



Преподаватель:

Коляда

Никита Владимирович

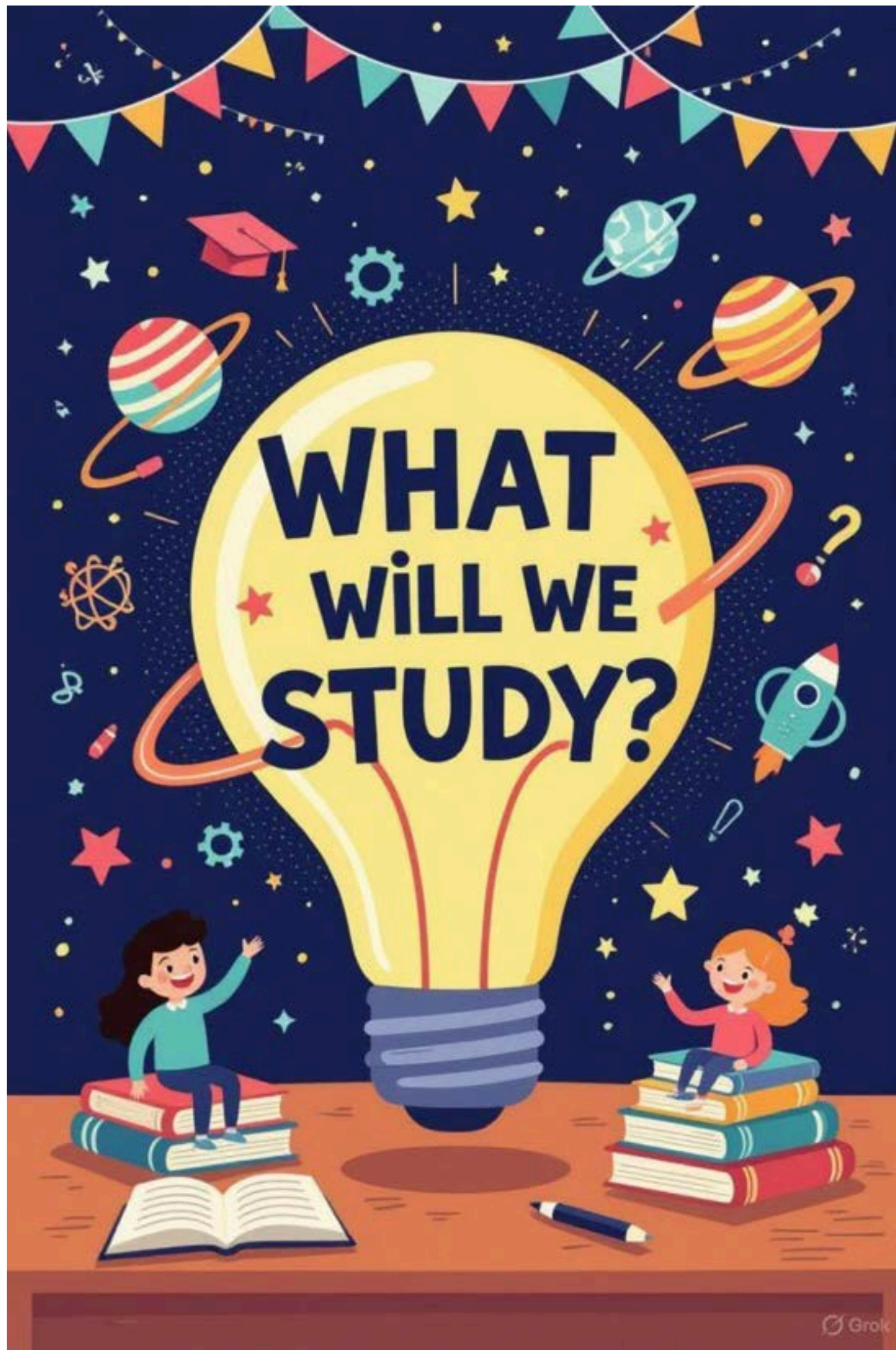
Обратная связь:

- сообщения на inStudy

# ВВЕДЕНИЕ В РАЗРАБОТКУ

ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА  
ИНТЕРФЕЙСОВ ПОЛЬЗОВАТЕЛЯ

2025 г.



## ЧТО ИЗУЧИМ СЕГОДНЯ?

1. Как работает Интернет и веб?
2. Клиент-серверная архитектура
3. Понятия frontend и backend
4. Браузеры и их инструменты разработчика
5. В чём писать код и что такое среда разработки
6. Установка среды разработки WebStorm и VS Code
7. Интерфейс программы WebStorm



1.Что такое веб-сайт?



## ВЕБ-САЙТ

2.Что такое веб-страница?



page1.html  
page2.html  
page3.html

3.Как изменить  
отображение (стиль) страницы?



style.css

5.Как открыть веб-сайт в браузере?



`https://www.google.com/`

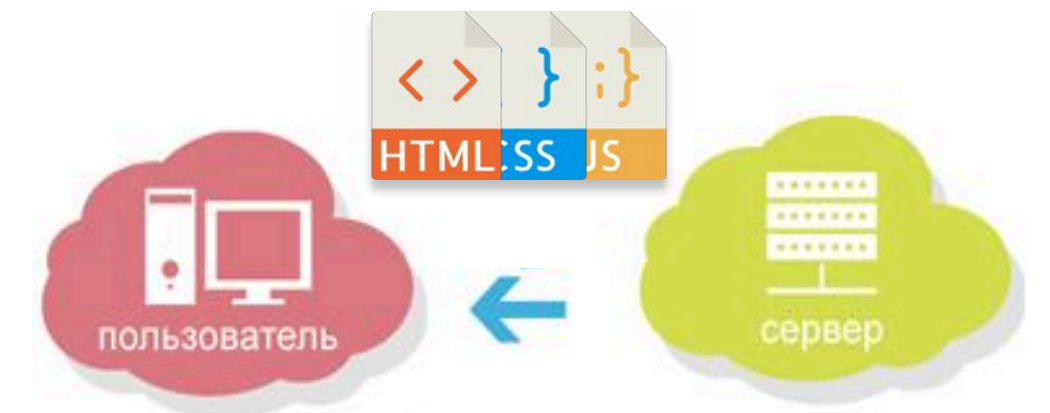
`file:///index.html`

4.Как обрабатываются действия/события?

