



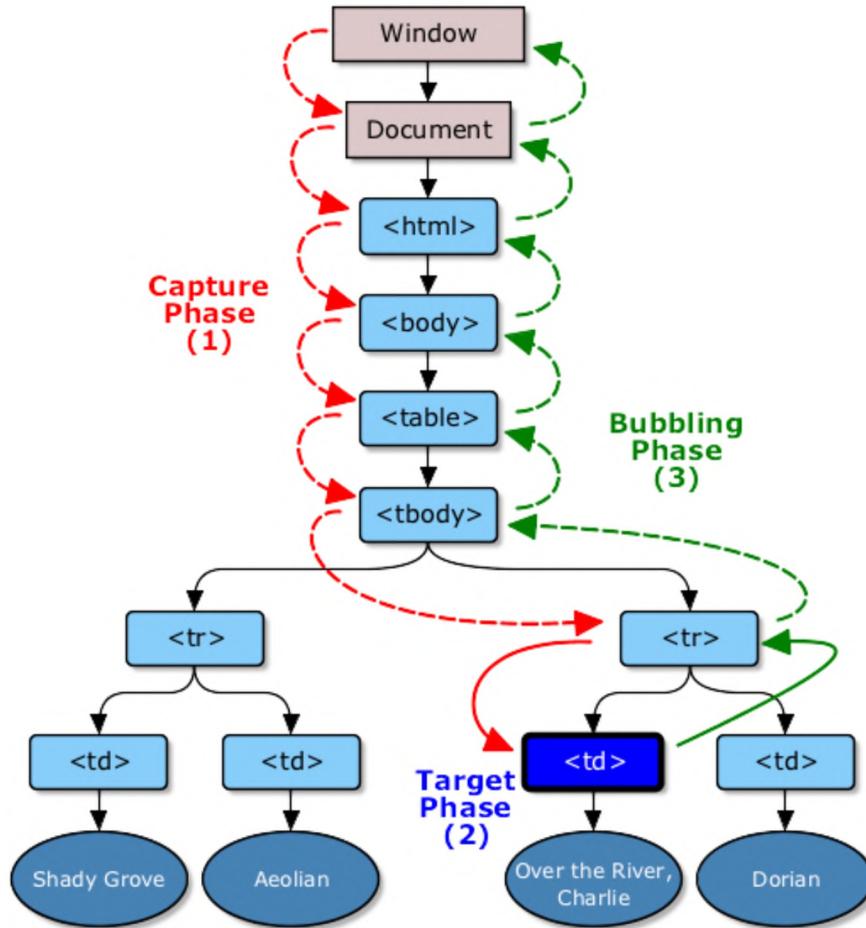
# Полный курс

Lesson 4

Работа с DOM в jQuery

# Работа с DOM в jQuery

## Что такое DOM?



HTML-документы имеют иерархическую структуру, представленную в DOM дереве. Узлы дерева представляют различные типы содержимого документа.

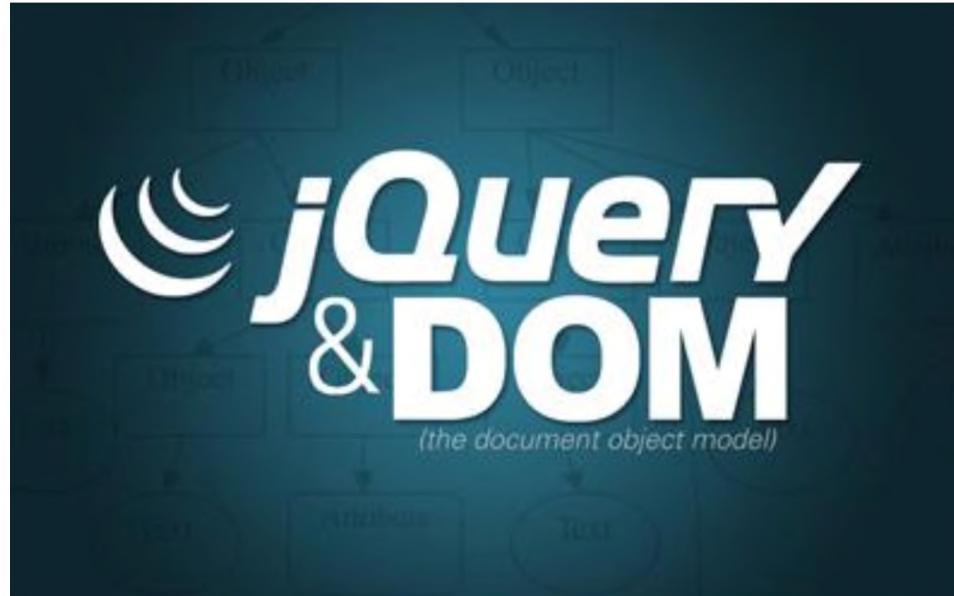
В первую очередь, древовидное представление HTML-документа содержит узлы, представляющие элементы, такие как `<a>`, `<p>`, `<div>` и узлы, представляющие строки текста.

# Работа с DOM в jQuery

## Организация работы с DOM в jQuery

jQuery — мощная библиотека JavaScript, ориентированная на упрощение взаимодействия с HTML через JavaScript.

В jQuery предусмотрена работа с DOM-деревом при помощи специальных методов.



