

## Выполнение заданий

- 1) Открыть браузер (например, Chrome)
- 2) Нажать F12 (вызвать консоль разработчика)
- 3) Перейти на вкладку «Elements»



Рисунок 1 – Просмотр дерева DOM

# Вариант 1

Смотреть раздел [Выполнение заданий](#)

## Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/get>, метод GET, установить заголовок Content-Type: audio/mp4

2) Полученный результат преобразовать к JSON

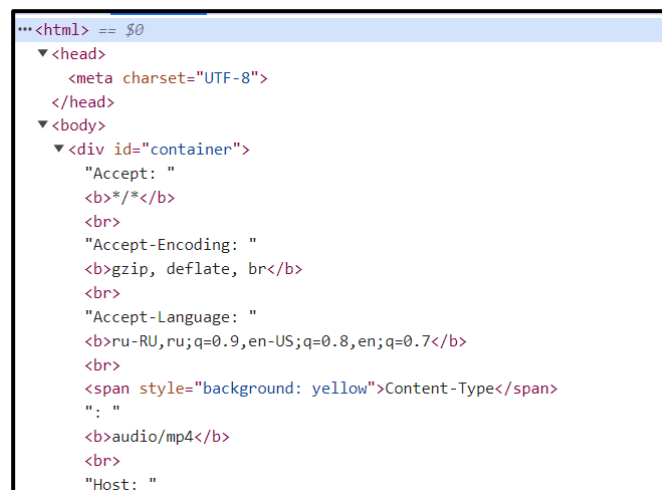
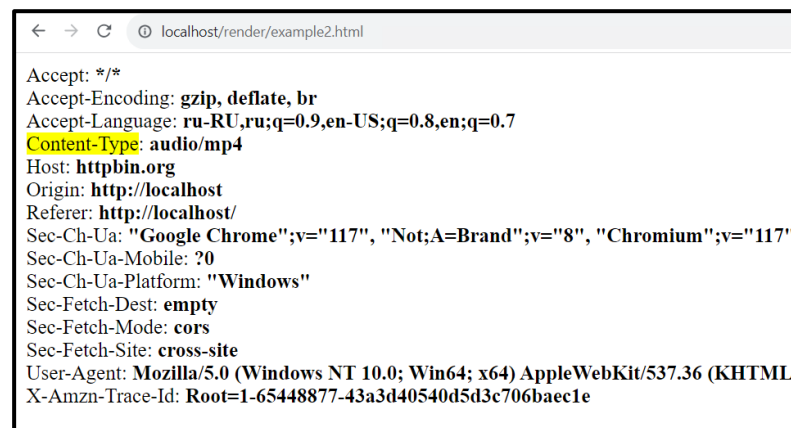
3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон зеленым. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:



## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

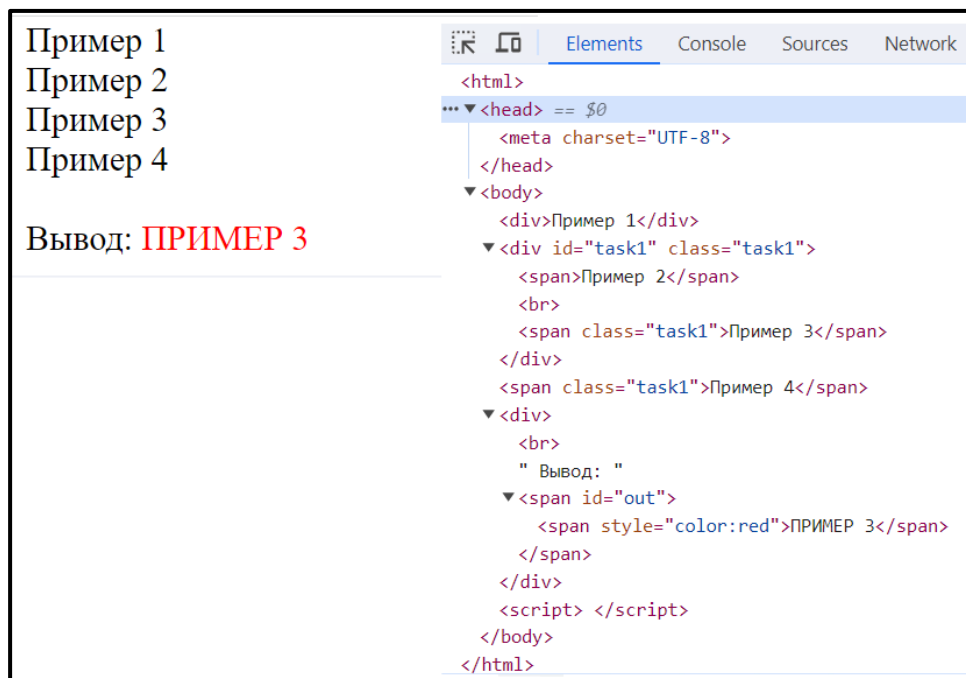
```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task1" class="task1">
    <span>Пример 2</span>
    <br/>
    <span class="task1">Пример 3</span>
  </div>
  <span class="task1">Пример 4</span>
  <div>
    <br/>
    Вывод:
    <span id="out"></span>
  </div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task1">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:



## Вариант 2

Смотреть раздел [Выполнение заданий](#)

### Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/get>, метод GET, установить заголовок Content-Type: application/pdf

2) Полученный результат преобразовать к JSON

3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

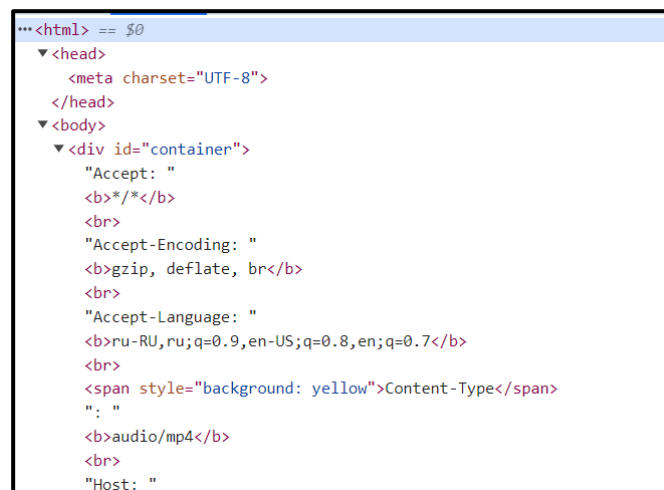
4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон зеленым. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:



```
Accept: */*
Accept-Encoding: gzip, deflate, br
Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7
Content-Type: audio/mp4
Host: httpbin.org
Origin: http://localhost
Referer: http://localhost/
Sec-Ch-Ua: "Google Chrome";v="117", "Not;A=Brand";v="8", "Chromium";v="117"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: cross-site
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
X-Amzn-Trace-Id: Root=1-65448877-43a3d40540d5d3c706baec1e
```



```
<html> == $0
  <head>
    <meta charset="UTF-8">
  </head>
  <body>
    <div id="container">
      "Accept: "
      <b>*/*/</b>
      <br>
      "Accept-Encoding: "
      <b>gzip, deflate, br</b>
      <br>
      "Accept-Language: "
      <b>ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7</b>
      <br>
      <span style="background: yellow">Content-Type</span>
      "; "
      <b>audio/mp4</b>
      <br>
      "Host: "
```

## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

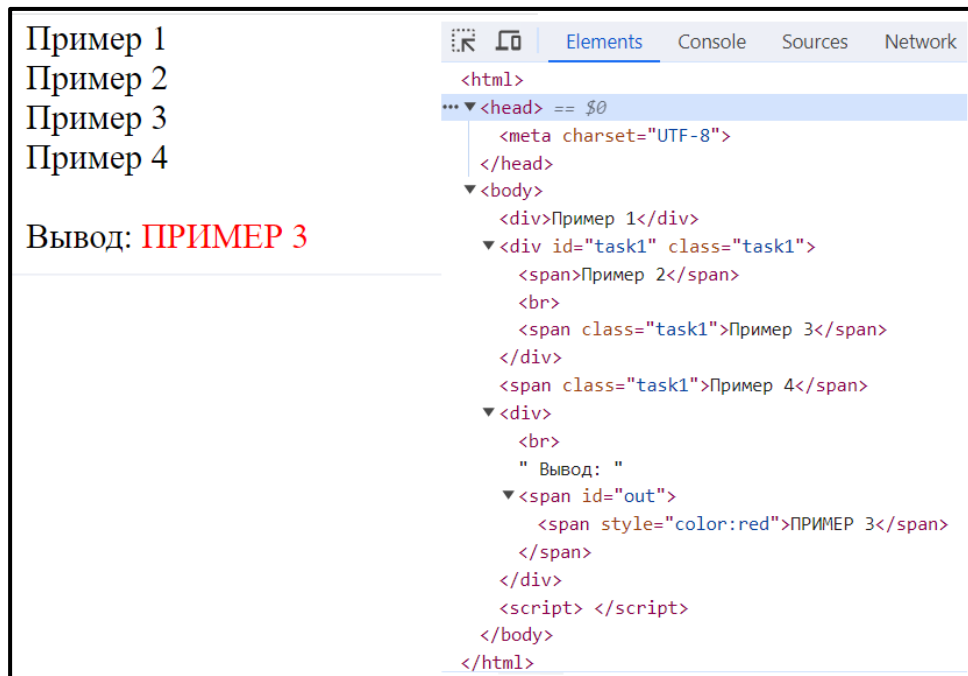
```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task2" class="task2">
    <span>Пример 2</span>
    <br/>
    <span class="task2">Пример 3</span>
  </div>
  <span class="task2">Пример 4</span>
</div>
  <br/>
  Вывод:
  <span id="out"></span>
</div>
<script>
</script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task2">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:



