

Типовые вопросы к собеседованию на должность Junior Frontend Developer

БЛОК 1: HTML

- За что отвечает тег `<doctype>`?
- Какие есть способы перевода страницы/сайта/приложения на другие языки?
- Для чего нужны `data`-атрибуты?
- Представьте, что вы решили создать веб-приложение с использованием HTML5. Какие блоки вы бы использовали и для чего?
- Объясните разницу между `cookie`, `sessionStorage` и `localStorage`.
- В чём разница использования тега `<script>`: `<script async>` и `<script defer>`?
- Почему считается хорошей практикой размещать тег/теги `<link>` между `<head></head>` и JS `<script>` до `</body>`? Знаете ли вы какие-либо исключения?
- Что такое прогрессивный рендеринг?
- Для чего нужен атрибут `src` у тега `<image>`? Объясните, что происходит, когда браузер “читает” значение данного атрибута.
- В чём суть использования шаблонизаторов для HTML?
- Какие бывают виды вёрстки?

БЛОК 2: CSS

- В чём разница между `reset css` и `normalize css`? Что бы вы использовали и почему?
- Что такое CSS-спрайты и как они используются?
- Какие есть особенности при отображении одинаковых свойств для разных браузеров?
- Как используются `media queries`?
- Какими плюсы и минусы существуют в использовании препроцессоров `css`?
- Какими препроцессорами пользовались вы?
- Как подключить шрифт в веб-приложение?
- Перечислите все виды селекторов `css`.
- Как образуется и функционирует вес селекторов?
- Объясните принцип `box-model`.
- Перечислите максимальное количество способов центрирования элементов.
- Каким образом происходит абсолютное позиционирование элемента?

- В чём преимущества CSS Flexbox и Grid?
- Сможете ли вы объяснить разницу между разработкой респонсив версии сайта и стратегией mobile-first?
- В чём разница между респонсив дизайном и адаптивным дизайном?
- Какие есть особенности работы с ретина-дисплеями?

БЛОК 3: JavaScript

- Что такое DOM?
- Что такое BOM и для чего она используется?
- Что такое Shadow Dom?
- Что такое цикл событий?
- Что такое замыкание?
- Как работает прототипное наследование и чем оно отличается от классической модели наследования?
- Как работает ключевое слово this? Приведите примеры использования.
- Что такое всплытие событий и как работает этот механизм?
- Опишите несколько способов обмена данными между клиентом и сервером. Расскажите, не вдаваясь в подробности, о том, как работают несколько сетевых протоколов (IP, TCP, HTTP/S/2, UDP, RTC, DNS, и так далее).
- Что такое REST и почему эта технология популярна?
- Ваш сайт тормозит. Расскажите о шагах по его диагностированию и исправлению. Опишите популярные подходы к оптимизации, и расскажите о том, когда их следует использовать.
- Какими фреймворками вы пользовались? Каковы их сильные и слабые стороны? Почему программисты пользуются фреймворками? Проблемы какого рода решают фреймворки?
- Объясните понятие делегирование событий.
- Объясните, почему следующее выражение не работает как IIFE: `function foo() { }();`. Что нужно сделать, чтобы она заработала как IIFE?
- Какие типы данных есть в JS?
- Какая принципиальная разница между null, undefined и необъявленной переменной?
- Приведите пример использования анонимных функций.
- Каким образом можно организовать свой код (паттерны модулей, классическое наследование)?
- В чём разница методов .call и .apply?
- Объясните разницу между `function Person() {}, var person = Person(),` и `var person = new Person().`
- Объясните, как работает `Function.prototype.bind.`
- Когда следует использовать `document.write()`?

- По какому принципу работает Ajax? Что можете сказать о его плюсах и минусах использования?
- Как работает JSONP?
- В чём принципиальная разница между атрибутом и свойством?
- Почему расширение встроенных объектов в JS не очень хорошая идея?
- В чём разница между `==` и `===` ?
- Почему в термине “тернарный оператор” используется слово “тернарный”?
- Почему считается хорошей практикой отделяться от глобального скоупа видимости?
- Для чего используется “use strict”? Какие у этого режима есть плюсы и минусы?
- Для чего нам может пригодиться событие `load`? Есть ли у него какие-либо недостатки? Есть ли альтернативы?
- Объясните, как работают промисы и/или их полифилы?
- Какие существуют способы борьбы с асинхронностью?
- Какие за и против существуют в использовании промисов вместо колбеков?
- В чём особенности стрелочных функций? Где их можно применить?
- Какие инструменты существуют для дебаггинга кода?
- Какие существуют методы перебора объектов?
- Объясните разницу между мутабельными и иммутабельными объектами.
- В каком случае можно изменить какое-либо значение, если оно объявлено с помощью `const`?
- Объясните разницу между синхронными и асинхронными функциями.
- Что такое event loop? Как он работает?
- Объясните разницу использования `foo` между `function foo() {}` и `var foo = function() {}`.
- В чём разница между объявлением переменных с помощью `let`, `var` и `const`?
- Какие отличия между ES6 классами и ES5 функциями-конструкторами?
- Что такое функция высшего порядка и где она используется?
- Приведите примеры деструктуризации массива или объекта.
- Можете ли вы привести пример каррирования и объяснить её преимущества?
- Какие существуют преимущества и различия между оператором `spread` и `rest`?