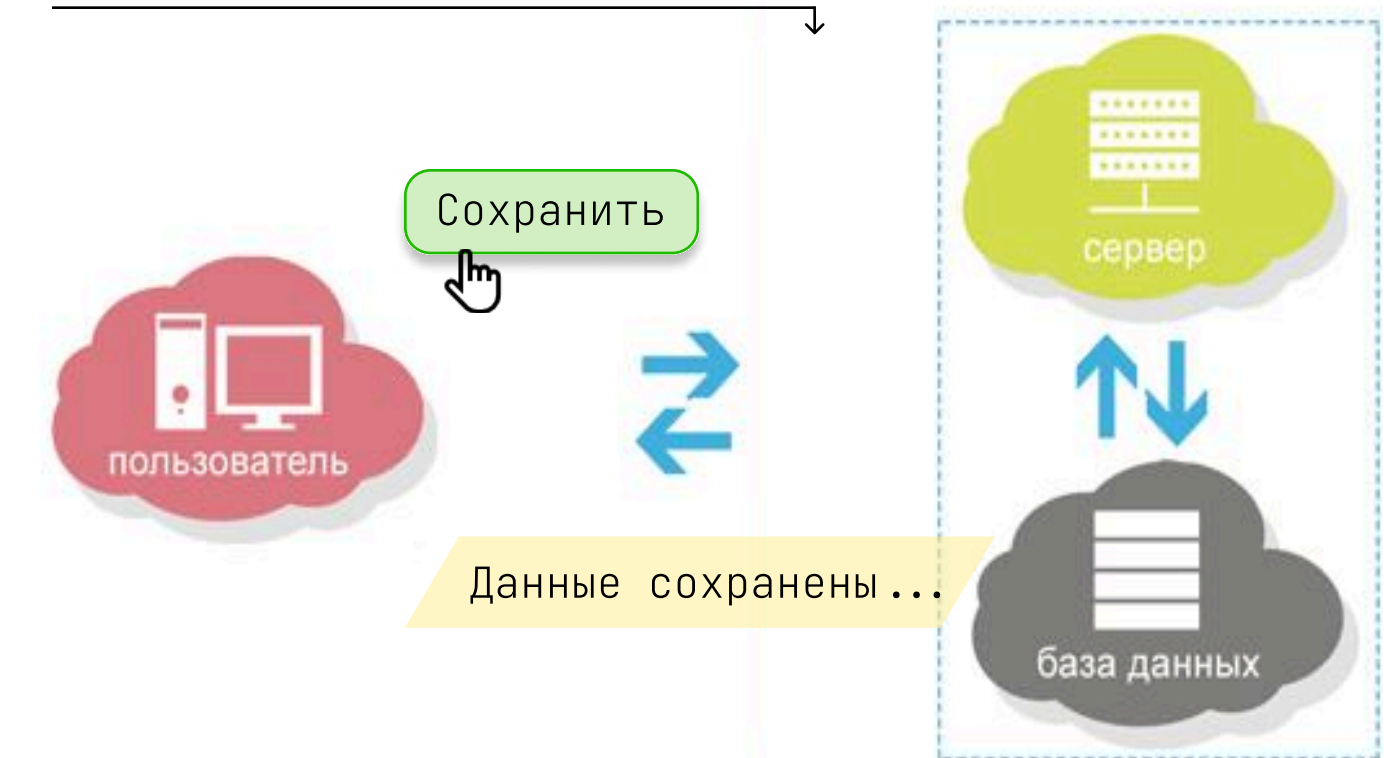


scripts.js

Статический

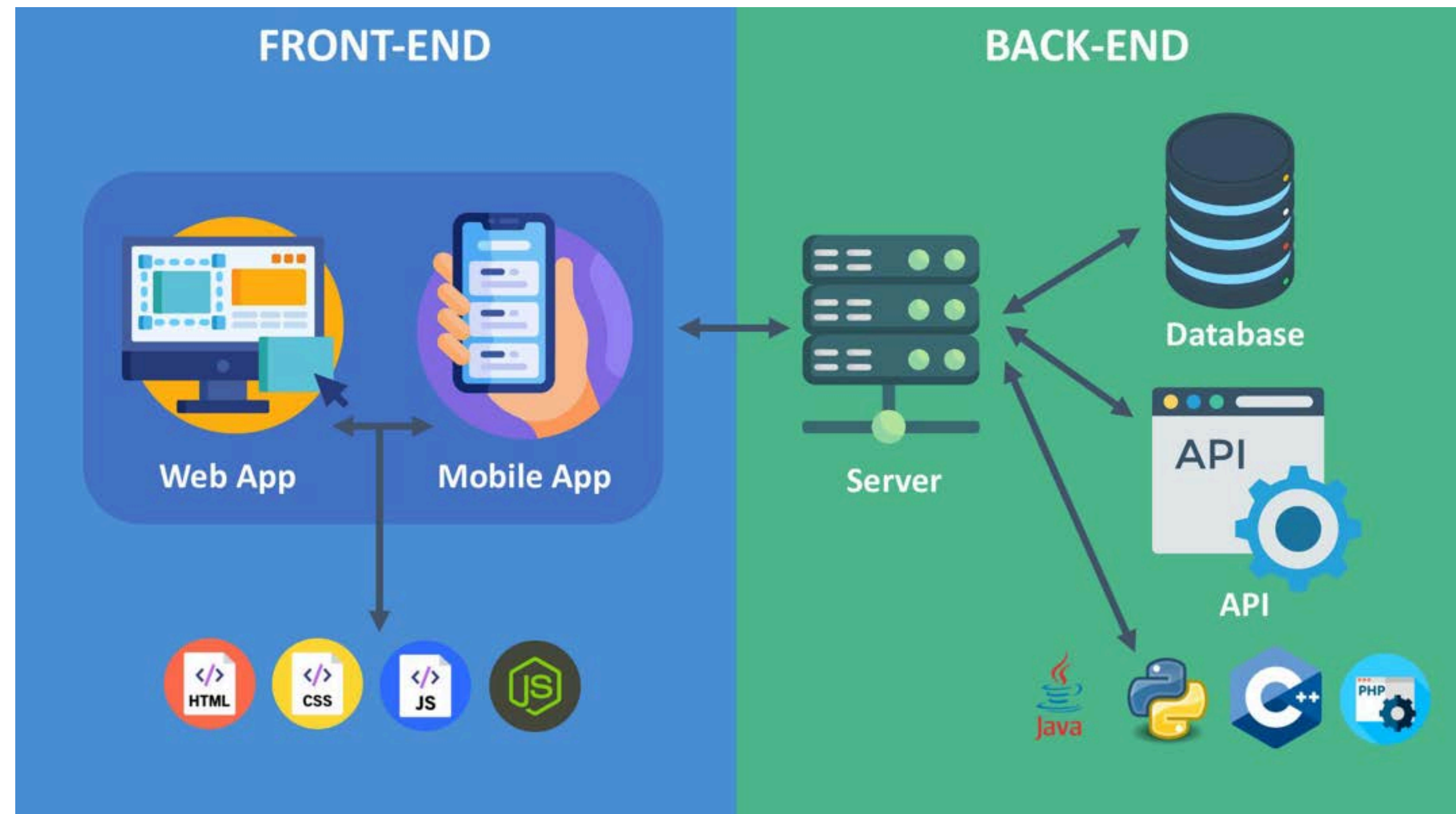


Динамический



# **FRONTEND & BACKEND** *КТО ОНИ ТАКИЕ?*

Клиентская и серверная часть веб-приложения



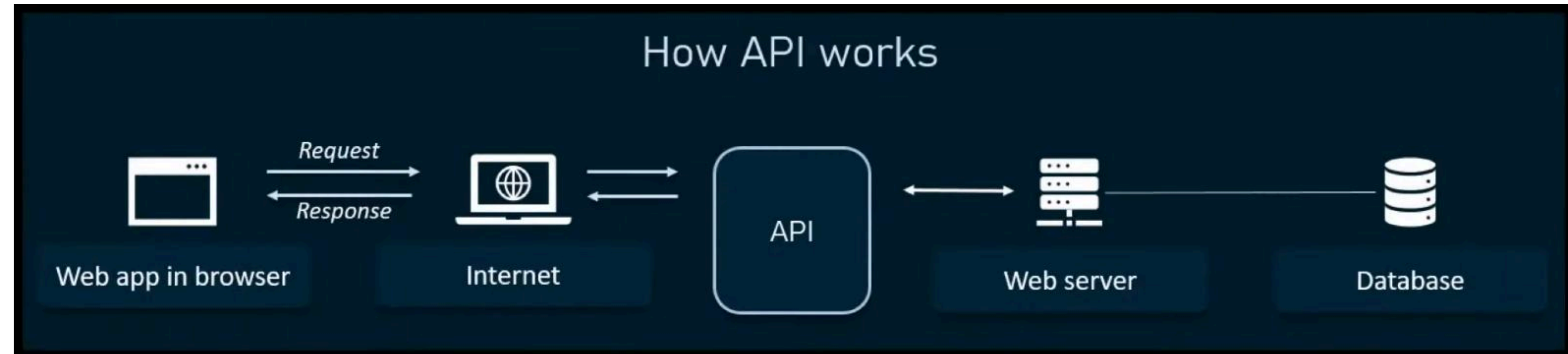
**Frontend** — это клиентская часть, с которой взаимодействует пользователь. Всё, что вы видите в браузере: кнопки, изображения, текст, анимации, формы — относится к фронтенду. Его задача — обеспечить удобный, красивый и понятный интерфейс.

**Backend** — это серверная часть приложения, скрытая от пользователя. Она отвечает за бизнес-логику, обработку данных, безопасность, взаимодействие с базами данных и внешними сервисами.

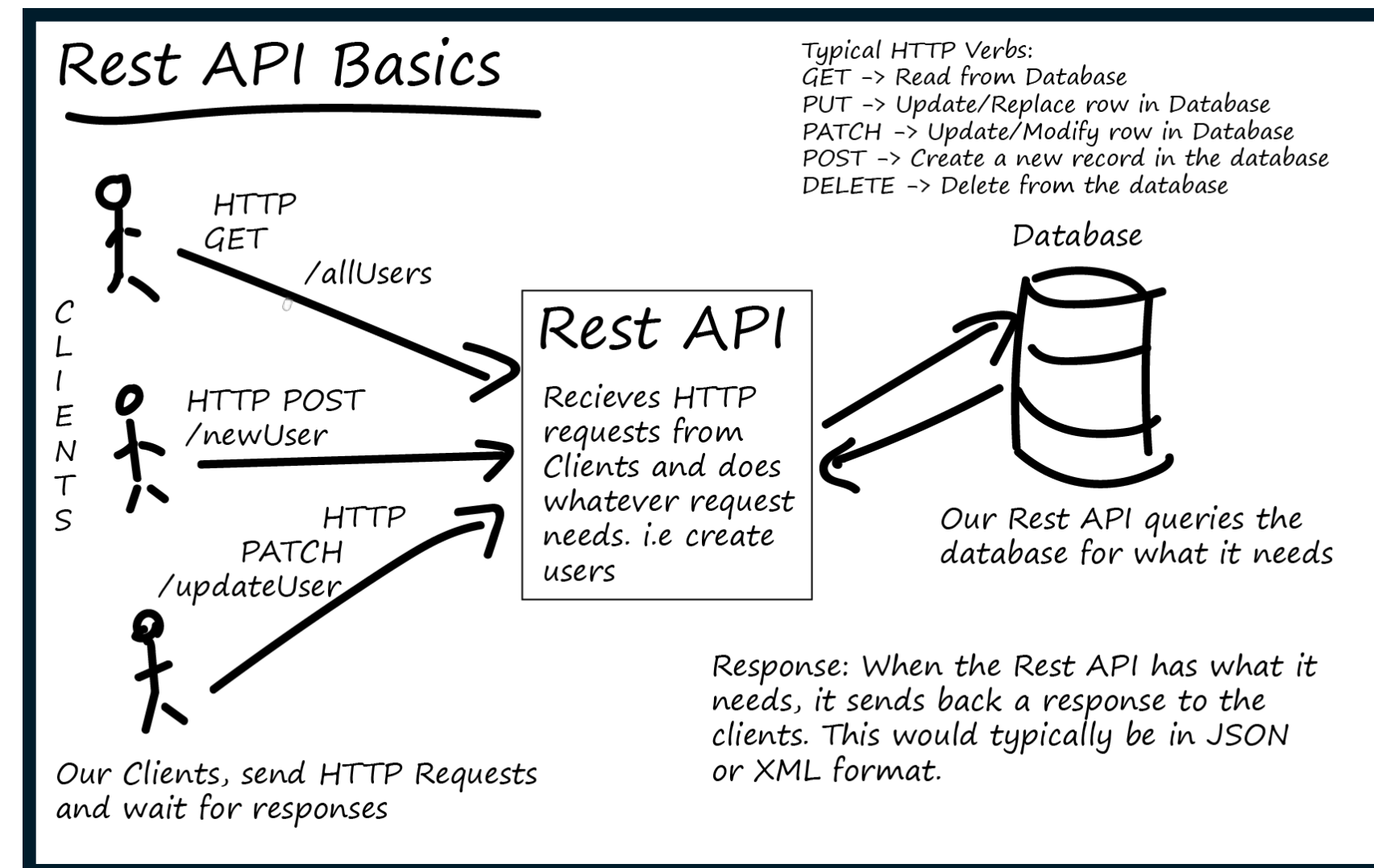


# ЧТО ТАКОЕ **API** И С КЕМ ОН ДРУЖИТ?

Почему API называют интерфейсом?



