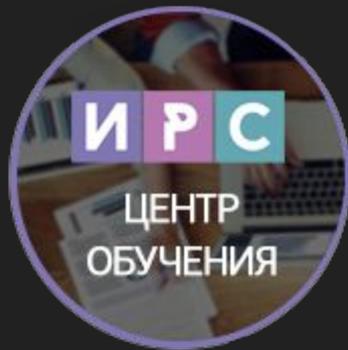


Курс вёрстки и программирования сайтов

goo.gl/hZdroj



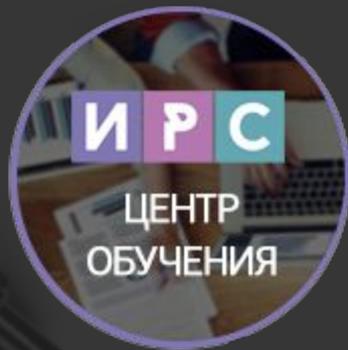
Ямасыпов Виталий

вконтакте: vk.com/snake_yava

почта: snake-yava@mail.ru

skype: [snake-yava](https://www.skype.com/people/snake-yava)

icq: 366696661



JavaScript

```
<!DOCTYPE html PUBLIC  
<html xmlns="http://www  
<head>  
  <meta name="TITLE"  
  <meta http-equiv="c
```

Размещение операторов языка JS на странице

1. Задание операторов языка внутри тэга <script> языка HTML.

```
<script language="JavaScript">  
    <!-- операторы //-->  
</script>
```

2. Задание файла с кодом JavaScript.

```
<script language="JavaScript" src="http://url/file.js"></script>
```

3. Использование выражений JavaScript в качестве значений параметров тэгов HTML.

```
<a href="javascript: window.open('name.htm', '_self')">  
      
</a>
```

4. Определение обработчика событий в тэге HTML.

Что такое jQuery?

jQuery - это библиотека, которая значительно упрощает и ускоряет написание JavaScript кода.

Девиз jQuery "write less, do more" (пиши меньше, делай больше) отражает ее главное предназначение.

jQuery позволяет создавать анимацию, обработчики событий, значительно облегчает выбор элементов в DOM и создание AJAX запросов.

Для jQuery написано огромное количество плагинов, которые позволяют расширить ее возможности еще больше.

Русская документация по ссылке jquery-docs.ru

Добавление jQuery на страницы

```
<script type="text/javascript" src="
путь_к_скачанному_файлу/jquery.js"></script>
```

```
<script type="text/javascript">
```

```
//здесь мы будем писать наш JS с использованием jQuery
```

```
</script>
```

Команды jQuery

Стандартный синтаксис jQuery команд:

```
$(селектор).метод();
```

Пример использования jQuery

```
$(document).ready(function(){
```

```
  $(".button").click(function(){
```

```
    $(".button").hide();
```

```
    $("#wrap1").addClass("add");
```

```
    $("#wrap1").animate({height:280}, 2000);
```

```
    $("#wrap1").animate({width:400}, 2000);
```

```
    $("#wrap1").animate({padding:20},2000,function(){
```

```
      $("#text1").hide(2000,function(){$("#text2").show(2000);});
```

```
    });
```

```
  });
```

```
});
```

Команды jQuery

Стандартный синтаксис jQuery команд:

```
$(селектор).метод();
```

- Знак \$ сообщает, что символы идущие после него являются jQuery кодом;
- Селектор позволяет выбрать элемент на странице;
- **Метод** задает **действие**, которое необходимо совершить над выбранным элементом.

Методы в jQuery

Методы в jQuery разделяются на следующие группы:

- Методы для манипулирования DOM;
- Методы для оформления элементов;
- Методы для создания AJAX запросов;
- Методы для создания эффектов;
- Методы для привязки обработчиков событий.

