

7

- Сделаем приложение для вывода имени на экран:

```
var name = prompt('Как тебя зовут?');
if(name) {
    document.write('Привет ' + name);
} else {
    alert('Не ввел имя!');
}
```

- В простом документе получим содержимое блока с id `myBlock`. Добавим к нему класс `new-class`, удалим класс, добавим несколько внутренних блоков, удалим первый из них, получим значения атрибутов, добавим обработчики событий.

ДЗ Нарисовать фигуры с помощью CSS (свойства устанавливать через JS):

- свойствами
 - классами
 - аппендить к боди `document.createElement`
 - менять выбранный элемент на нужную фигуру
- Выведем все типы данных в консоль, посмотрим как проверяются типы и как этим можно пользоваться. Пройдем циклом по массивам и объектам. Попробуем менять типы переменных разными способами.

ДЗ Функция рекурсивна выводящая все свойства объекта в `ul`

- функция проверки типа данных с выводом значения

```
[{
  "glossary": {
    "title": "example glossary",
    "GlossDiv": {
      "title": "S",
      "GlossList": {
        "GlossEntry": {
          "ID": "SGML",
          "SortAs": "SGML",
          "GlossTerm": "Standard Generalized Markup Language",
          "Acronym": "SGML",
          "Abbrev": "ISO 8879:1986",
          "GlossDef": {
            "para": "A meta-markup language, used to create markup lan
guages such as DocBook.",
            "GlossSeeAlso": ["GML", "XML"]
          },
          "GlossSee": true
        }
      }
    }
  }
},
...
]
```

- Проверим все примеры из презентации. Узнаем почему так.

ДЗ Написать функцию, которая выводит this, менять его через call / apply / bind и выводить деревом, выводить тип и значение

```

var app = (function(settings) {
    var settings = {
        name: "Default"
    };

    var tmpVar = false;

    return {
        init: function(settings) {

        },

        getTMP: function() {

        },

        setTMP: function() {

        },

        getThis: function() {

        }
    };
});

```

- Напишем 'класс', сделаем в нем инкапсулированные методы и переменные. Сделаем несколько доступных снаружи методов. Посмотри как инициировать класс, что потом с ним делать, как наследовать свойства.

ДЗ Сделать страницу и добавить events listeners для hover, focus, click, mouse events, вывести положение курсора, активный элемент, время ивента. Все listeners должны быть внутри класса.

```

<!doctype html>
<html>
<head>
<title>Listners</title>
<meta name="viewport" content="width=device-width">
</head>
<body>

<p><button id="hoverBtn">Hover event</button></p>
<p><button id="mouseoverBtn">Mouseover event</button></p>
<p><button id="mousemoveBtn">Mousemove event</button></p>
<p><button id="mouseleaveBtn">Mouseleave event</button></p>
<p><button id="clickBtn">Click event</button></p>
<p><input id="focusInput" type="text" value="Focus event"></p>
<p><input id="blurInput" type="text" value="Blur event"></p>
<p><input id="keyupInput" type="text" value="Keyup event"></p>
<p><input id="keydownInput" type="text" value="Keydown event"></p>

<div id="console">
    <strong>Events log:</strong><br/><br/>
</div>
</body>
</html>

```

- На основе предустановленных файлов с информацией попробуем различные способы обращения к данным с помощью AJAX. Используем для этого методы GET и POST. HTML:

```

<!doctype html>
<html>
<head>
<title>Justified Nav</title>
<meta name="viewport" content="width=device-width">
<link rel="stylesheet" href="https://netdna.bootstrapcdn.com/bootstrap/3.0.0/css/bootstrap.min.css"/>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/2.0.3/jquery.min.js"
    type="text/javascript"></script>
<script src="https://netdna.bootstrapcdn.com/bootstrap/3.0.0/js/bootstrap.min.js"
    type="text/javascript"></script>
<style type="text/css">
    body {
        padding-top: 20px;
    }

    .footer {
        border-top: 1px solid #eee;
        margin-top: 40px;
        padding-top: 40px;
        padding-bottom: 40px;
    }

```

```

}

/* Main marketing message and sign up button */
.jumbotron {
    text-align: center;
    background-color: transparent;
}
.jumbotron .btn {
    font-size: 21px;
    padding: 14px 24px;
}

/* Customize the nav-justified links to be fill the entire space of the .nav
bar */

.nav-justified {
    background-color: #eee;
    border-radius: 5px;
    border: 1px solid #ccc;
}
.nav-justified > li > a {
    padding-top: 15px;
    padding-bottom: 15px;
    color: #777;
    font-weight: bold;
    text-align: center;
    border-bottom: 1px solid #d5d5d5;
    background-color: #e5e5e5; /* Old browsers */
    background-repeat: repeat-x; /* Repeat the gradient */
    background-image: -moz-linear-gradient(top, #f5f5f5 0%, #e5e5e5 100%); /*
FF3.6+ */
    background-image: -webkit-gradient(linear, left top, left bottom, color-st
op(0%,#f5f5f5), color-stop(100%,#e5e5e5)); /* Chrome,Safari4+ */
    background-image: -webkit-linear-gradient(top, #f5f5f5 0%,#e5e5e5 100%); /
* Chrome 10+,Safari 5.1+ */
    background-image: -ms-linear-gradient(top, #f5f5f5 0%,#e5e5e5 100%); /* IE
10+ */
    background-image: -o-linear-gradient(top, #f5f5f5 0%,#e5e5e5 100%); /* Ope
ra 11.10+ */
    filter: progid:DXImageTransform.Microsoft.gradient( startColorstr='#f5f5f5
', endColorstr='#e5e5e5',GradientType=0 ); /* IE6-9 */
    background-image: linear-gradient(top, #f5f5f5 0%,#e5e5e5 100%); /* W3C */
}
.nav-justified > .active > a,
.nav-justified > .active > a:hover,
.nav-justified > .active > a:focus {
    background-color: #ddd;
    background-image: none;
    box-shadow: inset 0 3px 7px rgba(0,0,0,.15);

