

# FRONT-END

НАЧАЛО

# Что такое фронт-энд

Front-end разработка - создание пользовательского интерфейса для веб-приложения, часть веб-разработки, которую видит и с которой взаимодействует пользователь непосредственно через браузер.

# Как работает веб-приложение



# Что такое WEB (веб)

web 1 - только чтение информации, редактировать контент может только разработчик сайта (пример википедия и тд)

web 2 - централизованный интернет где можно обмениваться информацией но контролируется разработчиками (контролируется корпорациями)

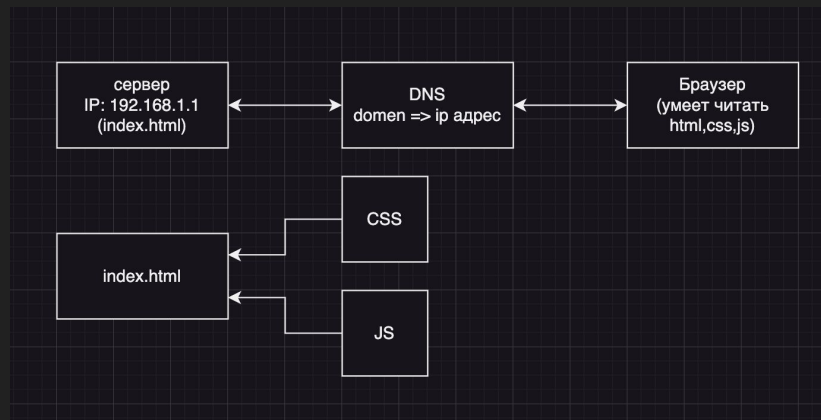
web 3 - децентрализованный интернет без особого контроля (пример NFT, блокчейн, криптовалюта и тд)

# Что такое браузер

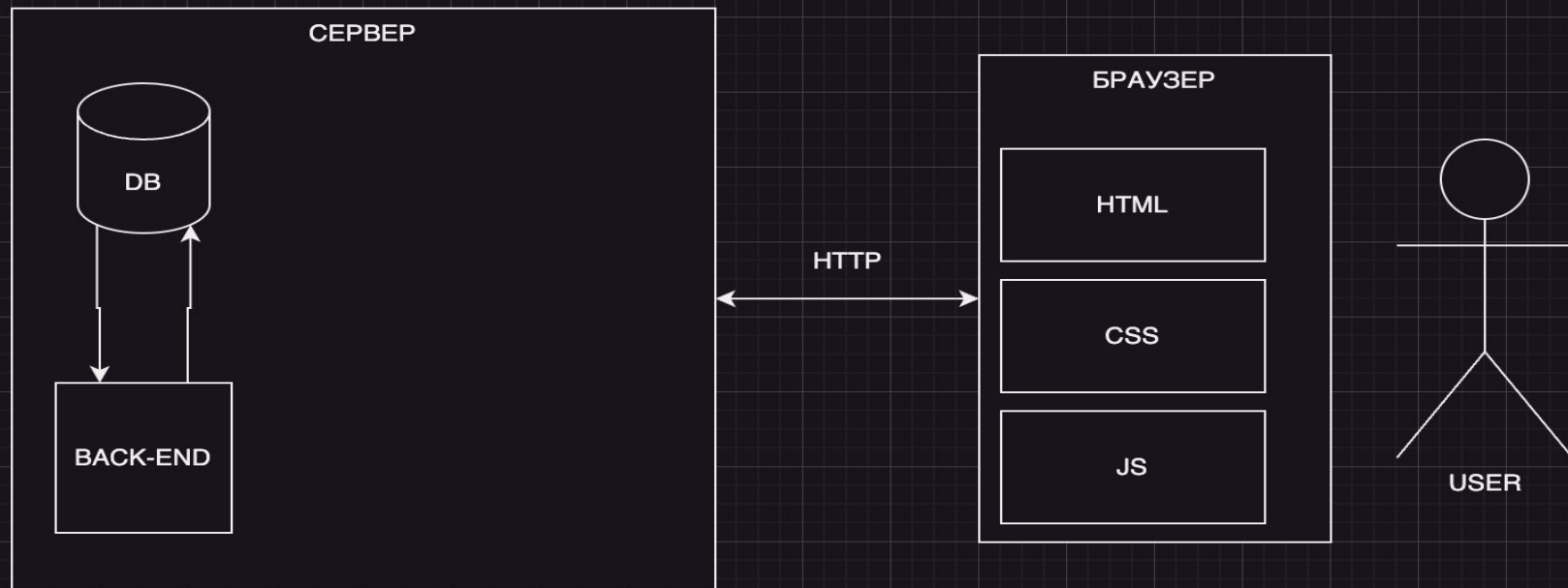
Браузер - это программное обеспечение, которое позволяет пользователям просматривать веб-страницы и взаимодействовать с интернет-ресурсами, отображая содержимое в виде текста, изображений и мультимедиа.

