

## Практическая работа №6.3

### Тема работы: «Создание анимации при помощи библиотеки jQuery»

#### 1 Цель работы

Формирование умений и навыков создания на веб-странице анимации с использованием библиотек jQuery.

#### 2 Задание

Реализовать анимационный эффект в соответствии с порядком выполнения работы.

#### 3 Оснащение работы

ПК, текстовый редактор, браузер, редактор исходного кода.

#### 4 Основные теоретические сведения

Библиотека jQuery содержит простейшие анимации (fadeIn(), hide(), slideDown() и т.д.) и позволяет создавать любую анимацию с помощью метода animate(), который имеет следующий синтаксис:

```
$(selector).animate({ params },speed,callback);
```

jQuery animate() – метод-оболочка, то есть работает с заранее выбранным набором элементов DOM, обернутых jQuery. Метод позволяет применять требуемые эффекты анимации к элементам в наборе. Осуществление вышесказанного на практике делается возможным благодаря наличию набора свойств CSS и значений, которые эти свойства примут по завершении исполнения анимации. Промежуточные значения, которые стиль будет иметь по достижении желаемого эффекта (автоматически контролируемые анимационным движком), определяются продолжительностью эффекта и функцией-easing.

Список подвергающихся анимации свойств CSS ограничивается теми, которые могут принимать численные значения. Значения могут быть абсолютными (например, 200), или относительными. При абсолютных значениях используемая по умолчанию единица jQuery – пиксели. Также можно использовать специфические единицы: em, rem или проценты. Чтобы задать относительное значение используются префиксы += или -=, префиксы направляют целевое относительное значение в положительном или отрицательном направлении, соответственно.

Нечисловые свойства, такие как цвет текста или цвет фона, можно анимировать с помощью сторонних плагинов.

Метод имеет две основные формы; большинство используемых параметров необязательные (заключены в квадратные скобки):

```
animate(properties[, duration][, easing][, callback])  
animate(properties[, options])
```

Параметры:

- `properties` (Объект): список CSS-свойств, содержащий достигаемые в конце анимации значения;

- `duration` (Число|Строка): продолжительность эффекта в миллисекундах или одна из заранее установленных строк: “slow” (600ms), “normal” (400ms), или “fast” (200ms). Значение по умолчанию – “normal”;

- `easing` (Строка): используемое при переходе (transition) имя функции-easing (которая определяет изменение скорости анимации). Значение по умолчанию “swing”;

- `callback` (Функция): функция, исполняемая по завершении анимации для каждого анимированного элемента;

- `options` (Объект): объект, содержащий набор свойств (дополнительные опции), которые будут переданы методу. Доступны следующие свойства:

- `always` (Функция): функция, вызываемая по завершении анимации или, когда исполнение анимации останавливается (но не завершается);

- `complete` (Функция): исполняемая функция по завершении анимации;

- `done` (Функция): вызываемая по завершении анимации функция;

- `duration` (Строка|Число): описана выше;

- `easing` (Строка): описана выше;

- `fail` (Функция): исполняется при неудачной попытке анимации;

- `progress` (Функция): Исполняется после каждого шага анимации.

Вызывается один раз для каждого элемента после исполнения его анимации;

- `queue` (Boolean): когда анимацию надо поместить в очередь эффектов/функций (effects queue (подробности ниже)). Значение по умолчанию – true;

- `specialEasing` (Объект): объект, параметрами которого являются CSS-свойства, чьими значениями являются функции-перехода;

- `start` (Функция): исполняется, когда анимации начинает совершаться;

- `step` (Функция): функция, вызываемая для каждого анимированного свойства каждого анимированного элемента.

Термин `easing` используется для описания способа, который управляет обработкой и задает скорость кадров анимации. Анимация выполняется последовательно, если опция `queue` задана как `true`, и параллельно (без очереди) – если как `false`.

```
<div class="rectangle">
  <div class="square-small"></div>
</div>
```

```
.rectangle
{
  width: 300px;
  height: 20px;
  display: block;
  position: relative;
  border: 1px solid black;
  margin: 20px 0;
}
```

```
.square-small
{
  display: block;
  width: 20px;
  height: 20px;
  position: absolute;
  background-color: red;
}
```

```
$('.rectangle')
  .find('.square-small')
  .animate({
    left: 280
  }, 'slow');
```

Возможен вариант анимирования нескольких свойств элемента одновременно, используя метод `animate()`. Все свойства анимируются одновременно, без каких-либо задержек.

Можно анимировать несколько свойств элемента по очереди используя возможность «сцепления» методов jQuery:

```
$("#button").click(function(){
  let div = $("#div");
  div.animate({height: '300px', opacity: '0.4'}, "slow");
  div.animate({width: '300px', opacity: '0.8'}, "slow");
  div.animate({height: '100px', opacity: '0.4'}, "slow");
  div.animate({width: '100px', opacity: '0.8'}, "slow");
});
```

В дополнение к числовым значениям, каждое свойство может принимать строки 'show', 'hide' и 'toggle'. Это можно использовать, когда свойства анимируются от их текущего значения до начального значения и наоборот.

## 5 Порядок выполнения работы

1. Создать html-документ.
2. Выполнить подключение библиотеки jQuery.
3. Реализовать анимационный эффект в соответствии с вариантом, таблица 8.1.

Таблица 8.1 – Варианты заданий

Номер варианта	Задание
1	реализовать сворачивание и разворачивание элементов списка
2	реализовать поочередное выделение тенью всех ссылок на странице
3	реализовать плавное выцветание и появление текста
4	реализовать плавное увеличение заголовка по очередному клику
5	реализовать плавное вращение текста по кругу
6	реализовать вращение круга по клику по часовой стрелке на один оборот
7	реализовать поочередное выделение элементов меню навигационной панели
8	реализовать плавное уменьшение заголовка по клику

4. Оформить отчет о проделанной работе.

## 6 Форма отчета о работе

*Практическая работа № \_\_\_\_\_*

*Номер учебной группы \_\_\_\_\_*

*Фамилия, инициалы учащегося \_\_\_\_\_*

*Дата выполнения работы \_\_\_\_\_*

*Тема работы: \_\_\_\_\_*

*Цель работы: \_\_\_\_\_*

*Оснащение работы: \_\_\_\_\_*

*Результат выполнения работы: \_\_\_\_\_*

\_\_\_\_\_

\_\_\_\_\_

## **7 Контрольные вопросы и задания**

1. Для чего предназначен метод `animate()`?
2. Какие свойства позволяет анимировать метод `animate()`?
3. Как можно реализовать одновременное анимирование свойств?
4. Как можно реализовать анимацию свойств по очереди?

## **8 Рекомендуемая литература**

**HTML5BOOK.RU** [Электронный ресурс] / Сайт для тех, кто изучает веб-технологии и создает сайты – Елена Назарова, 2014-2022. – Режим доступа: <https://html5book.ru>

**METANIT.COM** [Электронный ресурс] / Сайт о программировании – [metanit.com](https://metanit.com), 2012-2022. – Режим доступа: <https://metanit.com/web>

**Учебник jQuery для начинающих** [Электронный ресурс] / MSiter.ru HTML, CSS, XML – Режим доступа: <https://msiter.ru/tutorials/jquery>