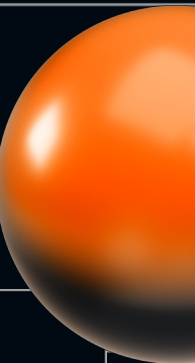
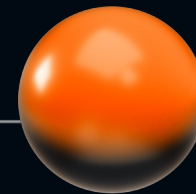


Браузерные инструменты на страже перформанса




**Иванов
Алексей**

Самолет



 @alexey_qa

 alexxx55509@ya.ru

HEISENBUG



**Иванов
Алексей**

✈ alexey_qa

Обо мне

😄 Люблю качественные продукты



самолет



**Иванов
Алексей**

✈ alexey_qa

Обо мне



Люблю качественные продукты



Пишу код, который проверяет код!



самолет



**Иванов
Алексей**

✈ alexey_qa

Обо мне

- 🌟 Люблю качественные продукты
- 💻 Пишу код, который проверяет код!
- ✍ Веду тг канал @dev_qa



самолет



**Иванов
Алексей**

✈ alexey_qa

Обо мне

- 👑 Люблю качественные продукты
- 💻 Пишу код, который проверяет код!
- ✍ Веду тг канал @dev_qa
- ☁ Мечтаю создать дружное московское QA сообщество



самолет

О чём будет доклад?

- Lighthouse



О чём будет доклад?

- Lighthouse
- Network Tab



О чём будет доклад?

- Lighthouse
- Network Tab
- Memory Tab



О чём будет доклад?

- Lighthouse
- Network Tab
- Memory Tab
- Performance



Почему performance важен

Walmart

Каждое улучшение на 100 мс приводит к 1% дополнительного дохода.

Cloudflare

Уменьшение времени загрузки с 5.7 секунд до 2.4 секунд утраивает коэффициент конверсии.

Akamai

Задержка во времени загрузки веб-страницы на две секунды увеличивает показатель отказов на 103%.



самолет



Manager

Клиенты жалуются, что наше приложение зависает. Надо срочно разобраться, что происходит.



самолет



Manager

Клиенты жалуются, что наше приложение зависает. Надо срочно разобраться, что происходит.



DEV

А что случилось?



самолет



Manager

Клиенты жалуются, что наше приложение зависает. Надо срочно разобраться, что происходит.



DEV

А что случилось?



Manager

О, а у нас еще и тестировщики здесь! Прекрасно, вы можете помочь нам выяснить, почему приложение зависает?





Manager

Клиенты жалуются, что наше приложение зависает. Надо срочно разобраться, что происходит.



DEV

А что случилось?



Manager

О, а у нас еще и тестировщики здесь! Прекрасно, вы можете помочь нам выяснить, почему приложение зависает?



QA

Конечно, мы рады помочь! Может быть, наше приложение просто решило сыграть в прятки со своими багами.



**Manager**

Клиенты жалуются, что наше приложение зависает. Надо срочно разобраться, что происходит.

**DEV**

А что случилось?

**Manager**

О, а у нас еще и тестировщики здесь! Прекрасно, вы можете помочь нам выяснить, почему приложение зависает?

**QA**

Конечно, мы рады помочь! Может быть, наше приложение просто решило сыграть в прятки со своими багами.

**Manager**

Ребята, будьте серьезней, давайте разберемся с проблемой.



Как определить,
что с приложением что-то не так ?



Как определить, что с приложением что-то не так ?

- Мониторинг



Как определить, что с приложением что-то не так ?

- Мониторинг
- Юзеры пришли и сказали



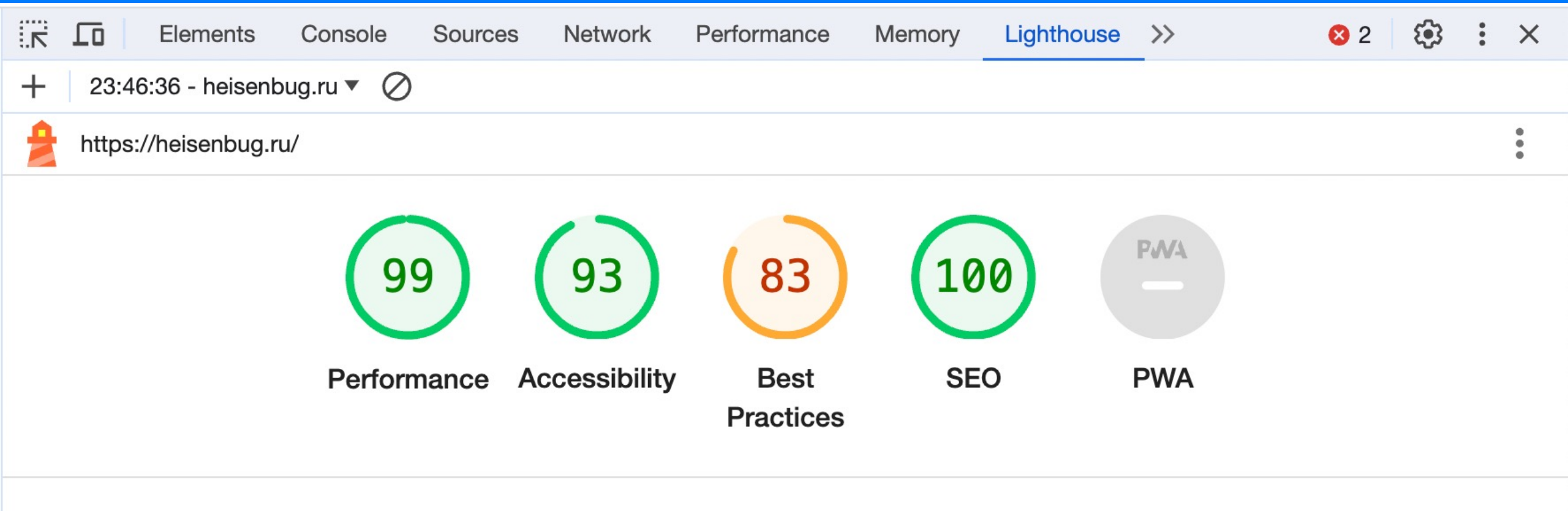
Как определить, что с приложением что-то не так ?

- Мониторинг
- Юзеры пришли и сказали
- Отловили на тестирование используя специальные инструменты

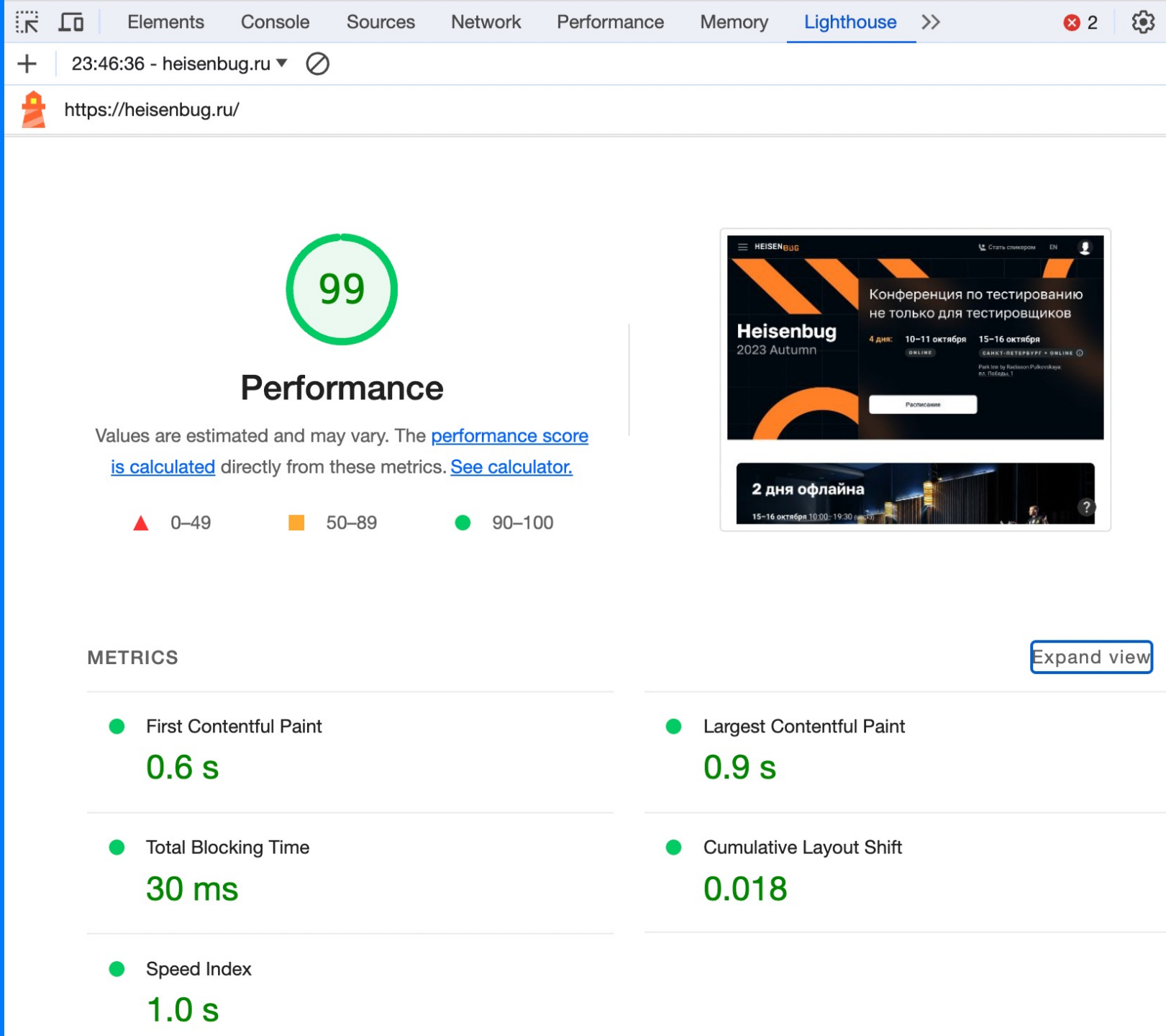
Есть ли универсальная пилюля?



самолет



самолет













самолет

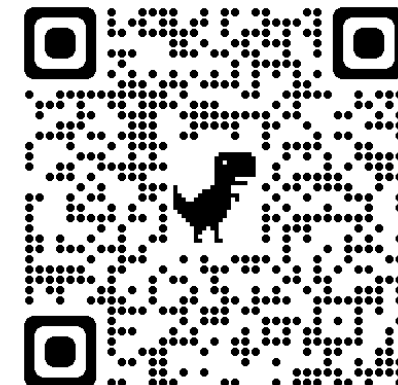
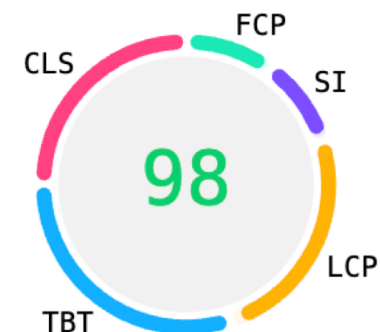
Метрики lighthouse

Lighthouse Scoring Calculator

Device type: Versions:

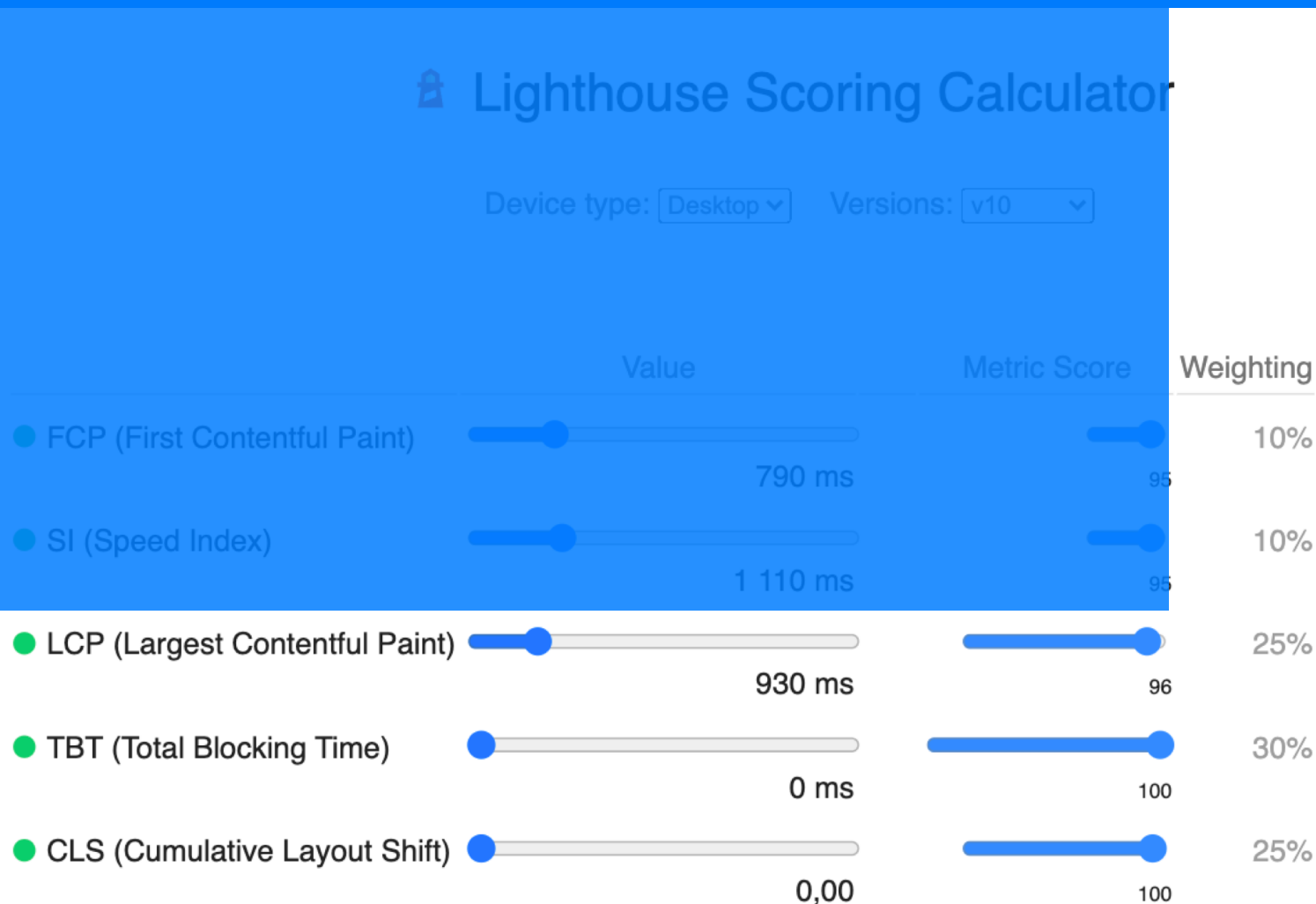
latest
[v10](#)

	Value	Metric Score	Weighting
● FCP (First Contentful Paint)	 790 ms	 95	10%
● SI (Speed Index)	 1 110 ms	 95	10%
● LCP (Largest Contentful Paint)	 930 ms	 96	25%
● TBT (Total Blocking Time)	 0 ms	 100	30%
● CLS (Cumulative Layout Shift)	 0,00	 100	25%



Метрики lighthouse

latest
[v10](#)



<https://googlechrome.github.io/lighthouse/scorecalc/>

самолет

Largest Contentful Paint (LCP)

самолет

+7 499 455-94-70



Ипотека от 2,9%

Своя квартира от 17 000 ₽ в месяц

4% скидка на покупку квартиры
в день посещения офиса продаж до 15.10

До окончания программы:

10 : 02 : 10 : 11
дней : часа : мин. : сек.