- Работа с HTML CSS и JS
- Среда выполнения JS кода
- Платформа для работы с ВЕБ-страницами/сайтами (расширения, закладки и тд)

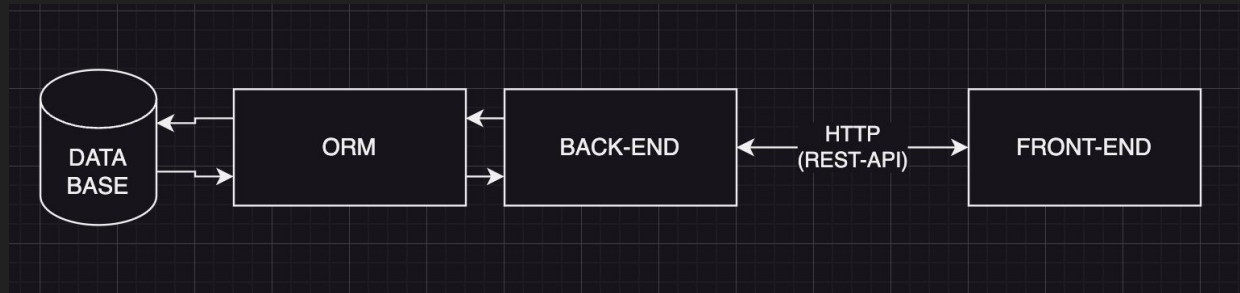
Как браузер отображает веб-страницу/веб-сайт



# Как работает браузер



# Пример онлайн магазин



DATABASE => BACK-END => HTTP request => FRONT-END => USER

USER => FRONT-END => HTTP request => BACK-END => DATABASE

В базе данных есть товар, бек мне дает данные о товаре, на фронте отображаю товар в виде карточки.

Тыкнул на карточку купить, отправил запрос на бек, бек записал в базу данных мой заказ

# Технологии front-end разработчика

- HTML
- CSS
- JavaScript
- TypeScript
- JSON - формат общения с back-end-ом
- библиотеки (lodash / ui libraries / axios)
- фреймворки (react / vue / angular / svelt и тд)
- модули для фреймворков (redux / vuex / RTK query и тд)
- webpack / vite / bun.js и тд - сборщики

# Что такое HTML (вкратце)

Язык разметки

HTML (от английского HyperText Markup Language) — это язык гипертекстовой разметки текста. Он нужен, чтобы размещать на веб-странице элементы: текст, картинки, таблицы и видео.

# Что такое CSS (вкратце)

## Язык стилей

CSS (Cascading Style Sheets «каскадные таблицы стилей») — формальный язык декорирования и описания внешнего вида документа (веб-страницы), написанного с использованием языка разметки (чаще всего HTML или XHTML). Также может применяться к любым XML-документам, например, к SVG или XUL.

# Что такое JavaScript (не путать с Java) (вкратце)

JavaScript (JS) — мультипарадигменный язык программирования. Поддерживает объектно-ориентированный, императивный и функциональный стили. Является реализацией спецификации ECMAScript.