# Фильтрация DOM-дерева



## Filtering DOM

# Работа с DOM в jQuery

## Функции фильтрации набора элементов

Функция	Описание
.eq(index)	Сокращает число совпадающих элементов до одного. Аргументом является позиция элемента в наборе совпадающих элементов.
.not(selector)	Удаляет элементы, соответствующие указанному выражению, из набора совпадающих элементов.
.filter(function)	Удаляет все элементы, которые не удовлетворяют требованиям функции, из набора совпадающих элементов.
.is(selector)	Проверяет текущий набор элементов на соответствие указанному выражению и возвращает true, если хотя бы один элемент соответствует выражению.
.slice(index, index)	Выделяет подмножество из набора совпадающих элементов.

# Методы фильтрации набора элементов

## Метод eq(index)

.eq(index) - сокращает число совпадающих элементов до одного. Аргументом является позиция элемента в наборе совпадающих элементов, начинается с 0 и продолжается до length-1. Указание некорректного индекса влечет за собой возвращение пустого набора элементов, а не нуля, поскольку запрос отфильтровывает все элементы, не соответствующие указанному индексу.

[0]	<p> Параграф №1 </p>	Параграф №1
[1]	<p> Параграф №2 </p>	Параграф №2
[2]	<p> Параграф №3 </p>	Параграф №3
[3]	<p> Параграф №4 </p>	Будет выбран элемент из предыдущей выборки под индексом “2”
[4]	<p> Параграф №5 </p>	Параграф №4
[5]	<p> Параграф №6 </p>	Параграф №5
		Параграф №6

```
$(“p”).eq(2).css(“color”, “Red”);
```

# Методы фильтрации набора элементов

## Метод not(selector)

.not(selector) - удаляет элементы, соответствующие указанному выражению, из набора совпадших элементов. Выражение может быть как отдельным селектором, так и сложной селекторной конструкцией.



```
$(“p”).not(“:eq(2)”).css(“color”, “Red”);
```

# Методы фильтрации набора элементов

## Методы filter()

.filter(selector) - удаляет все элементы, которые не соответствуют указанному выражению, из набора совпавших элементов.

```
$(“p”).filter(“.class1”).css(“background-color”, “Red”);
```

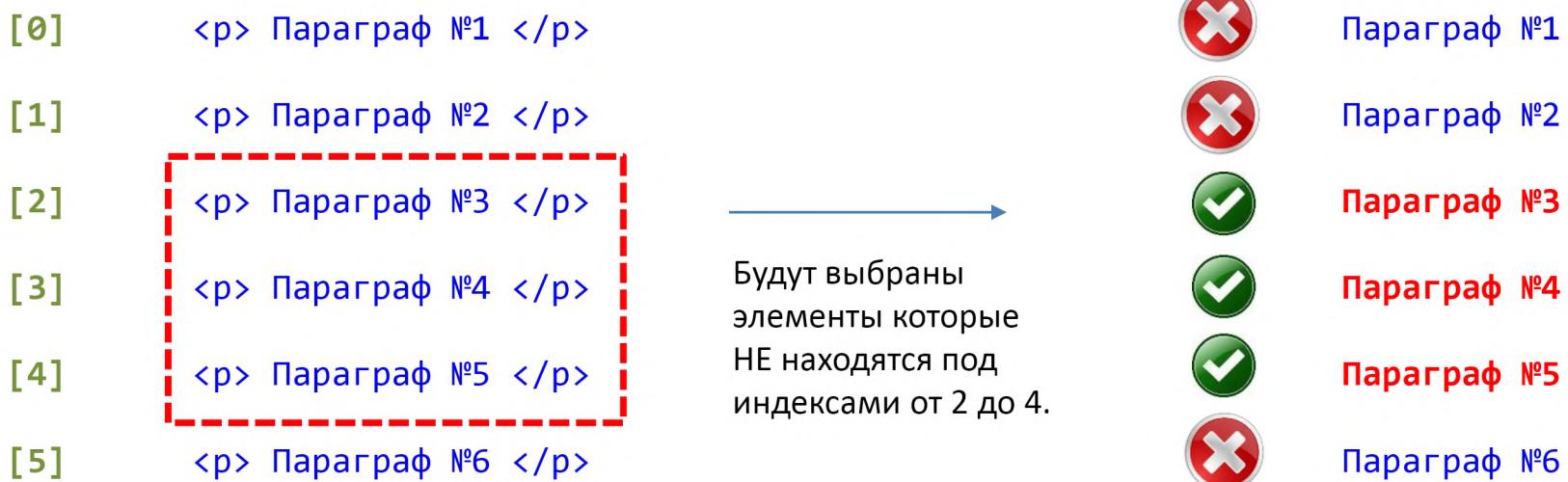
.filter(function). Если передать методу filter() некоторую функцию, он вызовет ее для каждого элемента в наборе и исключит те элементы, для которых эта функция вернет значение false.

```
$(“p”).filter(  
    function(i) {  
        return i != 3 && i < 7;  
    }  
).css(“background-color”, “Red”);
```