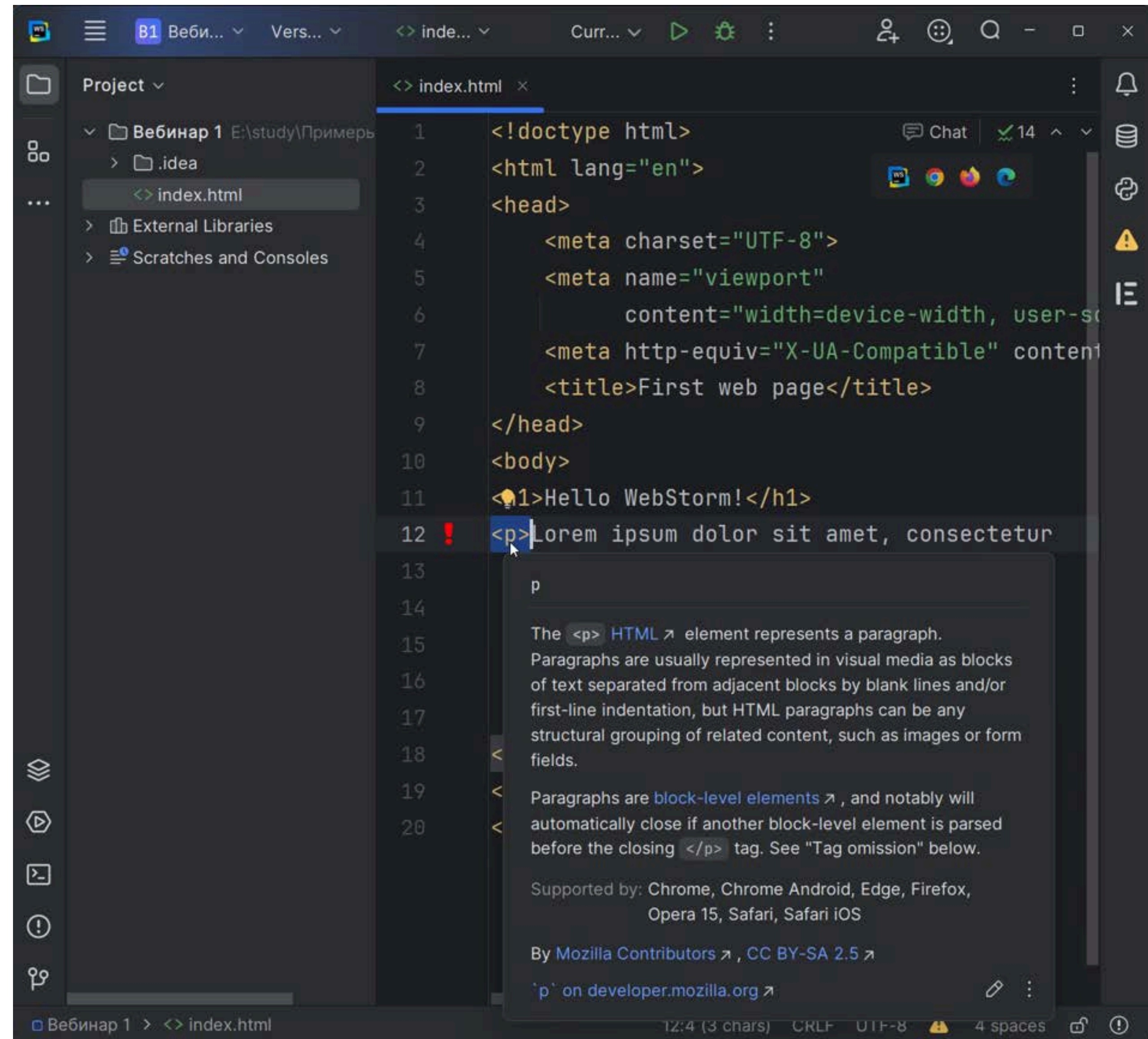
**API** (Application Programming Interface) – интерфейс, который позволяет программам обмениваться данными и функциями между собой.



**REST** (Representational State Transfer) – стиль архитектуры API, при котором взаимодействие происходит через стандартные HTTP-запросы (GET, POST и др.), обычно в формате JSON. REST API часто используется в веб-разработке для связи клиента и сервера.

# ЧТО ТАКОЕ **WEBSTORM** И КАК ИМ ПОЛЬЗОВАТЬСЯ?

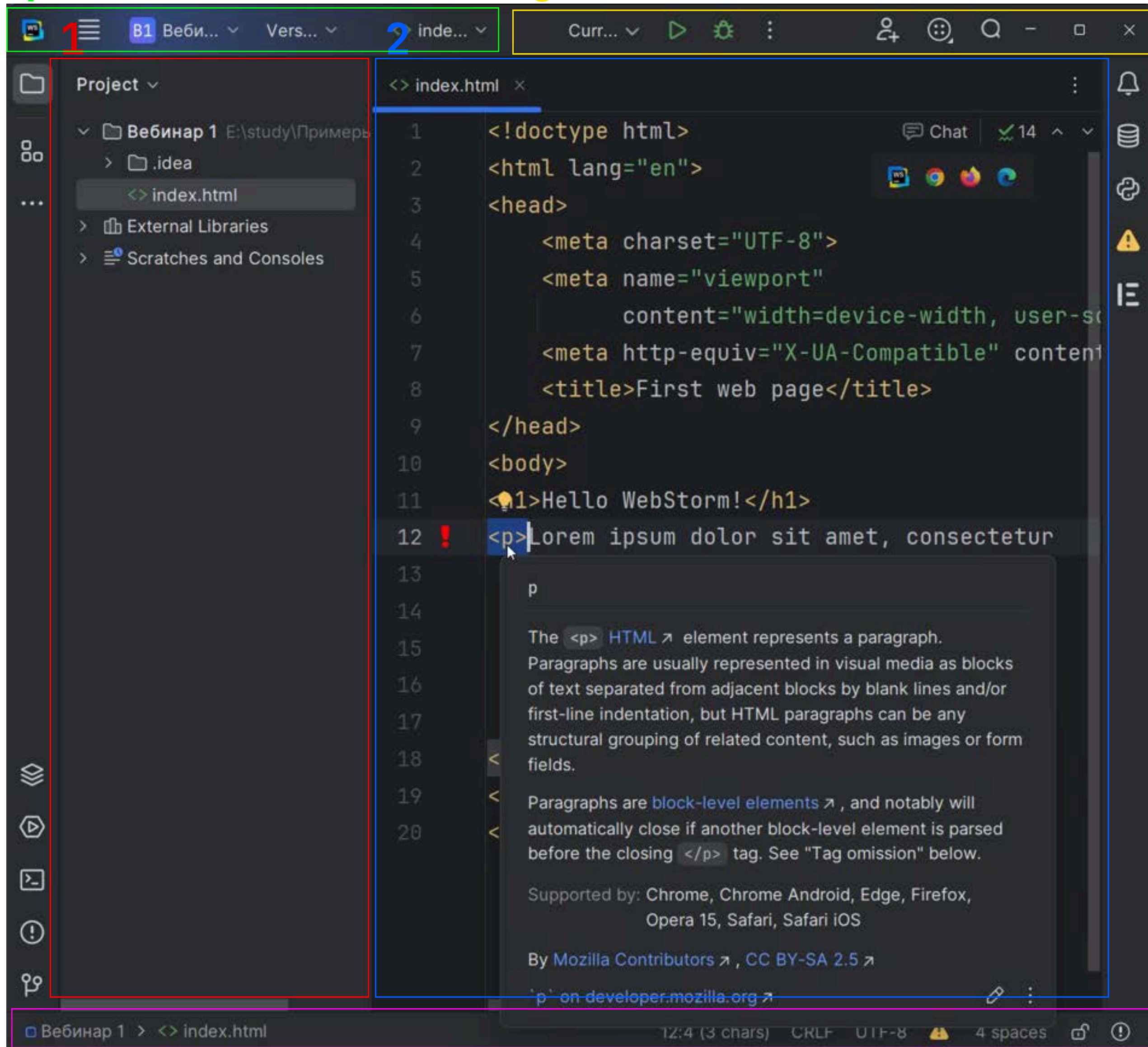
Интерфейс программы Webstorm





4

3



5

## Основные элементы интерфейса WebStorm

### Главное окно

#### 1. Панель проектов (Project View):

- Расположена слева.
- Отображает структуру проекта (папки, файлы).
- Позволяет быстро находить и открывать файлы.

#### 2. Рабочая область (Editor):

- Центральная часть интерфейса.
- Место для написания и редактирования кода.
- Поддержка подсветки синтаксиса, автодополнения и рефакторинга.

#### 3. Панель инструментов (Toolbar):

- Расположена сверху.
- Содержит кнопки для запуска, отладки, управления версиями (Git) и других операций.

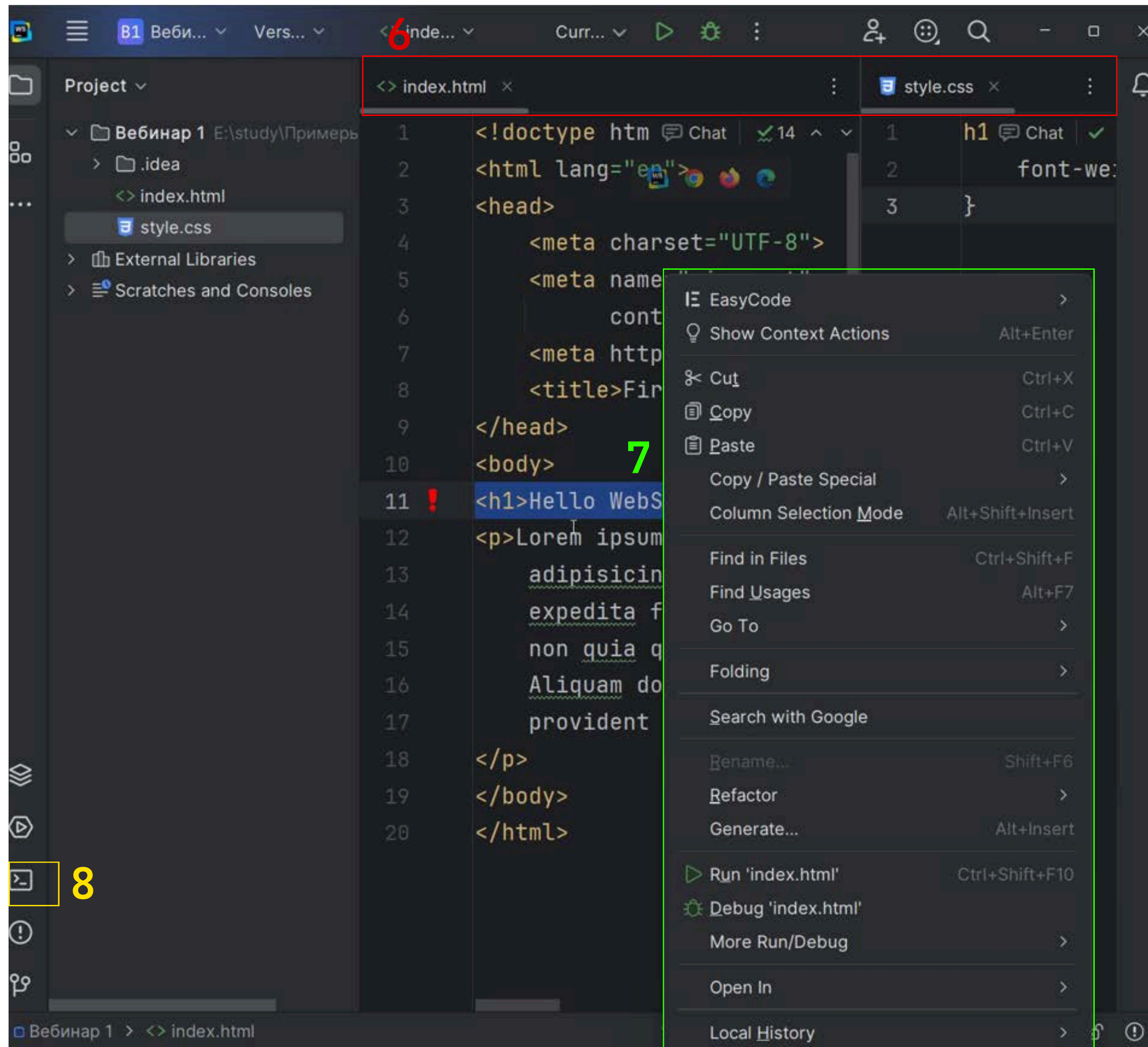
#### 4. Панель навигации (Navigation Bar):

- Находится над рабочей областью.
- Показывает путь к текущему файлу и позволяет быстро перемещаться по структуре проекта.