# Инструменты front-end разработчика

- dev-tools
- git - контроль версии
- редактор кода (vs code / webstorm / stack-blitz)
- документация
- chat-gpt
- NPM и NODE - среда исполнения javascript кода и пакетный менеджер
- jira / youtrack / trello / bitrix и остальные kanban (контроль выполнения задач)

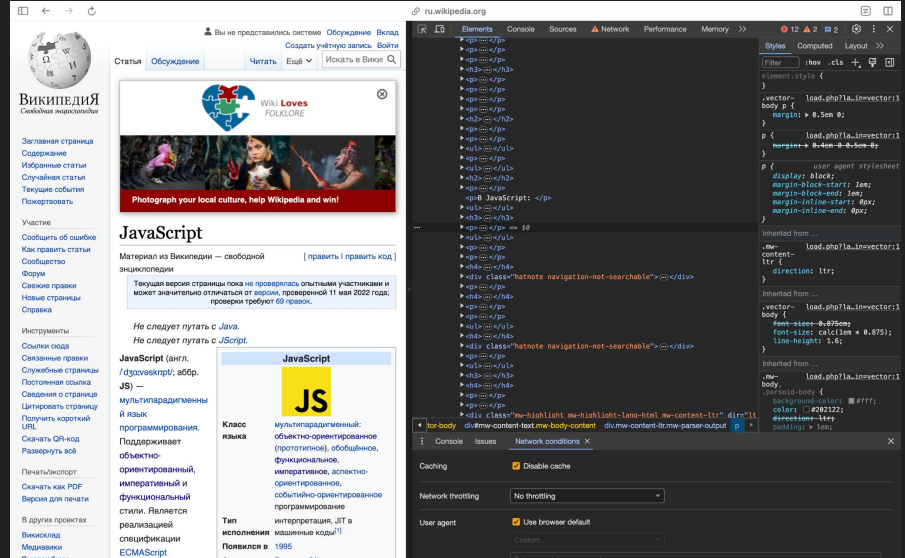
# “Задачи на фронте”

- ВЕРСТКА - базовая отрисовка контента
- АДАПТИВ - тонкая настройка CSS (для мобилок)
- ГИДРАЦИЯ - добавляем JS
- РАБОТА С СОСТОЯНИЕМ - контроль и синхронизация данных
- ОЦИФРОВКА БИЗНЕСА - перевод работы человека на код
- РАБОТА С API (СВЯЗЬ С BACK-END-ОМ) - связь с сервером и БД

ОТОБРАЖЕНИЕ ИНФОРМАЦИИ И РАБОТА ОТПРАВКА ДАННЫХ

# DEV-TOOLS

- ЭТО панель разработчика
- Доступ к HTML разметке и CSS стилям
- Можно “лазить” по сайту чуть с большими привилегиями
- Отображает все данные (статистика, код, сеть, нагруженность)
- Дает возможность дебага и тестирования (мануального)



# VS-CODE / WEB-STORM / STACK-BLITZ

Редактор кода - удобная среда для написания кода (блокнот на стероидах)

ПОЧЕМУ????:

- подсвечивает код
- авто-заполнения
- подсказки
- удобные плагины
- авто табуляция (автоматически ставит пробелы и табы)