# Методы фильтрации набора элементов

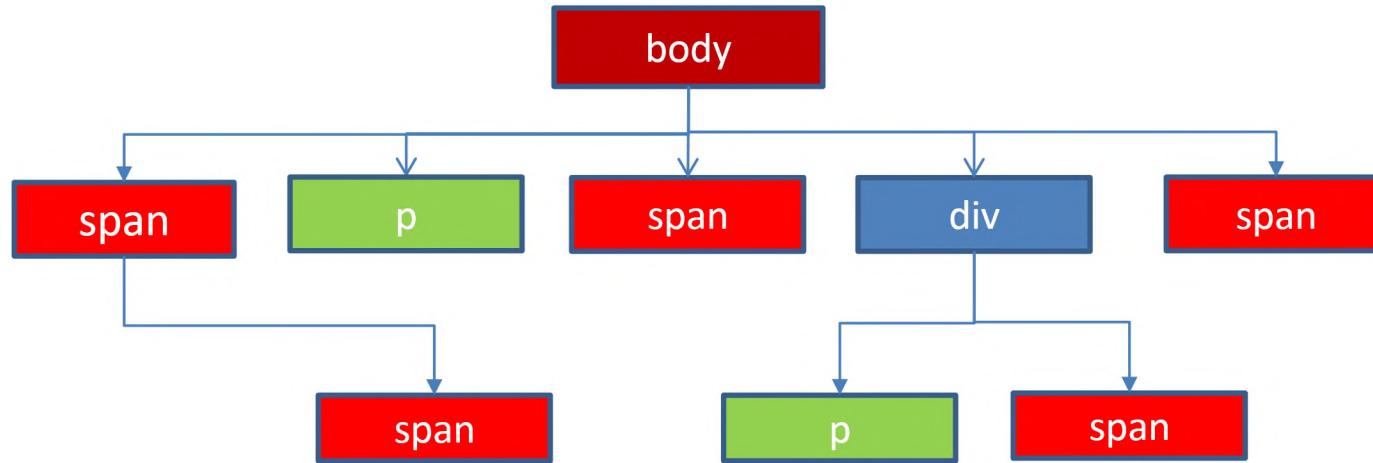
## Метод slice(start, stop)

.slice(start, stop) - выделяет подмножество из набора совпадших элементов. Ведет себя точно также как и встроенный метод разделения массивов. (допускаются отрицательные значения). Не включая последний индекс, но включая первый.



```
$(“p”).slice(2 , 5).css(“color”, “Red”);
```

# Перемещение по DOM-дереву



## Traversing DOM

# ФУНКЦИИ ПОИСКА ПО DOM В jQuery

## Traversing

Функция	Описание
.add(выражение)	добавляет в уже существующий набор дополнительные элементы, которые удовлетворяют указанному выражению.
.find(выражение)	отыскивает дочерние элементы, которые удовлетворяют указанному выражению.
.contents()	поиск всех дочерних узлов в наборе совпадших элементов (включая текстовые) или в содержимом документа, если он является фреймом.
.children()	получает набор элементов, содержащий всех непосредственных уникальных потомков для каждого совпадшего элемента.
.sibling()	получает набор элементов, содержащий все уникальные родственные элементы для набора совпадших элементов.
.andSelf()	Добавление предыдущего набора к текущему набору.
.end()	Отменяет последнее деструктивное действие.

# Перемещение по DOM-дереву

## Метод add(selector)

.add(выражение) - добавляет в уже существующий набор дополнительные элементы, которые удовлетворяют указанному выражению.

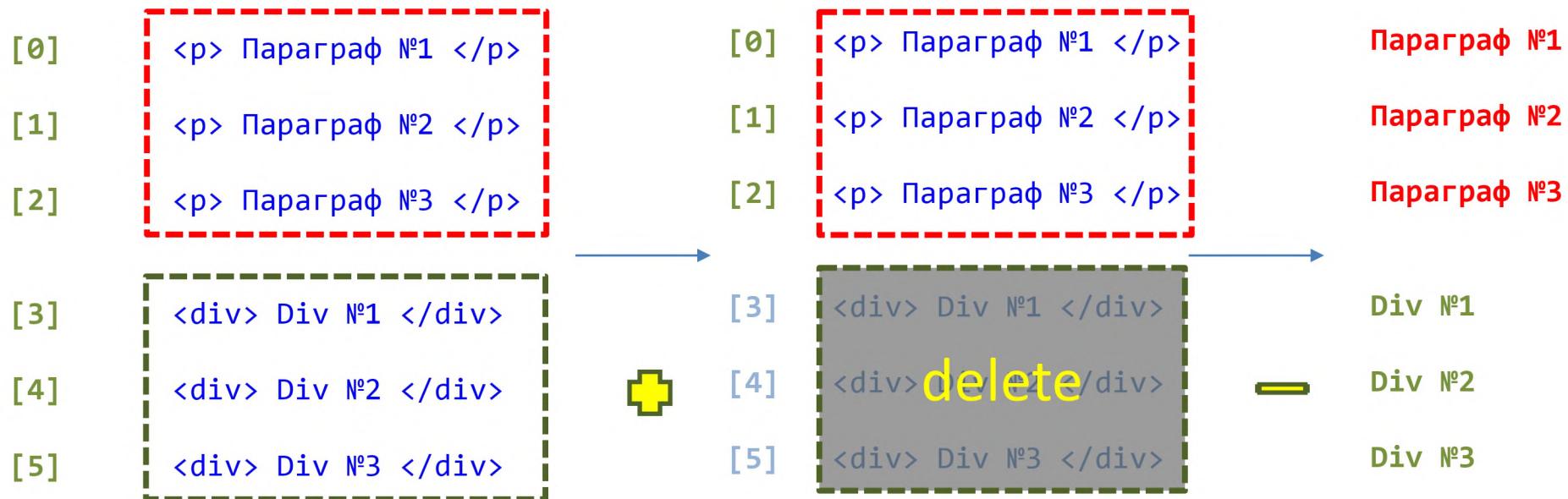


```
$(“p”).add(“div”).css(“color”, “Red”);
```

# Методы фильтрации набора элементов

## Метод end()

.end() – отменяет последний фильтр, возвращая тем самым набор элементов к его предыдущему состоянию (до фильтрации)