Селекторы jQuery аналогичны селекторам CSS

Пример	Результат
<code>\$("#p")</code>	Будут выбраны все элементы p, которые находятся на странице.
<code>\$(".par")</code>	Будут выбраны все элементы на странице с class="par".
<code>\$("#par")</code>	Будет выбран первый элемент на странице с id="par".
\$(this)	Позволяет выбрать текущий HTML элемент. Щелкните на <code>\$(this)</code> слева, чтобы посмотреть пример использования данного селектора в онлайн редакторе.
<code>\$("#p.par")</code>	Будут выбраны все элементы p на странице с class="par".
<code>\$("#p#par")</code>	Будут выбраны все элементы p на странице с id="par".
<code>\$(".par,.header,#heat")</code>	Будут выбраны все элементы на странице со значениями атрибутов class="par", class="header" и id='heat'.
<code>\$("#[src]")</code>	Будут выбраны все элементы на странице, имеющие атрибут src.
<code>\$("#[src='значение']")</code>	Будут выбраны все элементы со значениям атрибута src="значение".
<code>\$("#[src\$='.gif']")</code>	Будут выбраны все элементы со значениями атрибута src заканчивающимися на .gif.
<code>\$("#div#wrap .par1")</code>	Будут выбраны все элементы с class=par1, которые находятся внутри элементов div с id=wrap.
<code>\$("#:input")</code>	Будут выбраны все input элементы на странице.
<code>\$("#:тип")</code>	Будут выбраны все input элементы с <code><input type='тип' /></code> . Например <code>:button</code> выберет все <code><input type='button' /></code> элементы, <code>:text</code> выберет все <code><input type='text' /></code> элементы и т.д.

Предотвращение преждевременного выполнения кода

```
$(document).ready(function(){
```

Код который будет выполнен после полной загрузки документа

```
});
```

Изменение содержимого элементов с помощью jQuery

С помощью метода `html()` Вы можете изменить или узнать внутреннее содержимое выбранного элемента.

С помощью метода `append()` Вы можете вставить произвольный текст после внутреннего содержимого выбранного элемента.

С помощью метода `prepend()` Вы можете вставить произвольный текст перед внутренним содержимым выбранного элемента.

С помощью метода `attr()` Вы можете узнать или изменить содержимое указанного атрибута у выбранного элемента.

С помощью метода `removeAttr()` Вы можете удалить указанный атрибут у выбранного элемента.

Метод `wrap()` позволяет "обернуть" выбранный элемент указанными тэгами.

Изменение содержимого элементов с помощью jQuery

```
//Узнаем содержимое выбранного элемента
```

```
$("#селектор").html();
```

```
//Изменим содержимое выбранного элемента
```

```
$("#селектор").html("новое содержимое");
```

Цепочки команд в jQuery

Для того, чтобы сократить размер кода Вы можете соединять команды jQuery в цепочки.

Команды в цепочке будут выполняться поочередно слева направо.

```
<script type='text/javascript'>
```

```
//Код без сокращения
```

```
$("p").css("color","green");
```

```
$("p").css("font-size","30px");
```

```
//Сокращенный код
```

```
$("p").css("color","green").css("font-size","30px");
```

```
</script>
```

jQuery введение. Задания

1. Откройте верстку <http://codepen.io/htmllab/pen/NNaGOV>
2. Выберите при помощи селекторов jQuery
3. — все HTML-элементы `strong` и окрасьте их в зеленый цвет
4. — найдите все HTML-элементы `em` и добавьте им класс `.selected`
5. — Найдите все элементы `mark`, которые находятся в `div` с классом `gow` и задайте им класс `.selected`
6. — Найдите все гиперссылки и удалите у них подчеркивания
7. — Найдите все HTML-элементы, который содержат слово «Задания» и находятся в элементе с классом `.container`
8. — Переключите элементы `strong` с классом `some` в состояние без этого класса, а тем элементам (`strong`), у которых не было этого класса — добавьте.

Задание

Обработчики событий jQuery

Обработчики событий - это функции, код которых выполняется только после совершения определенных действий.

Обработчики событий есть и в чистом JavaScript, но jQuery облегчает их использование и расширяет их функциональность.

