



Джуновский Сергей Петрович

Junior frontend-разработчик

Москва, 30 лет

Основные навыки:

HTML, CSS, JavaScript

Профессиональные навыки:



Верстка и профессиональная разработка веб-интерфейсов



Написание кода на JavaScript: DOM, встроенные API, обработка событий, модули и принцип DRY, работа с сетью, структуры данных, сборщики и другие нюансы языка.



Инструменты

- Редакторы кода
- Системы контроля версий: Git, GitHub
- Chrome DevTools.
- Препроцессоры: Less, Sass
- Инструменты автоматизации: Gulp, Webpack
- Графические редакторы: Figma, Photoshop
- Консоль

Дополнительные навыки:

- Английский язык
- Навыки общения
- Self-management (управление собой)



*** Трудоустройство:

1. Резюме

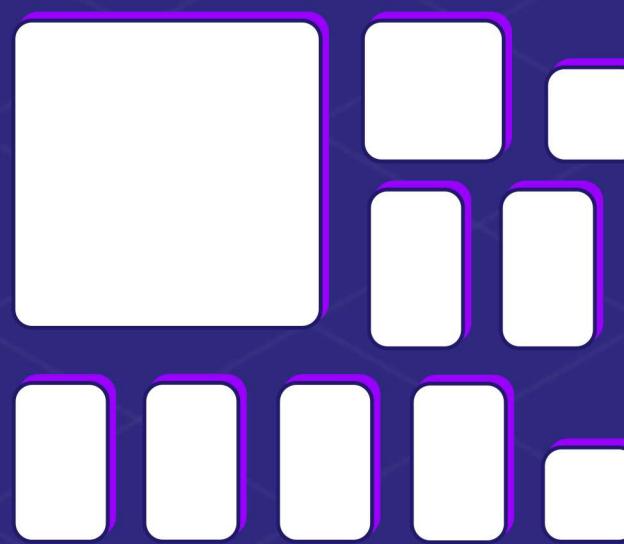
2. Зарплата

3. Собеседование

HTML CSS JavaScript Навыки на новн о н

HTML:

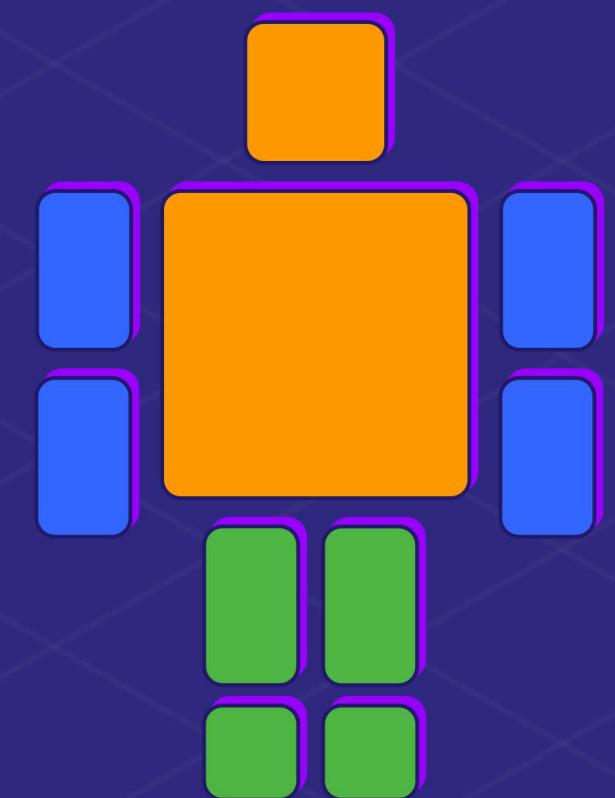
Разметка и оформление текста
Как размечать страницу по макету
Как ставить ссылки и вставлять изображения



HTML – набор кубиков, из которых можно собрать что-то классное

CSS:

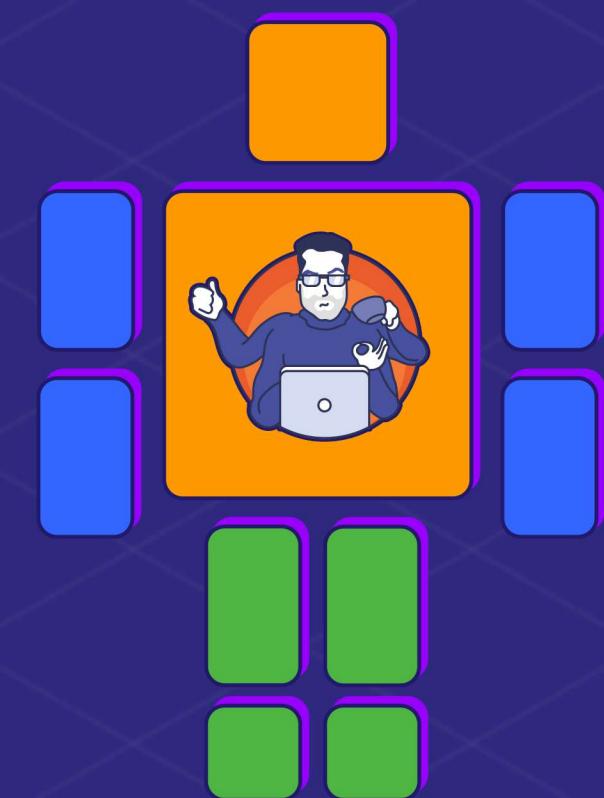
Правила
Селекторы
Свойства
Наследование и каскадирование
Построение сеток
Декоративные эффекты
Базовая стилизация страниц
Встроенные стили



Собрали кубики, раскрасили, поставили по местам – **получился робот**

JavaScript:

Типы данных
Операторы
Приведение типов
Переменные
Функции
Условия



Поселили внутри робота что-то, что им руководит – теперь он живой и можно как угодно им управлять. Так работает **JavaScript**

Что почитать?

- ⊖ Тренажёры для начинающих «Основы HTML и CSS»
- ⊖ Как написать и запустить HTML на компьютере
- ⊖ Как писать хороший HTML
- ⊖ Книжки по JavaScript
- ⊖ Ошибки при обучении программированию, и как их избежать
- ⊖ 8 шагов, чтобы получать 100 тысяч за код



ПРОФЕССИОНАЛЬНЫЕ НАВЫКИ

Написание кода на JavaScript:



Алексей Симоненко

JavaScript с нуля: можно ли научиться программировать без опыта

Вёрстка, профессиональная разработка веб-интерфейсов:

Вёрстка страницы по макету
Выразительная и доступная разметка
Построение адаптивных сеток
Оформление текста и декоративных элементов
Оптимизация кода
Автоматизация сборки проекта
Вёрстка по методологии БЭМ
Использование препроцессоров
Работа с адаптивной и ретиновой графикой

Концепции:

Работа с DOM-деревом
Встроенные API
Шаблонизация
Асинхронность
Валидация форм
Области видимости функций
Модульность
Объекты, иммутабельность и компонентный подход
Связывание данных

Аббревиатуры:

HTTP: XML, JSON, JSONP, XMLHttpRequest
REST API
DRY
ООП



Совет от наставников:

«Если программа выдаёт неожиданный результат, значит алгоритм описан некорректно.

В программировании вообще легко запутаться, особенно, если проект большой, главное не зацикливаться на сложной задаче.



Попробуйте отдохнуть, забыть о задаче, а потом с новыми силами попробовать написать код с нуля.

Если не получается и так — задайте вопрос знающему человеку.»

Быстрые упражнения для прокачки навыков



- ✓ Простая анимация на чистом CSS
- ✓ Интерактивная SVG-диаграмма
- ✓ Список задач с drag & drop
- ✓ Как написать Телеграм-бота на JavaScript

Инструменты (редакторы кода, графические редакторы, системы контроля версий)



Инструменты помогают программисту работать. В их числе не только редактор кода, но и **программы отладки, графические редакторы, системы контроля версий и средства автоматизации**. Каждый из инструментов решает свою задачу — и чем большим количеством владеет разработчик, тем выше его **профессиональный уровень**.