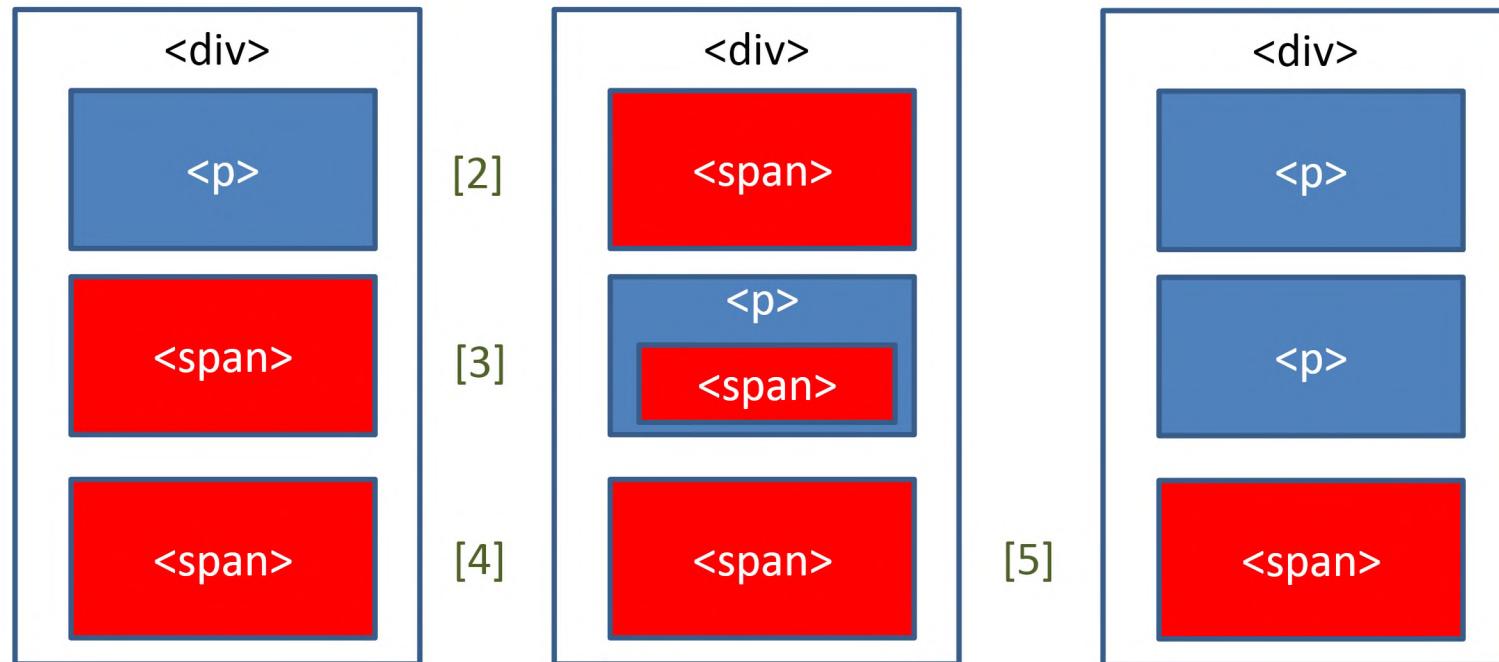


```
$(“p”).add(“div”).css(“color”, “green”).end().css(“color”, “Red”);
```

# Методы фильтрации набора элементов

## Метод `find(selector)`

`.find(выражение)` - отыскивает дочерние элементы, которые удовлетворяют указанному выражению.

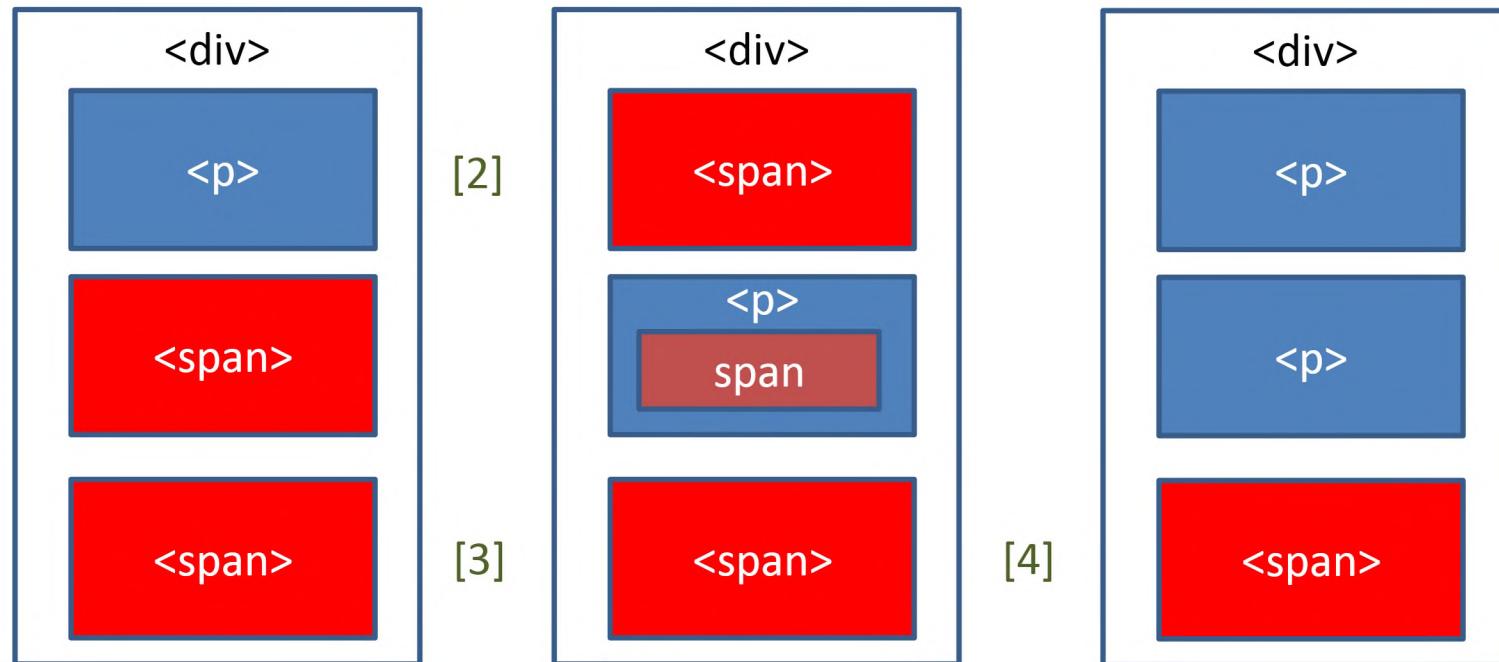


```
$(“div”).find(“span”).css(“background-color”, “Red”);
```