#### 5. Нижняя панель (Status Bar):

- Отображает информацию о файле (кодировка, формат строк).
- Доступ к терминалу, задачам, проблемам кода.





## Основные элементы интерфейса WebStorm

### Вкладки и окна

#### 6. Вкладки файлов:

- Открытые файлы отображаются в виде вкладок над рабочей областью.
- Можно закреплять часто используемые файлы.
- Разделение окон:
  - Возможность разделить рабочую область для одновременной работы с несколькими файлами.
  - Пример: просмотр HTML и CSS одновременно.

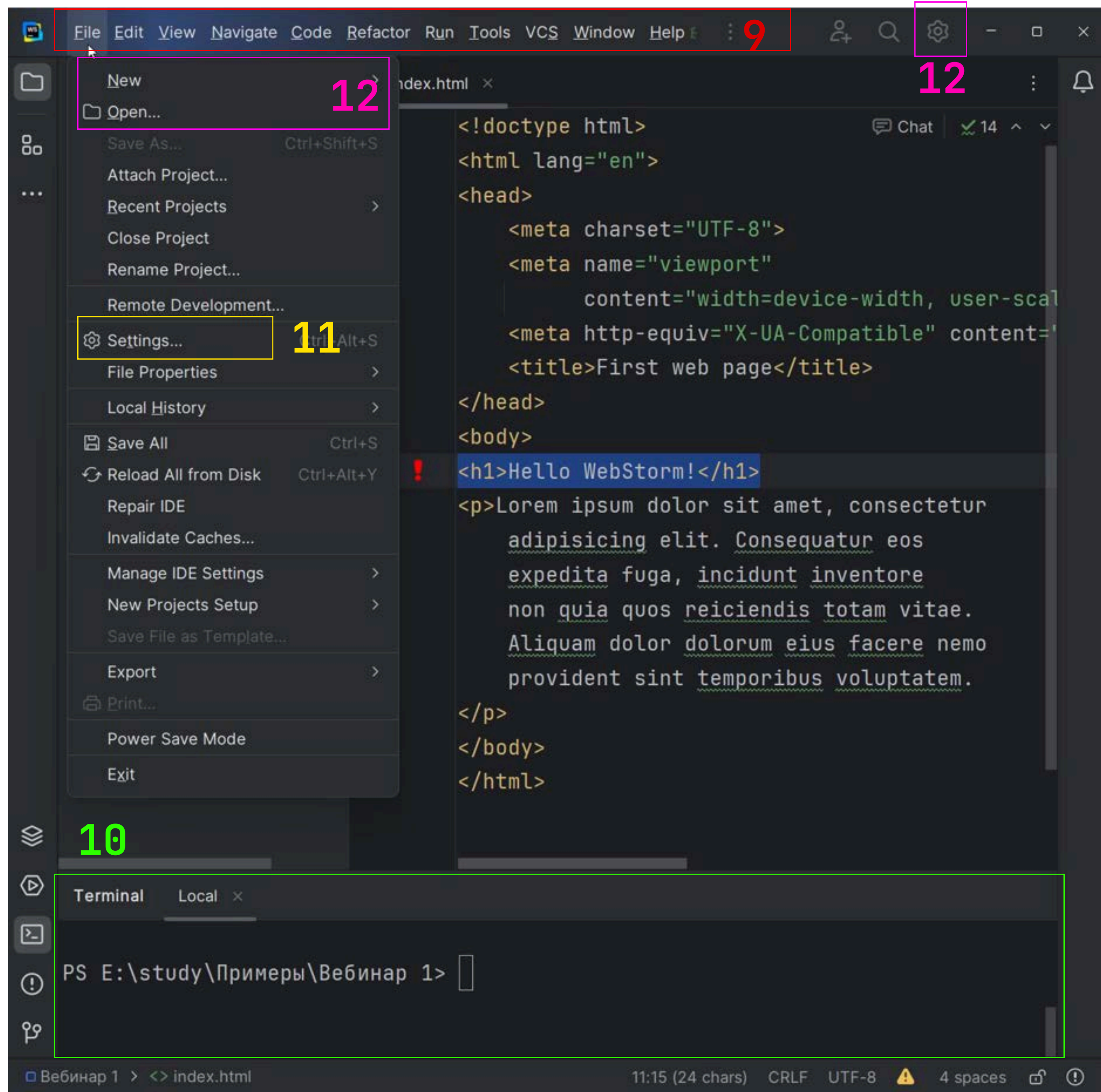
#### 7. Контекстные меню и горячие клавиши

- Контекстное меню (ПКМ на файле или в редакторе) предоставляет быстрый доступ к рефакторингу, поиску, настройкам.
- Горячие клавиши:
  - Ctrl + S – сохранить файл.
  - Ctrl + / – закомментировать строку.
  - Alt + Enter – быстрые исправления (Quick Fix).
  - Ctrl + B – переход к определению функции или переменной.

#### 8. Терминал (консоль)

- а. терминал открывается с помощью соответствующей кнопки в левой части интерфейса программы.



