на что делать акцент, как пройти собес уверенно.

Вот ссылка на мой ТГ, напиши туда, если хочешь **получить бесплатную консультацию или записаться на МОК** 👉 <https://t.me/tikhoncare>