[Подробнее](#)



Откройте Целепорт

Начните копить на
первоначальный взнос с
Целепортом и получите до
24% г

Мы используем cookie, чтобы улучшить работу сайта.
[Как это работает](#)

Ок



Проекты



Квартиры



Избранное



Ипотека



Меню

Cumulative Layout Shift (CLS)



РЕКЛАМА

самолет

+7 495 126-16-38



Своя квартира от 17 000 Р в месяц

Подробнее



Бонусы к квартире мечты от Самолет

Купите квартиру в ипотеку от Сбера, получите скидку 3% и бонусы «спасибо» в размере 0,3% от стоимости кредита

Подробнее

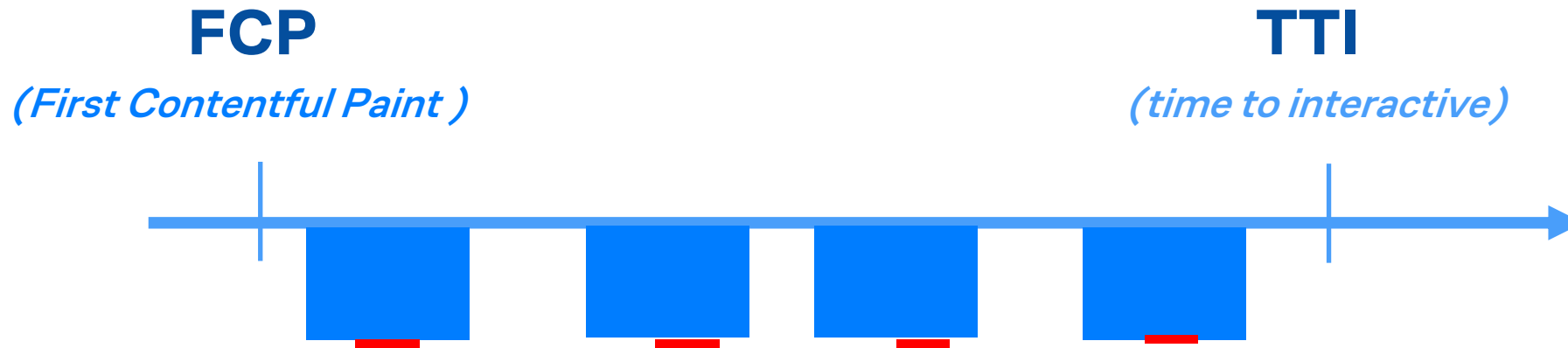
Мы используем cookie, чтобы улучшить работу сайта. [Как это работает](#)

Ок

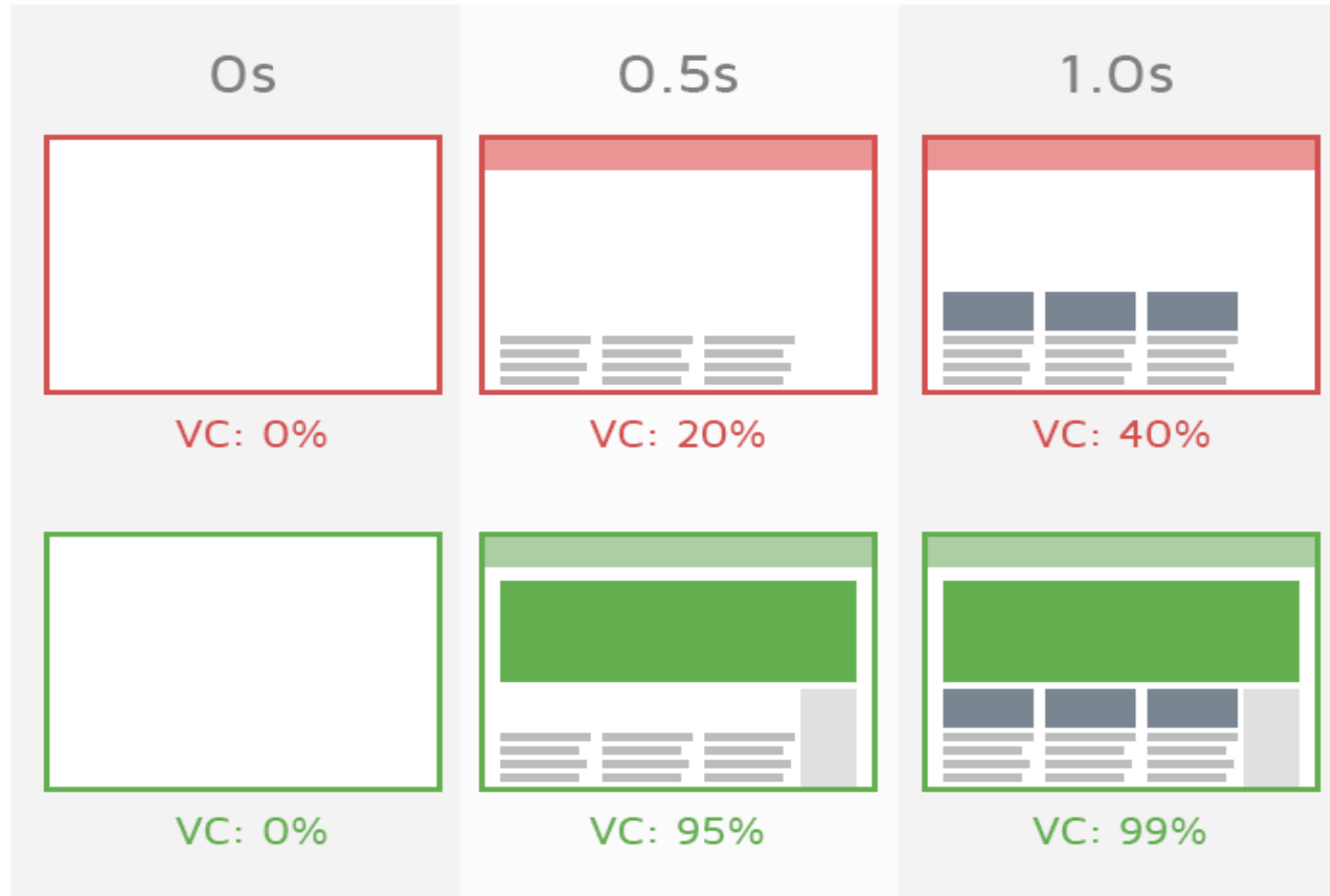
First Contentful Paint (FCP)



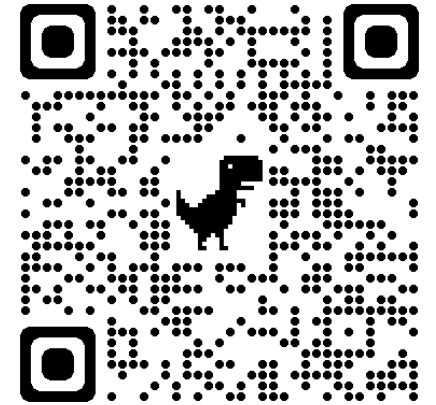
Total Blocking Time (TBT)



Speed Index (SI)



VC = Visual Completeness



Что можно сделать сейчас

Топ-продукты

События

Теханализ

Личный кабинет

Статистика

Котировки

Аналитика

Торговое ПО

Ав

Открыть счет

Тест-драйв

15:44 • Аналитики Финама • Девять идей аналитиков «Финама», которые принесли высокую доходность

РусГидро 2,00%

Совкомфлот 2,45%

МосБиржа 4,21%

AbbVie Inc. -0,98%

Amgen Inc. -0,58%

Halliburton Company -0,25%

Global Payments Inc. -1,08%

15:22 • Новости международных рынков • SoftBank готов вложиться в OpenAI

15:00 • Новости и комментарии • Кризис на пороге

Еще новости

Котировки, индексы

Лидеры роста и падения

Мировые индексы

Фьючерсы

Акции РФ

Сырьевые рынки

Валютные рынки

Россия

США

ETF

15.09.2023

ДЭК ао

9.27%

АЛРОСА ао*

-1.43%

МосБиржа*

4.21%

GLTR-гдр*

-0.08%

29

Performance

Values are estimated and may vary. The [performance score is calculated](#) directly from these metrics. [See calculator.](#)

▲ 0–49

■ 50–89

● 90–100

METRICS

Expand view

▲ First Contentful Paint

4.3 s

▲ Total Blocking Time

1,360 ms

▲ Speed Index

13.3 s

▲ Largest Contentful Paint

19.6 s

■ Cumulative Layout Shift

0.109

Получите доступ к данным США, Евро, рублем

Топ-продукты

События

Теханализ

Личный кабинет

Открыть счет

Тест-драйв

15:44 • Аналитики Финама • Девять идей аналитиков «Финама», которые принесли высокую доходность

РусГидро 2,00% Совкомфлот 2,45% МосБиржа 4,21% AbbVie Inc. -0,98% Amgen Inc. -0,58% Halliburton Company -0,25% Global Payments Inc. -1,08%

15:22 • Новости международных рынков • SoftBank готов вложиться в OpenAI






15:00 • Новости и комментарии • Кризис на пороге

Еще новости

Котировки, индексы

Что можно сделать сейчас

OPPORTUNITIES

Opportunity		Estimated Savings
▲ Reduce unused JavaScript		7.34s ▼
▲ Use HTTP/2		2.38s ▼
▲ Minify JavaScript		1.27s ▼
■ Eliminate render-blocking resources		0.57s ▼
■ Avoid serving legacy JavaScript to modern browsers		0.30s ▼

Оптимизация показателей Lighthouse за счет уменьшения неиспользуемого JavaScript





Description

Цель:

Улучшить производительность сайта, уменьшив неиспользуемый JavaScript, как указано в отчете Lighthouse. Это поможет снизить объем данных, потребляемых сетевой активностью, и улучшить время загрузки контента (LCP).

Исходные данные:

Отчет Lighthouse указывает на значительное количество неиспользуемого JavaScript, загрузка которого занимает до 7,34 секунд. Конкретные файлы и их потенциальная экономия:

<input checked="" type="checkbox"/>	TEAM-2	Выявление неиспользуемого кода	=		TO DO ▾
<input checked="" type="checkbox"/>	TEAM-3	Разделение кода	=		TO DO ▾
<input checked="" type="checkbox"/>	TEAM-4	Отложенная загрузка	=		TO DO ▾
<input checked="" type="checkbox"/>	TEAM-5	Оптимизация сторонних библиотек	=		TO DO ▾



Оптимизация показателей Lighthouse за счет перехода на HTTP/2

Description

Цель:

Улучшить производительность сайта, перейдя на протокол HTTP/2, который предлагает ряд преимуществ по сравнению с HTTP/1.1, включая бинарные заголовки и мультиплексирование.

Исходные данные:

Отчет Lighthouse указывает, что сайт использует протокол HTTP/1.1, что занимает дополнительное время на загрузку — 2,38 секунды. Список URL, использующих HTTP/1.1:

- <https://www.site.ru>
- ...classic/jquery-1.11.2.min.cp.js(www.site.ru)
- ...infinity/global.js(www.site.ru)
- ...bootstrap/bootstrap.min.css(www.site.ru)
- ...css/site-call.min.css?ver=1(www.site.ru)

Какие проблемы может подсветить lighthouse

- Задержки во времени ответа сервера



Какие проблемы может подсветить lighthouse

- Задержки во времени ответа сервера
- Ненужное или избыточное использование JavaScript



Какие проблемы может подсветить lighthouse

- Задержки во времени ответа сервера
- Ненужное или избыточное использование JavaScript
- Оптимизация изображений



Какие проблемы может подсветить lighthouse

- Задержки во времени ответа сервера
- Ненужное или избыточное использование JavaScript
- Оптимизация изображений
- Блокирование основного потока



Какие проблемы может подсветить lighthouse

- Задержки во времени ответа сервера
- Ненужное или избыточное использование JavaScript
- Оптимизация изображений
- Блокирование основного потока
- Неправильное использование кэширования






Lighthouse

Memory

Network

Performance

Performance insights 

Какие проблемы lighthouse не сможет решить

Избыточные или неэффективные сетевые запросы



самолет

Какие проблемы lighthouse не сможет решить

Избыточные или неэффективные сетевые запросы

Проблемы с отзывчивостью приложения



Какие проблемы lighthouse не сможет решить

Избыточные или неэффективные сетевые запросы

Проблемы с отзывчивостью приложения

Задержки в анимациях и интерактивных элементах



Какие проблемы lighthouse не сможет решить

Избыточные или неэффективные сетевые запросы

Проблемы с отзывчивостью приложения

Задержки в анимациях и интерактивных элементах

Утечки памяти



А теперь бонус как сделать
lighthouse 100



А теперь бонус как сделать lighthouse 100

(не повторять)



самолет

ава ОПЕК предупредил об
ую промышленность



Performance

Values are estimated and may vary. The [performance score is calculated](#) directly from these metrics. [See calculator.](#)



0–49



50–89



90–100

ксы | Фьючерсы | Акции РФ | Сырьевые |

METRICS

Expand view



First Contentful Paint

2.0 s



Largest Contentful Paint

4.7 s



Total Blocking Time

70 ms



Cumulative Layout Shift

0.243



Speed Index

8.0 s

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Performance 100</title>
</head>
<body>
  <iframe src="https://www.site.ru/" width="100%" height="100%"></iframe>
</body>
</html>
```





Performance

Values are estimated and may vary. The [performance score is calculated](#) directly from these metrics. [See calculator.](#)



0–49



50–89



90–100

METRICS

[Expand view](#)



First Contentful Paint

0.2 s



Largest Contentful Paint

0.2 s



Total Blocking Time

10 ms



Cumulative Layout Shift

0



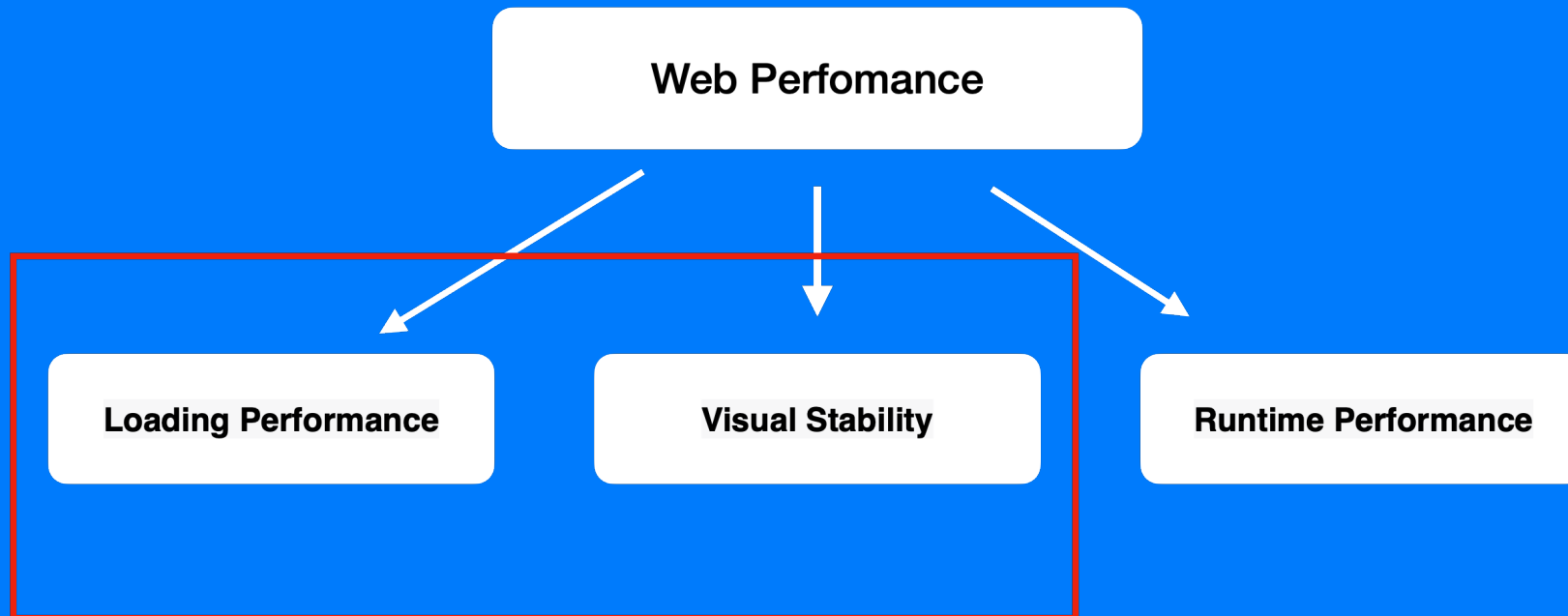
**Первое правило тестировщика —
не верь на слова разработчику**



**Второе правило тестировщика —
не верь метрикам lighthouse**



Типы web performance



Нужен dev tools



самолет

Network



самолет

Какие проблемы можно найти

Задержки в сетевых запросах



самолет

Какие проблемы можно найти

Задержки в сетевых запросах

Множественные или дублированные
запросы



самолет

Какие проблемы можно найти

Задержки в сетевых запросах

Множественные или дублированные
запросы

Загрузка больших/блокирующих
ресурсов



самолет

Какие проблемы можно найти

Задержки в сетевых запросах

Множественные или дублированные
запросы

Загрузка больших/блокирующих
ресурсов

Неэффективное кэширование



Кейс #1 Страница списка новостей



Требования:

1. Разработать страницу со списком новостей и аннотацией
2. Страница должна содержать кнопку "Reload Posts", при нажатии на которую произойдет повторный запрос списка постов.



Наше тестовое приложение



[Home](#) [Reload Posts](#)

- **sunt aut facere repellat provident occaecati excepturi optio reprehenderit**

quia et suscipit suscipit recusandae consequuntur expedita et cum reprehenderit molestiae ut ut quas totam nostrum rerum est autem sunt rem eveniet architecto

- **qui est esse**

est rerum tempore vitae sequi sint nihil reprehenderit dolor beatae ea dolores neque fugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis qui aperiam non debitis possimus qui neque nisi nulla

- **ea molestias quasi exercitationem repellat qui ipsa sit aut**

et iusto sed quo iure voluptatem occaecati omnis eligendi aut ad voluptatem doloribus vel accusantium quis pariatur molestiae porro eius odio et labore et velit aut

- **eum et est occaecati**

ullam et saepe reiciendis voluptatem adipisci sit amet autem assumenda provident rerum culpa quis hic commodi nesciunt rem tenetur doloremque ipsam iure quis sunt voluptatem rerum illo velit

- **nesciunt quas odio**

repudiandae veniam quaerat sunt sed alias aut fugiat sit autem sed est voluptatem omnis possimus esse voluptatibus quis est aut tenetur dolor neque

- **dolorem eum magni eos aperiam quia**

ut aspernatur corporis harum nihil quis provident sequi mollitia nobis aliquid molestiae perspiciatis et ea nemo ab reprehenderit accusantium quas voluptate dolores velit et doloremque molestiae

- **magnum facilis autem**

dolore placeat quibusdam ea quo vitae magni quis enim qui quis quo nemo aut saepe quidem repellat excepturi ut quia sunt ut sequi eos ea sed quas

• sunt aut facere repellat provident occaecati excepturi optio reprehenderit

quia et suscipit suscipit recusandae consequuntur expedita et cum reprehenderit molestiae ut ut quas totam nostrum rerum est autem sunt rem eveniet architecto

• qui est esse

est rerum tempore vitae sequi sint nihil reprehenderit dolor beatae ea dolores neque fugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis qui aperiam non debitis possimus qui neque nisi nulla

• ea molestias quasi exercitationem repellat qui ipsa sit aut

et iusto sed quo iure voluptatem occaecati omnis eligendi aut ad voluptatem doloribus vel accusantium quis pariatur molestiae porro eius odio et labore et velit aut

• eum et est occaecati

ullam et saepe reiciendis voluptatem adipisci sit amet autem assumenda provident rerum culpa quis hic commodi nesciunt rem tenetur doloremque ipsam iure quis sunt voluptatem rerum illo velit

• nesciunt quas odio

repudiandae veniam quaerat sunt sed alias aut fugiat sit autem sed est voluptatem omnis possimus esse voluptatibus quis est aut tenetur dolor neque

• dolorem eum magni eos aperiam quia

ut aspernatur corporis harum nihil quis provident sequi mollitia nobis aliquid molestiae perspiciatis et ea nemo ab reprehenderit accusantium quas voluptate dolores velit et doloremque molestiae

• magnam facilis autem

dolore placeat quibusdam ea quo vitae magni quis enim qui quis quo nemo aut saepe quidem

ElementsConsoleNetwork

Preserve logDisable cacheNo throttling

Filter

All

Fetch/XHR

JS

CSS

Img

Media

Font

Doc

WS

Wasm

Manifest

Other

Has blocked cookies

Blocked Requests

3rd-party requests

100000 ms

200000 ms

300000 ms

400000 ms

500000 ms

Name	Status	Type	Initiator	Size	Time	Waterfall
posts?userId=1	200	fetch	manyRe...	1.6...	53 ...	
posts?userId=1	200	fetch	manyRe...	1.6...	53 ...	
posts?userId=1	200	fetch	manyRe...	1.6...	54 ...	
posts?userId=1	200	fetch	manyRe...	1.6...	10...	
posts?userId=1	200	fetch	manyRe...	1.6...	54 ...	
posts?userId=1	200	fetch	manyRe...	1.6...	56 ...	
posts?userId=1	200	fetch	manyRe...	1.6...	56 ...	
posts?userId=1	200	fetch	manyRe...	1.6...	59 ...	
posts?userId=1	200	fetch	manyRe...	1.6...	88 ...	
posts?userId=1	200	fetch	manyRe...	1.6...	53 ...	
posts?userId=1	200	fetch	manyRe...	1.6...	52 ...	
posts?userId=1	200	fetch	manyRe...	1.6...	22...	
posts?userId=1	200	fetch	manyRe...	1.6...	53 ...	
posts?userId=1	(pe...	fetch	manyRe...	0 B	Pe...	

Приходит фикс от разработчика

[Home](#) Reload Posts

- **sunt aut facere repellat provident occaecati excepturi optio reprehenderit**

quia et suscipit suscipit recusandae consequuntur expedita et cum reprehenderit molestiae ut ut quas totam nostrum rerum est autem sunt rem eveniet architecto
- **qui est esse**

est rerum tempore vitae sequi sint nihil reprehenderit dolor beatae voluptate blanditiis voluptate porro vel nihil molestiae ut reiciendis qui aperiam non ve
- **ea molestias quasi exercitationem repellat qui ipsa sit nihil nulla sed unde dolore**

et iusto sed quo iure voluptatem occaecati omnis eligendi autem accusantium quis pariatur molestiae porro eius odio et labore et dolore aut mollit esse
- **eum et est occaecati**

ullam et saepe reiciendis voluptatem adipisci sit amet autem accusantium

Elements Console Sources **Network** >> 1 ⚙ ⋮ ✕

⊞ ⊘ 🔍

☒ Preserve log ☒ Disable cache No throttling ⌵ ⚙

📶 ⬆ ⬇

Filter ☐ Invert ☐ Hide data URLs

All **Fetch/XHR** JS CSS Img Media Font Doc WS Wasm Manifest Other

☐ Has blocked cookies ☐ Blocked Requests ☐ 3rd-party requests

20 ms40 ms60 ms80 ms100 ms

Name	Sta...	Type	Initiator	Size	Time	Waterfall
<input type="checkbox"/> posts?userId=1	200	fetch	manyRe...	1.7...	17...	<div></div>
<input type="checkbox"/> posts?userId=1	200	fetch	manyRe...	1.6...	22...	<div></div>



В чем проблема и как ее исправить?