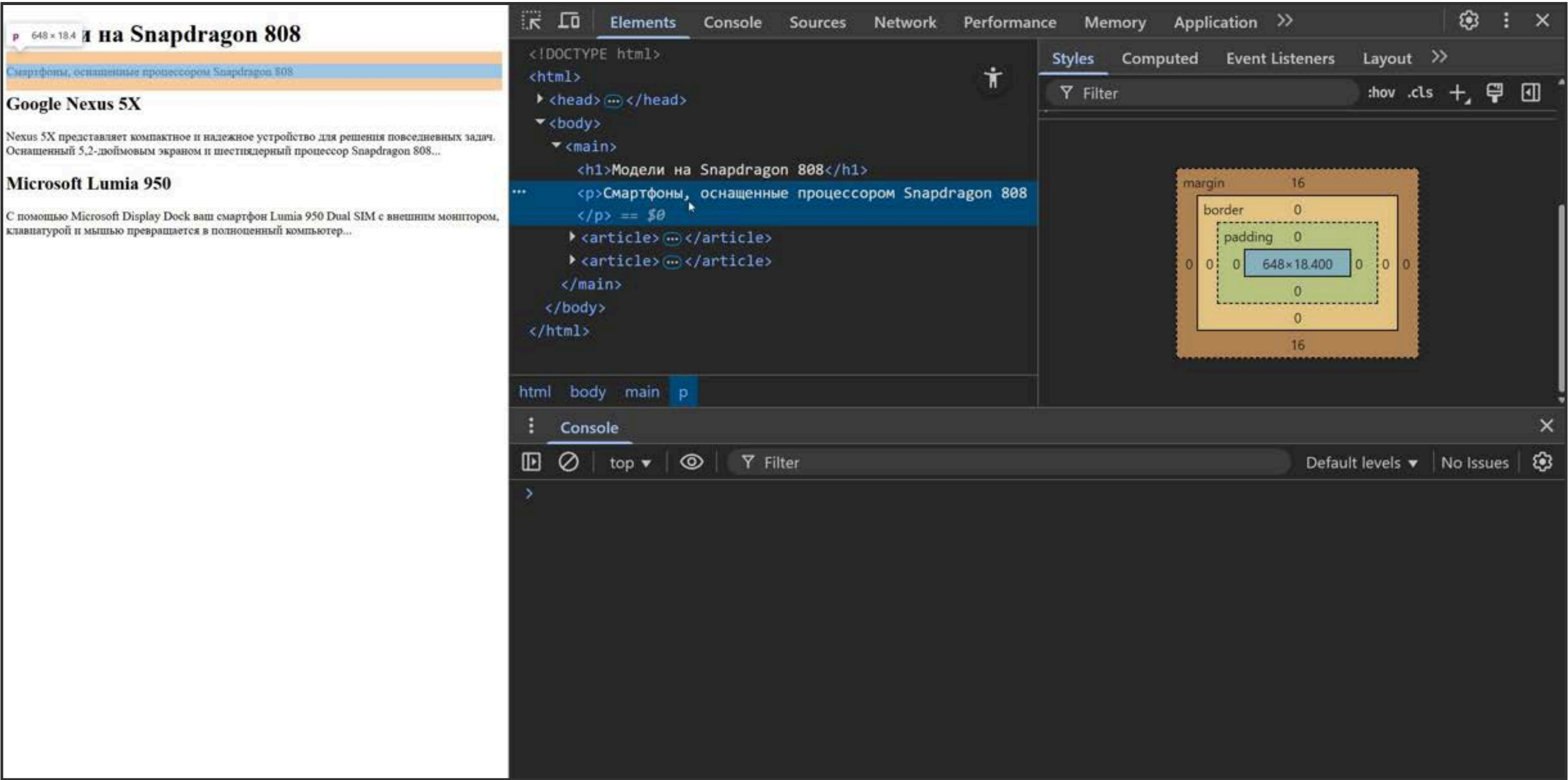


## Основные элементы интерфейса WebStorm

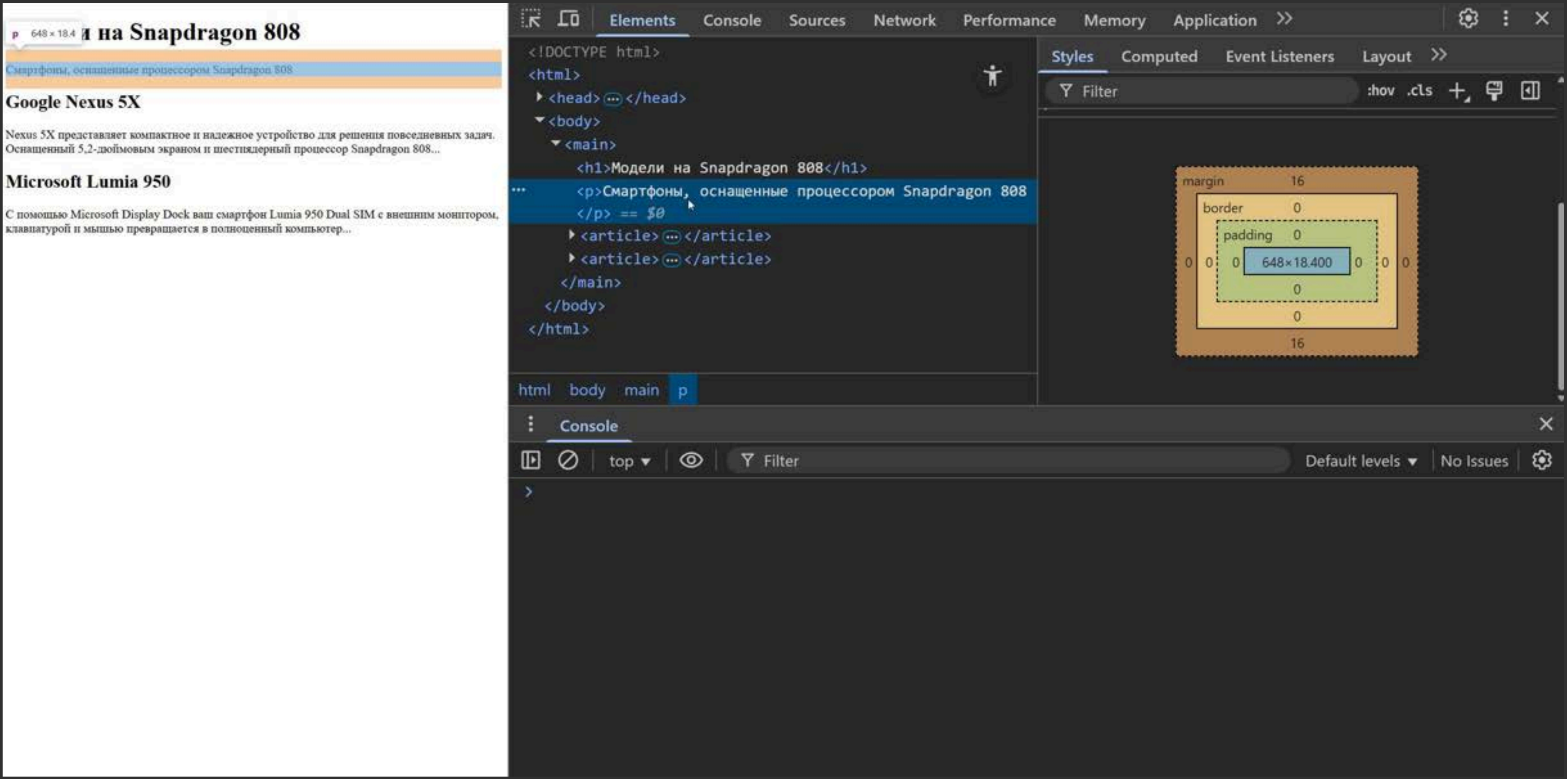
- 9. Панель настроек и инструментов
- 10. Терминал (консоль)
- 11. Настройки программы
- 12. Открыть или создать новый файл/проект

# ИНСТРУМЕНТЫ ДЛЯ ОТЛАДКИ КОДА В БРАУЗЕРЕ

Инструменты веб-разработчика в браузере

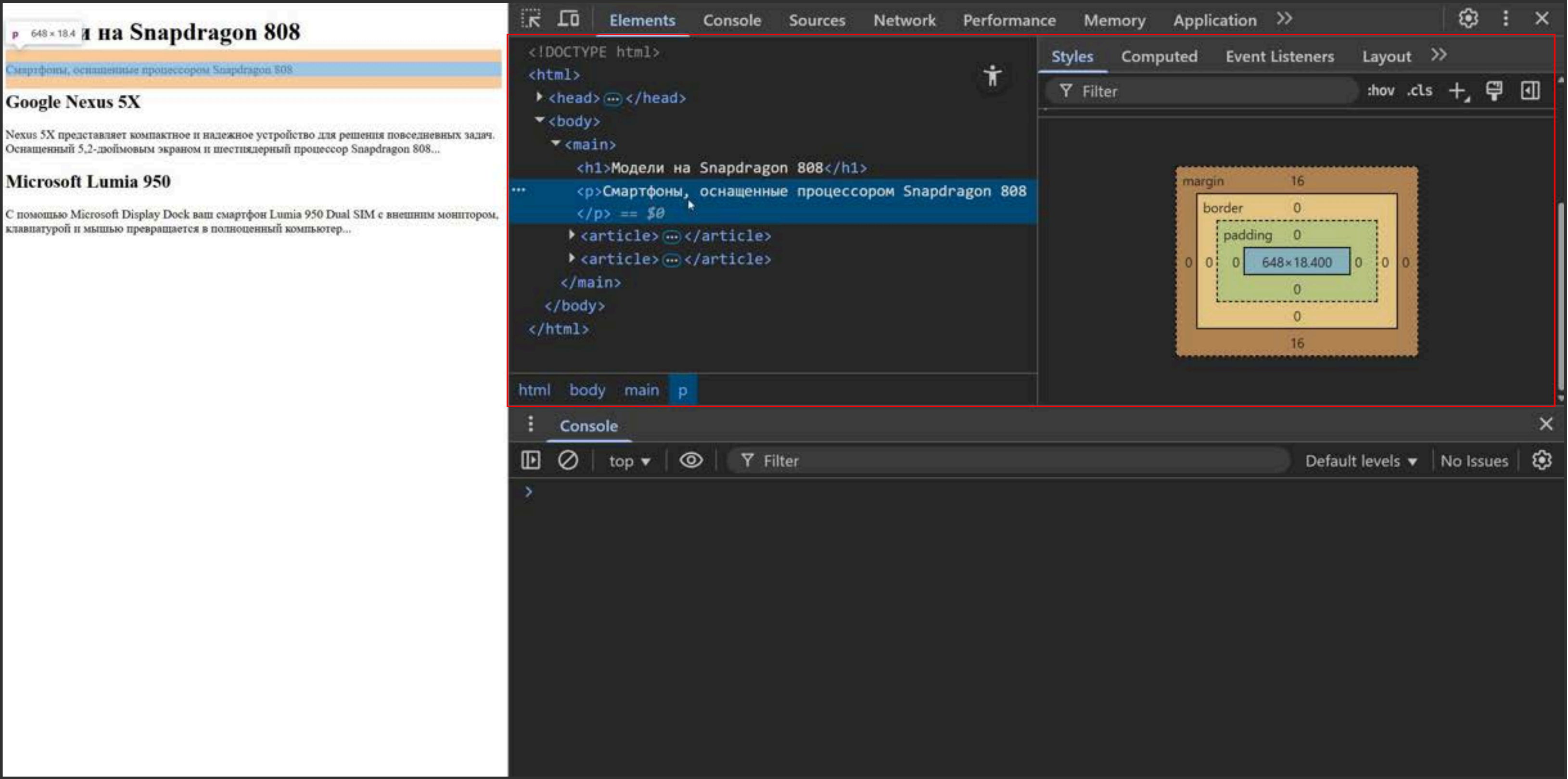






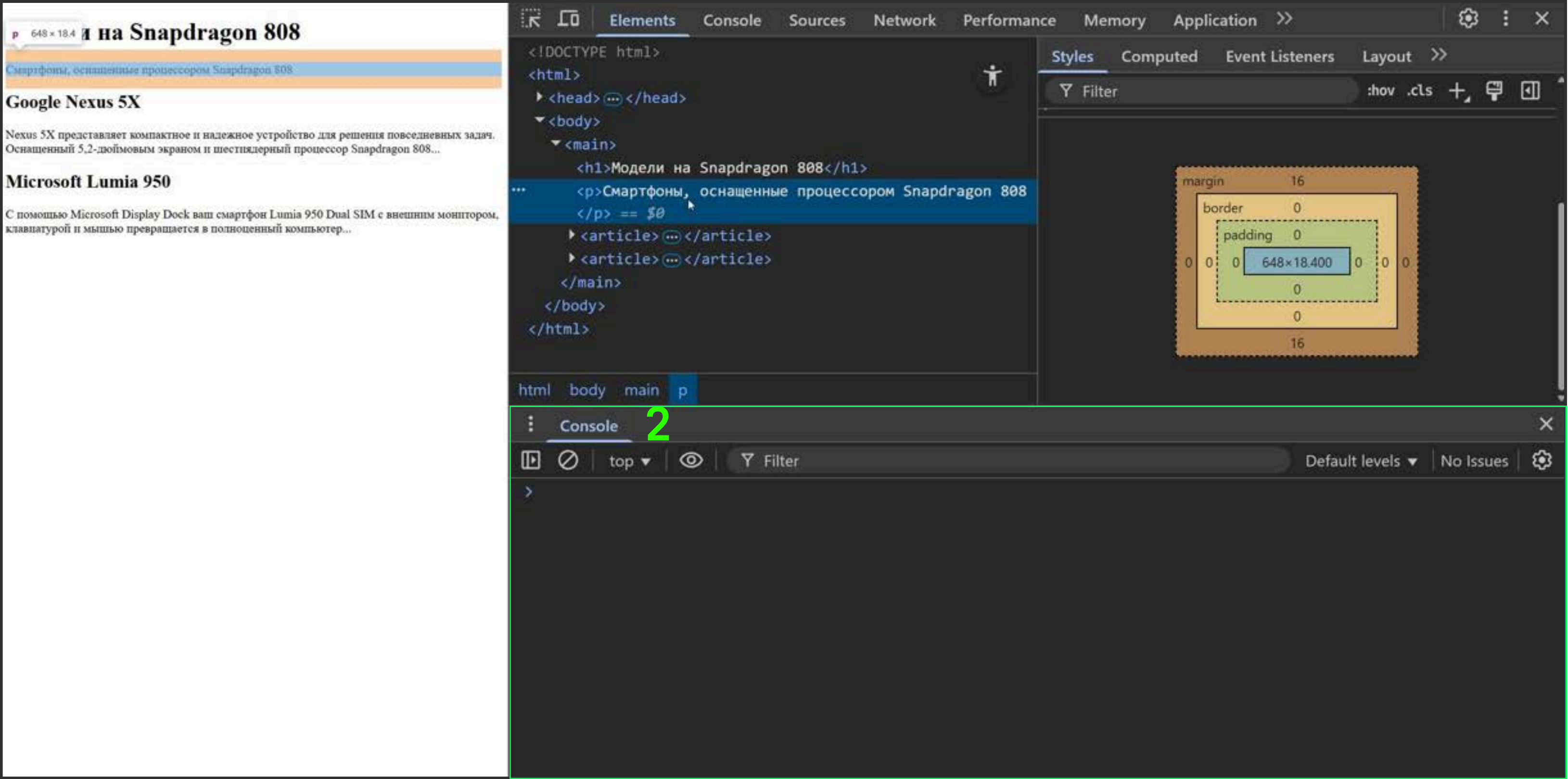
Инструкция по открытию панели "Инструменты разработчика" в Google Chrome

- а. Через контекстное меню:
  - Откройте браузер Google Chrome.
  - Перейдите на веб-страницу, которую хотите исследовать.
  - Щёлкните правой кнопкой мыши на любом элементе страницы.
  - В контекстном меню выберите "Просмотреть код" (или "Inspect" в английской версии).
- б. Через меню браузера:
  - Откройте Google Chrome и перейдите на нужную страницу.
  - Нажмите на значок меню (три точки) в правом верхнем углу.
  - Выберите "Дополнительные инструменты" → "Инструменты разработчика".
- в. С помощью горячих клавиш:
  - На Windows/Linux: Нажмите Ctrl + Shift + I или F12.
  - На Mac: Нажмите Cmd + Option + I.
  - Это мгновенно откроет панель DevTools.
- г. Открытие на определённой вкладке:
  - После открытия DevTools внизу или сбоку (в зависимости от настроек) появятся вкладки: Elements, Console, Network и т.д.
  - Кликните на нужную вкладку (например, "Elements" для просмотра HTML/CSS).