# Методы фильтрации набора элементов

## Метод children(selector)

.children(выражение) - получает набор элементов, содержащий всех непосредственных уникальных потомков для каждого совпадшего элемента.



```
$(“div”).children(“span”).css(“background-color”, “Red”);
```

# ФУНКЦИИ ПОИСКА ПО DOM В jQuery

## Traversing

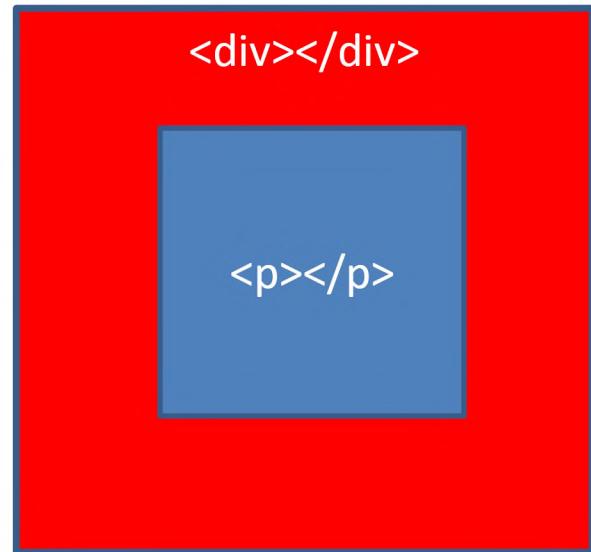
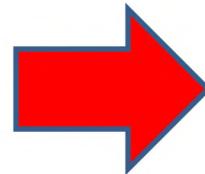
Функция	Описание
.parent(выражение)	Получает прямого “родителя” элемента.
.parents(выражение)	Получает набор элементов, содержащий уникальных “родителей” для совпадших элементов (кроме корневого элемента).
.closest()	Получает набор, содержащий ближайшие родительские элементы, которые удовлетворяют указанному селектору, включая начальный элемент.
.offsetParent()	Возвращает коллекцию jQuery с “родителем” по позиционированию первого совпадшего элемента.
.next(), .prev()	Получает набор элементов, содержащий уникальные последующие(предыдущие) родственные элементы для всех элементов существующего набора.
.nextAll(), .prevAll()	Поиск всех родственных элементов после текущего элемента (перед текущим элементом).
.nextUntil(селектор), .prevUntil(селектор)	Поиск всех родственных элементов после текущего элемента (перед текущим элементом) до указанного селектора.

# Методы фильтрации набора элементов

## Метод parent()

.parent(выражение) - возвращает ссылку на ближайшего родителя.

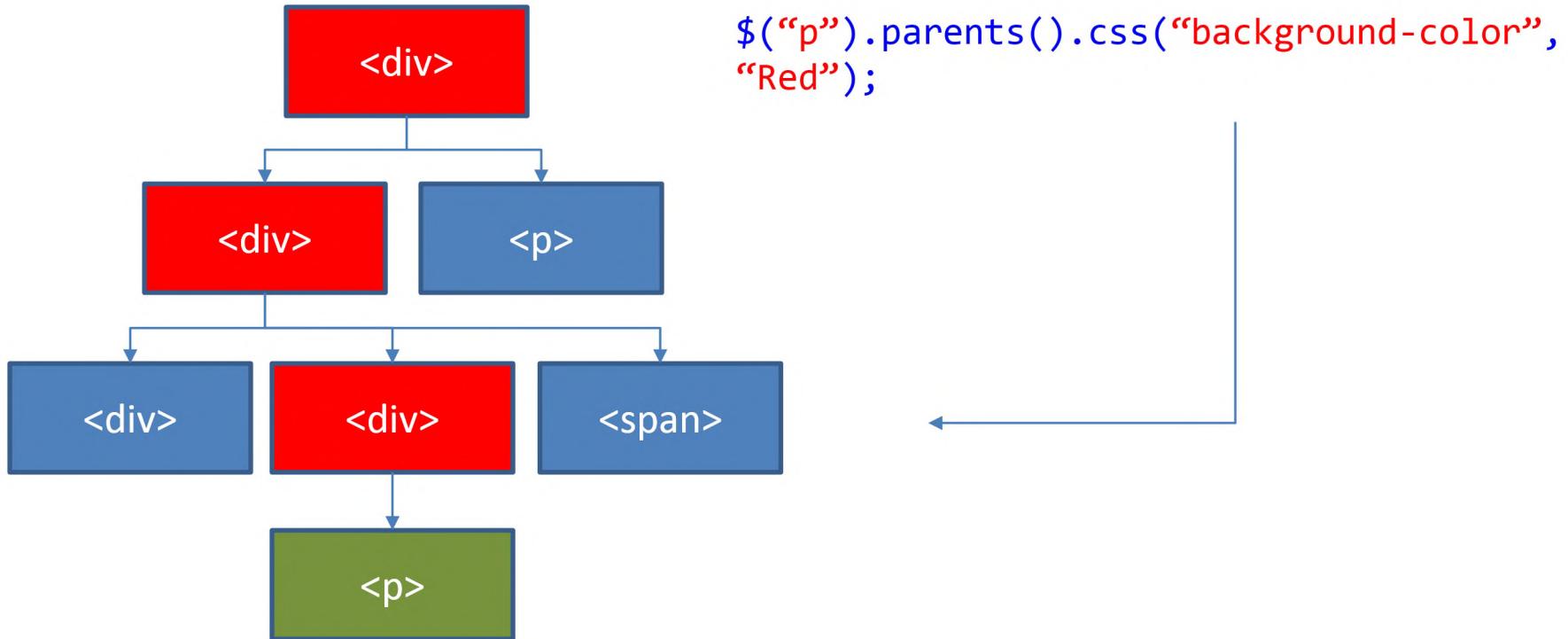
```
$(“p”).parent()  
.css(“background-color”, “Red”);
```