```
useEffect(() => {  
  fetch(url)  
    .then(response => response.json())  
    .then((data: IPostID[]) => {  
      setPosts(data);  
    })  
}, [reload]);
```



В чем проблема и как ее исправить?

```
useEffect(() => {  
  fetch(url)  
    .then(response => response.json())  
    .then((data: IPostID[]) => {  
      setPosts(data);  
    })  
}, [reload]);
```



В чем проблема и как ее исправить?

```
useEffect(() => {  
  fetch(url)  
    .then(response => response.json())  
    .then((data: IPostID[]) => {  
      setPosts(data);  
    })  
}, [reload]);
```



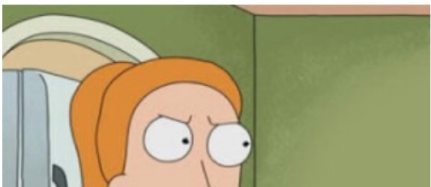
Rick Sanchez



Morty Smith



Summer Smith



Кейс #2

Персонажи

Рика и Морти

Alien Morty



Jerry Smith



Abadango Cluster Princess



Abradolf Lincler



Требования

- Загружает список персонажей из API "Rick and Morty".
- Можно вернуться на главную страницу

[Home](#)

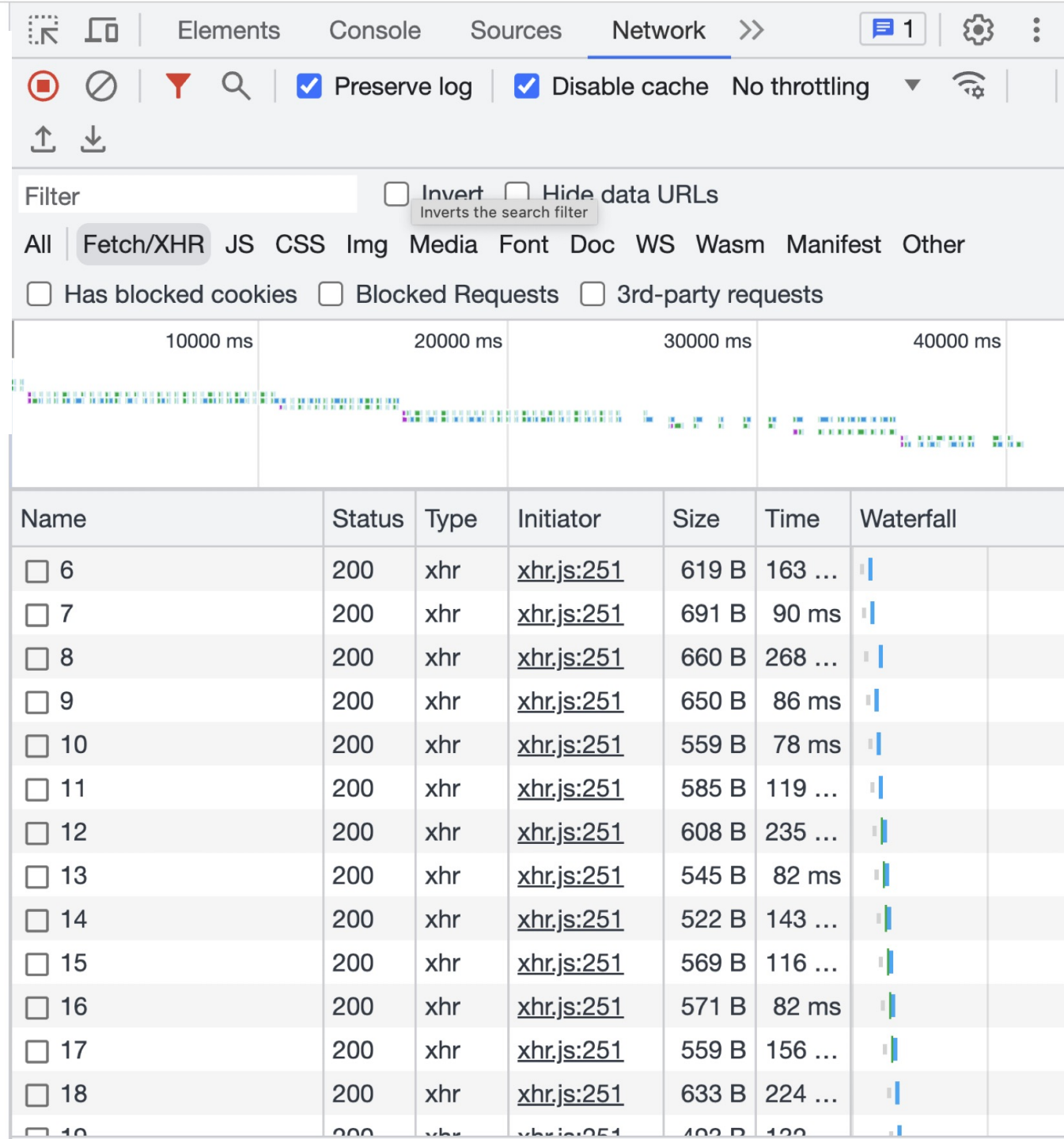
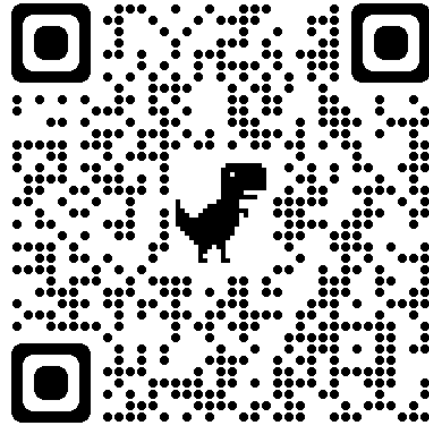
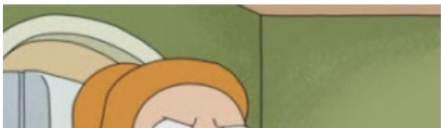
Rick Sanchez



Morty Smith



Summer Smith



главная

analytics

Рик и морти

Посты сломанные

Посты немного сломанные

Network

Filter ☐ Invert ☐ Hide data URLs

All Fetch/XHR JS CSS Img Media Font Doc WS Wasm Manifest Other

☐ Has blocked cookies ☐ Blocked Requests ☐ 3rd-party requests

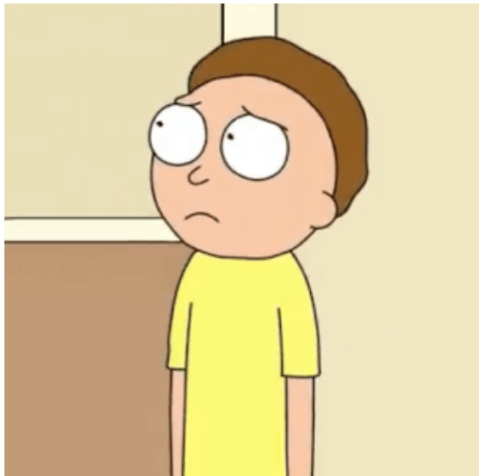
Name	Status	Type	Initiator	Size	Time	Waterfall
[] 72	200	xhr	xhr.js:251	507 B	220 ...	
[] 73	200	xhr	xhr.js:251	545 B	109 ...	
[] 74	200	xhr	xhr.js:251	656 B	442 ...	
[] 75	200	xhr	xhr.js:251	550 B	86 ms	
[] 76	200	xhr	xhr.js:251	571 B	84 ms	
[] 77	200	xhr	xhr.js:251	544 B	255 ...	
[] 78	200	xhr	xhr.js:251	588 B	161 ...	
[] 79	200	xhr	xhr.js:251	579 B	82 ms	
[] 80	200	xhr	xhr.js:251	562 B	214 ...	
[] 81	200	xhr	xhr.js:251	539 B	117 ...	
[] 82	200	xhr	xhr.js:251	648 B	96 ms	
[] 83	200	xhr	xhr.js:251	676 B	216 ...	
[] 84	200	xhr	xhr.js:251	527 B	119 ...	
[] 85	200	xhr	xhr.js:251	600 B	119 ...	

46 / 57 requests 26.9 kB / 396 kB transferred 21.7 kB / 389 kB resources

Rick Sanchez



Morty Smith



Summer Smith



Elements

Console

Sources

Network >>

Preserve log

Disable cache

No throttling

Filter

Invert

Hide data URLs

Hide ex...

All

Fetch/XHR

JS

CSS

Img

Media

Font

Doc

WS

Wasm

Blocked response cookies

Blocked requests

3rd-party request

2000 ms

4000 ms

6000 ms

8000 ms

10000 ms

12000 ms

Name	Sta...	Ty...	Initiator	S	T...	Waterfall	
<input type="checkbox"/> 47	200	xhr	main....	8.	2...		
<input type="checkbox"/> 28	200	xhr	main....	6.	1...		
<input type="checkbox"/> 48	200	xhr	main....	6.	5...		
<input type="checkbox"/> 29	200	xhr	main....	6.	4...		
<input type="checkbox"/> 49	200	xhr	main....	5.	1...		
<input type="checkbox"/> 30	200	xhr	main....	4.	4...		
<input type="checkbox"/> 31	200	xhr	main....	5.	4...		
<input type="checkbox"/> 50	200	xhr	main....	5.	2...		
<input type="checkbox"/> 32	200	xhr	main....	5.	8...		
<input type="checkbox"/> 51	200	xhr	main....	5.	1...		
<input type="checkbox"/> 33	200	xhr	main....	5.	1...		
<input type="checkbox"/> 52	200	xhr	main....	5.	1...		
<input type="checkbox"/> 34	200	xhr	main....	6.	5...		
<input type="checkbox"/> 35	200	xhr	main....	7.	1...		
<input type="checkbox"/> 53	200	xhr	main....	5.	2...		
<input type="checkbox"/> 36	200	xhr	main....	4.	4...		

Как можно это предотвратить?

- Добавить функцию очистки, чтобы правильно отменять все запросы при размонтировании компонента

Как можно это предотвратить?

- Добавить функцию очистки, чтобы правильно отменять все запросы при размонтировании компонента
- Рассмотреть возможность пагинации или ленивой загрузки(переделать апи)

Исправление (Отмена запросов после размонтирование компонента)

```
const source : CancelTokenSource = axios.CancelToken.source();
useEffect( effect: () => {
  no usages  👤 dragin96 *
  const fetchCharacters = async () : Promise<void> => {...};
  fetchCharacters();

  return () : void => {
    source.cancel( message: "Component unmounted");
  };
}, deps: []);
```



Выводы по кейсам

Если логика кода кажется
прямолинейной, могут возникнуть
неожиданное поведение



Выводы по кейсам

Если логика кода кажется
прямолинейной, могут возникнуть
неожиданное поведение

Всегда следите за сетевой
активностью, особенно при
изменение/обновление состояние



Выводы по кейсам

Если логика кода кажется
прямолинейной, могут возникнуть
неожиданное поведение

Всегда следите за сетевой
активностью, особенно при
изменении/обновлении состояние

Современные фреймворк
(react/vue/angular) облегчают
работу
с состояниями, но надо быть
внимательным при тестировании



Выводы по кейсам

Если логика кода кажется прямолинейной, могут возникнуть неожиданное поведение

Всегда следите за сетевой активностью, особенно при изменении/обновление состояние

Современные фреймворк (react/vue/angular) облегчают работу с состояниями, но надо быть внимательным при тестировании

Автотесты не всегда смогут выявлять такие проблемы.



Поговорим о проблеме с памятью



Утечки памяти



самолет



Симптомы

Замедление работы
приложения

Повышенное
потребление ресурсов

Краши браузера или
приложения



самолет




Опаньки...

При загрузке этой страницы возникли неполадки.

Код ошибки: 5

[Подробнее](#)















Перезагрузить

Мониторинг системы	
Мои процессы	
Имя процесса	Пам...
Google Chrome Helper (GPU)	471,5 МБ
Google Chrome Helper (Renderer)	400,2 МБ
Google Chrome Helper (Renderer)	313,6 МБ
 Google Chrome	300,4 МБ
Google Chrome Helper (Renderer)	265,9 МБ
Google Chrome Helper (Renderer)	251,9 МБ
Google Chrome Helper (Renderer)	135,9 МБ
Google Chrome Helper (Renderer)	104,0 МБ
Google Chrome Helper (Renderer)	89,1 МБ
Google Chrome Helper (Renderer)	73,3 МБ
Google Chrome Helper (Renderer)	65,0 МБ
Google Chrome Helper (Renderer)	61,4 МБ
Google Chrome Helper (Renderer)	59,9 МБ
Google Chrome Helper (Renderer)	54,6 МБ
Google Chrome Helper (Renderer)	52,8 МБ
Google Chrome Helper	49,3 МБ
Google Chrome Helper (Renderer)	47,8 МБ
Google Chrome Helper (Renderer)	36,6 МБ



самолет

Дополнительные инструменты -> диспетчер задач

Диспетчер задач				
Задача	Объем потребляемой памяти ▾	Процессор	Сеть	Идентификатор процесса
 Процесс GPU	344 МБ	14.0	0	787
 Браузер	296 МБ	1.4	0	779
 Вкладка: Notebook List view	179 МБ	0.2	0	828
 Вкладка: Google	147 МБ	0.0	0	825
 Вкладка: Яндекс Музыка — собираем н	121 МБ	5.7	0	837
 Вкладка: Receive SMS	96,2 МБ	0.0	0	811
 Вкладка: chatgpt - Поиск в Google	83,4 МБ	0.0	0	826
Субфрейм: https://google.com/				
 Вкладка: sim online - Поиск в Google	81,5 МБ	0.0	0	813
 Расширение: uBlock Origin	81,1 МБ	0.0	0	799
 Вкладка: ChatGPT	68,3 МБ	0.0	0	795
 Расширение: Browsec VPN - бесплатн	59,3 МБ	0.0	0	800
 Вкладка в режиме инкогнито: DevTools	57,6 МБ	0.0	0	16107
 Вкладка: React App	53,8 МБ	0.1	0	829
 Утилита: Network Service	44,8 МБ	0.4	0	788
Субфрейм: https://accounts.google.com				
Субфрейм: https://accounts.google.com				

Завершить процесс

Как работает память в js?

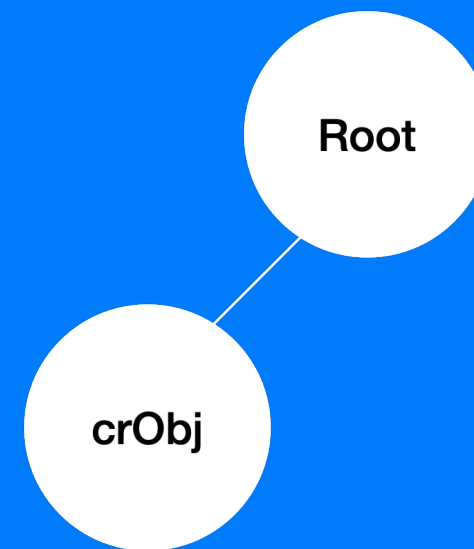


Как устроена память в js

- Выделение памяти
- Освобождение памяти: Сборка мусора (GC)

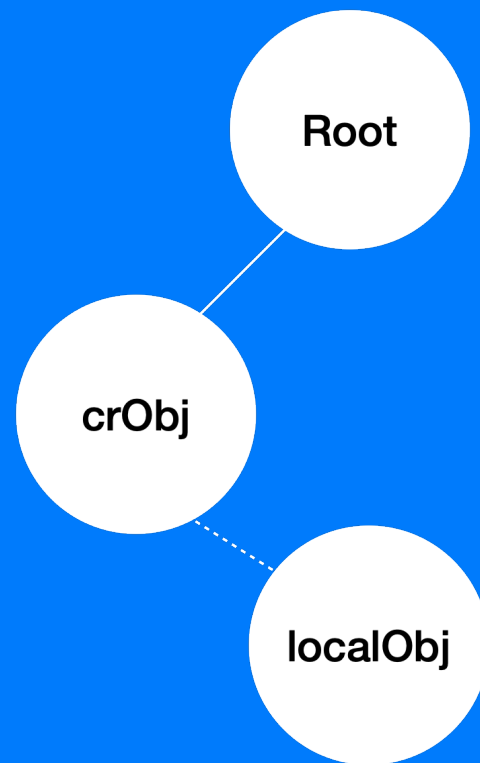
Выделение памяти