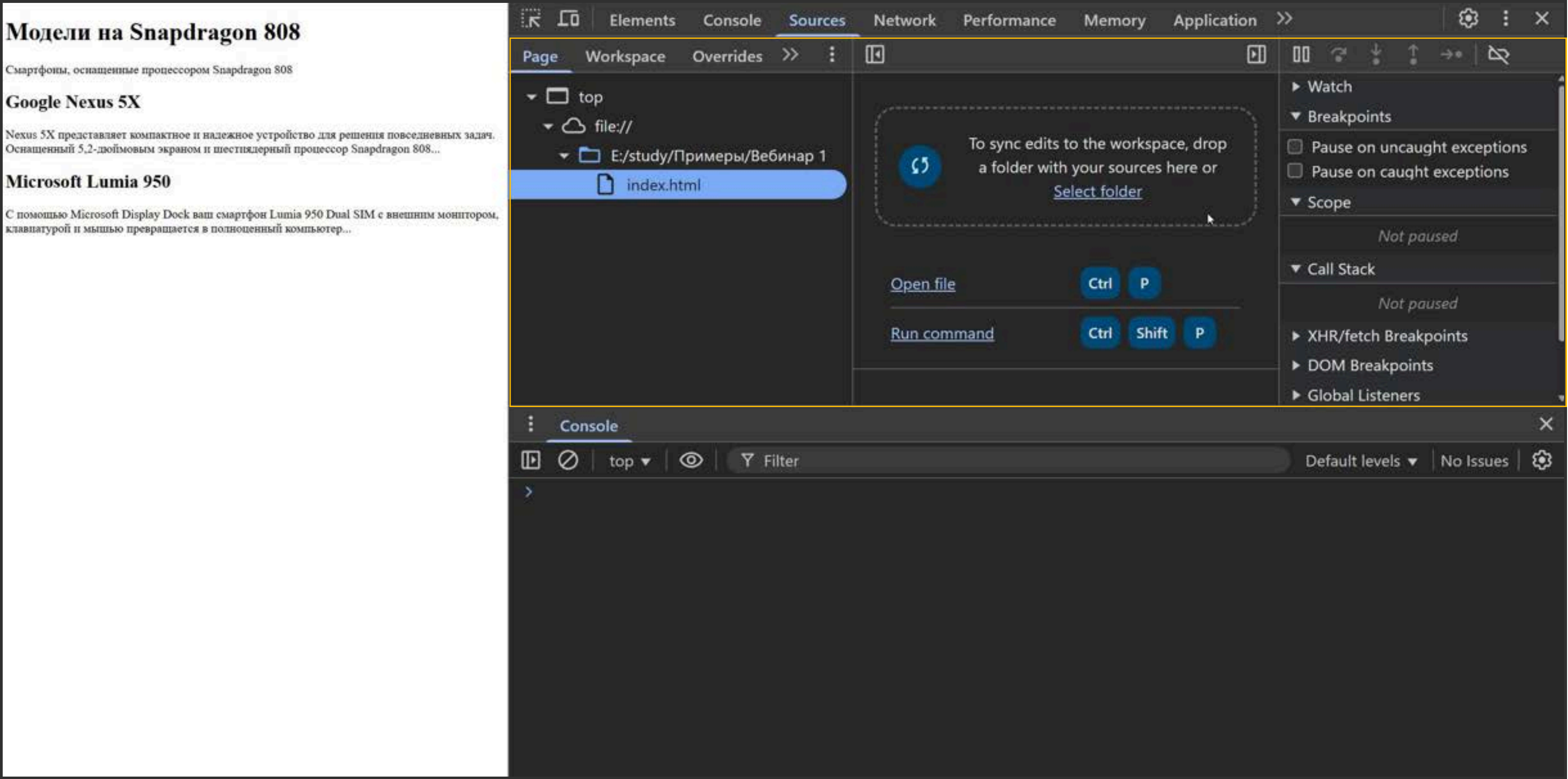


1. **Elements (Элементы):** Просмотр и редактирование DOM-дерева, CSS-стилей, просмотр событий и свойств элементов.
2. **Console (Консоль):** Вывод ошибок, логов, предупреждений, выполнение JavaScript-кода.
3. **Sources (Источники):** Доступ к файлам страниц (HTML, CSS, JS), отладка JavaScript с точками останова.
4. **Network (Сеть):** Анализ сетевых запросов, времени загрузки, статуса, заголовков и содержимого ресурсов.
5. **Performance (Производительность):** Запись и анализ производительности страницы, профилирование загрузки и рендеринга.
6. **Memory (Память):** Инструменты для анализа использования памяти, поиска утечек и профилирования кучи.
7. **Application (Приложение):** Управление localStorage, sessionStorage, cookies, кэшем, сервис-воркерами.
8. **Security (Безопасность):** Информация о сертификатах HTTPS, проблемах безопасности и защищенности соединения.
9. **Lighthouse:** Аудит страницы по производительности, SEO, доступности и PWA.
10. **Device Toolbar (Эмуляция устройств):** Тестирование адаптивности с выбором устройств, разрешений и симуляцией сенсорного ввода.



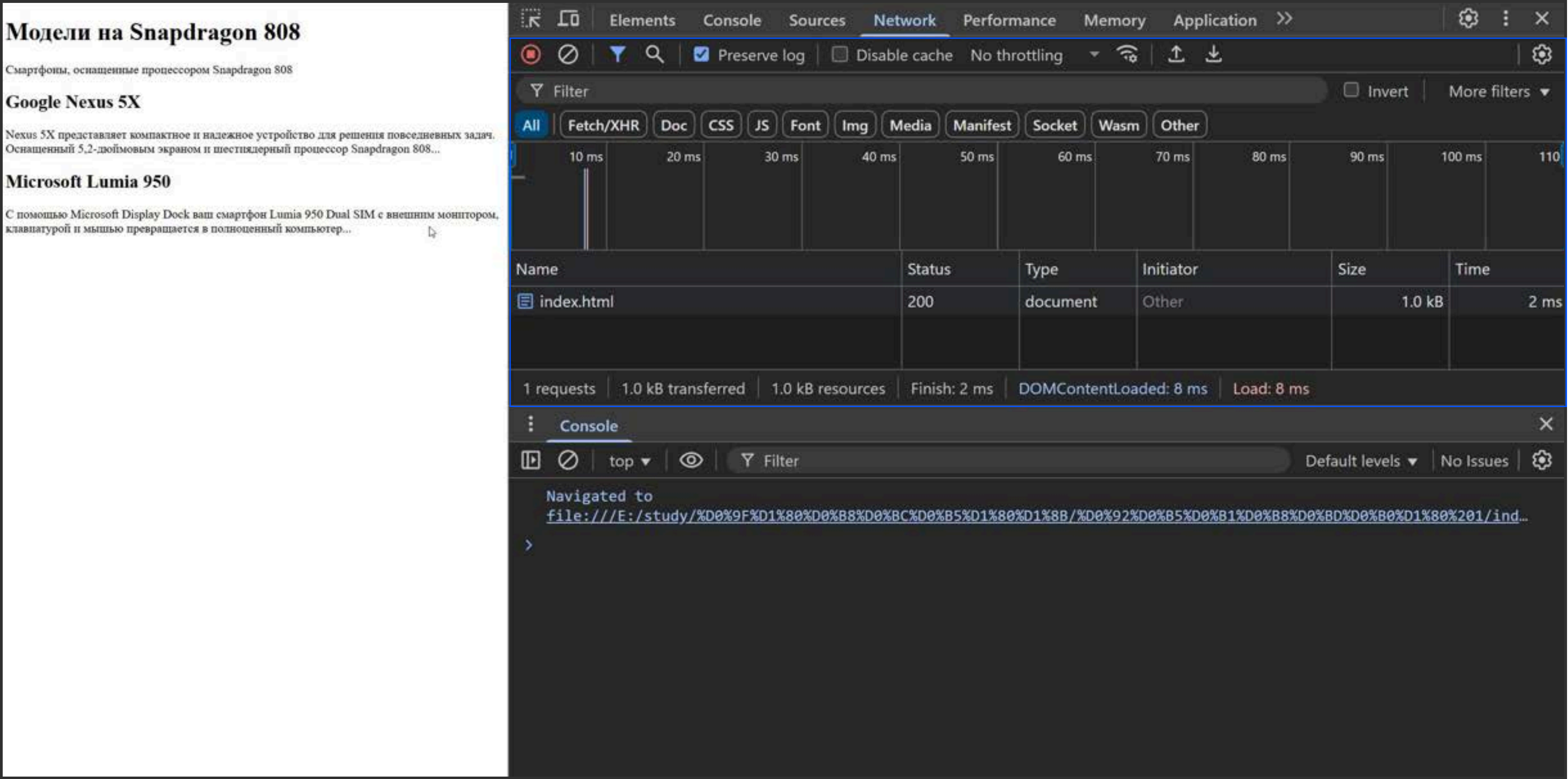


1. Elements (Элементы): Просмотр и редактирование DOM-дерева, CSS-стилей, просмотр событий и свойств элементов.
2. Console (Консоль): Вывод ошибок, логов, предупреждений, выполнение JavaScript-кода.
3. Sources (Источники): Доступ к файлам страниц (HTML, CSS, JS), отладка JavaScript с точками останова.
4. Network (Сеть): Анализ сетевых запросов, времени загрузки, статуса, заголовков и содержимого ресурсов.
5. Performance (Производительность): Запись и анализ производительности страницы, профилирование загрузки и рендеринга.
6. Memory (Память): Инструменты для анализа использования памяти, поиска утечек и профилирования кучи.
7. Application (Приложение): Управление localStorage, sessionStorage, cookies, кэшем, сервис-воркерами.
8. Security (Безопасность): Информация о сертификатах HTTPS, проблемах безопасности и защищенности соединения.
9. Lighthouse: Аудит страницы по производительности, SEO, доступности и PWA.
10. Device Toolbar (Эмуляция устройств): Тестирование адаптивности с выбором устройств, разрешений и симуляцией сенсорного ввода.