# Методы фильтрации набора элементов

## Метод `parents()`

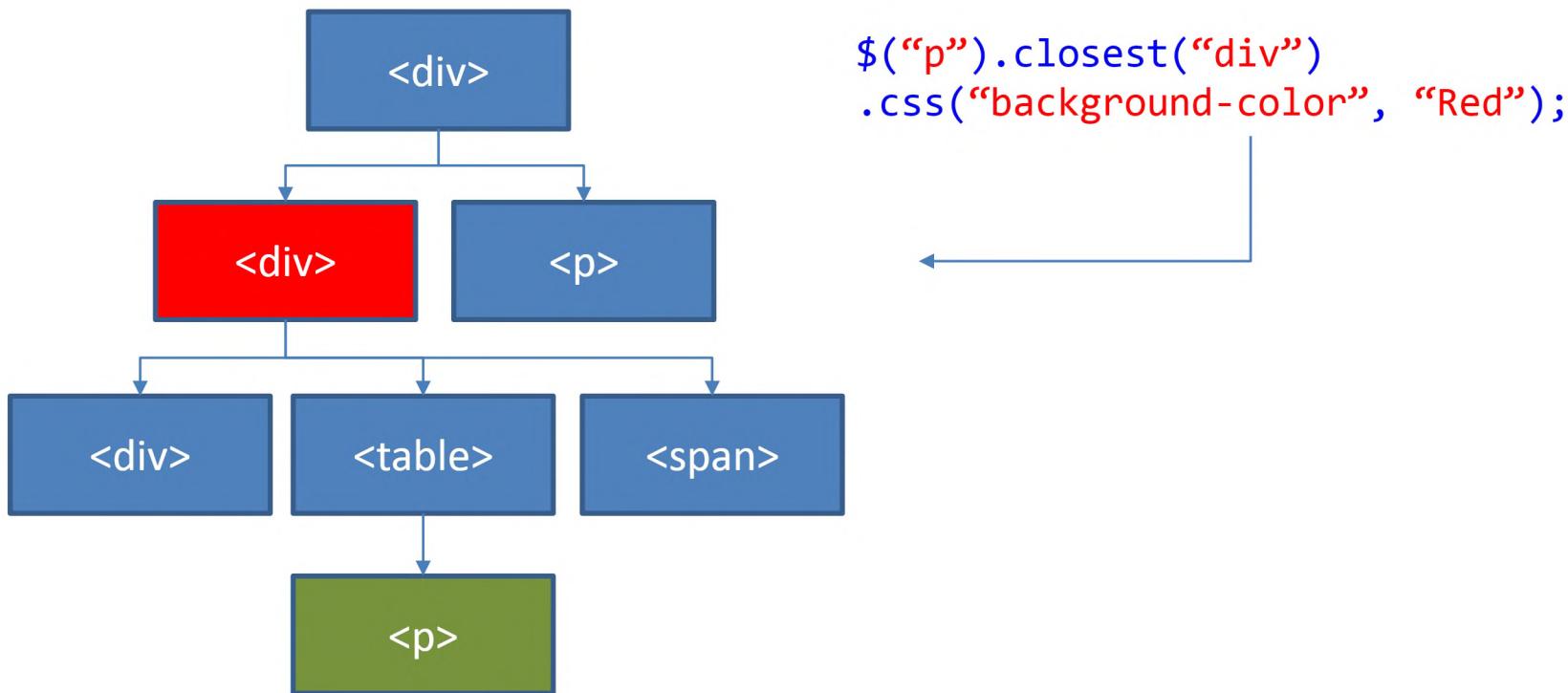
`.parents()` - поиск всех предков выбранных элементов. Будут выбраны не только прямые родители, но и прародители, прапрародители и так далее до начала DOM-дерева.