```
const crObj = {} => {  
  let localObj = {  
    name: "Temporary"  
  };  
}
```



Освобождение памяти

```
const crObj = () => {  
  let localObj = {  
    name: "Temporary"  
  };  
}  
crObj();
```



самолет



А как случаются
проблемы с памятью?



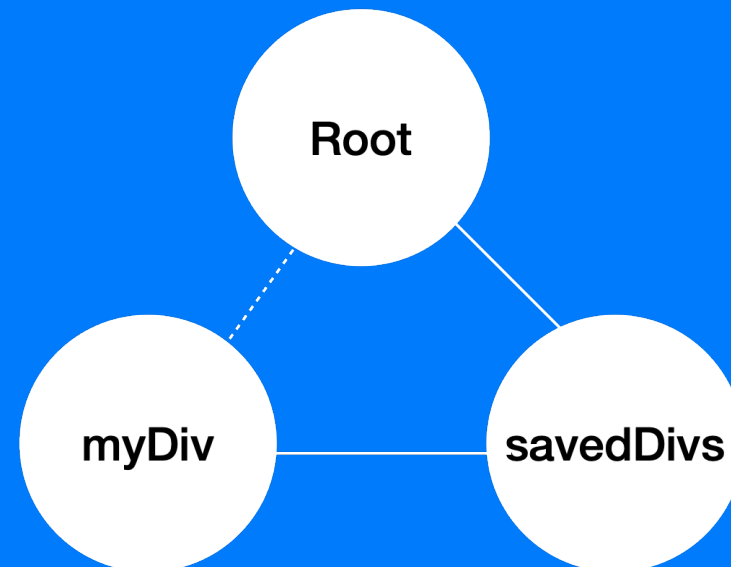
Выделение памяти

```
let savedDivs = [];  
let myDiv = document.getElementById("myDiv");
```



Освобождение памяти