# Visual Studio Code

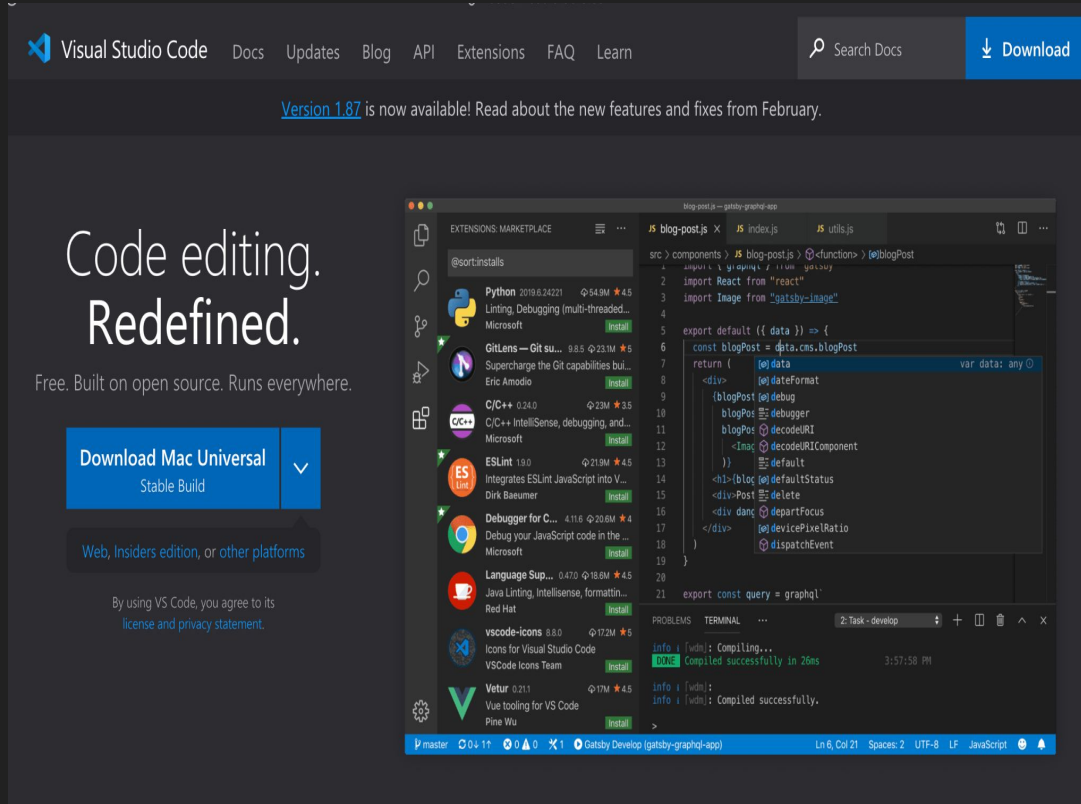
## Плюсы:

- БЕСПЛАТНО
- Удобно настроить под себя

## Минусы:

- Нужно настроить под себя
- Не очень удобно сразу “из под коробки”

[Тык - оф.сайт](#)



# Web Storm

[ТЫК - ССЫЛКА](#)

Плюсы:

- Сразу все работает
- Меньше заморочек
- Много фишек

Минусы:

- ПЛАТНО



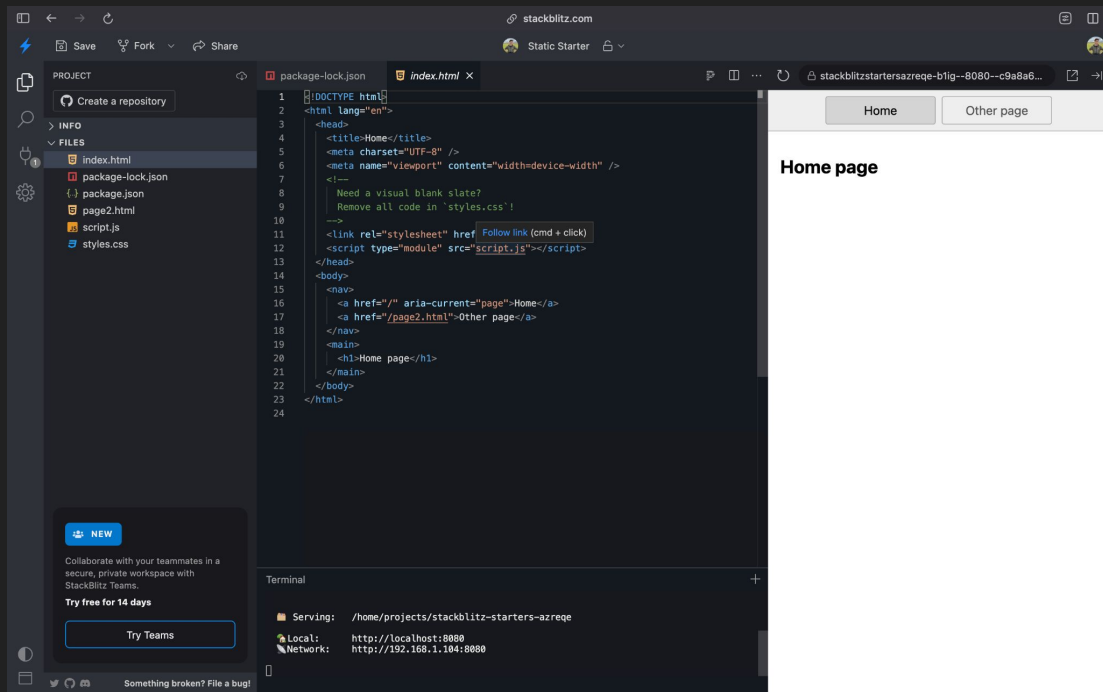
# Stack-blitz

## Плюсы

- Сохраняем все онлайн
- Сразу запускает демку
- Не нужно разворачивать локально

## Минусы

- Не очень удобно
- Иногда лагает
- требует подключения к интернету



# HTML - что такое, зачем, почему?

Язык разметки

Условный скелет сайта (по аналогии: css - кожа волосы, то есть внешность, а js - уже мышцы, то есть движущая сила, которая придает динамики скелету, без css и js сайт на голом html как скелет, не красивый, грубый и статичный)

Удобный формат описания контента и деления его в виде древа

HTML имеет древовидную схему у всего есть корень/родитель, внутренние составляющие - ветки являются дочерними элементами

# Структура HTML

`<HEAD>` - условная голова веб страницы - отвечает за информацию которую мы не видим конкретно в области окна страницы браузера, но можем видеть в других частях браузера (например информация в адресной строке, иконка страницы/сайта, информация о сайте/странице в поисковике и тд)

Так же в head можем подключать стили

`<BODY>` - тело веб страницы - отвечает за ту часть что мы видим на сайте

`<SCRIPT>` - тег для подключения js (пока не важно)

# Базовая структура HTML

`<!DOCTYPE html>` - тип файла

`<html lang="en">` - базовый язык

`<head>` - то что мы не видим

`<meta charset="UTF-8" />` - кодировка символов (пока не паримся)

`<meta name="viewport" content="width=device-width, initial-scale=1.0" />` - (мета данные для отображения контента)

`<title>Document</title>` - титульный заголовок страницы (у вкладки браузера)

`</head>`

`<body></body>` - то что мы видим, тут должен быть контент

`</html>`

# Теги для задания

заголовок тег <h1>, <h2> ... <h6> заголовки от 1 до 6 уровня чем ближе к 1 тем важнее заголовок (как в word)

текст (параграф) <p>{{какой-то текст}}</p>

<img> - тег для изображения

<title> - должен быть внутри <head> - отвечает за титульное название страницы

# Задание на дом

Сверстать страницу о себе

- создаем index.html и заполняем его
- используем теги `<h1>...<h6>` `<p>` `<img>`
- МОЖНО использовать интернет для подсказок

## Iskander



## My work

### Google

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Eius, in similique? Expedita voluptatum sint eos, iusto velit ducimus saepe laborum quo, nisi ea distinctio similique aliquid aperiam atque exercitationem dolor.

### Yandex

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Esse quo repudiandae autem explicabo ullam consectetur, maxime sequi, praesentium labore velit eos id delectus vitae provident fuga quas! Animi, nisi corporis!

## About me

Lorem ipsum, dolor sit amet consectetur adipisicing elit. Explicabo, molestiae fuga eveniet illum esse quaerat cumque sapiente excepturi consequuntur possimus consectetur, alias maxime exercitationem laboriosam quas nobis perspiciatis repudiandae perferendis!