# Методы фильтрации набора элементов

## Метод closest(selector)

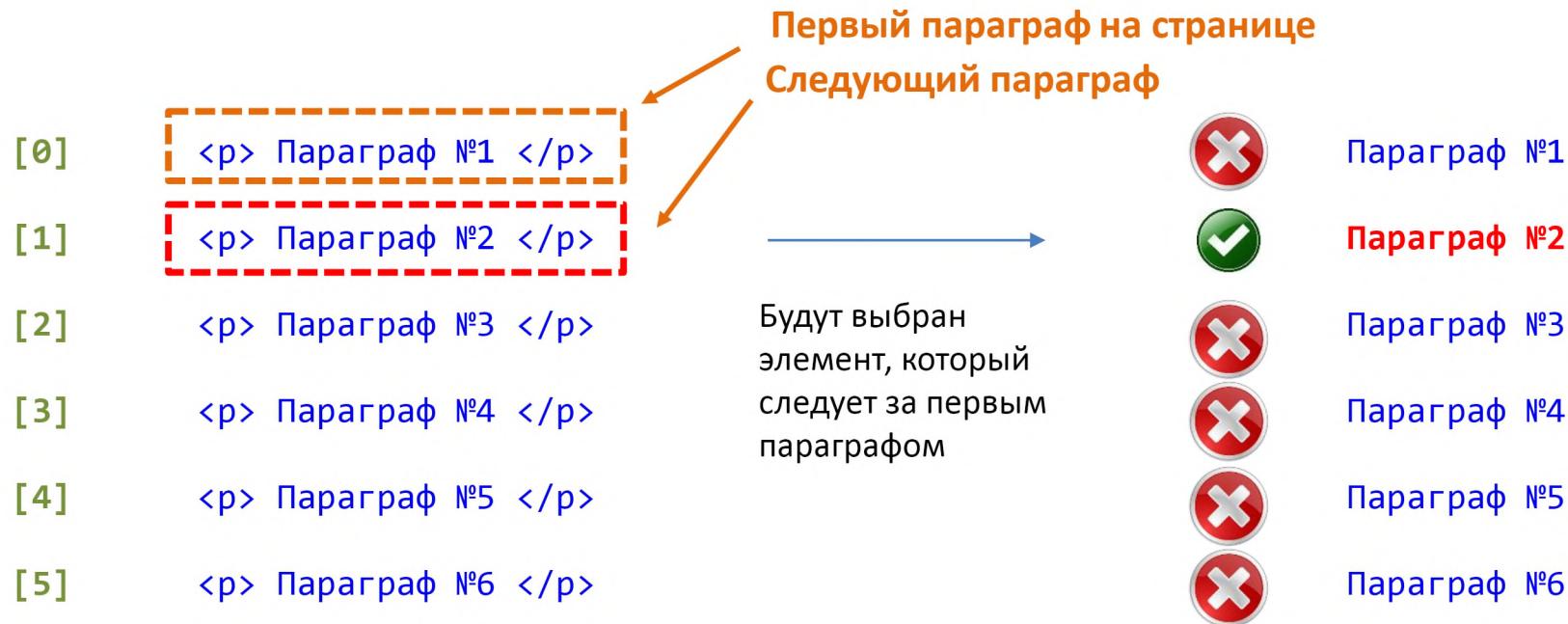
.closest(выражение) - начинает поиск непосредственно с выбранного (выбранных) элемента и движется вверх по иерархии. Двигаясь вверх по иерархии останавливает поиск после первого подходящего элемента (после чего начинает осуществлять поиск для следующего выбранного элемента). Поэтому, находит не более одного элемента для каждого из выбранных.



# Методы фильтрации набора элементов

## Метод next()

.next() - получает набор элементов, содержащий только соседние элементы, следующие непосредственно за указанным элементом.

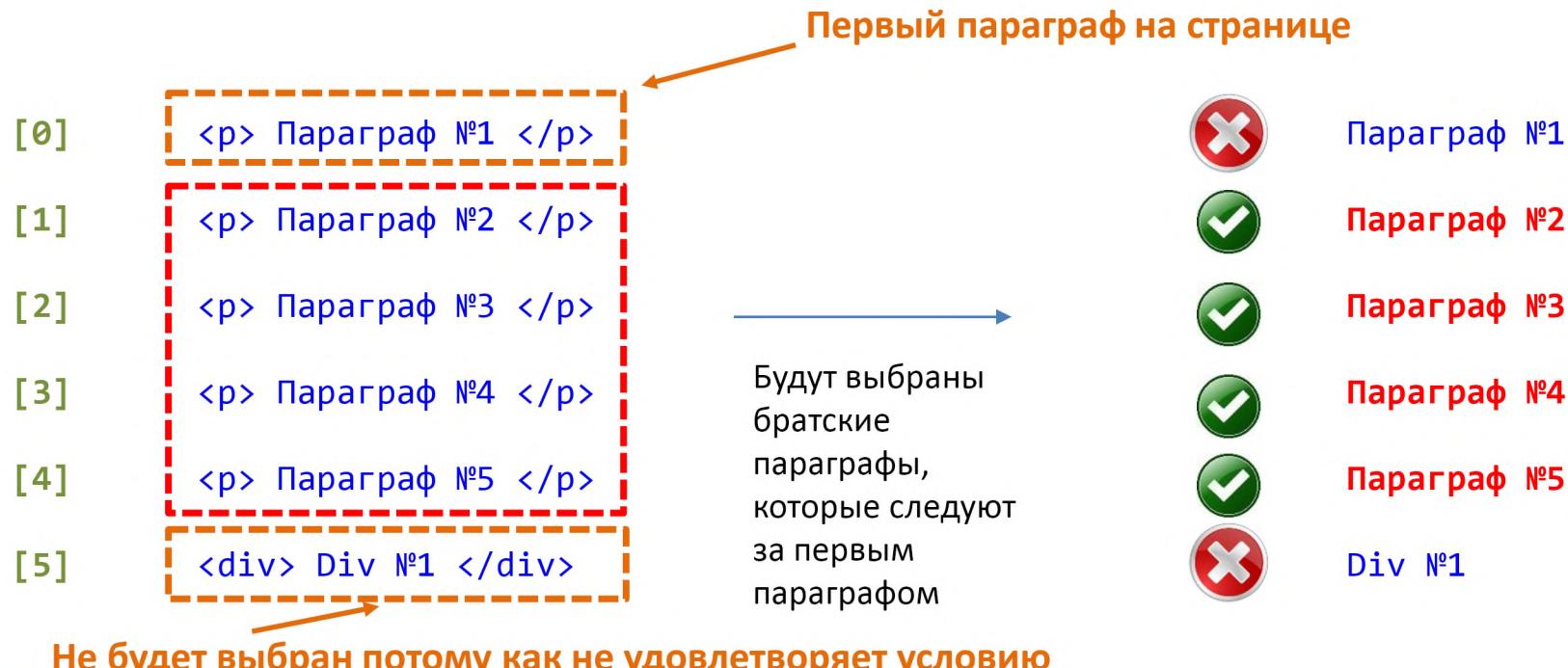


```
$(“p:first”).next().css(“color”, “Red”);
```