```
let savedDivs = [];  
let myDiv = document.getElementById("myDiv");  
savedDivs.push(myDiv);  
myDiv.remove();
```



Проблемы с памятью в js —
это память, которая остается
занятой после того, как она
становится ненужной

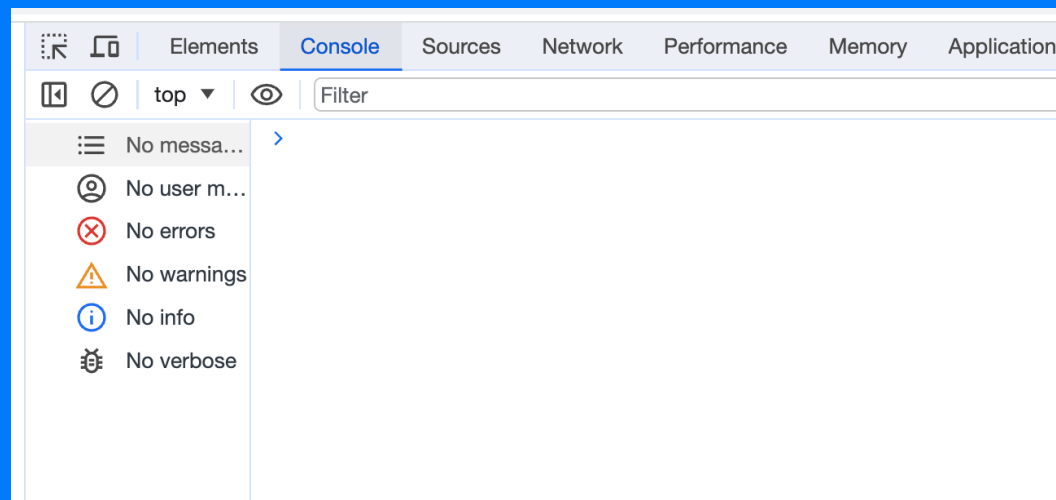


Как такое находить?

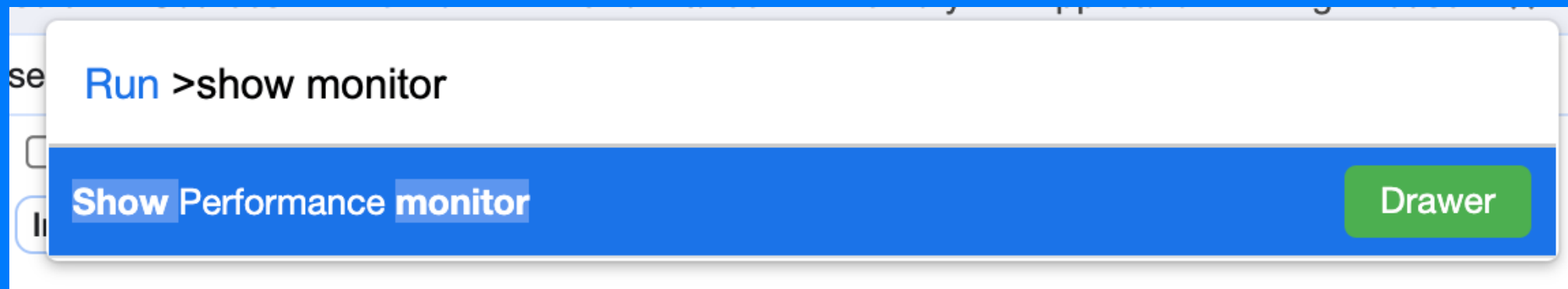


самолет

Performance Monitor



Ctrl+Shift+I или $\text{⌘}+\text{⌘}+\text{I}$



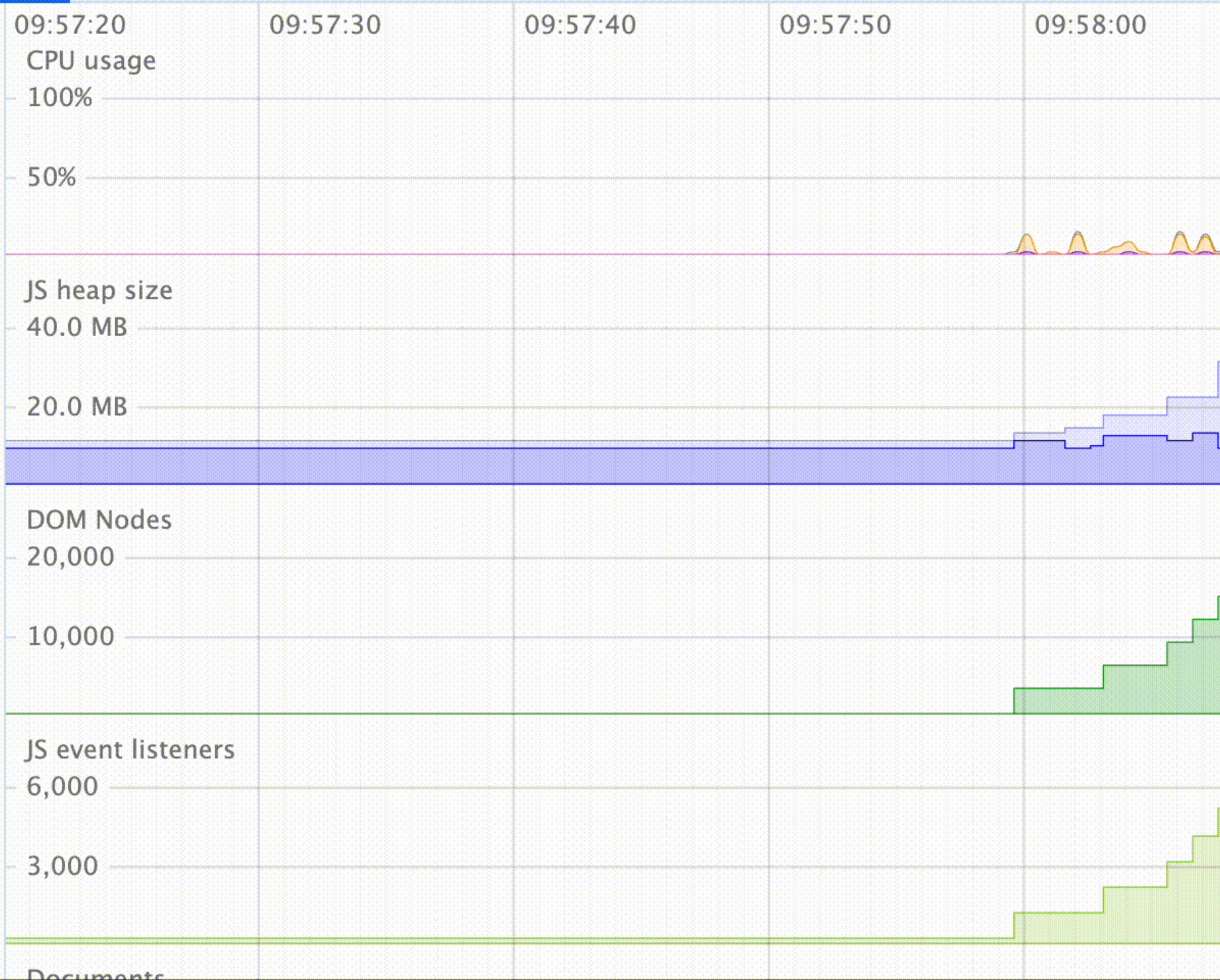
Shift+cmd+p

И вводим show monitor



самолет

- ✓ CPU usage 0.8%
- ✓ JS heap size 31.2 MB
- ✓ DOM Nodes 3,128
- ✓ JS event listeners 1,160
- ✓ Documents 1
- ✓ Document Frames 1
- ✓ Layouts / sec
- ✓ Style recalcs / sec



(i) No info		
⋮ Console Performance monitor X		
✓ CPU usage	0%	2:59:10 CPU usage
✓ JS heap size	11.5 MB	100% —
✓ DOM Nodes	160	50% —
✓ JS event listeners	164	
✓ Documents	5	JS heap si 20.0 MB —
✓ Document Frames	5	10.0 MB —
✓ Layouts / sec		
✓ Style recalcs / sec		DOM Nod 200 —



(i) No info		
⋮ Console		Performance monitor X
✓ CPU usage	0%	2:59:10 CPU usage
✓ JS heap size	11.5 MB	100%
✓ DOM Nodes	160	50%
✓ JS event listeners	164	
✓ Documents	5	JS heap si 20.0 MB
✓ Document Frames	5	10.0 MB
✓ Layouts / sec		
✓ Style recalcs / sec		DOM Nod 200



(i) No info		
⋮ Console		Performance monitor X
✓ CPU usage	0%	2:59:10 CPU usage
✓ JS heap size	11.5 MB	100% —
✓ DOM Nodes	160	50% —
✓ JS event listeners	164	
✓ Documents	5	JS heap si 20.0 MB
✓ Document Frames	5	10.0 MB
✓ Layouts / sec		
✓ Style recalcs / sec		DOM Nod 200



(i) No info		
⋮	Console	Performance monitor X
✓ CPU usage	0%	2:59:10 CPU usage
✓ JS heap size	11.5 MB	100% —
✓ DOM Nodes	160	50% —
✓ JS event listeners	164	
✓ Documents	5	JS heap si 20.0 MB —
✓ Document Frames	5	10.0 MB —
✓ Layouts / sec		
✓ Style recalcs / sec		DOM Nod 200 —



— А что дальше?

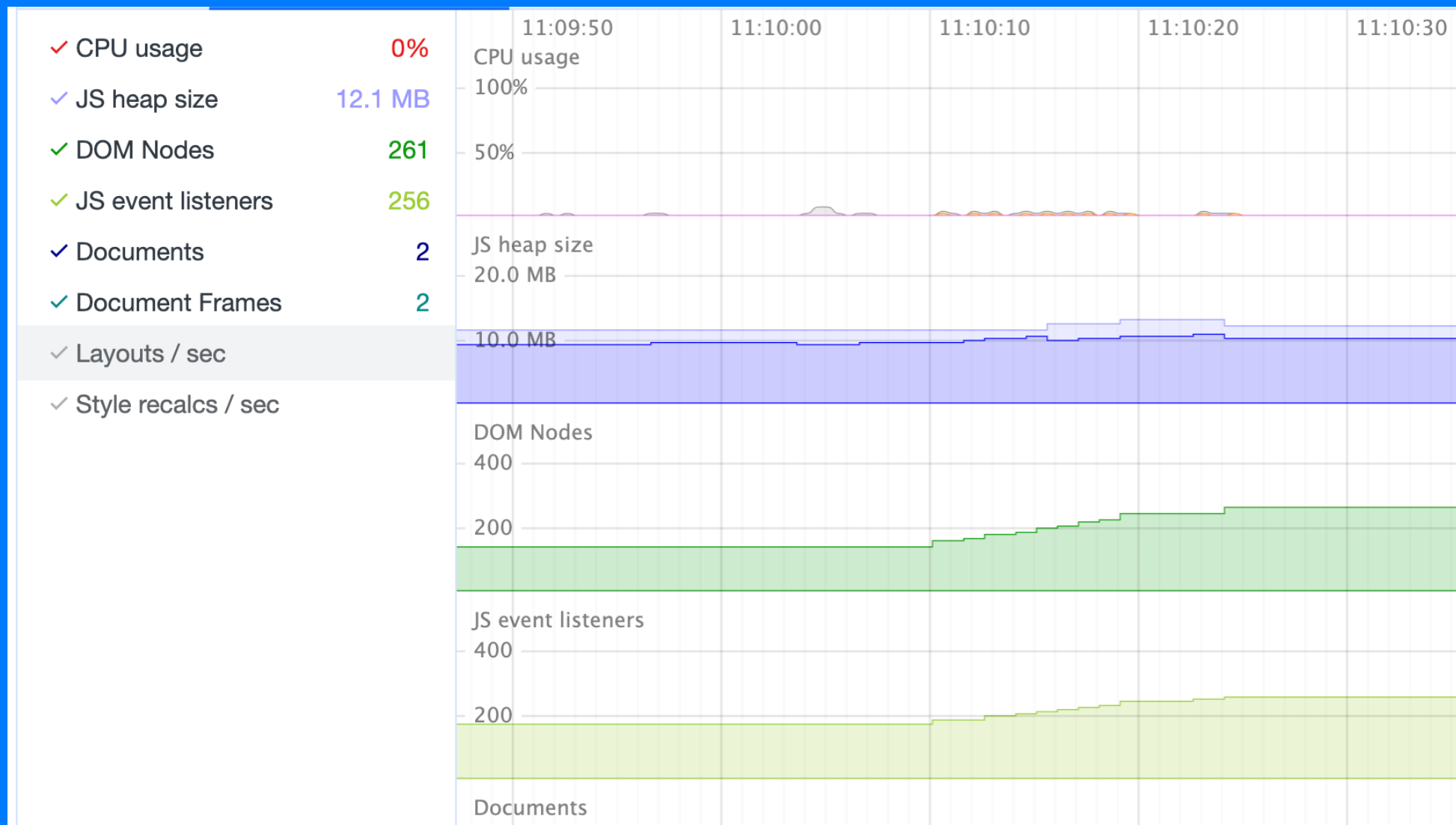


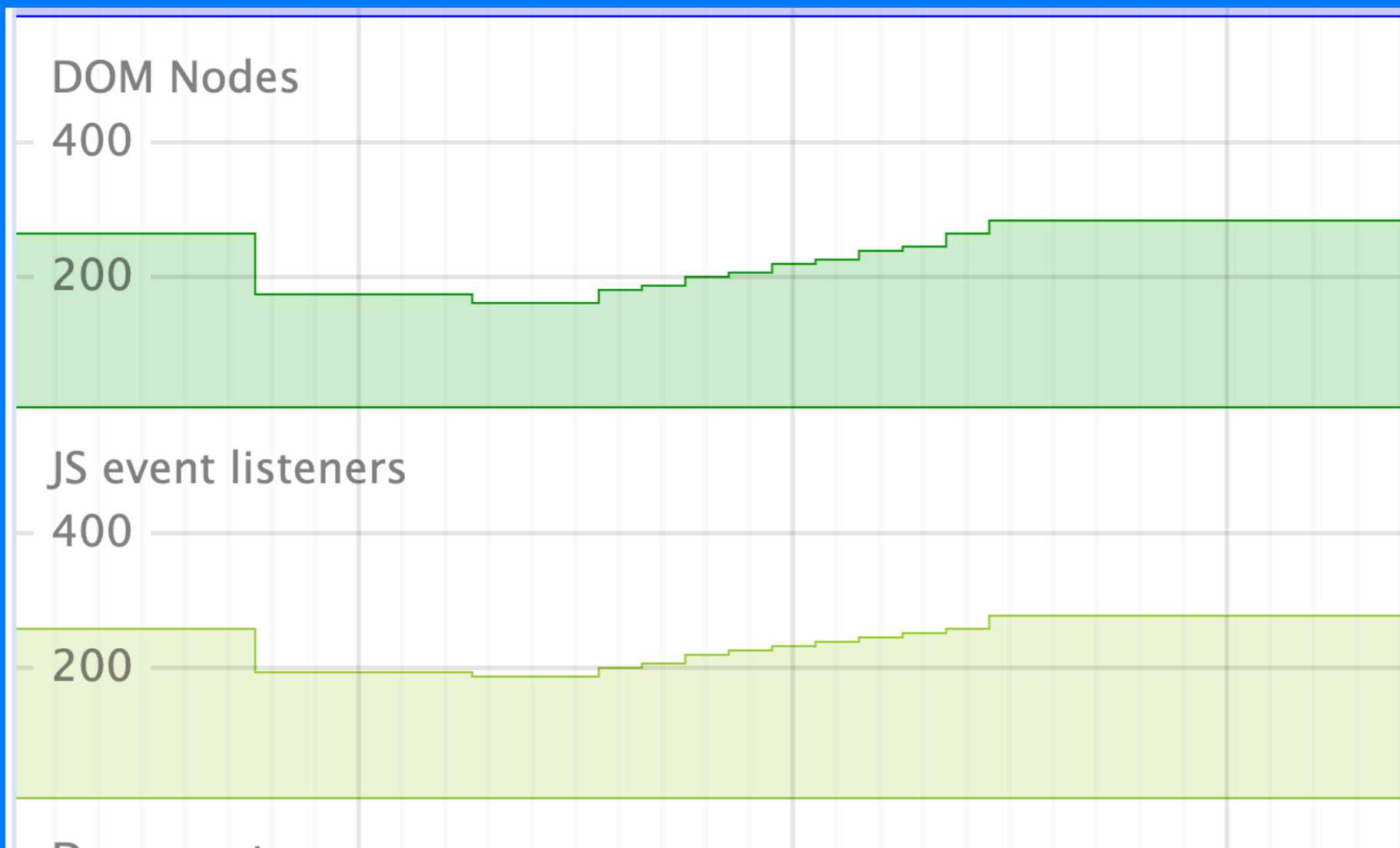
самолет

— А что дальше?

— Выполняем свои обычные тест кейсы, и если график растёт, и не возвращается к своему начальному состоянию, то есть проблема

Есть ли тут утечка?





Достоинства и Ограничения "Performance Monitor"

+Плюсы:

- **Реальное Время:** Мгновенный мониторинг и анализ данных производительности.
- **Широкий Спектр Метрик:** От CPU и памяти до счетчиков DOM и слушателей событий
- Не замедляет приложение



Достоинства и Ограничения "Performance Monitor"

+Плюсы:

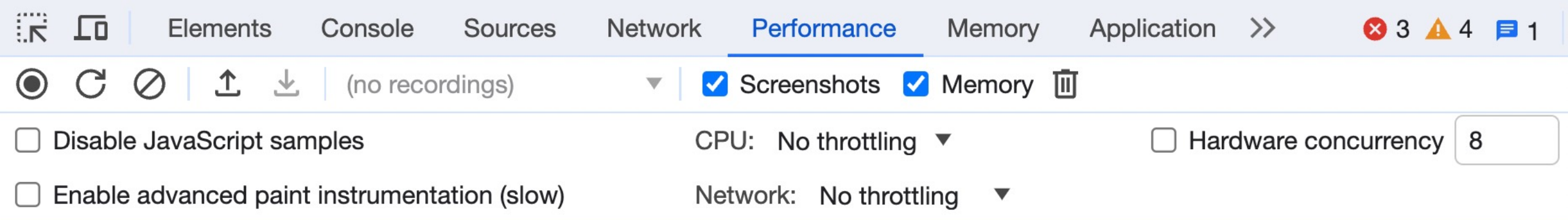
- **Реальное Время:** Мгновенный мониторинг и анализ данных производительности.
- **Широкий Спектр Метрик:** От CPU и памяти до счетчиков DOM и слушателей событий
- Не замедляет приложение


- Минусы:

- **Ограниченный анализ:** Не предоставляет глубокий уровень детализации
- **Отсутствие Истории:** Нет возможности сохранить или экспортировать данные для долгосрочного анализа.

Вкладка performance



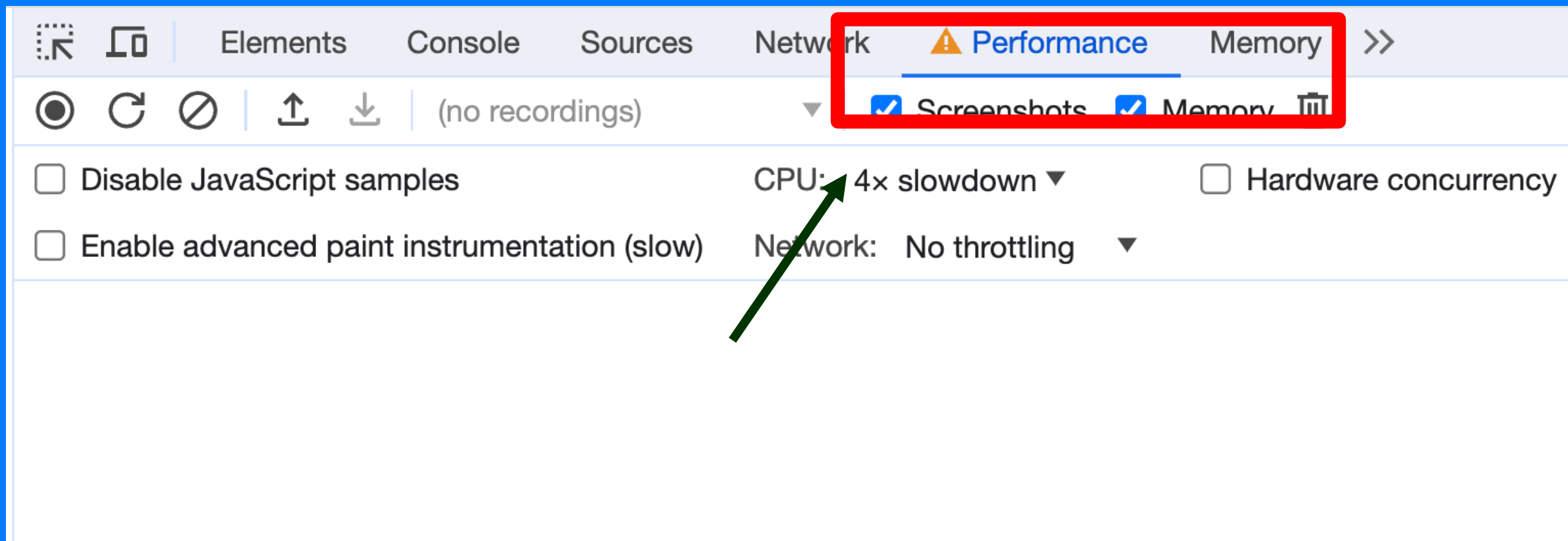


Click the record button  or hit **⌘ E** to start a new recording.

Click the reload button  or hit **⌘ ⇧ E** to record the page load.

After recording, select an area of interest in the overview by dragging. Then, zoom and pan the timeline with the mousewheel or **WASD** keys. [Learn more](#)

Вкладка performance



Какие Проблемы Можем Найти с Помощью 'Performance'?

Задержки в Отрисовке

Медленная отрисовка элементов и анимаций.

Высокая Нагрузка на CPU

Избыточные вычислительные операции, задержки в обработке событий.



Какие Проблемы Можем Найти с Помощью 'Performance'?

Задержки в Отрисовке

Медленная отрисовка элементов и анимаций.

Высокая Нагрузка на CPU

Избыточные вычислительные операции, задержки в обработке событий.

Проблемы с Памятью

Неправильное освобождение памяти, утечки.

Неоптимальное Использование Сети

Множественные перезапросы, долгое ожидание ответа от сервера.



Какие Проблемы Можем Найти с Помощью 'Performance'?

Задержки в Отрисовке

Медленная отрисовка элементов и анимаций.

Высокая Нагрузка на CPU

Избыточные вычислительные операции, задержки в обработке событий.

Проблемы с Памятью

Неправильное освобождение памяти, утечки.

Неоптимальное Использование Сети

Множественные перезапросы, долгое ожидание ответа от сервера.

Ненужный или Медленный JavaScript/CSS

Большие блоки неиспользуемого или медленно работающего кода.