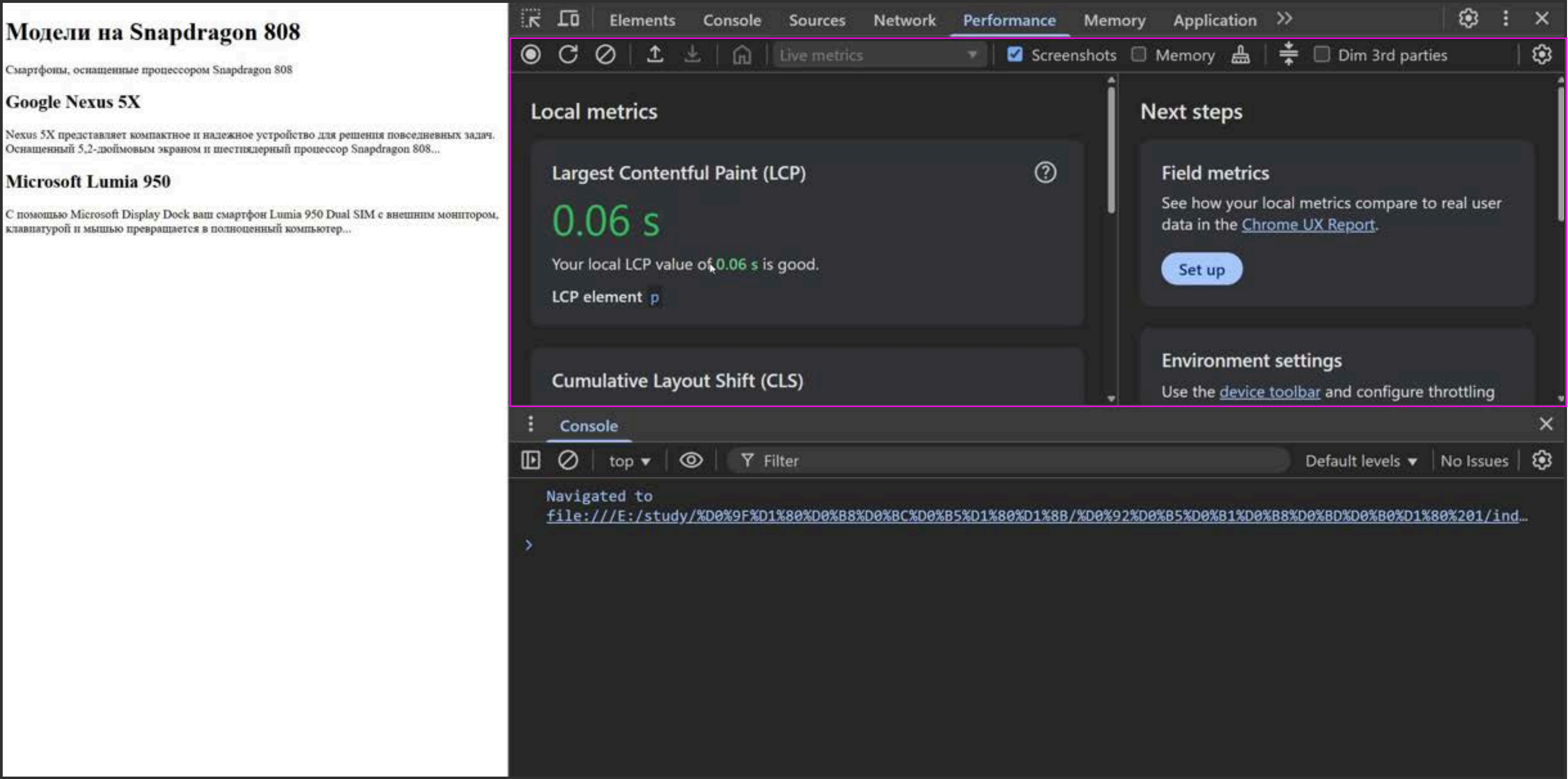


1. Elements (Элементы): Просмотр и редактирование DOM-дерева, CSS-стилей, просмотр событий и свойств элементов.
2. Console (Консоль): Вывод ошибок, логов, предупреждений, выполнение JavaScript-кода.
3. Sources (Источники): Доступ к файлам страниц (HTML, CSS, JS), отладка JavaScript с точками останова.
4. Network (Сеть): Анализ сетевых запросов, времени загрузки, статуса, заголовков и содержимого ресурсов.
5. Performance (Производительность): Запись и анализ производительности страницы, профилирование загрузки и рендеринга.
6. Memory (Память): Инструменты для анализа использования памяти, поиска утечек и профилирования кучи.
7. Application (Приложение): Управление localStorage, sessionStorage, cookies, кэшем, сервис-воркерами.
8. Security (Безопасность): Информация о сертификатах HTTPS, проблемах безопасности и защищенности соединения.
9. Lighthouse: Аудит страницы по производительности, SEO, доступности и PWA.
10. Device Toolbar (Эмуляция устройств): Тестирование адаптивности с выбором устройств, разрешений и симуляцией сенсорного ввода.



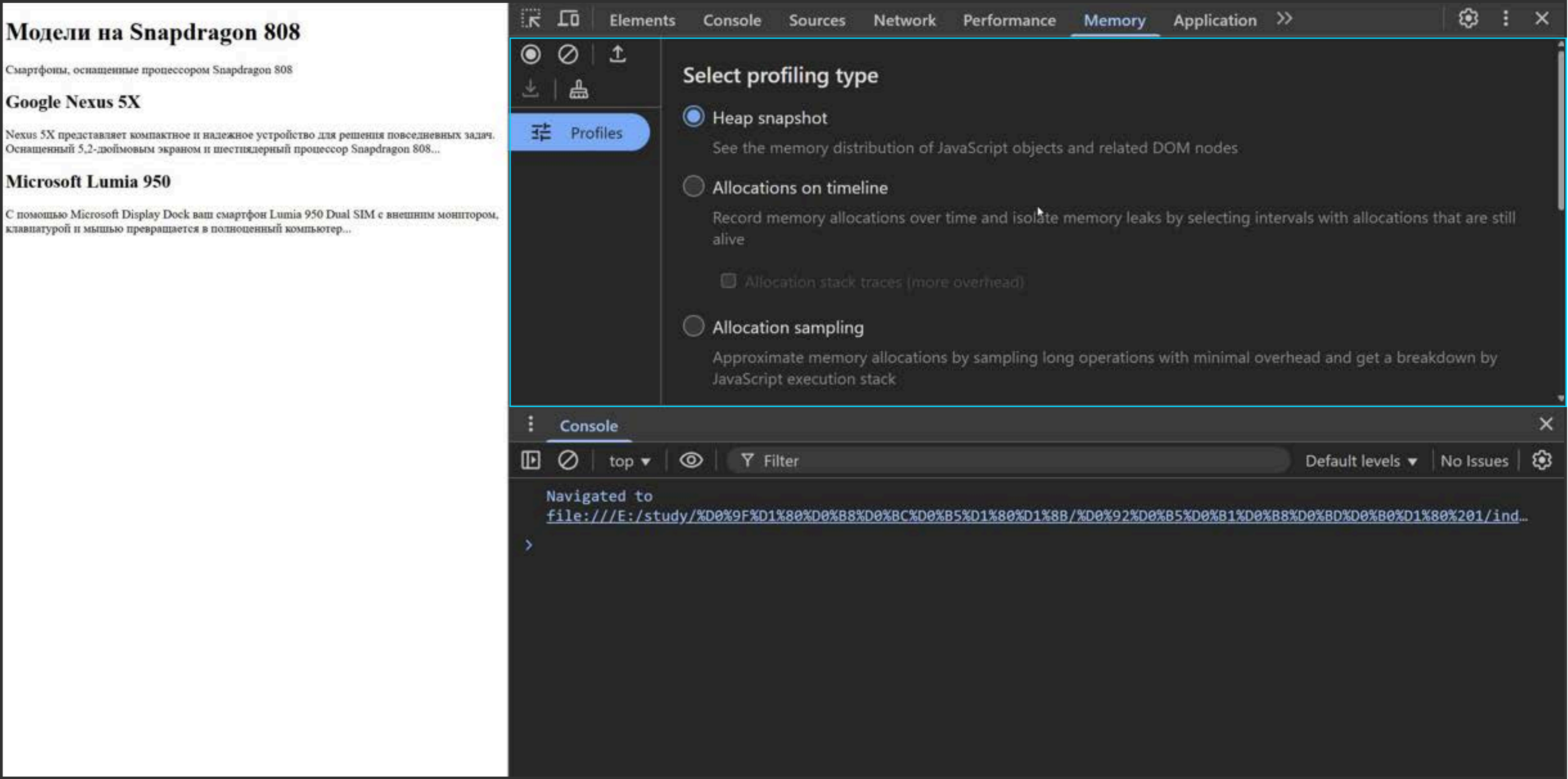


1. Elements (Элементы): Просмотр и редактирование DOM-дерева, CSS-стилей, просмотр событий и свойств элементов.
2. Console (Консоль): Вывод ошибок, логов, предупреждений, выполнение JavaScript-кода.
3. Sources (Источники): Доступ к файлам страниц (HTML, CSS, JS), отладка JavaScript с точками останова.
4. Network (Сеть): Анализ сетевых запросов, времени загрузки, статуса, заголовков и содержимого ресурсов.
5. Performance (Производительность): Запись и анализ производительности страницы, профилирование загрузки и рендеринга.
6. Memory (Память): Инструменты для анализа использования памяти, поиска утечек и профилирования кучи.
7. Application (Приложение): Управление localStorage, sessionStorage, cookies, кэшем, сервис-воркерами.
8. Security (Безопасность): Информация о сертификатах HTTPS, проблемах безопасности и защищенности соединения.
9. Lighthouse: Аудит страницы по производительности, SEO, доступности и PWA.
10. Device Toolbar (Эмуляция устройств): Тестирование адаптивности с выбором устройств, разрешений и симуляцией сенсорного ввода.