Примеры действий, после которых выполняются обработчики:

- Курсор мыши наведен на элемент;
- Веб-страница или картинка полностью загружена;
- Изменено содержимое поля формы;
- HTML форма отправлена;
- Нажата клавиша на клавиатуре;

Обработчики событий jQuery

Обработчики событий - это функции, код которых выполняется только после совершения определенных действий.

Обработчики событий есть и в чистом JavaScript, но jQuery облегчает их использование и расширяет их функциональность.

Примеры действий, после которых выполняются обработчики:

- Курсор мыши наведен на элемент;
- Веб-страница или картинка полностью загружена;
- Изменено содержимое поля формы;
- HTML форма отправлена;
- Нажата клавиша на клавиатуре;

Обработчики событий jQuery

```
$(селектор).обработчик_события(  
    function() {  
        код_обработчика_события  
    }  
);
```

Обработчики событий jQuery. Пример.

```
$(document).ready(function(){
```

```
    $("p").mouseover(function(){
```

```
        $("p").css("color","green");
```

```
    });
```

```
    $("p").mouseout(function(){
```

```
        $("p").css("color","black");
```

```
    });
```

```
});
```

Обработчики событий jQuery. Пример.

```
$(document).ready(function(){
```

```
$(".logo").click(function(){alert("Вы нажали один раз на логотип!");});
```

```
$("#button1").dblclick(function(){alert("Вы нажали два раза на кнопку!");});
```

```
});
```

Код обработчика **focus()** будет выполнен, когда элемент станет активным.

Код обработчика **blur()** будет выполнен, когда элемент перестанет быть активным.

Код обработчика **change()** будет выполнен, при изменении содержимого элемента.

Задания по работе с событиями в jQuery

1. Создайте обработчик нажатия на кнопку, который будет менять цвет фона страницы.
2. Создайте HTML-форму и по нажатию на кнопку проверяйте заполненность всех полей. Если останутся незаполненные поля — выводите предупреждение и не отправляйте форму.
3. Создайте обработчик события на нажатие кнопок направлений (влево, вправо) так, чтобы HTML-элемент при однократном нажатии на кнопку анимированно сдвигался влево или вправо на 100px
4. Навесьте на событие движения мыши обработчик, который будет в консоль выводить координаты движения. Примечание: выводите их как x и y. Событие надо «повесить» на document
5. * Создайте эффект перетаскивания HTML-элементов с классом .drag. Примечание к задаче: напишите обработчики событий mousedown, mouseup и mousemove

Задание

Эффекты jQuery

С помощью jQuery методов `fadeOut()`, `fadeIn()` и `fadeTo()` Вы можете постепенно скрывать и отображать элементы анимировано.

//Позволяет постепенно скрыть выбранный элемент

```
$("#селектор").fadeOut(скорость,функция);
```

//Позволяет постепенно отобразить выбранный элемент

```
$("#селектор").fadeIn(скорость,функция);
```

//Позволяет постепенно скрыть/отобразить элемент до указанного значения прозрачности

```
$("#селектор").fadeTo(скорость,прозрачность,функция);
```

Эффекты jQuery. Пример

```
$(document).ready(function(){
```

```
$("#but1").click(function(){$("#ar1p").fadeOut(3000)});
```

```
$("#but2").click(function(){$("#par1").fadeIn(3000)});
```

```
$("#but3").click(function(){$("#par1").fadeTo(3000,0.3)});
```

```
$("#but4").click(function(){$("#par1").fadeTo(3000,0.8)});
```

```
$("#but5").click(function(){$("#par1").fadeOut(3000,function(){
```

```
    alert("Абзац был полностью скрыт.");
```

```
});
```

```
});
```

```
});
```

Эффекты jQuery

С помощью jQuery методов `slideUp`, `slideDown` и `slideToggle` Вы можете плавно изменять высоту выбранных элементов.