Плохая Отзывчивость

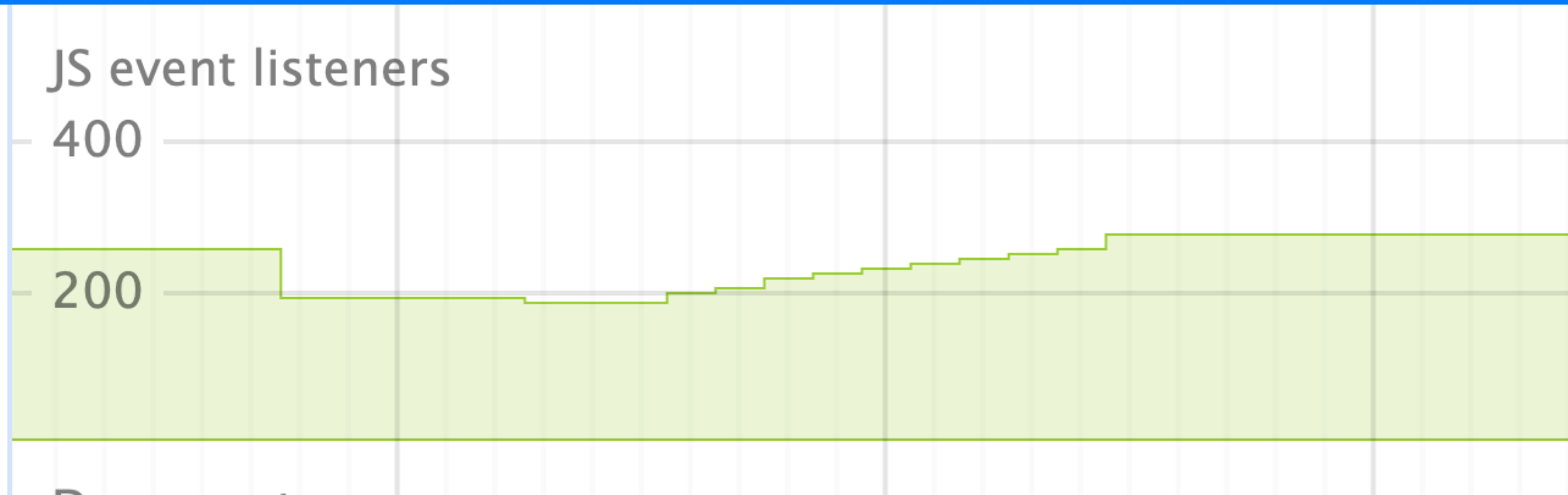
Задержки при обработке событий ввода (клики, скролл, ввод текста).



JS event listeners

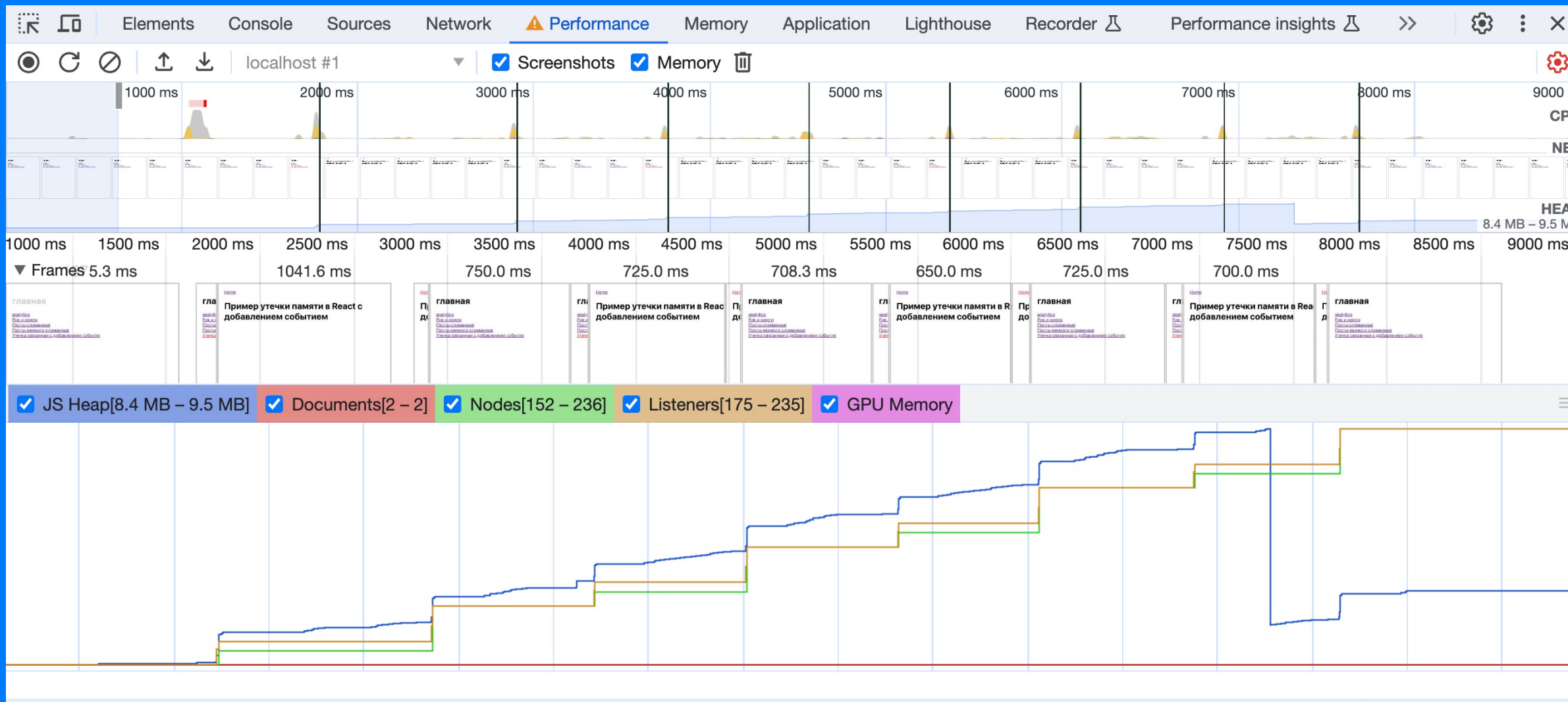
400

200

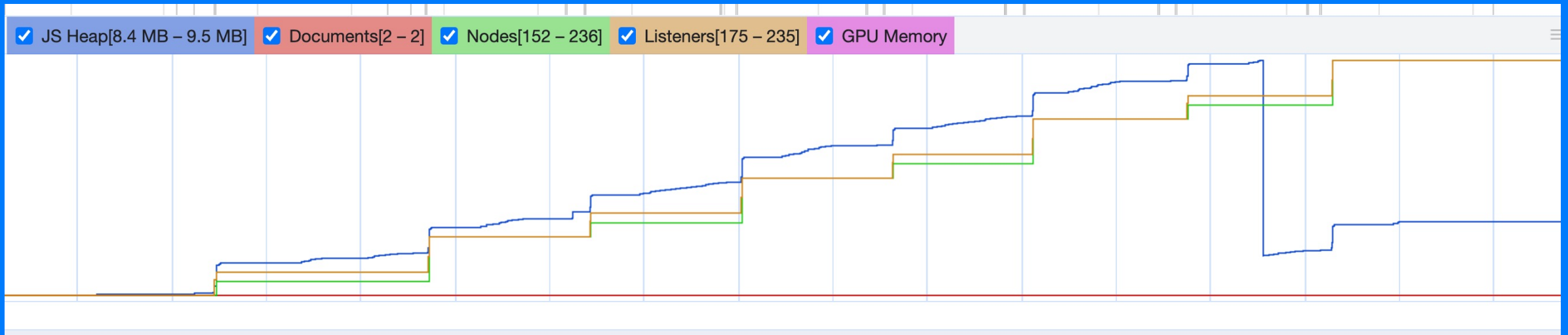


самолет

Запускаем профайлер и проходим сценарий возможной утечки

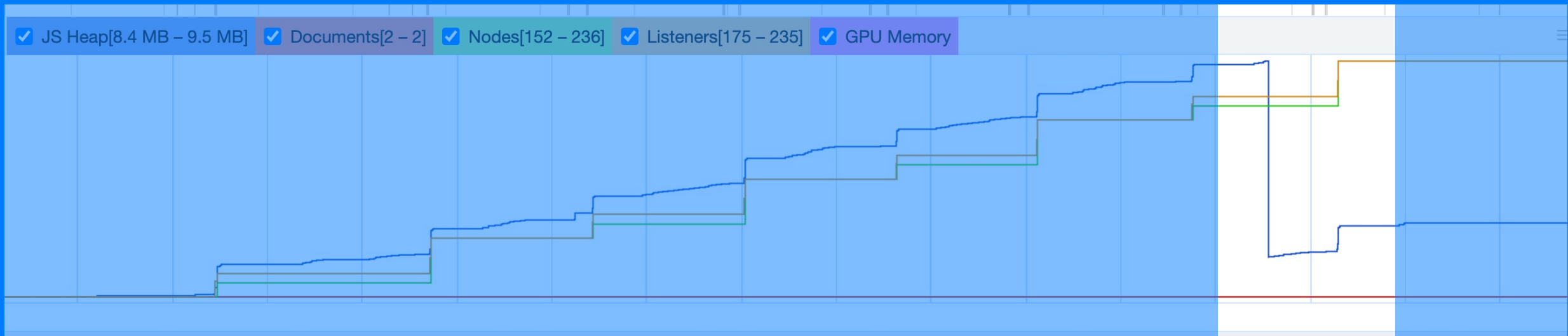


Запускаем профайлер и проходим сценарий возможной утечки



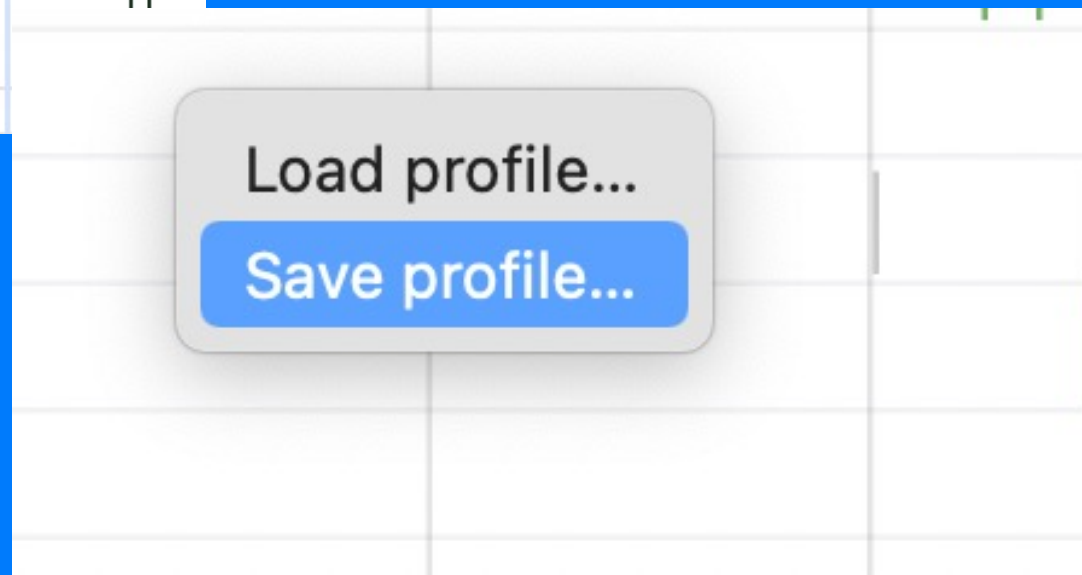
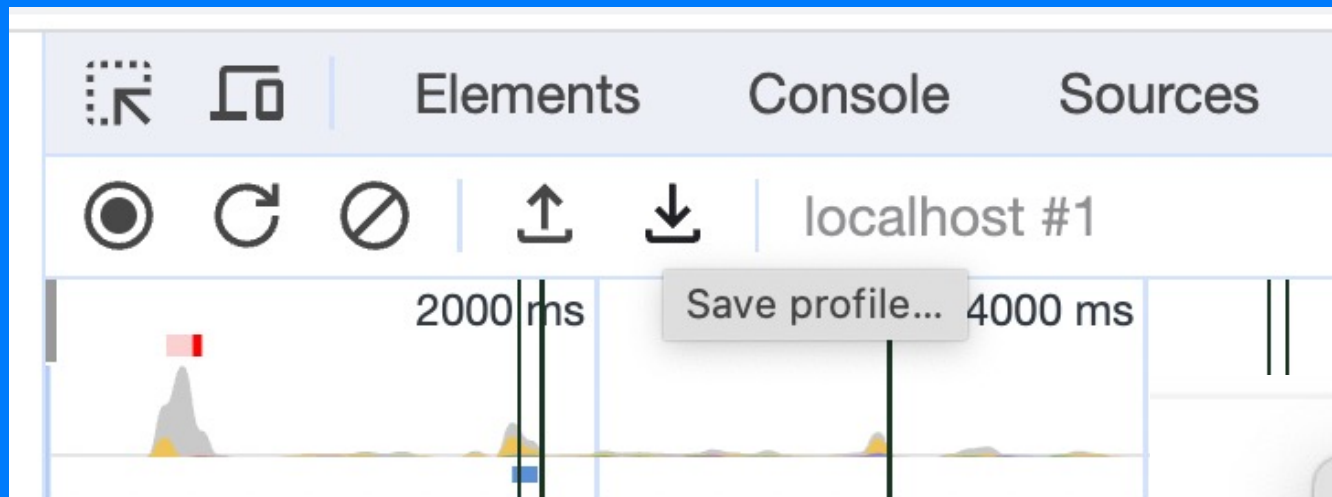
самолет

Запускаем профайлер и проходим сценарий возможной утечки



самолет

Выгружаем профиль



**И составляем баг репорт,
прикрепляем результаты из выгрузки**

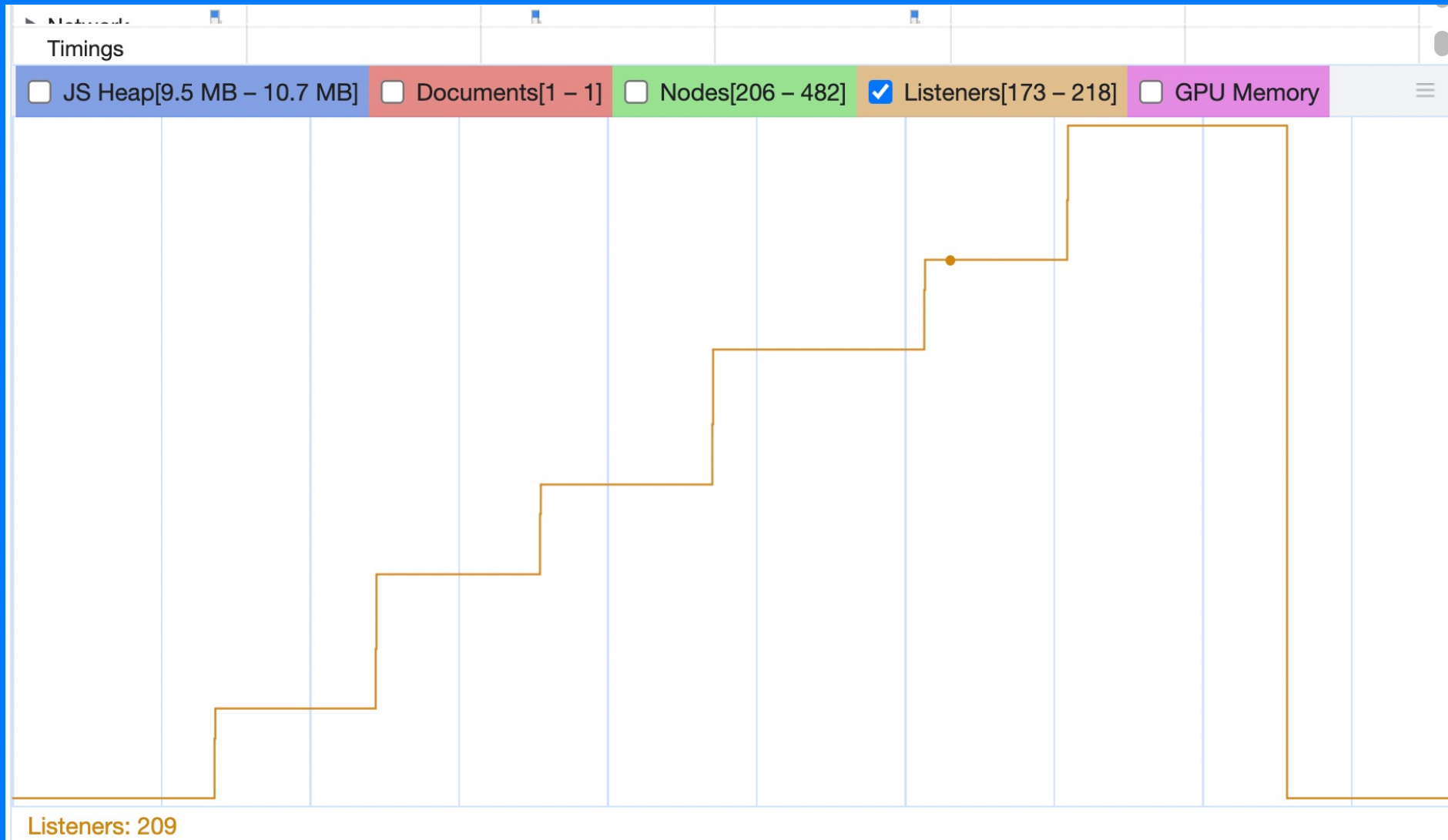


Небольшой интерактив

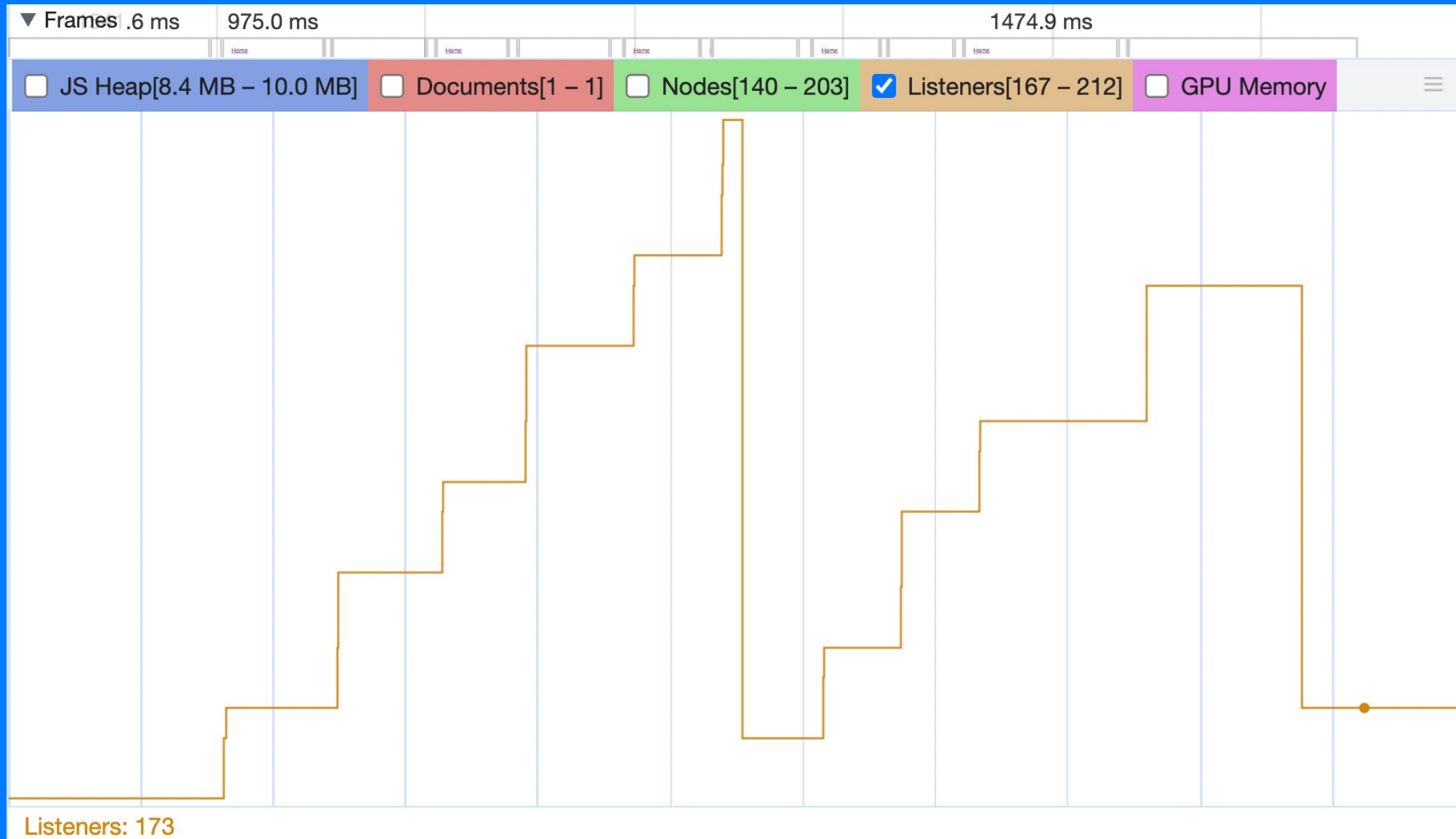


самолет

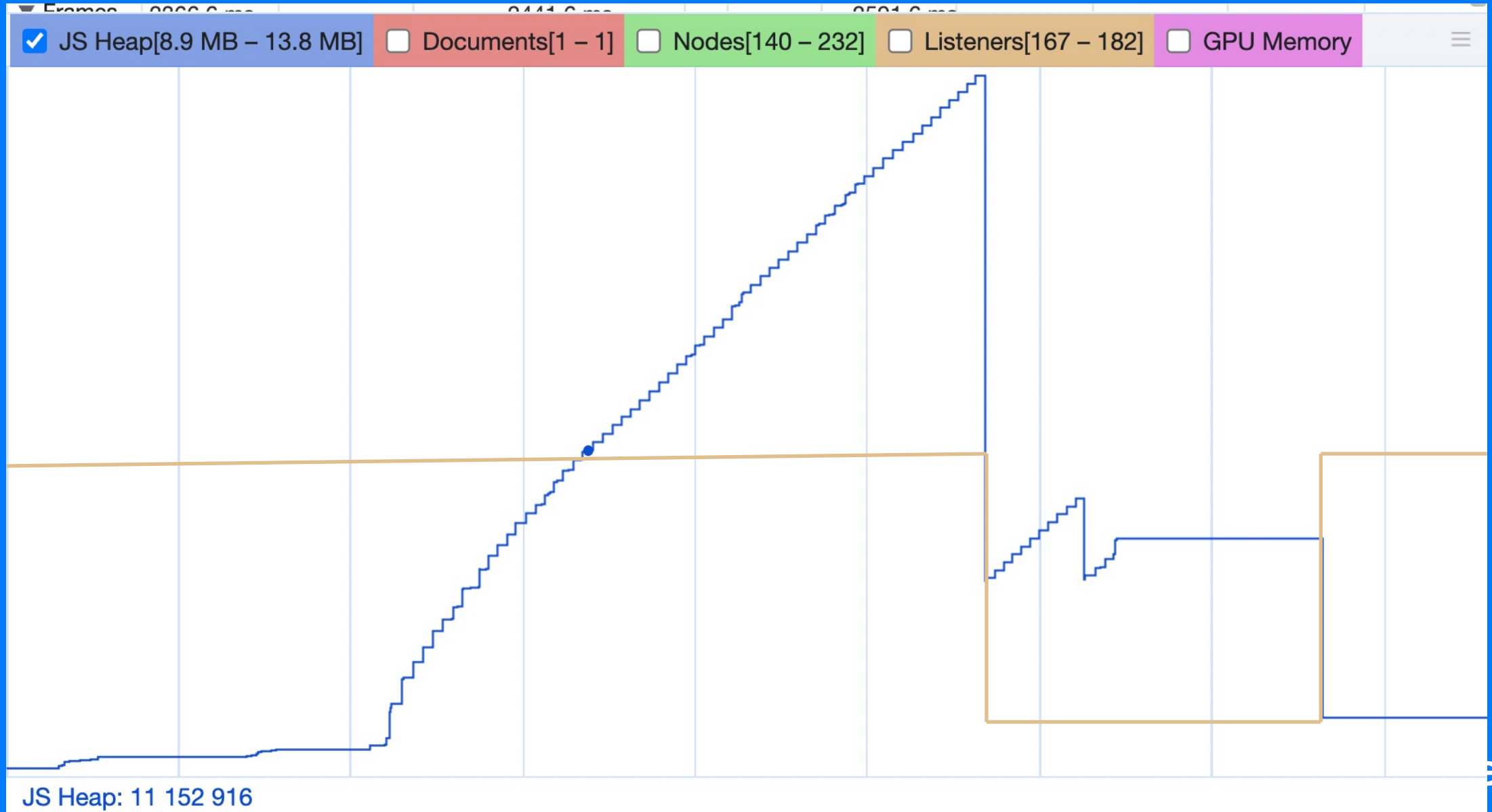
Тут есть утечка #1?



Тут есть утечка #2?



Тут есть утечка #3?



ET

Когда Использовать Вкладку 'Performance'

1. После 'Performance Monitor'
2. Детализация Проблем
3. Профилирование Кода
4. Для баг репорта



Вкладка memory



самолет



Profiles

Select profiling type

☒ Heap snapshot

Heap snapshot profiles show memory distribution among your page's JavaScript objects and related DOM nodes.

☐ Include numerical values in capture

☐ Allocation instrumentation on timeline

Allocation timelines show instrumented JavaScript memory allocations over time. Once profile is recorded you can select a time interval objects that were allocated within it and still alive by the end of recording. Use this profile type to isolate memory leaks.

☒ Record stack traces of allocations (extra performance overhead)

☐ Allocation sampling

Record memory allocations using sampling method. This profile type has minimal performance overhead and can be used for long running operations. It provides good approximation of allocations broken down by JavaScript execution stack.

Select JavaScript VM instance

7.6 MB ↑1.2 kB/s localhost:3000: Main

7.6 MB ↑1.2 kB/s Total JS heap size

Take snapshot

Load

Режимы работы профайлера

- Heap Snapshots

Режимы работы профайлера

- Heap Snapshots
- Allocation Instrumentation on Timeline



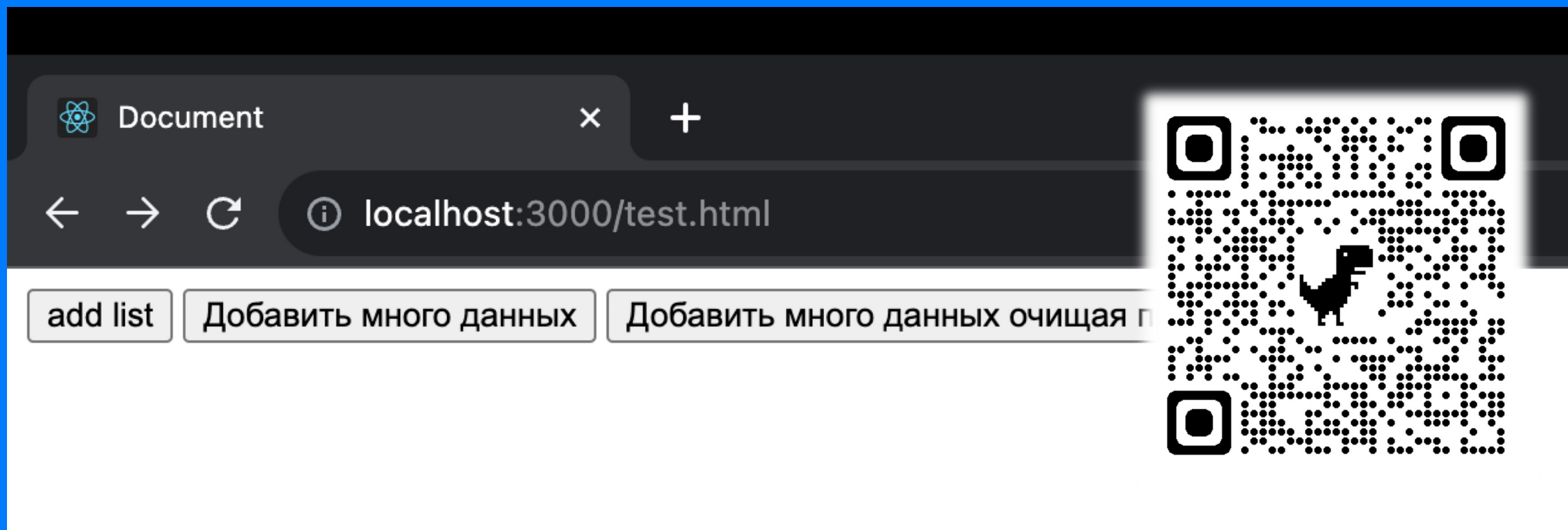
Режимы работы профайлера

- Heap Snapshots
- Allocation Instrumentation on Timeline
- Allocation Sampling



Анализ Heap Snapshot для выявления проблем с памятью

1. открываем тестируемый сайт и девтулз на вкладке memory



Анализ Heap Snapshot для выявления проблем с памятью

1. Открываем тестируемый сайт и девтулз на вкладке memory
2. Нажмите кнопку "Take snapshot" для создания начального снимка кучи.

Take heap snapshot

HEAP SNAPSHOTS

Snapshot 1

1.7 MB

Save

Summary

Class filter

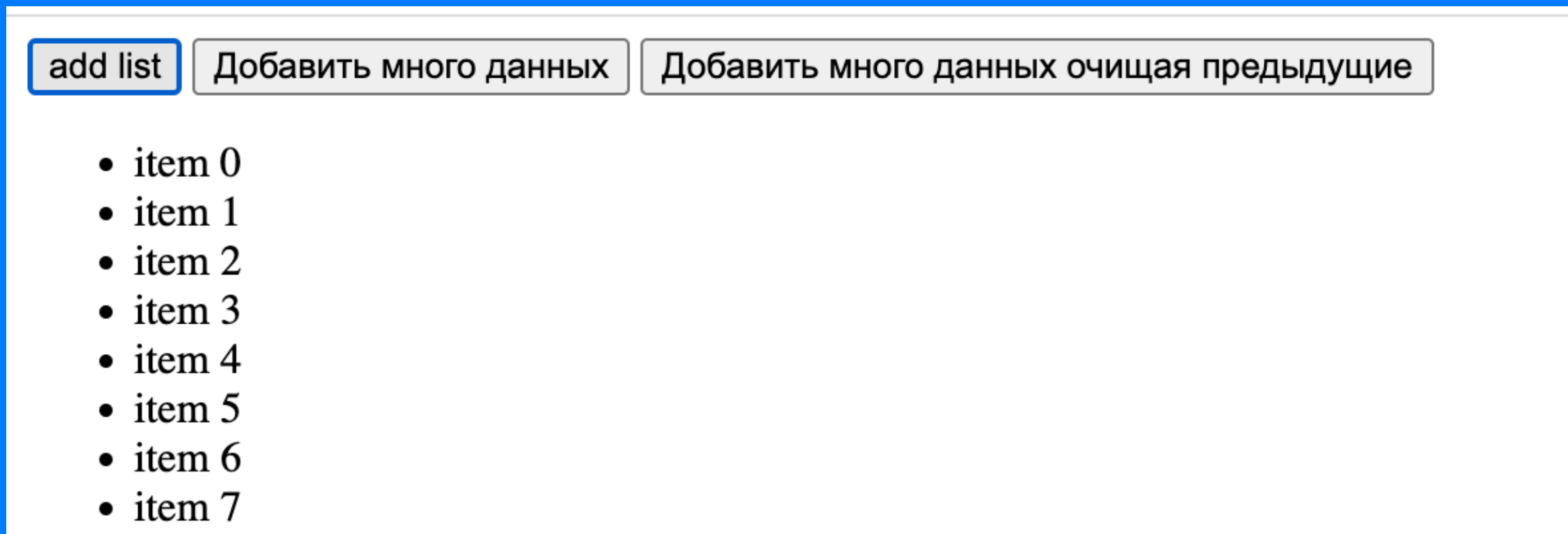
All objects

Constructor	Distance	Shallow Size	Retained Size
▶ (system) ×11905	2	446 184 26 %	634 552 37 %
▶ system / Context ×66	3	1 936 0 %	427 360 25 %
▶ Window / http://localhost:3000	1	36 0 %	421 004 24 %
▶ (array) ×213	2	285 088 17 %	404 184 24 %
▶ (closure) ×2934	2	83 824 5 %	317 408 18 %
▶ (compiled code) ×4910	3	283 060 16 %	313 844 18 %
▶ Array ×29	3	464 0 %	193 604 11 %
▶ (concatenated string) ×8506	5	170 120 10 %	183 832 11 %
▶ Window /	1	36 0 %	156 144 9 %
▶ (object shape) ×2242	2	151 044 9 %	154 900 9 %
▶ (string) ×5157	3	132 660 8 %	132 780 8 %



Анализ Heap Snapshot для выявления проблем с памятью

1. Открываем тестируемый сайт и девтулз на вкладке memory
2. Нажмите кнопку "Take snapshot" для создания начального снимка кучи.
3. Проходим сценарий предполагающий утечку



Анализ Heap Snapshot для выявления проблем с памятью

1. Открываем тестируемый сайт и девтулз на вкладке memory
2. Нажмите кнопку "Take snapshot" для создания начального снимка кучи.
3. Проходим сценарий предполагающий утечку
4. Снова нажмите "Take snapshot" для создания второго снимка кучи.

И вот у нас результат

<div><div><div></div><div></div><div></div></div></div>	Summary ▼	Class filter	All objects ▼	
Profiles	Constructor	Distance	Shallow Size	Retained Size
HEAP SNAPSHOTS	▶ (system) x54734	2	1 729 308 23 %	2 729 368 37 %
Snapshot 1 7.4 MB	▶ (array) x2708	2	1 087 560 15 %	2 110 664 29 %
	▶ (closure) x18449	–	542 724 7 %	1 845 324 25 %
	▶ (compiled code) x32267	–	1 632 768 22 %	1 842 312 25 %