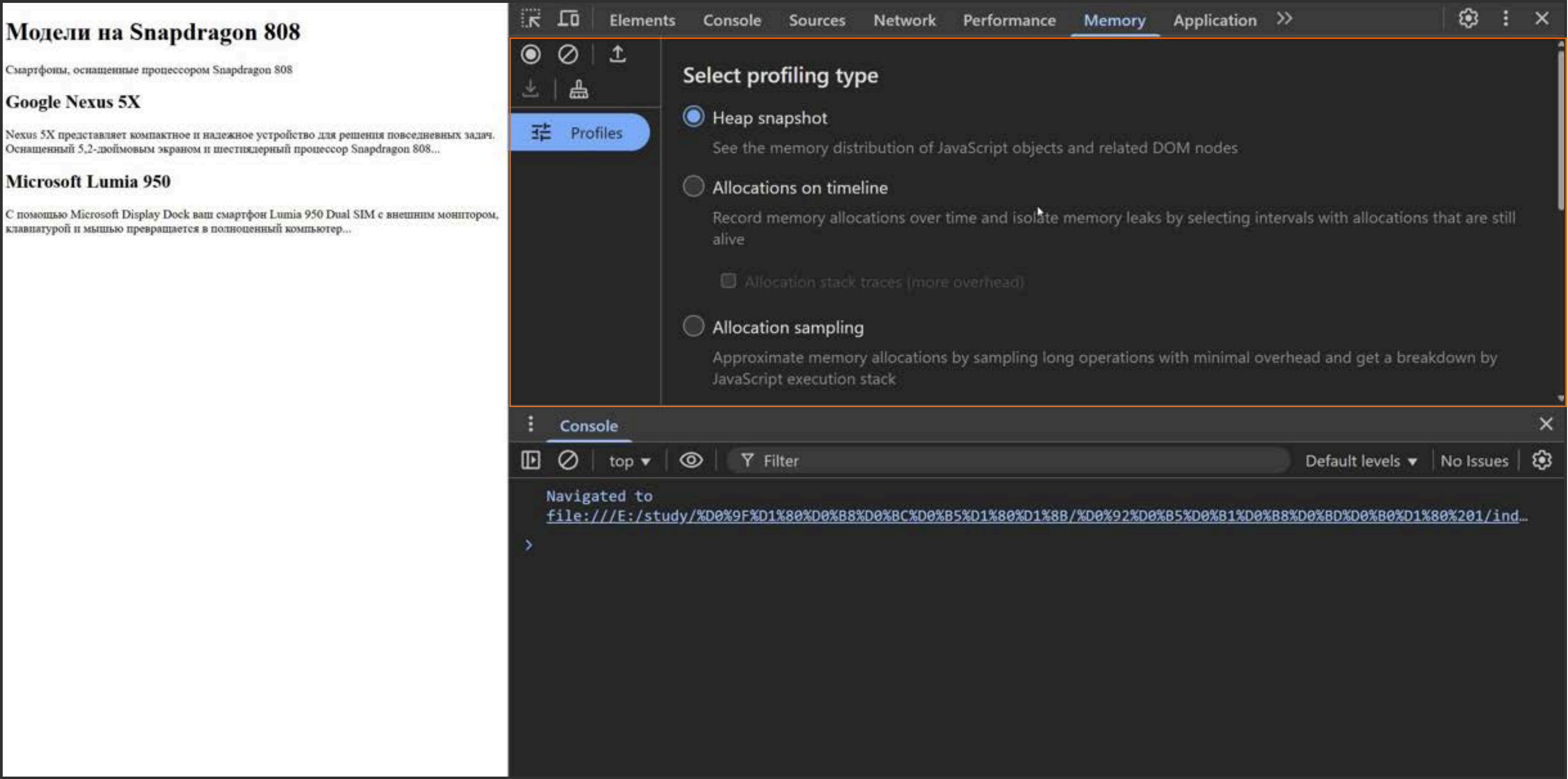


1. Elements (Элементы): Просмотр и редактирование DOM-дерева, CSS-стилей, просмотр событий и свойств элементов.
2. Console (Консоль): Вывод ошибок, логов, предупреждений, выполнение JavaScript-кода.
3. Sources (Источники): Доступ к файлам страниц (HTML, CSS, JS), отладка JavaScript с точками останова.
4. Network (Сеть): Анализ сетевых запросов, времени загрузки, статуса, заголовков и содержимого ресурсов.
5. Performance (Производительность): Запись и анализ производительности страницы, профилирование загрузки и рендеринга.
6. Memory (Память): Инструменты для анализа использования памяти, поиска утечек и профилирования кучи.
7. Application (Приложение): Управление localStorage, sessionStorage, cookies, кэшем, сервис-воркерами.
8. Security (Безопасность): Информация о сертификатах HTTPS, проблемах безопасности и защищенности соединения.
9. Lighthouse: Аудит страницы по производительности, SEO, доступности и PWA.
10. Device Toolbar (Эмуляция устройств): Тестирование адаптивности с выбором устройств, разрешений и симуляцией сенсорного ввода.



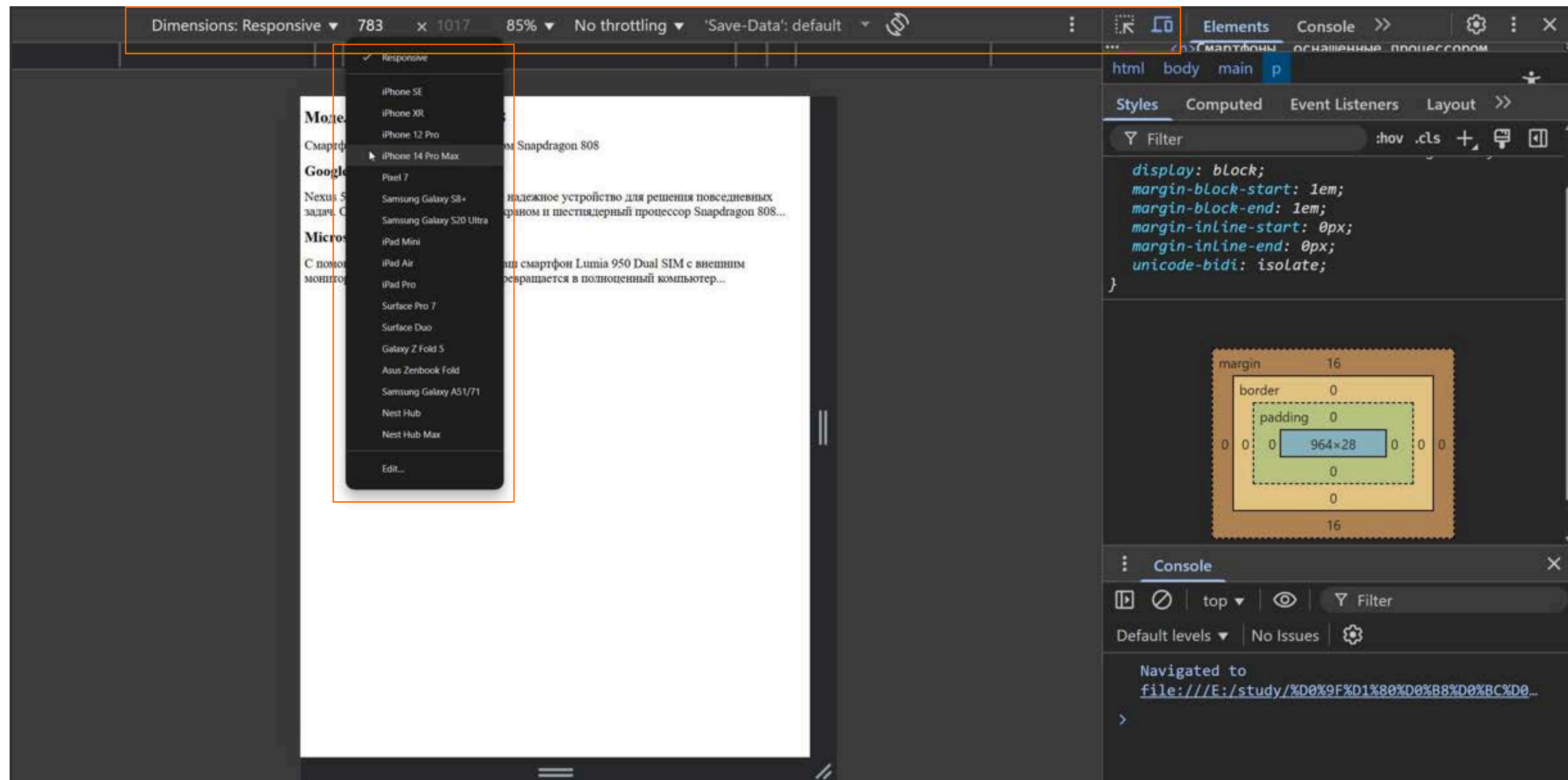


- 1. Elements (Элементы): Просмотр и редактирование DOM-дерева, CSS-стилей, просмотр событий и свойств элементов.
- 2. Console (Консоль): Вывод ошибок, логов, предупреждений, выполнение JavaScript-кода.
- 3. Sources (Источники): Доступ к файлам страниц (HTML, CSS, JS), отладка JavaScript с точками останова.
- 4. Network (Сеть): Анализ сетевых запросов, времени загрузки, статуса, заголовков и содержимого ресурсов.
- 5. Performance (Производительность): Запись и анализ производительности страницы, профилирование загрузки и рендеринга.
- 6. Memory (Память): Инструменты для анализа использования памяти, поиска утечек и профилирования кучи.
- 7. Application (Приложение): Управление localStorage, sessionStorage, cookies, кэшем, сервис-воркерами.
- 8. Security (Безопасность): Информация о сертификатах HTTPS, проблемах безопасности и защищенности соединения.
- 9. Lighthouse: Аудит страницы по производительности, SEO, доступности и PWA.
- 10. Device Toolbar (Эмуляция устройств): Тестирование адаптивности с выбором устройств, разрешений и симуляцией сенсорного ввода.



1. Elements (Элементы): Просмотр и редактирование DOM-дерева, CSS-стилей, просмотр событий и свойств элементов.
2. Console (Консоль): Вывод ошибок, логов, предупреждений, выполнение JavaScript-кода.
3. Sources (Источники): Доступ к файлам страниц (HTML, CSS, JS), отладка JavaScript с точками останова.
4. Network (Сеть): Анализ сетевых запросов, времени загрузки, статуса, заголовков и содержимого ресурсов.
5. Performance (Производительность): Запись и анализ производительности страницы, профилирование загрузки и рендеринга.
6. Memory (Память): Инструменты для анализа использования памяти, поиска утечек и профилирования кучи.
7. Application (Приложение): Управление localStorage, sessionStorage, cookies, кэшем, сервис-воркерами.
8. Device Toolbar (Эмуляция устройств): Тестирование адаптивности с выбором устройств, разрешений и симуляцией сенсорного ввода.





1. **Elements (Элементы):** Просмотр и редактирование DOM-дерева, CSS-стилей, просмотр событий и свойств элементов.
2. **Console (Консоль):** Вывод ошибок, логов, предупреждений, выполнение JavaScript-кода.
3. **Sources (Источники):** Доступ к файлам страницы (HTML, CSS, JS), отладка JavaScript с точками останова.
4. **Network (Сеть):** Анализ сетевых запросов, времени загрузки, статуса, заголовков и содержимого ресурсов.
5. **Performance (Производительность):** Запись и анализ производительности страницы, профилирование загрузки и рендеринга.
6. **Memory (Память):** Инструменты для анализа использования памяти, поиска утечек и профилирования кучи.
7. **Application (Приложение):** Управление localStorage, sessionStorage, cookies, кэшем, сервис-воркерами.
8. **Device Toolbar (Эмуляция устройств):** Тестирование адаптивности с выбором устройств, разрешений и симуляцией сенсорного ввода.



*ВОПРОСЫ?*