//Позволяет изменять высоту элемента до 0

```
$("#селектор").slideUp(скорость,функция);
```

//Позволяет плавно вернуть элементу его изначальную высоту

```
$("#селектор").slideDown(скорость,функция);
```

//При первом вызове будет действовать как `slideUp`, а при втором как `slideDown`

```
$("#селектор").slideToggle(скорость,функция);
```

С помощью метода `slideToggle` Вы можете создавать на Ваших страницах удобные выпадающие меню.

Задание

1. Добавить на странице 4 кнопки с соответствующими id.
2. При нажатии на кнопку с id=but1 абзац с id=par1 должен исчезать, а при нажатии на кнопку с id=but2 появляться.
3. При наведении курсора мыши на абзац с id=par2 он должен стать прозрачным (можно задавать любые значения прозрачности, но необходимо чтобы текст при этом был видим) в течении 3 секунд (3000 миллисекунд). При выведении курсора мыши за пределы абзаца он должен вернуть стандартные значения прозрачности.
4. При нажатии на кнопку с id=but3 высота элемента id=par1 должна быть уменьшена до 0 в течении 5 секунд. При повторном нажатии на кнопку должна быть возвращена стандартная высота.

Задание

1. После нажатия на кнопку с id=but1 содержимое невидимого абзаца с id=par1 должно быть считано и вставлено в абзац с id=par2.
2. После нажатия на кнопку с id=but2 в абзацы par3, par4, par5 и par6 должны добавляться недостающие слова. Слова при вставке должны быть выделены жирным шрифтом.
3. После нажатия на кнопку с id=but3 элементы с id=href1 и id=par7 должны стать оформленными соответственно своему содержимому.
4. После нажатия на кнопку с id=but4 элементы должны стать оформленными соответственно своему содержанию. (Необходимые стили уже заданы, нужно только обернуть абзацы в тэги с соответствующими id - style1, style2, style3, style4.)

Управление стилями в jQuery

jQuery имеет группу различных методов значительно упрощающих оформление элементов.

Одним из самых важных методов в этой группе является метод `css()`.

С помощью метода `css` Вы можете узнавать текущие или устанавливать новые значения свойств оформления элементов.

```
//Узнаем значение указанного CSS свойства выбранного элемента
```

```
$("#селектор").css("свойство");
```

```
//Установим новое значение указанному CSS свойству выбранного элемента
```

```
$("#селектор").css("свойство", "значение");
```

```
//Установим произвольные значения нескольким CSS свойствам выбранного элемента
```

```
$("#селектор").css({свойство1:значение1, свойствоN:значениеN});
```

Управление стилями в jQuery

С помощью метода `addClass` Вы можете добавить указанный класс выбранному элементу (данный метод, добавляя новый класс, не удаляет и не заменяет существующие).

С помощью метода `removeClass` Вы можете удалить указанный класс у выбранного элемента.

Метод `toggleClass` позволяет переключаться между удалением и добавлением класса выбранного элемента.

```
//Добавим указанный класс выбранному элементу
```

```
$("#селектор").addClass("имя_класса");
```

```
//Удалим указанный класс у выбранного элемента
```

```
$("#селектор").removeClass('имя_класса');
```

```
//При первом вызове указанный класс будет добавлен, а при втором удален.
```

```
$("#селектор").toggleClass('имя_класса');
```

Задания по работе с событиями в jQuery

1. Создайте несколько параграфов с элементами `strong` и `em`. Задайте обработчики событий наведения мышью на эти элементы: при наведении, цвет элементов должен отличаться от обычного, а размер плавно увеличиваться.
2. При наведении на элемент с идентификатором `#box`, поворачивать элемент с классом `.some` на 90 градусов, при событии выведения мыши с элемента возвращать `.some` в исходную позицию
3. Назначить обработчики событий для кнопок, такие, чтобы после нажатия на кнопку элемент должен двигаться на 100px в направлении, указанном кнопкой
4. При нажатии на любую точку страницы, плавно с анимацией перемещать к этому месту элемент.
5. Для изображений создать обработчики событий, чтобы работа скрипта была похожа на работу слайдера
6. Создать эффект лайтбокса средствами jQuery
7. Создать эффект карусели