Snapshot 2 7.4 MB	▶ system / Context x2288	3	66 096 1 %	1 833 772 25 %
<div>Save</div>	▶ Object x2650	2	66 904 1 %	1 406 840 19 %
	▶ (object shape) x12607	2	1 008 816 14 %	1 116 748 15 %
	▶ (string) x24365	2	826 828 11 %	826 908 11 %
	▶ InternalNode x1323	–	0 0 %	441 980 6 %
	▶ Window x50	2	3 376 0 %	434 108 6 %
	▶ Window / http://localhost:3000	1	36 0 %	296 868 4 %
	▶ EventListener x20	7	1 584 0 %	293 052 4 %
	▶ V8EventListener x16	8	640 0 %	290 472 4 %
	▶ HTMLDocument x38	2	6 728 0 %	248 800 3 %



самолет

Что можно посмотреть?

<div><div><div></div><div></div><div></div></div></div>		Comparison ▼	Class filter	Snapshot 1 ▼				
Profiles		Constructor	# New	# Deleted	# Delta	Alloc. Size	Freed Size	Size Delta
HEAP SNAPSHOTS		▶ Detached HTMLLIElement	1 000	0	+1 000	124 000	0	+124 000
Snapshot 1 1.1 MB		▶ (object shape)	46	7	+39	8 304	256	+8 048
Snapshot 2 1.2 MB		▶ (array)	9	6	+3	3 052	14 112	−11 060
Save		▶ (system)	60	0	+60	1 284	0	+1 284
		▶ (compiled code)	20	1	+19	768	16	+752
		▶ (closure)	17	0	+17	504	0	+504
		▶ (string)	7	0	+7	188	0	+188
		▶ Detached HTMLULListElem...	1	0	+1	124	0	+124
		▶ PerformanceEventTiming	1	0	+1	120	0	+120
		▶ HTMLLIElement	3	0	+3	84	0	+84
		▶ HTMLULListElement	3	0	+3	84	0	+84
		▶ PointerEvent	3	0	+3	84	0	+84
		▶ Range	1	0	+1	80	0	+80
		▶ system / Context	2	0	+2	40	0	+40



Allocation Timeline



Profiles

Select profiling type

☐ Heap snapshot

Heap snapshot profiles show memory distribution among your page's JavaScript objects and related DOM nodes.

☐ Include numerical values in capture

☒ Allocation instrumentation on timeline

Allocation timelines show instrumented JavaScript memory allocations over time. Once profile is recorded you can select a time interval to see objects that were allocated within it and still alive by the end of recording. Use this profile type to isolate memory leaks.

☒ Record stack traces of allocations (extra performance overhead)

☐ Allocation sampling

Record memory allocations using sampling method. This profile type has minimal performance overhead and can be used for long running operations. It provides good approximation of allocations broken down by JavaScript execution stack.



Allocation Timeline

add list Добавить много данных Добавить много данных очищая предыдущие

Profiles

ALLOCATION TIMELINES

Snapshot 1
Recording...

Summary 5.00 s 10.00 s

1.0 kB

Constructor

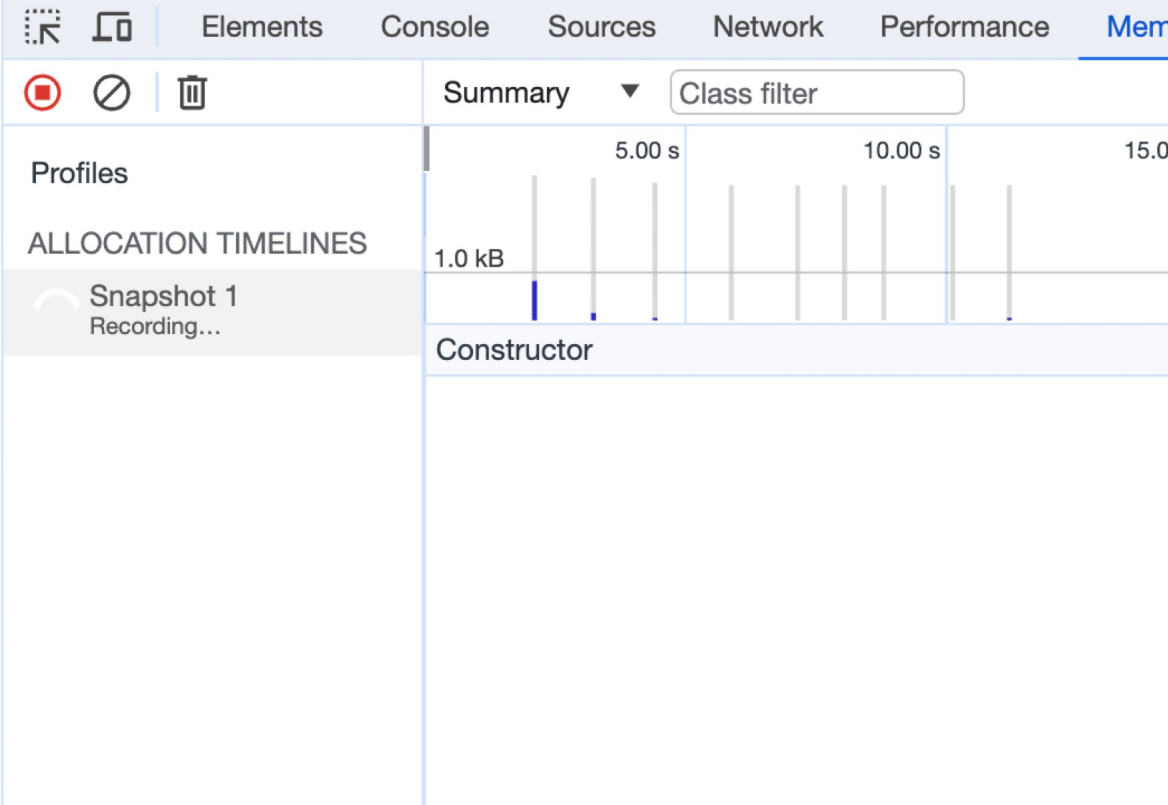


самолет

Allocation Timeline

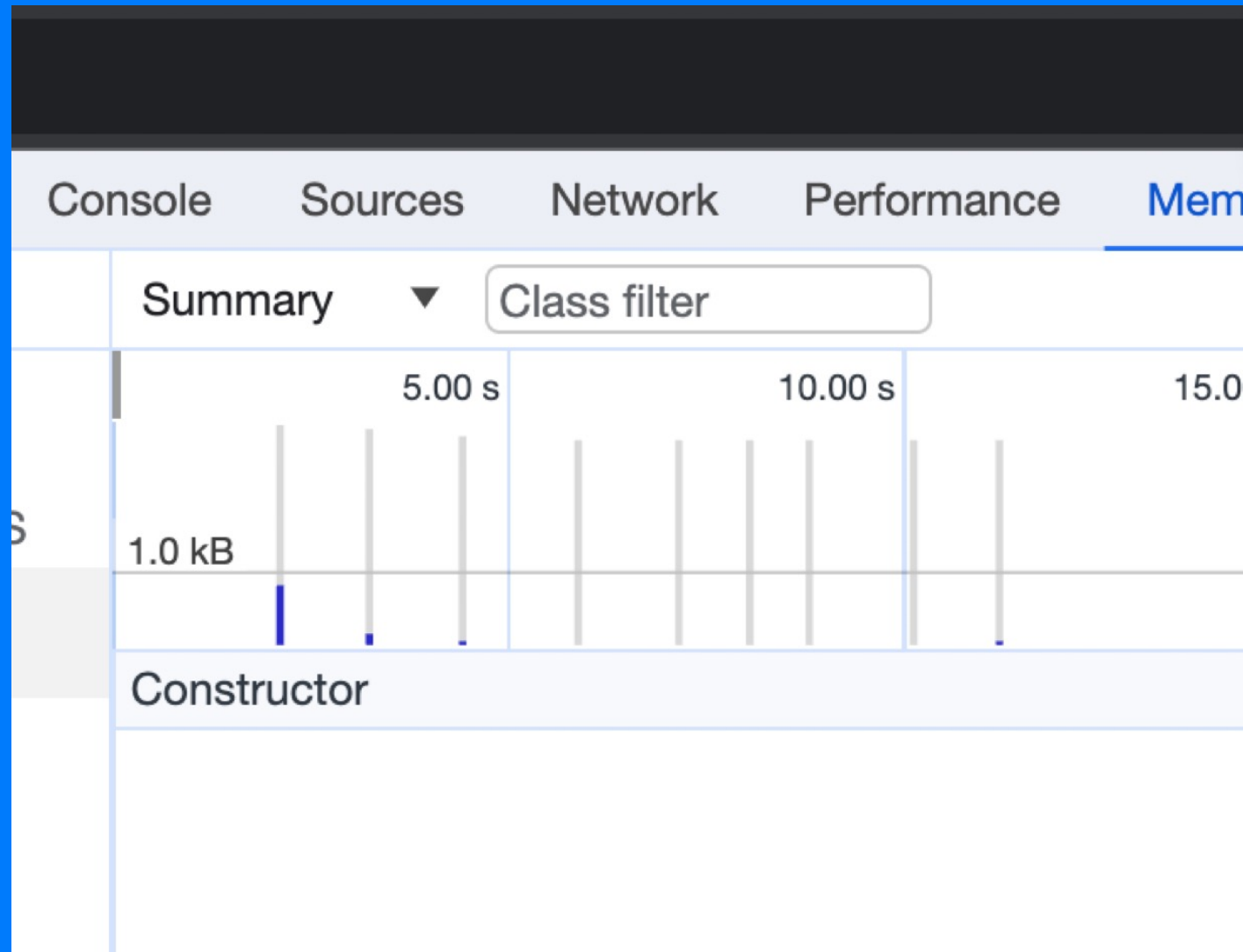
← → ↻ ⓘ localhost:3000/test.html

add list Добавить много данных **Добавить много данных очищая предыдущие**



самолет

Allocation Timeline






Какие проблемы можно найти с помощью Allocation Timeline

- Частые выделения и освобождения
- Memory Leak
- Интенсивные операции




Allocation Sampling



Profiles

SAMPLING PROFILES

 Profile 1
Recording...

Select profiling type

☐ Heap snapshot
Heap snapshot profiles show memory distribution among your page's JavaScript objects and related DOM nodes.

☒ Include numerical values in capture

☐ Allocation instrumentation on timeline
Allocation timelines show instrumented JavaScript memory allocations over time. Once profile is recorded you can select a time interval to see objects that were allocated within it and still alive by the end of recording. Use this profile type to isolate memory leaks.

☒ Record stack traces of allocations (extra performance overhead)

☒ Allocation sampling
Record memory allocations using sampling method. This profile type has minimal performance overhead and can be used for long running operations. It provides good approximation of allocations broken down by JavaScript execution stack.

Select JavaScript VM instance

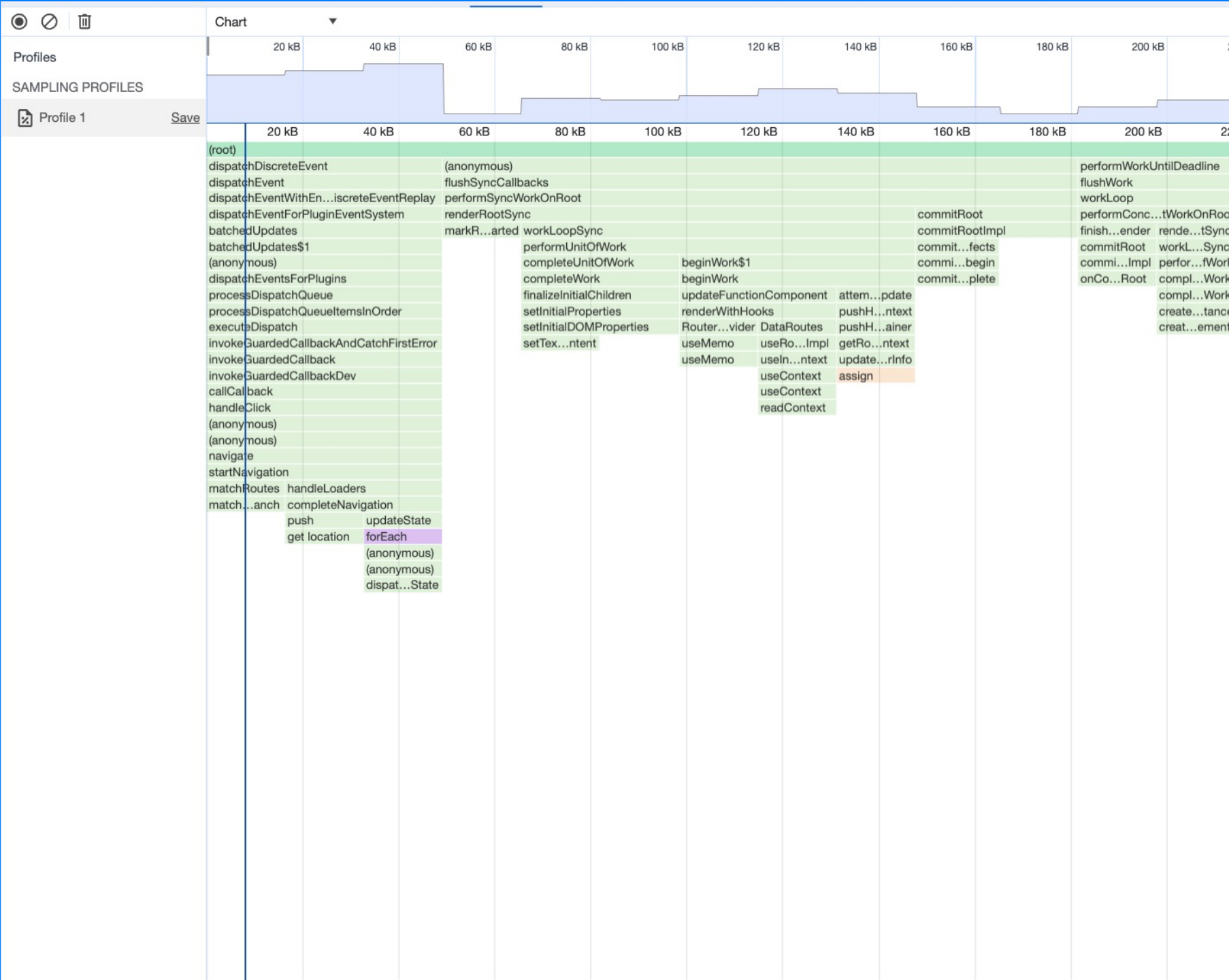
8.8 MB ↓41.3 kB/s localhost:3000: Main

8.8 MB ↓41.3 kB/s Total JS heap size

Stop

Load

Allocation Sampling



Когда использовать

- При появлении «лагов» или замедлений в приложении
- Во время длительных тестов



Итог по вкладке memory

- **Heap Snapshots:** Для статического анализа состояния памяти. Особенно полезно для поиска утечек памяти.

Итог по вкладке memory

- **Heap Snapshots:** Для статического анализа состояния памяти. Особенно полезно для поиска утечек памяти.
- **Allocation Timeline:** Для динамического анализа выделения и освобождения памяти в реальном времени.

Итог по вкладке memory

- **Heap Snapshots:** Для статического анализа состояния памяти. Особенно полезно для поиска утечек памяти.
- **Allocation Timeline:** Для динамического анализа выделения и освобождения памяти в реальном времени.
- **Allocation Sampling:** когда надо долго анализировать память

Выводы

Lighthouse — это
не универсальная пилуля

Важно понимать
как искать утечки памяти

Инструменты
хрома обширны

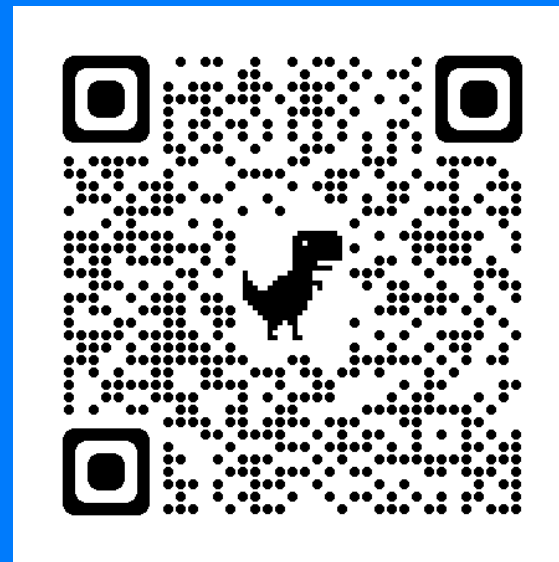
Перфоманс тесты проще,
чем кажется



Спасибо за внимание



tg @alexey_qa



Все ссылки и чуть больше



самолет

— Другие проблемы у веб приложения есть?

— Да и их много

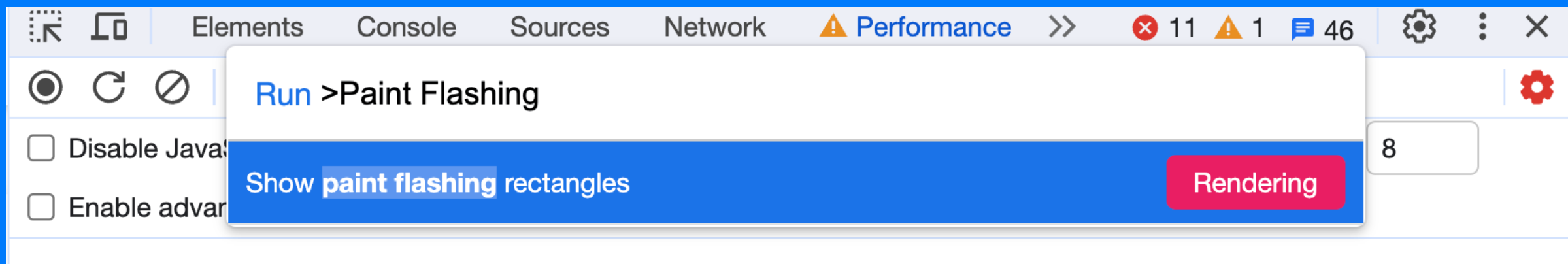


Другие полезные инструменты




Проблемы с частой перерисовкой

Shift + cmd(ctrl) + P



самолет

Искать проблемы где постоянные перерисовки







 **Котировки** > **Избранное**

График 

Поделиться 

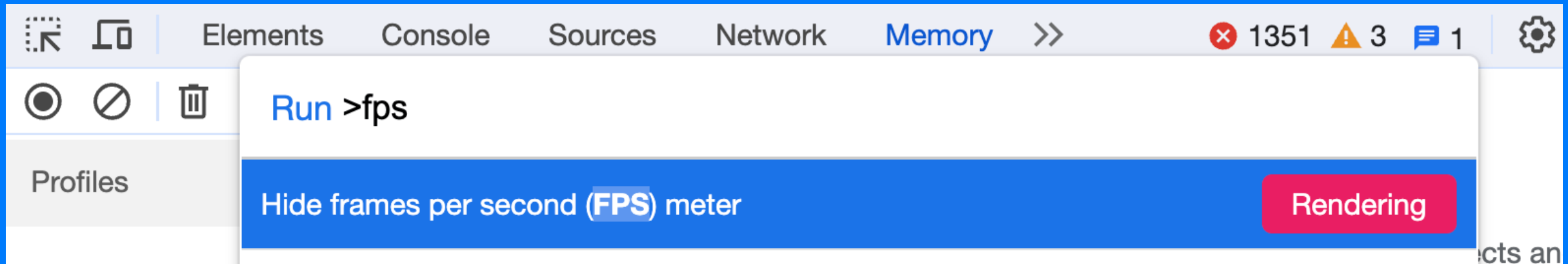
Отсортировать 	Посл. сделка	% Изм. цены	Открытие	Макс.	Мин.
★ AMDRUB курс ЦБ	0.2450P	2.88%			
★ ЛУКОЙЛ	6797.0P	1.12%	6749.5	6797.0	6717.0
★ ГМКНорНик	16268P	0.59%	16174	16276	16056
★ ГАЗПРОМ ао	166.59P	0.32%	166.29	166.86	165.10
★ ВТБ ао	0.025150P	1.43%	0.024980	0.025170	0.024640
★ BIDU-адр	9822P	-10.79%			
★ ETLN-гдр	86.00P	0.09%	85.92	87.10	84.04
★ BABA-адр	7400P	-4.52%			
★ AGRO-гдр	1157.0P	0.40%	1158.8	1165.0	1141.8
★ CADRUB курс ЦБ	73.1749P	0.89%			
★ BYNRUB курс ЦБ	29.9036P	0.13%			

Если нужного инструмента нет в списке, воспользуйтесь поиском

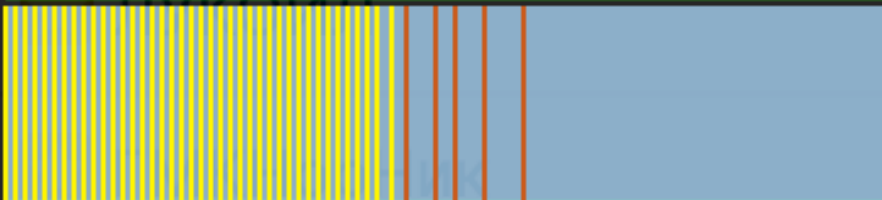



самолет

FPS



самолет

<div> <div>★</div> <div>МДВ курс ЦБ</div> </div>	0.2450₽
<div> <div>★</div> <div>ПУКОЙД</div> </div> <div> <div>116.4 fps</div>  </div>	6797.0₽
<div> <div>★</div> <div>ГАЗПРОМ ао</div> </div> <div> <div>GPU raster</div> <div>on</div> </div>	16268₽
<div> <div>★</div> <div>ВТБ ао</div> </div> <div> <div>GPU memory</div> <div>59.3 MB used</div> <div>655.4 MB max</div>  </div>	166.59₽
<div> <div>★</div> <div>ВТБ ао</div> </div> <div> <div>GPU memory</div> <div>59.3 MB used</div> <div>655.4 MB max</div> </div>	0.025150₽
<div> <div>★</div> <div>ВТБ ао</div> </div> <div> <div>GPU memory</div> <div>59.3 MB used</div> <div>655.4 MB max</div> </div>	9822₽
<div> <div>★</div> <div>ВТБ ао</div> </div> <div> <div>GPU memory</div> <div>59.3 MB used</div> <div>655.4 MB max</div> </div>	86.00₽



самолет

Core web vitals



самолет

главная

[analytics](#)

[Рик и морти](#)

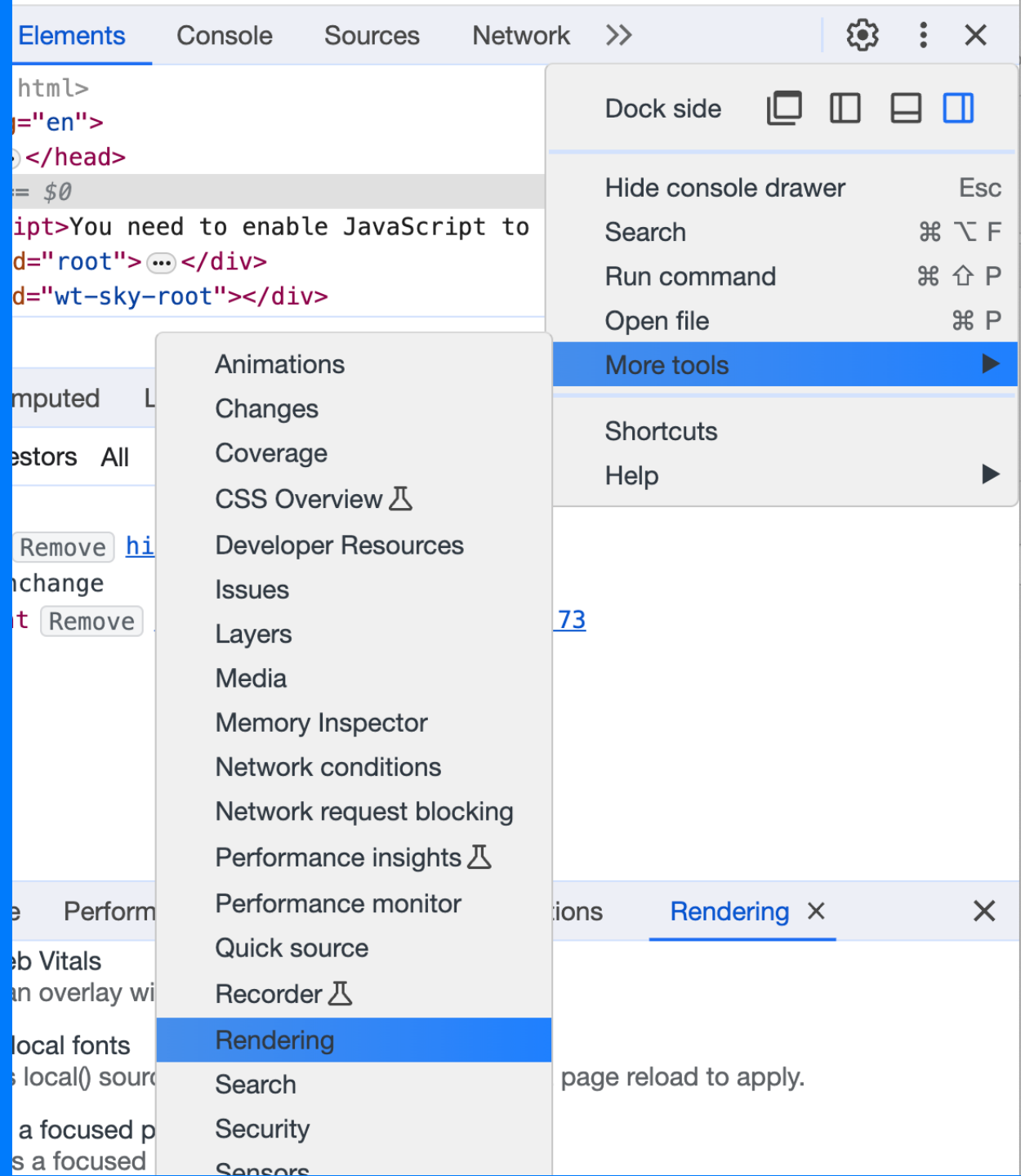
[Посты сломанные](#)

[Посты немного сломанные](#)

[Утечка связанная с добавлением события](#)

- Largest Contentful Paint 0.11 s
- First Input Delay 3.61 ms
- Cumulative Layout Shift -

Больше разных
инструментов, можно
найти тут



Больше разных
инструментов, можно
найти тут



Styles Computed Layout Event Listeners DOM Breakpoints Properties >>

⋮ Console Performance monitor Network conditions Rendering × ×

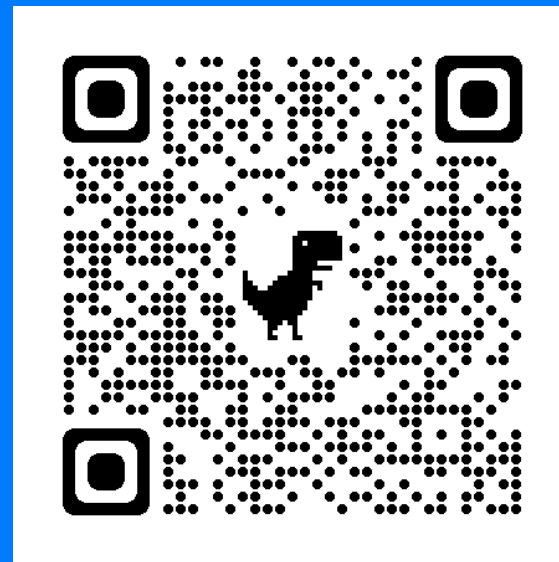
- Paint flashing
 - ☐ Highlights areas of the page (green) that need to be repainted. May not be suitable for people prone to photosensitive epilepsy.
- Layout Shift Regions
 - ☐ Highlights areas of the page (blue) that were shifted. May not be suitable for people prone to photosensitive epilepsy.
- Layer borders
 - ☐ Shows layer borders (orange/olive) and tiles (cyan).
- Frame Rendering Stats
 - ☐ Plots frame throughput, dropped frames distribution, and GPU memory.
- Scrolling performance issues
 - ☐ Highlights elements (teal) that can slow down scrolling, including touch & wheel event handlers and other main-thread scrolling situations.
- Highlight ad frames
 - ☐ Highlights frames (red) detected to be ads.
- Core Web Vitals
 - ☐ Shows an overlay with Core Web Vitals.
- Disable local fonts
 - ☐ Disables local() sources in @font-face rules. Requires a page reload to apply.
- Emulate a focused page
 - ☐ Emulates a focused page.
- Enable automatic dark mode
 - ☐ Enables automatic dark mode and sets prefers-color-scheme to dark.

Emulate CSS media feature prefers-color-scheme
Forces CSS prefers-color-scheme media feature

Спасибо за внимание



tg @alexey_qa



Все ссылки и чуть больше



самолет