Редакторы кода

Инструменты разработчика – **DevTools**.

[Введение в Chrome DevTools. Панель Elements](#)

[Введение в Chrome DevTools. Console, Sources, Network](#)

Препроцессоры (Less, Sass)

Gulp или **WebPack** для автоматизации сборки

Git для контроля версий:

[Введение в Гитхаб и контроль версий](#)

[Регистрация на Гитхабе](#)

[Полезные команды для работы с Git](#)

[Работа с Git через консоль](#)

Figma или Photoshop для графики

Консоль



Совет от наставников:

По большому счёту **софт – это инструмент**, который может подходить или не подходить для конкретной задачи. Как молоток для шурупов или отвёртка для открывания замков. Поэтому **выбирайте сердцем**, используйте с умом, но не относитесь к выбору слишком серьёзно. В конце концов, это просто компьютерные программы.



Soft skills (нетехнические скиллы)

Self-management (управление собой)

- Управление эмоциями
- Управление собственным развитием
- Навыки планирования и управления приоритетами задач

Мышление

- Самостоятельность и инициативность
- Поиск и анализ новой для себя информации
- Умение адаптироваться и меняться
- Работа над ошибками (а они будут постоянно)

Навыки общения

- Работа в команде
- Умение слушать
- Умение аргументировать свою точку зрения
- Работа с обратной связью

Английский язык уровня Intermediate (B1) и выше

Что почитать?

О коммуникативности:

- Гэвин Кеннеди «Договориться можно обо всем! Как добиваться максимума в любых переговорах»
- Джим Кэмп «Сначала скажите „Нет“»

О тайм-менеджменте:

- Том ДеМарко «Deadline. Роман об управлении проектами»
- Дэвид Аллен «Как привести дела в порядок»
- Даниел Канеман «Думай медленно... решай быстро»
- Максим Дорофеев «Джедайские техники»



Совет от наставников



Знания, навыки и владение инструментами ничего не стоят без **умения общаться с людьми**. Это важный навык для решения большинства рабочих задач.



Времена хакеров, в одиночку сидящих в тёмном подвале, прошли – и теперь большими проектами занимаются большие команды. Главное – **погуглите, прежде чем задавать вопросы**.

И ещё немного полезностей:



- Как стать ценным сотрудником и вырасти в компании
- 10 soft skills, которые нужно прокачать ИТ-специалисту
- 6 ошибок при обучении программированию
- Какие софт-скиллы прокачивать начинающему разработчику

Поиск первой работы в IT – сложная задача. Её можно упростить, если внимательно собирать всё, что вы сделали во время учёбы. Никогда не знаешь, где поможет калькулятор на JavaScript, сделанный для тренировки.

Что пригодится при поиске работы:

- ✔ Все знания и навыки, перечисленные выше
- ✔ Активный профиль на GitHub
- ✔ Знакомые в IT-компаниях
- ✔ Правильное резюме без ошибок

В некоторых школах есть стажировки и каталоги вакансий, так что всегда есть шанс найти работу сразу после обучения.

Что почитать?

- ✔ [Как найти работу. 6 советов для джуна](#)
- ✔ [Как провалить собеседование](#)
- ✔ [Сколько получают веб-разработчики](#)
- ✔ [Практика, опыт и работа. Где их брать?](#)



Вакансии есть почти в любом городе и для специалистов с любым опытом. **Зарплаты фронтендеров (по данным hh.ru за 2020 год)** говорят сами за себя:

По России

51–119 тыс ₽,
средняя – 80 000 ₽

В Москве

100–180 тыс ₽,
средняя – 140 тыс ₽

В Санкт-Петербурге

80–150 тыс ₽,
средняя – 120 тыс ₽