## Вариант 3

Смотреть раздел [Выполнение заданий](#)

### Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/get>, метод GET, установить заголовок Content-Type: text/plain

2) Полученный результат преобразовать к JSON

3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

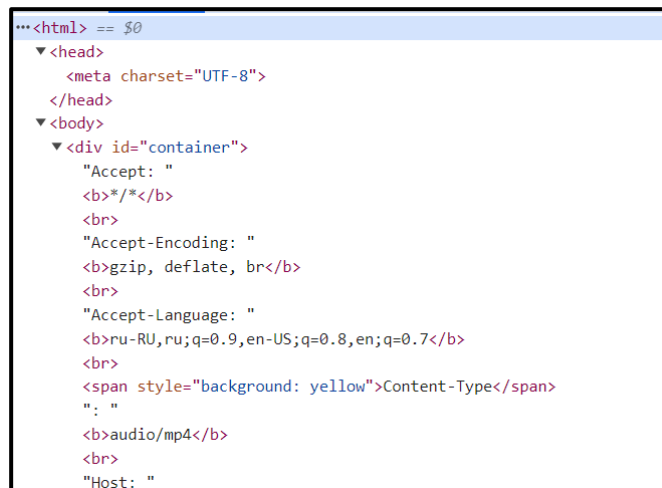
4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон желтым. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:



```
Accept: */*
Accept-Encoding: gzip, deflate, br
Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7
Content-Type: audio/mp4
Host: httpbin.org
Origin: http://localhost
Referer: http://localhost/
Sec-Ch-Ua: "Google Chrome";v="117", "Not;A=Brand";v="8", "Chromium";v="117"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: cross-site
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
X-Amzn-Trace-Id: Root=1-65448877-43a3d40540d5d3c706baec1e
```



```
<html> == $0
  <head>
    <meta charset="UTF-8">
  </head>
  <body>
    <div id="container">
      "Accept: "
      <b>*/*/</b>
      <br>
      "Accept-Encoding: "
      <b>gzip, deflate, br</b>
      <br>
      "Accept-Language: "
      <b>ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7</b>
      <br>
      <span style="background: yellow">Content-Type</span>
      "; "
      <b>audio/mp4</b>
      <br>
      "Host: "
```

## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

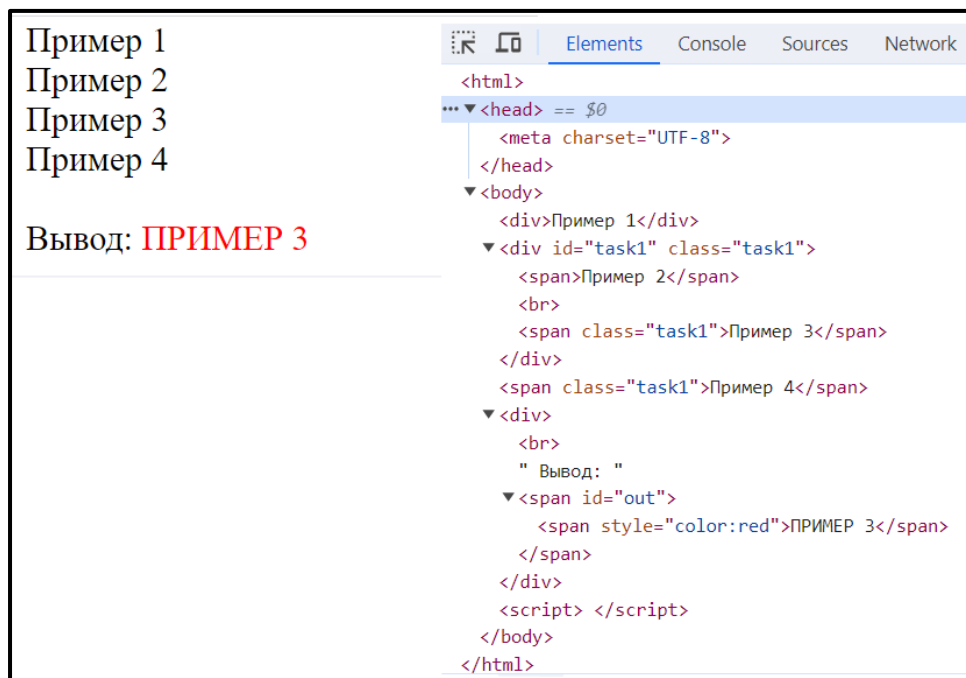
```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task3" class="task3">
    <span>Пример 2</span>
    <br/>
    <span class="task3">Пример 3</span>
  </div>
  <span class="task3">Пример 4</span>
  <div>
    <br/>
    Вывод:
    <span id="out"></span>
  </div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task3">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:



## Вариант 4

Смотреть раздел [Выполнение заданий](#)

### Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/get>, метод GET, установить заголовок Content-Type: image/svg+xml

2) Полученный результат преобразовать к JSON

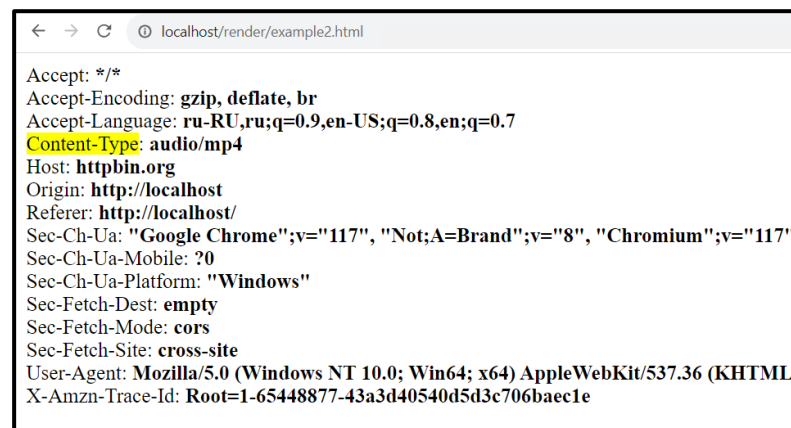
3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

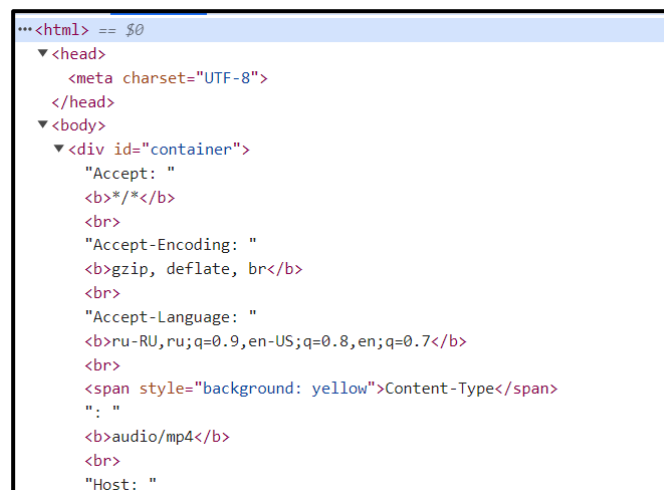
4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон красным. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:



```
Accept: */*
Accept-Encoding: gzip, deflate, br
Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7
Content-Type: audio/mp4
Host: httpbin.org
Origin: http://localhost
Referer: http://localhost/
Sec-Ch-Ua: "Google Chrome";v="117", "Not;A=Brand";v="8", "Chromium";v="117"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: cross-site
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
X-Amzn-Trace-Id: Root=1-65448877-43a3d40540d5d3c706baec1e
```