# Методы фильтрации набора элементов

## Метод nextAll()

.nextAll() – отыскивает все последующие братские элементы, которые соответствуют селектору и идут после выбранного элемента..

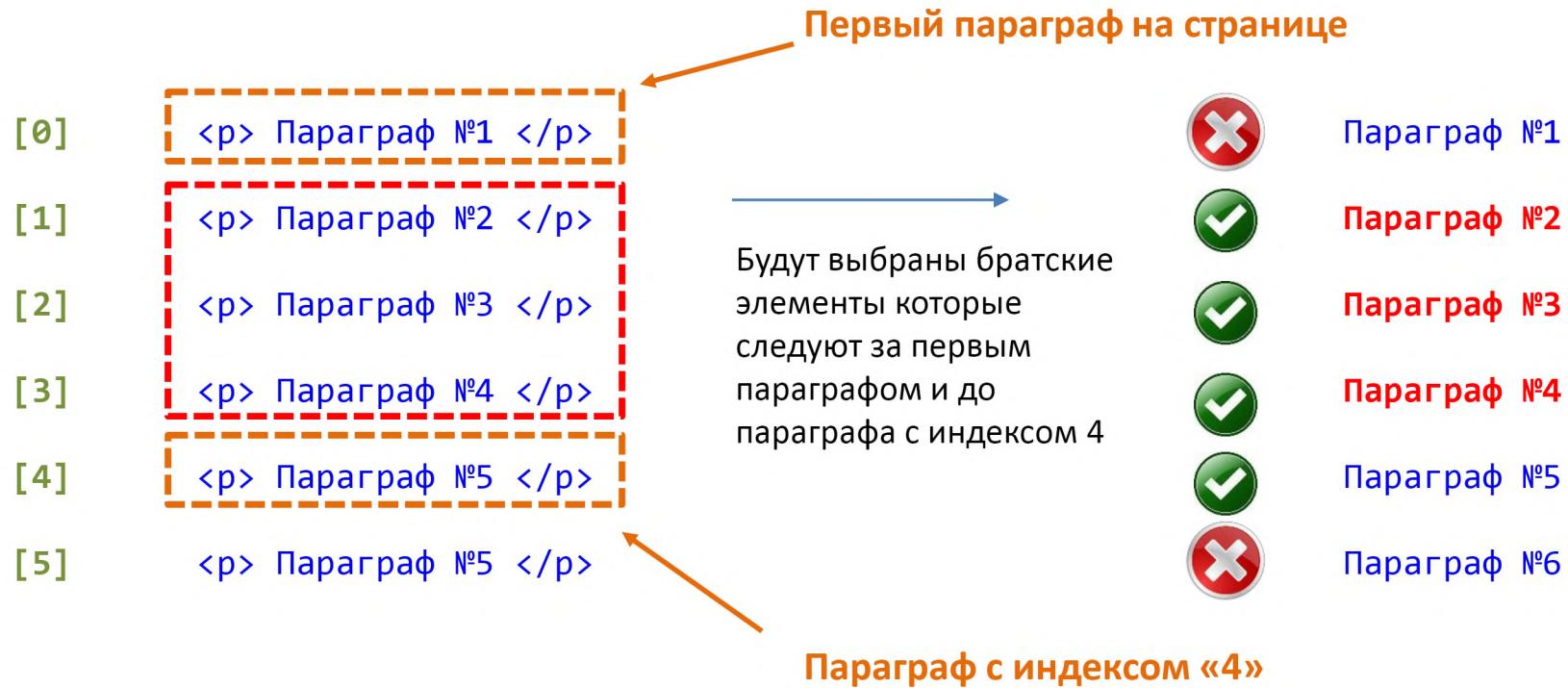


```
$(“p:first”).nextAll(“p”).css(“color”, “Red”);
```

# Методы фильтрации набора элементов

## Метод nextUntil()

.nextUntil() – получает всех последующих братьев каждого элемента до, но не включая элемент, соответствующий переданному селектору, узлу DOM или объекту jQuery.



```
$(“p:first”).nextUntil(“p:eq(4)”).css(“color”, “Red”);
```