```

```

}
.nav-justified > li:first-child > a {
  border-radius: 5px 5px 0 0;
}
.nav-justified > li:last-child > a {
  border-bottom: 0;
  border-radius: 0 0 5px 5px;
}

@media (min-width: 768px) {
  .nav-justified {
    max-height: 52px;
  }
  .nav-justified > li > a {
    border-left: 1px solid #fff;
    border-right: 1px solid #d5d5d5;
  }
  .nav-justified > li:first-child > a {
    border-left: 0;
    border-radius: 5px 0 0 5px;
  }
  .nav-justified > li:last-child > a {
    border-radius: 0 5px 5px 0;
    border-right: 0;
  }
}

/* Responsive: Portrait tablets and up */
@media screen and (min-width: 768px) {
  /* Remove the padding we set earlier */
  .masthead,
  .marketing,
  .footer {
    padding-left: 0;
    padding-right: 0;
  }
}
</style>
</head>
<body>

<div class="container">

  <div class="masthead">
    <h3 class="text-muted">Project name</h3>
    <ul class="nav nav-justified">
      <li><a href="/data/text.json">Projects</a></li>
      <li><a href="/data/text.html">Services</a></li>
      <li><a href="/data/text.txt">Downloads</a></li>

```

```

        <li><a href="/data/text.js">About</a></li>
    </ul>
</div>

<!-- Jumbotron -->
<div class="jumbotron">
    <h1>Marketing stuff!</h1>
    <p class="lead">Cras justo odio, dapibus ac facilisis in, egestas eget quam. Fusce dapibus, tellus ac cursus commodo, tortor mauris condimentum nibh, ut fermentum massa justo sit amet.</p>
    <p><a class="btn btn-lg btn-success" href="#">Get started today</a></p>
</div>

<!-- Example row of columns -->
<div class="row">
    <div class="col-lg-4">
        <h2>Heading</h2>
        <p>Donec id elit non mi porta gravida at eget metus. Fusce dapibus, tellus ac cursus commodo, tortor mauris condimentum nibh, ut fermentum massa justo sit amet risus. Etiam porta sem malesuada magna mollis euismod. Donec sed odio dui. </p>
        <p><a class="btn btn-primary" href="#">View details &raquo;</a></p>
    </div>
    <div class="col-lg-4">
        <h2>Heading</h2>
        <p>Donec id elit non mi porta gravida at eget metus. Fusce dapibus, tellus ac cursus commodo, tortor mauris condimentum nibh, ut fermentum massa justo sit amet risus. Etiam porta sem malesuada magna mollis euismod. Donec sed odio dui. </p>
        <p><a class="btn btn-primary" href="#">View details &raquo;</a></p>
    </div>
    <div class="col-lg-4">
        <h2>Heading</h2>
        <p>Donec sed odio dui. Cras justo odio, dapibus ac facilisis in, egestas eget quam. Vestibulum id ligula porta felis euismod semper. Fusce dapibus, tellus ac cursus commodo, tortor mauris condimentum nibh, ut fermentum massa.</p>
        <p><a class="btn btn-primary" href="#">View details &raquo;</a></p>
    </div>
</div>

<!-- Site footer -->
<div class="footer">
    <p>&copy; Company 2014</p>
</div>

</div> <!-- /container -->
</body>
</html>

```

- В конце будет домашнее задание, на котором нужно будет сделать калькулятор на javascript, используя полученные ранее знания. Внешний вид сделать согласно прилагаемому изображению.

8

- Поработаем со структурой документа:

```
<!doctype html>
<head>
  <title>Document</title>
</head>
<body>
  <div class="wrapper">
    <header>
      <h1>Title</h1>
    </header>
    <ul class="navigation">
      <li><a href="">Link 1</a></li>
      <li><a href="">Link 2</a></li>
      <li><a href="">Link 3</a></li>
      <li><a href="">Link 4</a></li>
      <li><a href="">Link 5</a></li>
    </ul>
    <section>

    </section>
    <footer>
      &copy; Copyright
    </footer>
  </div>
</body>
</html>
```

ДЗ 1 К странице со структурой, описанной выше добавить кнопки, которые позволяют вставлять новые элементы списка со ссылками (первая кнопка), выводят в консоль количество элементов списка (вторая кнопка), выделяют третий элемент списка жирным и добавляют класс `active` (третья кнопка), следующий элемент списка после `active` делают желтым фоном (четвертая кнопка), удаляют последний элемент из списка (пятая кнопка). Сделать табы внутри `section`, три таба со случайным текстом внутри.

ДЗ 2 Построить дерево ДОМА по селектору из инпута

(Отдельный файл)

- Кнопки разных анимация для jquery и css. + бегущий человек из спрайта (fig06.jpg)
- Поработаем над AJAX запросами, заменив чистый JS на методы в jQuery.

ДЗ Наш ОРА перевести на hashchange + jquery ajax + handlebars (date helper)

- Сделаем управление ивентами с jQuery. Разберем, что такое делегирование. попробуем работать с мобильными ивентами, используя эмулятор в Google Chrome. Сделаем карусель, управляемую с помощью свайпов.
- Календарь из готового дизайна.