```
...<html> == $0
  <head>
    <meta charset="UTF-8">
  </head>
  <body>
    <div id="container">
      "Accept: "
      <b>*/*/</b>
      <br>
      "Accept-Encoding: "
      <b>gzip, deflate, br</b>
      <br>
      "Accept-Language: "
      <b>ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7</b>
      <br>
      <span style="background: yellow">Content-Type</span>
      "; "
      <b>audio/mp4</b>
      <br>
      "Host: "
```



## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task4" class="task4">
    <span>Пример 2</span>
    <br/>
    <span class="task4">Пример 3</span>
  </div>
  <span class="task4">Пример 4</span>
</div>
  <br/>
  Вывод:
  <span id="out"></span>
</div>
<script>
</script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task4">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:

## Вариант 5

Смотреть раздел [Выполнение заданий](#)

### Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/get>, метод GET, установить заголовок Content-Type: application/zip

2) Полученный результат преобразовать к JSON

3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

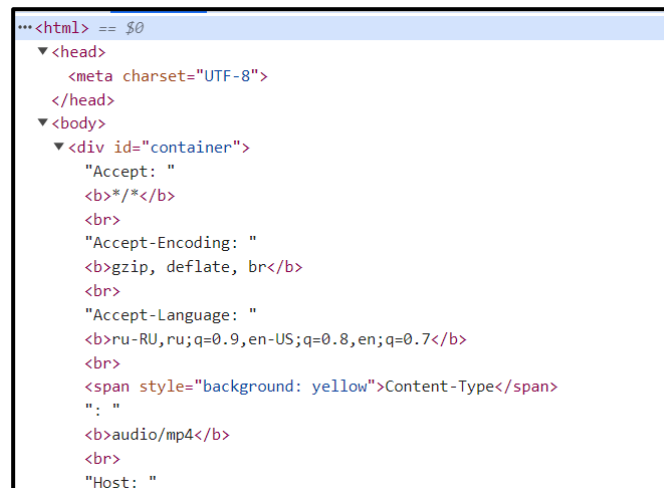
4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон зеленым. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:



```
Accept: */*
Accept-Encoding: gzip, deflate, br
Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7
Content-Type: audio/mp4
Host: httpbin.org
Origin: http://localhost
Referer: http://localhost/
Sec-Ch-Ua: "Google Chrome";v="117", "Not;A=Brand";v="8", "Chromium";v="117"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: cross-site
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
X-Amzn-Trace-Id: Root=1-65448877-43a3d40540d5d3c706baec1e
```



```
<html> == $0
  <head>
    <meta charset="UTF-8">
  </head>
  <body>
    <div id="container">
      "Accept: "
      <b>*/*/</b>
      <br>
      "Accept-Encoding: "
      <b>gzip, deflate, br</b>
      <br>
      "Accept-Language: "
      <b>ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7</b>
      <br>
      <span style="background: yellow">Content-Type</span>
      "; "
      <b>audio/mp4</b>
      <br>
      "Host: "
```

## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

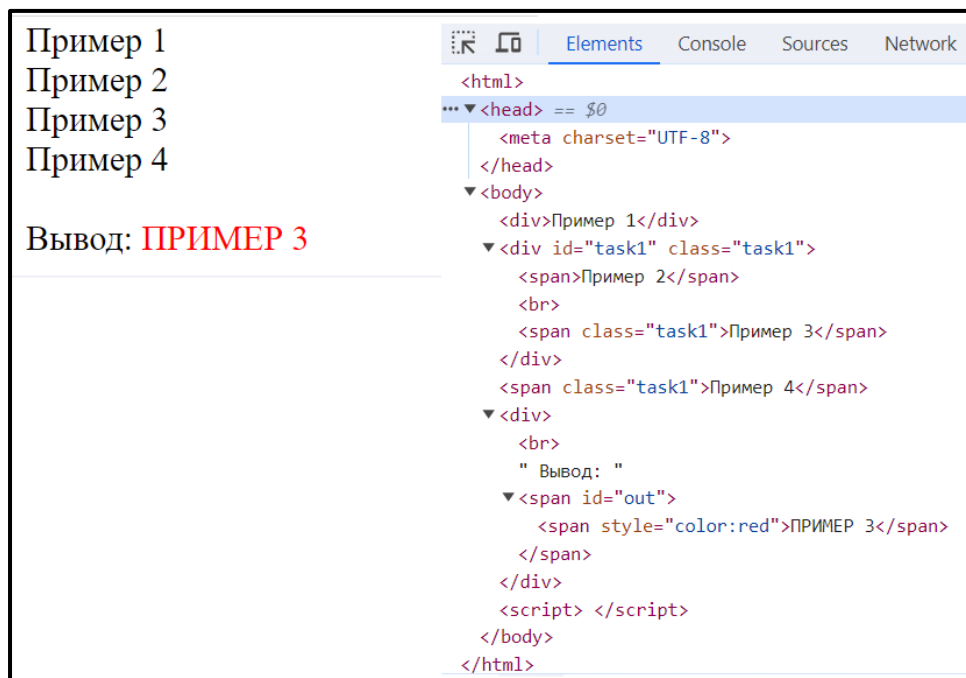
```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task5" class="task5">
    <span>Пример 2</span>
    <br/>
    <span class="task5">Пример 3</span>
  </div>
  <span class="task5">Пример 4</span>
  <div>
    <br/>
    Вывод:
    <span id="out"></span>
  </div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task5">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:



## Вариант 6

Смотреть раздел [Выполнение заданий](#)

### Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/post>, метод POST, установить заголовок Content-Type: audio/mp4

2) Полученный результат преобразовать к JSON

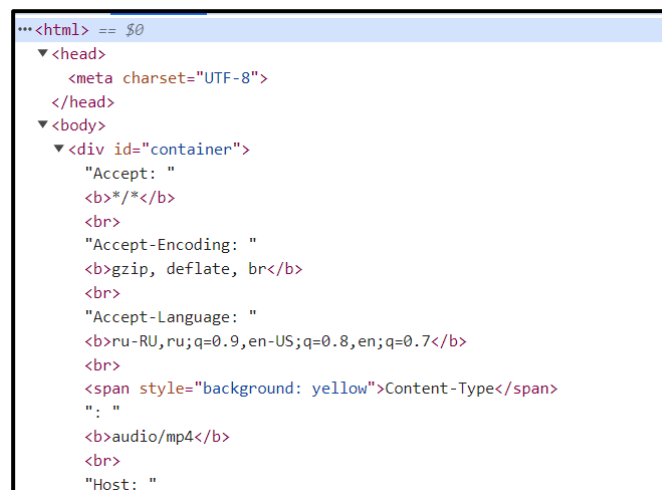
3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон зеленым. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:



## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

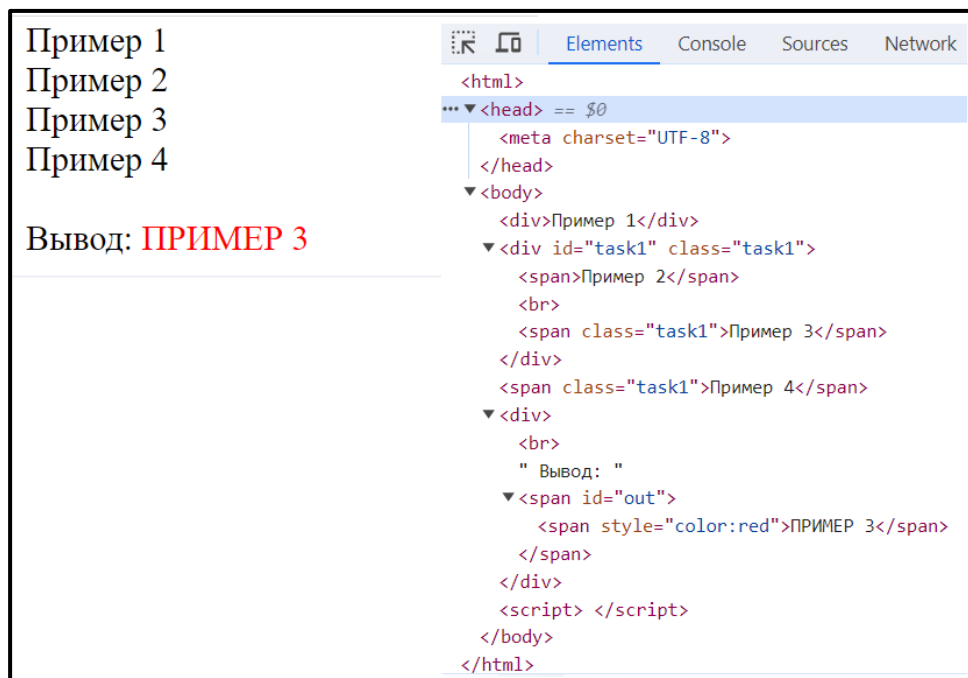
```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task6" class="task6">
    <span>Пример 2</span>
    <br/>
    <span class="task6">Пример 3</span>
  </div>
  <span class="task6">Пример 4</span>
  <div>
    <br/>
    Вывод:
    <span id="out"></span>
  </div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task6">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:



# Вариант 7

Смотреть раздел [Выполнение заданий](#)

## Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/post>, метод POST, установить заголовок Content-Type: application/pdf

2) Полученный результат преобразовать к JSON

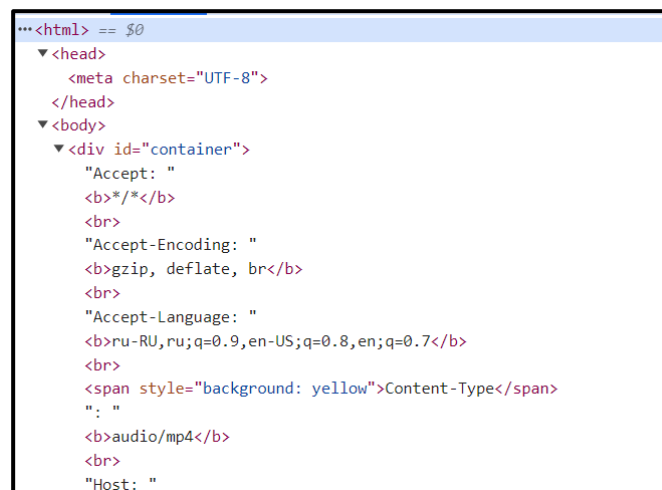
3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон зеленым. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:



## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task7" class="task7">
    <span>Пример 2</span>
    <br/>
    <span class="task7">Пример 3</span>
  </div>
  <span class="task7">Пример 4</span>
  <div>
    <br/>
    Вывод:
    <span id="out"></span>
  </div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task7">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:

## Вариант 8

Смотреть раздел [Выполнение заданий](#)

### Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/post>, метод POST, установить заголовок Content-Type: image/gif

2) Полученный результат преобразовать к JSON

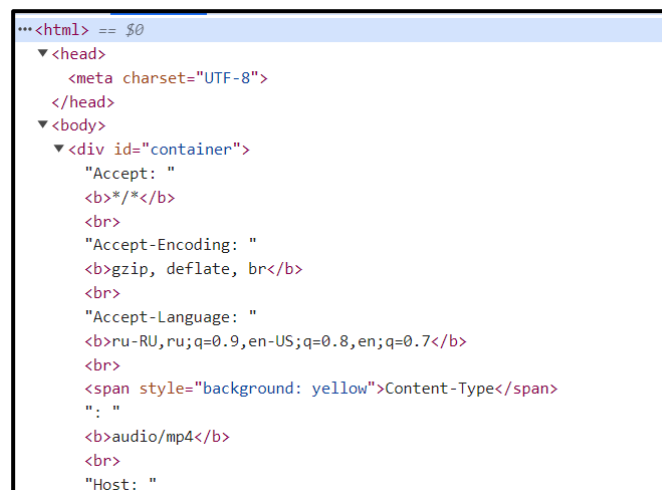
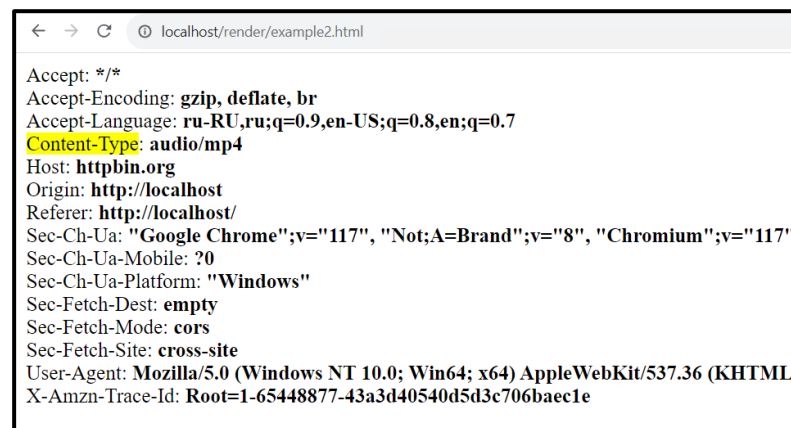
3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон желтым. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:





## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task8" class="task8">
    <span>Пример 2</span>
    <br/>
    <span class="task8">Пример 3</span>
  </div>
  <span class="task8">Пример 4</span>
  <div>
    <br/>
    Вывод:
    <span id="out"></span>
  </div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task8">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:

## Вариант 9

Смотреть раздел [Выполнение заданий](#)

### Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/post>, метод POST, установить заголовок Content-Type: image/svg+xml

2) Полученный результат преобразовать к JSON

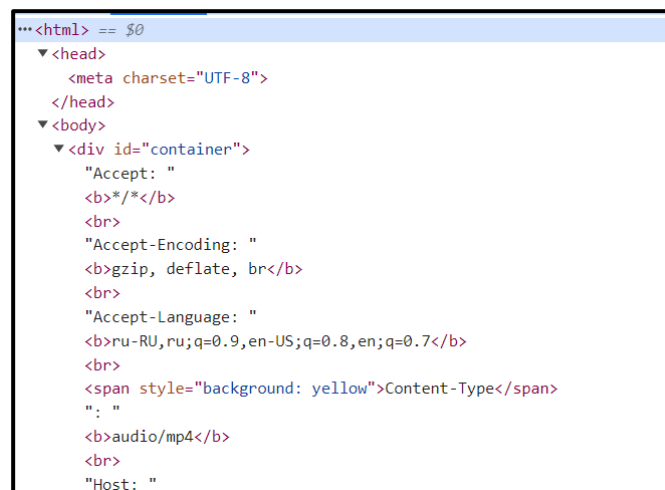
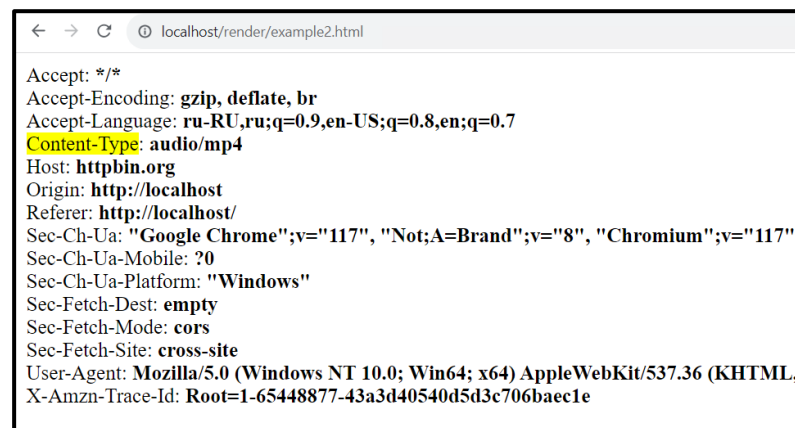
3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон красным. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:



## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task9" class="task9">
    <span>Пример 2</span>
    <br/>
    <span class="task9">Пример 3</span>
  </div>
  <span class="task9">Пример 4</span>
</div>
  <br/>
  Вывод:
  <span id="out"></span>
</div>
<script>
</script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task9">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:

# Вариант 10

Смотреть раздел [Выполнение заданий](#)

## Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/post>, метод POST, установить заголовок Content-Type: application/zip

2) Полученный результат преобразовать к JSON

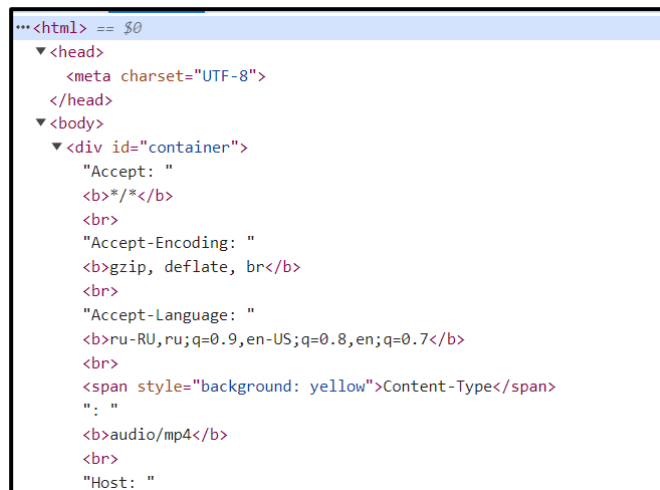
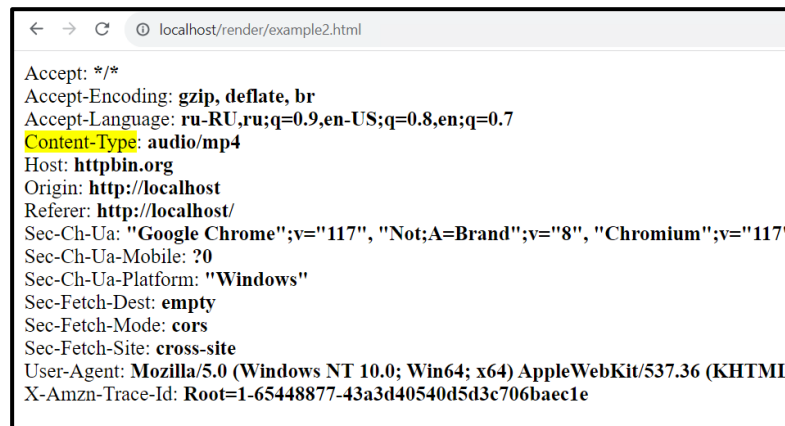
3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон зеленым. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:



## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task10" class="task10">
    <span>Пример 2</span>
    <br/>
    <span class="task10">Пример 3</span>
  </div>
  <span class="task10">Пример 4</span>
  <div>
    <br/>
    Вывод:
    <span id="out"></span>
  </div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task10">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:

# Вариант 11

Смотреть раздел [Выполнение заданий](#)

## Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/put>, метод PUT, установить заголовок Content-Type: audio/mp4

2) Полученный результат преобразовать к JSON

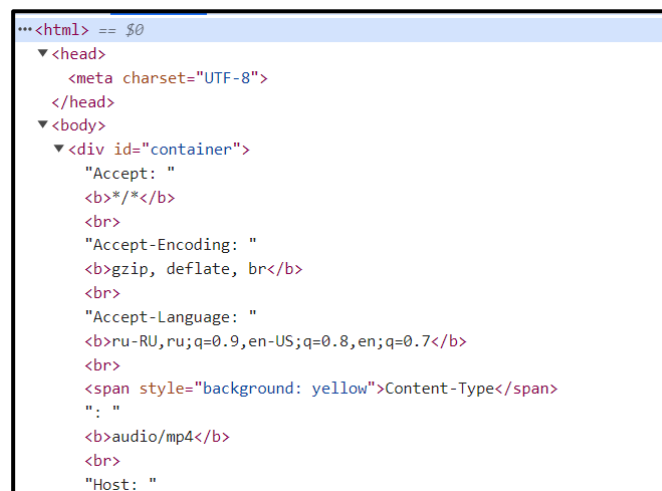
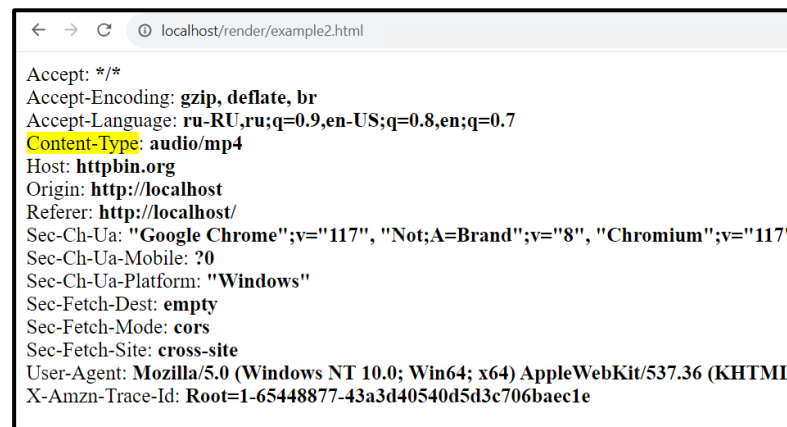
3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон зеленым. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:



## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

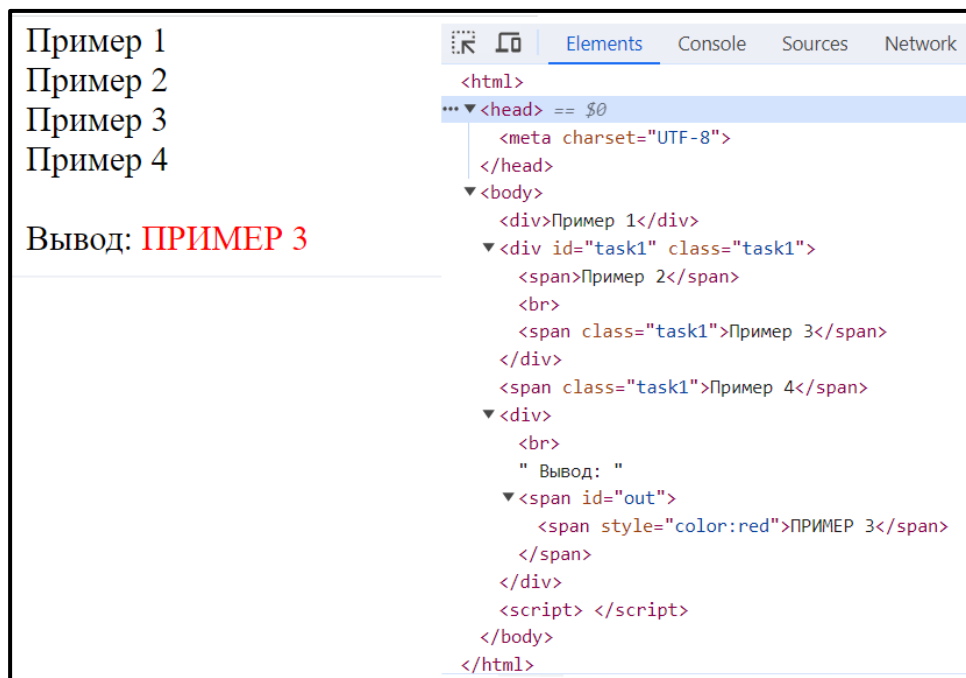
```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task1" class="task1">
    <span>Пример 2</span>
    <br/>
    <span class="task1">Пример 3</span>
  </div>
  <span class="task1">Пример 4</span>
  <div>
    <br/>
    Вывод:
    <span id="out"></span>
  </div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task1">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:



# Вариант 12

Смотреть раздел [Выполнение заданий](#)

## Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/put>, метод PUT, установить заголовок Content-Type: application/pdf

2) Полученный результат преобразовать к JSON

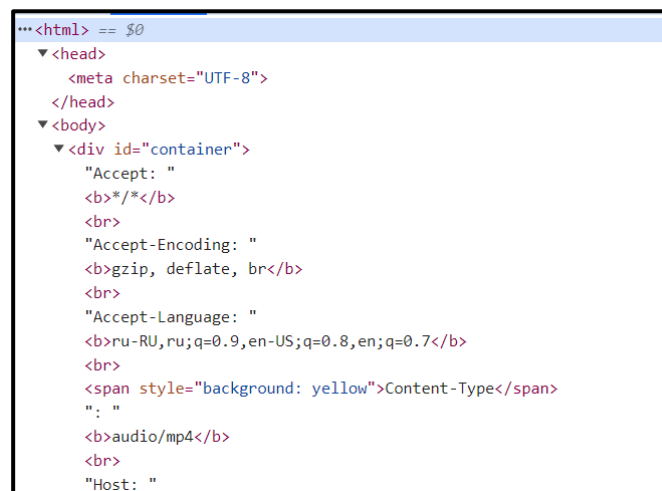
3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон зеленым. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:





## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task12" class="task12">
    <span>Пример 2</span>
    <br/>
    <span class="task12">Пример 3</span>
  </div>
  <span class="task12">Пример 4</span>
</div>
  <br/>
  Вывод:
  <span id="out"></span>
</div>
<script>
</script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task12">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:

# Вариант 13

Смотреть раздел [Выполнение заданий](#)

## Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/put>, метод PUT, установить заголовок Content-Type: text/plain

2) Полученный результат преобразовать к JSON

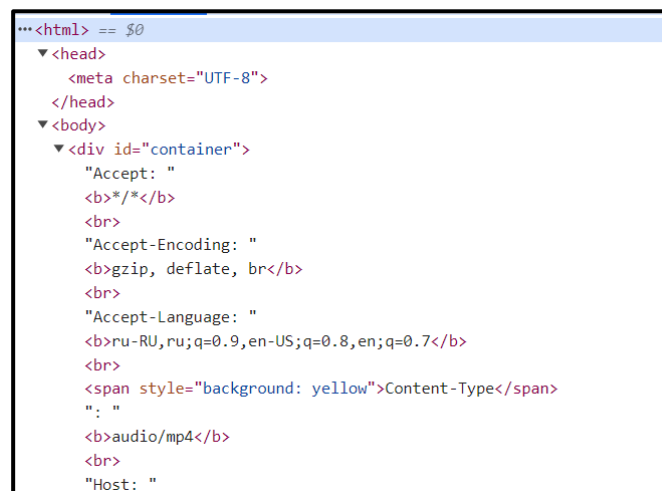
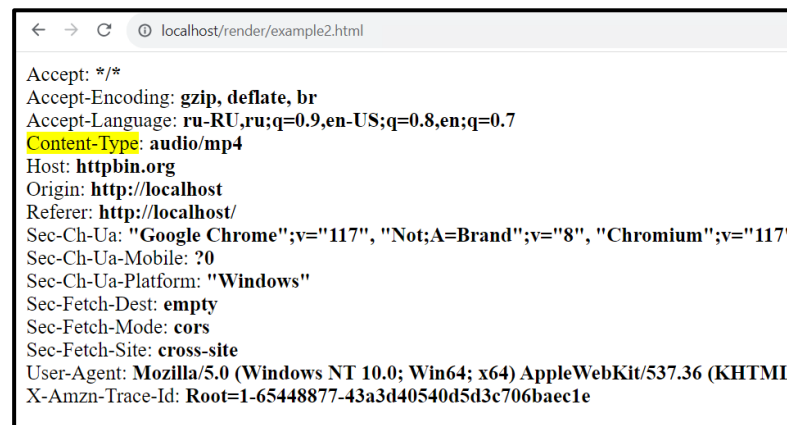
3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон желтым. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:



## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task13" class="task13">
    <span>Пример 2</span>
    <br/>
    <span class="task13">Пример 3</span>
  </div>
  <span class="task13">Пример 4</span>
</div>
  <br/>
  Вывод:
  <span id="out"></span>
</div>
<script>
</script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task13">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:

# Вариант 14

Смотреть раздел [Выполнение заданий](#)

## Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/put>, метод PUT, установить заголовок Content-Type: application/graphql

2) Полученный результат преобразовать к JSON

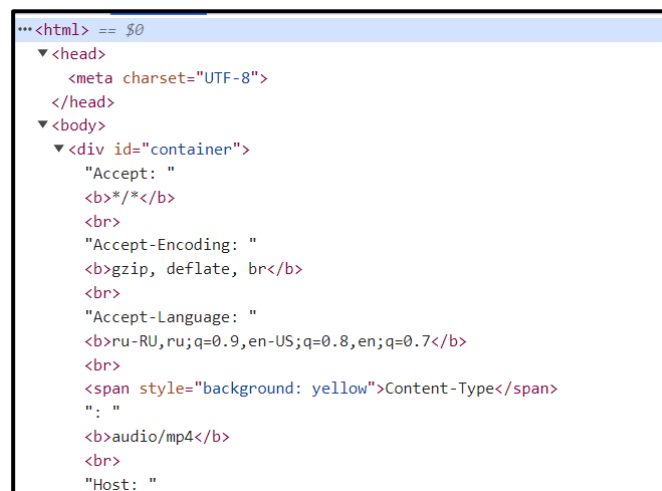
3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон красным. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:



## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

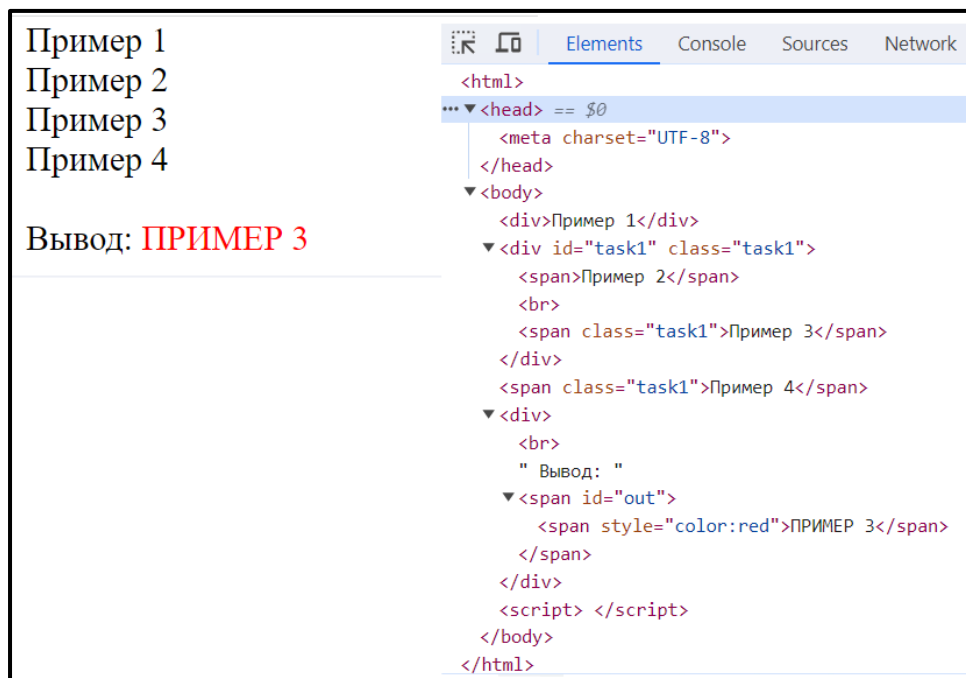
```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task14" class="task14">
    <span>Пример 2</span>
    <br/>
    <span class="task14">Пример 3</span>
  </div>
  <span class="task14">Пример 4</span>
  <div>
    <br/>
    Вывод:
    <span id="out"></span>
  </div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task14">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:



# Вариант 15

Смотреть раздел [Выполнение заданий](#)

## Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/put>, метод PUT, установить заголовок Content-Type: audio/webm

2) Полученный результат преобразовать к JSON

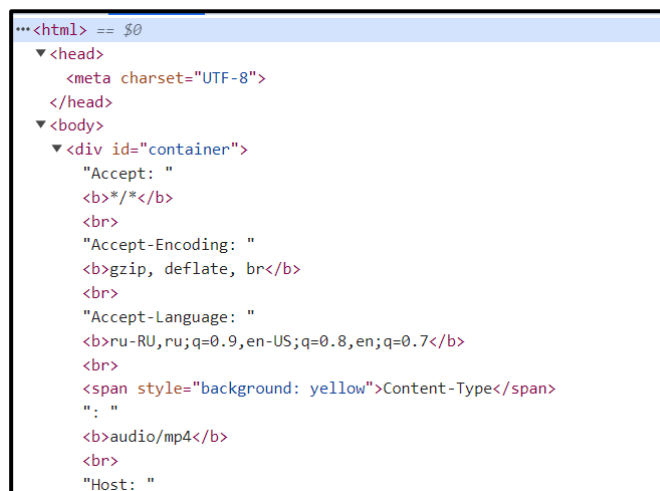
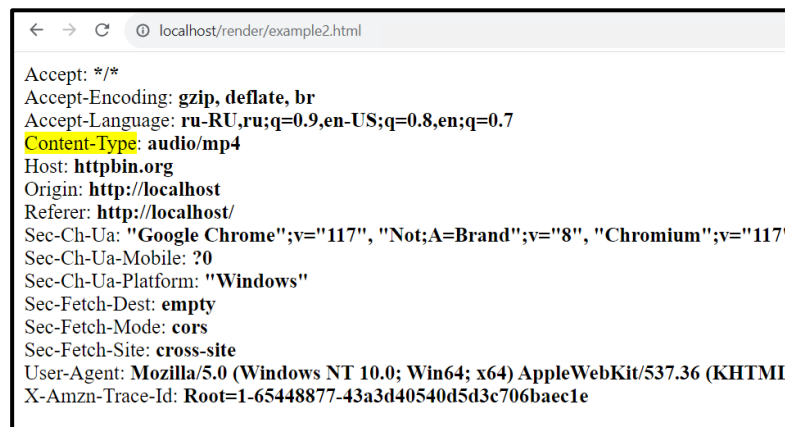
3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон зеленым. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:



## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

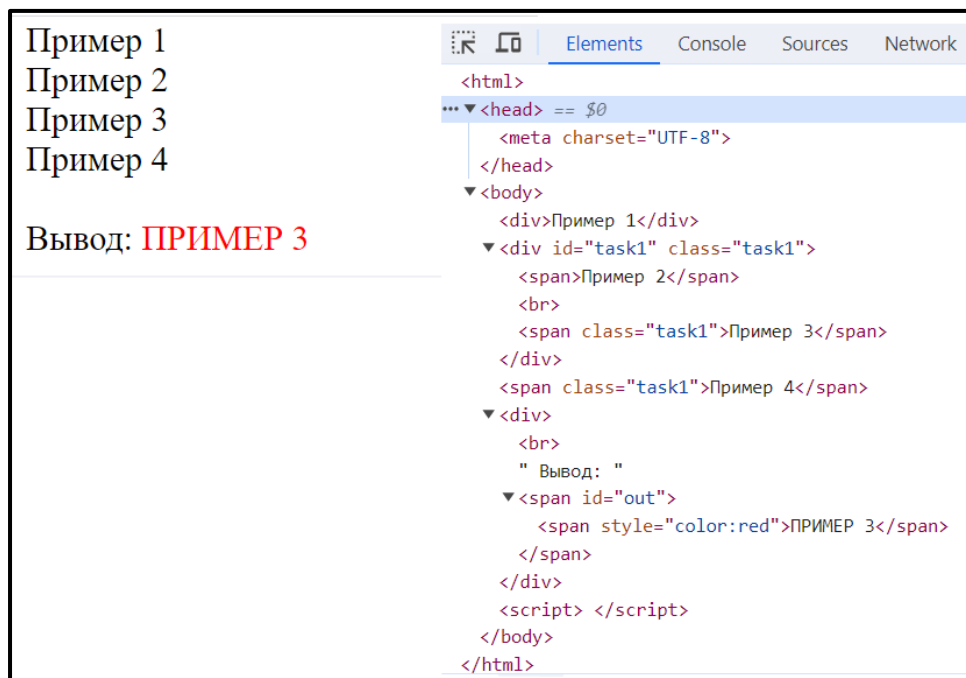
```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task15" class="task15">
    <span>Пример 2</span>
    <br/>
    <span class="task15">Пример 3</span>
  </div>
  <span class="task15">Пример 4</span>
</div>
  <br/>
  Вывод:
  <span id="out"></span>
</div>
<script>
</script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task15">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:



# Вариант 16

Смотреть раздел [Выполнение заданий](#)

## Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/delete>, метод DELETE, установить заголовок Content-Type: audio/mp4

2) Полученный результат преобразовать к JSON

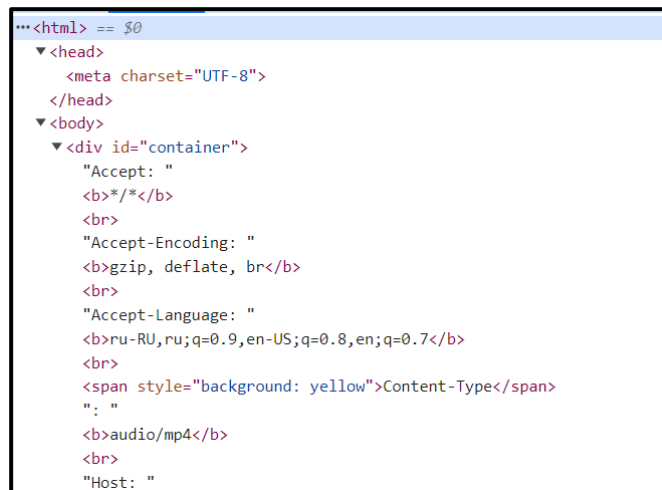
3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон зеленым. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:





## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task16" class="task16">
    <span>Пример 2</span>
    <br/>
    <span class="task16">Пример 3</span>
  </div>
  <span class="task16">Пример 4</span>
  <div>
    <br/>
    Вывод:
    <span id="out"></span>
  </div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task16">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:

# Вариант 17

Смотреть раздел [Выполнение заданий](#)

## Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/delete>, метод DELETE, установить заголовок Content-Type: application/pdf

2) Полученный результат преобразовать к JSON

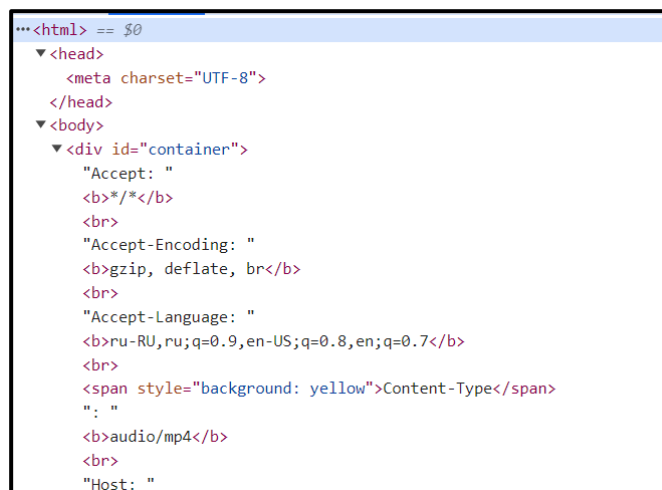
3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон зеленым. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:



## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

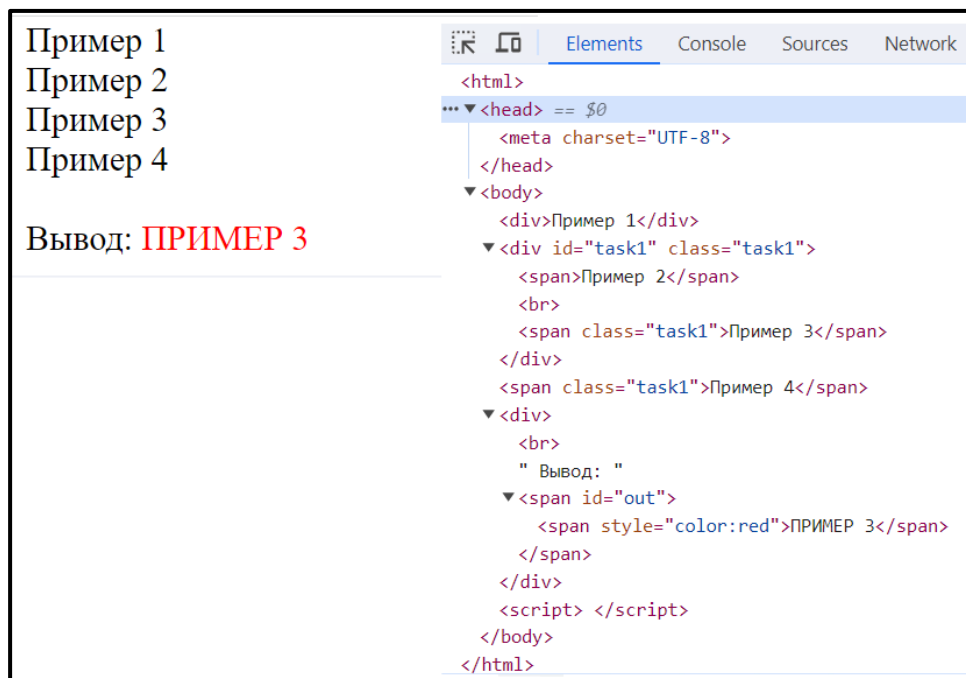
```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task17" class="task17">
    <span>Пример 2</span>
    <br/>
    <span class="task17">Пример 3</span>
  </div>
  <span class="task17">Пример 4</span>
  <div>
    <br/>
    Вывод:
    <span id="out"></span>
  </div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task17">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:



# Вариант 18

Смотреть раздел [Выполнение заданий](#)

## Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/delete>, метод DELETE, установить заголовок Content-Type: text/plain

2) Полученный результат преобразовать к JSON

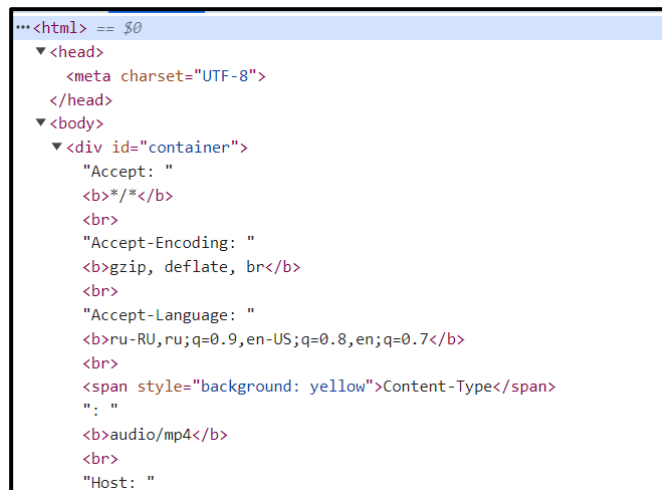
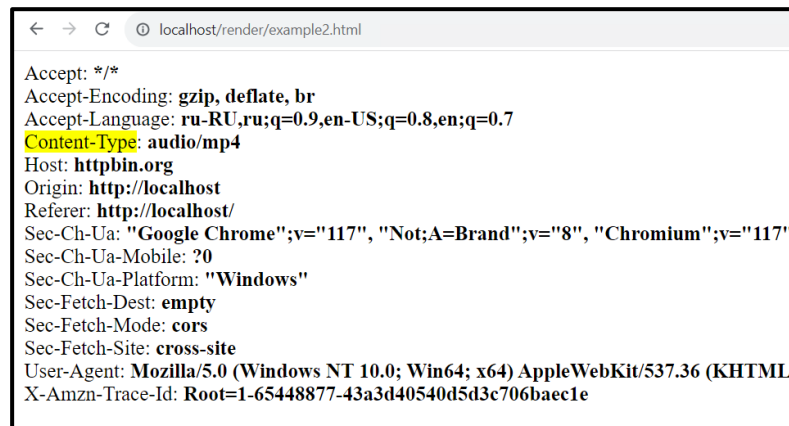
3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон желтым. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:



## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task18" class="task18">
    <span>Пример 2</span>
    <br/>
    <span class="task18">Пример 3</span>
  </div>
  <span class="task18">Пример 4</span>
</div>
  <br/>
  Вывод:
  <span id="out"></span>
</div>
<script>
</script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task18">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:

Пример 1  
Пример 2  
Пример 3  
Пример 4

Вывод: ПРИМЕР 3

Elements Console Sources Network

```
<html>
  <head> == $0
    <meta charset="UTF-8">
  </head>
  <body>
    <div>Пример 1</div>
    <div id="task1" class="task1">
      <span>Пример 2</span>
      <br>
      <span class="task1">Пример 3</span>
    </div>
    <span class="task1">Пример 4</span>
  </div>
  <br>
  " Вывод: "
  <span id="out">
    <span style="color:red">ПРИМЕР 3</span>
  </span>
</div>
<script> </script>
</body>
</html>
```

# Вариант 19

Смотреть раздел [Выполнение заданий](#)

## Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/delete>, метод DELETE, установить заголовок Content-Type: image/svg+xml

2) Полученный результат преобразовать к JSON

3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

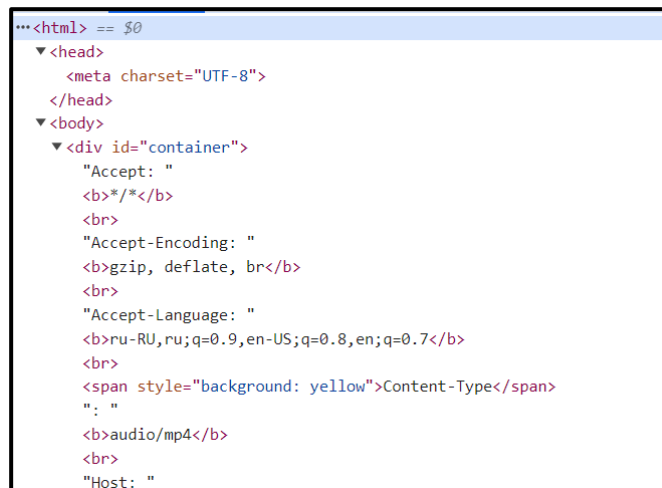
4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон красным. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:



```
Accept: */*
Accept-Encoding: gzip, deflate, br
Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7
Content-Type: audio/mp4
Host: httpbin.org
Origin: http://localhost
Referer: http://localhost/
Sec-Ch-Ua: "Google Chrome";v="117", "Not;A=Brand";v="8", "Chromium";v="117"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: cross-site
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
X-Amzn-Trace-Id: Root=1-65448877-43a3d40540d5d3c706baec1e
```



```
<html> == $0
  <head>
    <meta charset="UTF-8">
  </head>
  <body>
    <div id="container">
      "Accept: "
      <b>*/*/</b>
      <br>
      "Accept-Encoding: "
      <b>gzip, deflate, br</b>
      <br>
      "Accept-Language: "
      <b>ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7</b>
      <br>
      <span style="background: yellow">Content-Type</span>
      "; "
      <b>audio/mp4</b>
      <br>
      "Host: "
```

## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

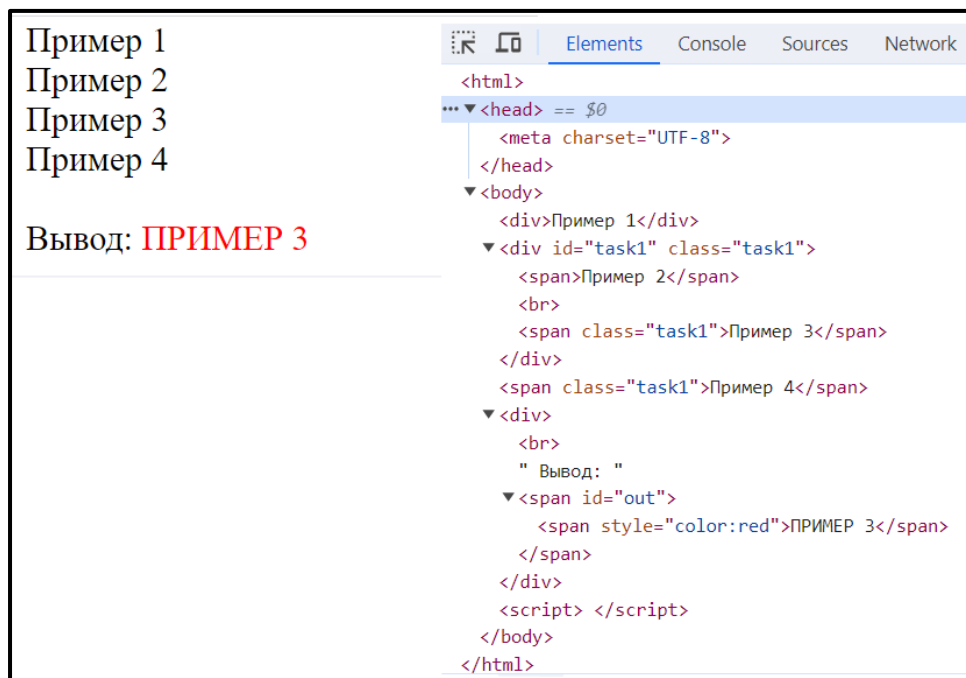
```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task19" class="task19">
    <span>Пример 2</span>
    <br/>
    <span class="task19">Пример 3</span>
  </div>
  <span class="task19">Пример 4</span>
</div>
  <br/>
  Вывод:
  <span id="out"></span>
</div>
<script>
</script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task19">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета:



# Вариант 20

Смотреть раздел [Выполнение заданий](#)

## Задание 1

Создать файл index1.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div id="container"></div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

1) Выполнить запрос через fetch к <https://httpbin.org/delete>, метод DELETE, установить заголовок Content-Type: application/zip

2) Полученный результат преобразовать к JSON

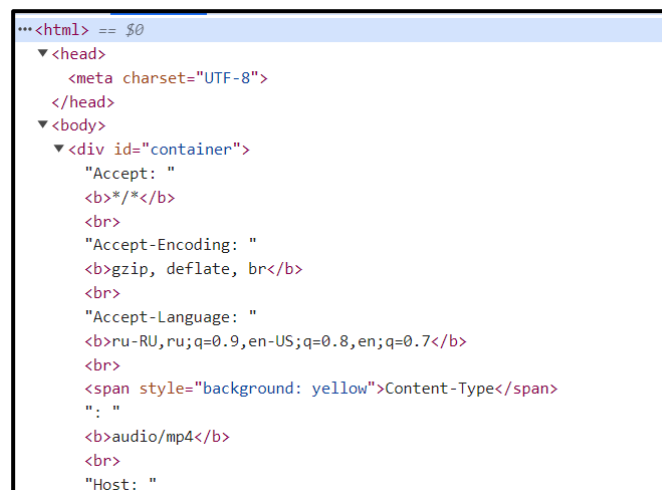
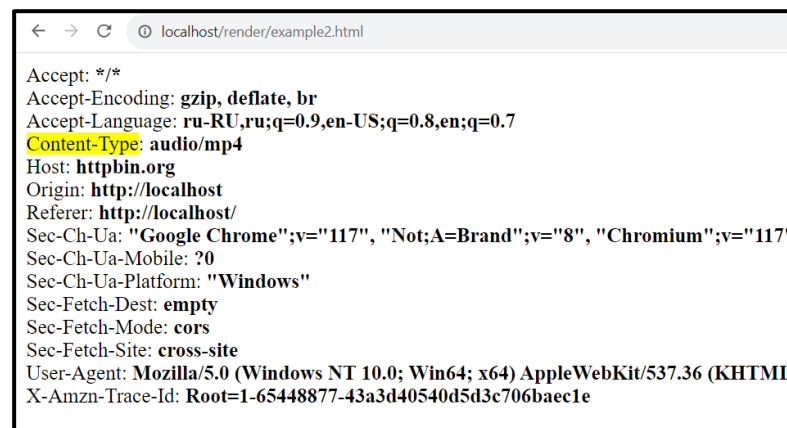
3) Получить свойство headers (data.headers). Заголовки будут представлять из себя объект в формате

```
{
  Заголовок1: "Значение1",
  Заголовок2: "Значение2",
  ...
}
```

4) Вывести заголовки в элемент с id="container" каждый с новой строки в формате Заголовок: значение (для перебора можно использовать for (let key in data.headers) или Object.keys() или Object.entries()). Если заголовок равен Content-Type, то сделать его фон зеленым. Значения выводить жирным.

Открыть index1.html в браузере. В отчет исходный код, скриншот результата.

Пример для заголовка Content-Type: audio/mp4 и желтого фона:





## Задание 2

Создать файл index2.html, открыть в редакторе (notepad++, VS Code, ...), вставить содержимое:

```
<meta charset="UTF-8">
<body>
  <div>Пример 1</div>
  <div id="task20" class="task20">
    <span>Пример 2</span>
    <br/>
    <span class="task20">Пример 3</span>
  </div>
  <span class="task20">Пример 4</span>
  <div>
    <br/>
    Вывод:
    <span id="out"></span>
  </div>
  <script>
  </script>
</body>
```

В теге <script></script> написать скрипт:

- 1) Получить элемент <span class="task20">Пример 3</span> при помощи querySelector, присвоить его в переменную element.
- 2) Получить элемент с id="out" при помощи querySelector в переменную out.
- 3) Вывести в элемент out содержимое элемента element, преобразовать значение к верхнему регистру, цвет текста – зеленый.

Открыть index2.html в браузере. В отчет исходный код, скриншот результата и скриншот дерева DOM со всеми раскрытыми элементами.

Пример для верхнего регистра и красного цвета: