|  |
| --- |
| Пермский филиал федерального государственного автономного образовательного учреждения высшего образования  «Национальный исследовательский университет  «Высшая школа экономики»  *Факультет экономики, менеджмента и бизнес-информатики* |
|  |
| Чепоков Елизар Сергеевич  **WEB-Разработка**  *Лабораторные работы*  студента образовательной программы «Программная инженерия»  по направлению подготовки *09.03.04 Программная инженерия*   |  |  | | --- | --- | |  | Преподаватель кафедры инф. тех. в бизнесе НИУ ВШЭ- Пермь  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Д. Б. Кузнецов | |

Пермь, 2022 год

**Оглавление**

[**HTML** 3](#_Toc100012748)

[**Формы** 3](#_Toc100012749)

[**Таблицы** 5](#_Toc100012750)

[**Списки** 6](#_Toc100012751)

[**Блоки** 7](#_Toc100012752)

[**CSS** 8](#_Toc100012753)

[**Виды адресов** 8](#_Toc100012754)

[**Блочная модель и grid** 9](#_Toc100012755)

[**JavaScript** 10](#_Toc100012756)

[**События** 10](#_Toc100012757)

[**DOM** 11](#_Toc100012758)

[**Массивы** 13](#_Toc100012759)

[**Циклы** 13](#_Toc100012760)

[**Отображения** 13](#_Toc100012761)

[**Arrow functions** 13](#_Toc100012762)

[**JQuery** 13](#_Toc100012763)

[**DOM** 13](#_Toc100012764)

[**Плагины** 13](#_Toc100012765)

[**Ajax** 14](#_Toc100012766)

[**AngularJS** 15](#_Toc100012767)

[**Связывание** 15](#_Toc100012768)

[**Вывод «Таблицы»** 15](#_Toc100012769)

[**Фильтры по «Таблице»** 16](#_Toc100012770)

[**React** 16](#_Toc100012771)

[**Состояние** 16](#_Toc100012772)

[**Рендер** 17](#_Toc100012773)

[**Bootstrap** 19](#_Toc100012774)

[**Сетка** 19](#_Toc100012775)

[**Адаптивная сетка** 19](#_Toc100012776)

[**Глифы** 20](#_Toc100012777)

[**Листинг сайта** 20](#_Toc100012778)

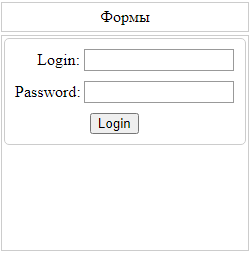
# **HTML**

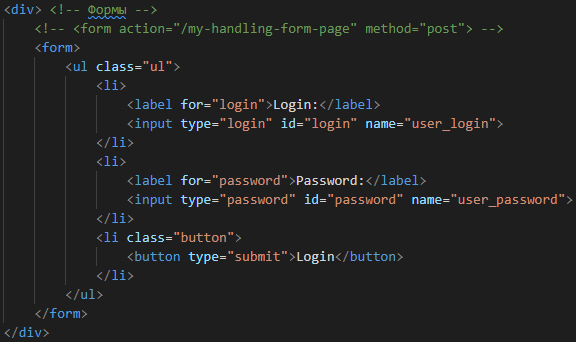
## **Формы**

HTML-формы — это простые элементы управления HTML, которые применяются для сбора информации от посетителей веб-сайта.

Типы:

* <input> с разными типами
  + type="text"
  + type="password"
  + type="checkbox"
  + type="radio"
  + type="button"
* <label>
* <input>
* <textarea>
* <button>

**Отображение в браузере:**

**Листинг верстки:**

**Листинг стилей:**

form {

     margin: 0 auto; /\* Расположим форму по центру страницы \*/

     width: 220px;

     padding: 10px; /\* контур формы \*/

     border: 1px solid #CCCCCC;

     border-radius: 5px;

}

.ul {

     list-style: none;

     padding: 0;

     margin: 0;

}

form li + li {

     margin-top: 10px;

}

label {

     display: inline-block;

     width: 65px;

     text-align: right;

}

input,

textarea {

     font: 1em sans-serif;

     width: 150px; /\* размер полей \*/

     box-sizing: border-box;

     border: 1px solid #999999; /\* границы полей \*/

}

input:focus,

textarea:focus { /\* подсветка для элементов в фокусе \*/

     border-color: #000000;

}

.button {

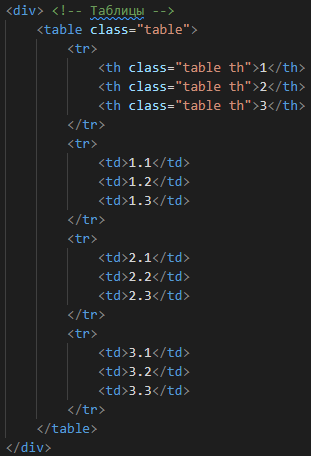
     padding-left: 65px;

     margin-left: 10px;

}

## **Таблицы**

**Отображение в браузере:**

**Листинг верстки:**

**Листинг стилей:**

.table {

    width: 500px;

    margin-bottom: 20px;

    border: 1px solid #dddddd;

    border-collapse: collapse;

    margin: 0 auto;

}

.table th {

    font-weight: bold;

    padding: 5px;

    background: #efefef;

    border: 1px solid #dddddd;

}

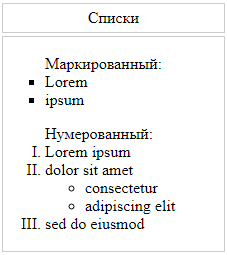
table td {

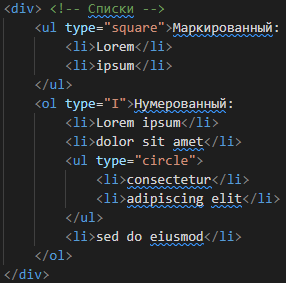
    border: 1px solid #dddddd;

    padding: 5px;

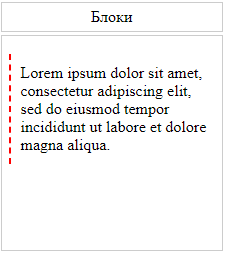
}

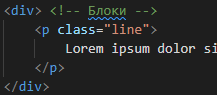
## **Списки**

**Отображение в браузере:**

**Листинг верстки:**

## **Блоки**

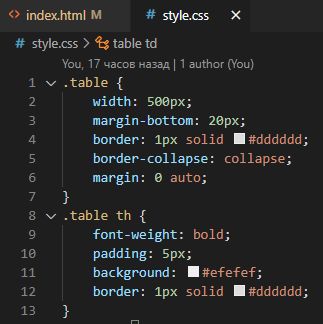
**Отображение в браузере:**

**Листинг верстки:**

# **CSS**

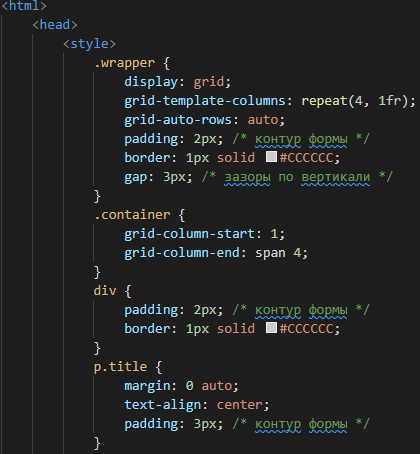
## **Виды адресов**

Внутри тэга:

В виде файла css:

C подключением в <head> страницы

<link rel="stylesheet" href="style.css"/>

Глобальные стили:

## **Блочная модель и grid**

Grid:

.wrapper {

                display: grid;

                grid-template-columns: repeat(4, 1fr);

                grid-auto-rows: auto;

                padding: 2px; /\* контур формы \*/

                border: 1px solid #CCCCCC;

                gap: 3px; /\* зазоры по вертикали \*/

            }

Блоки:

.container {

                grid-column-start: 1;

                grid-column-end: span 4;

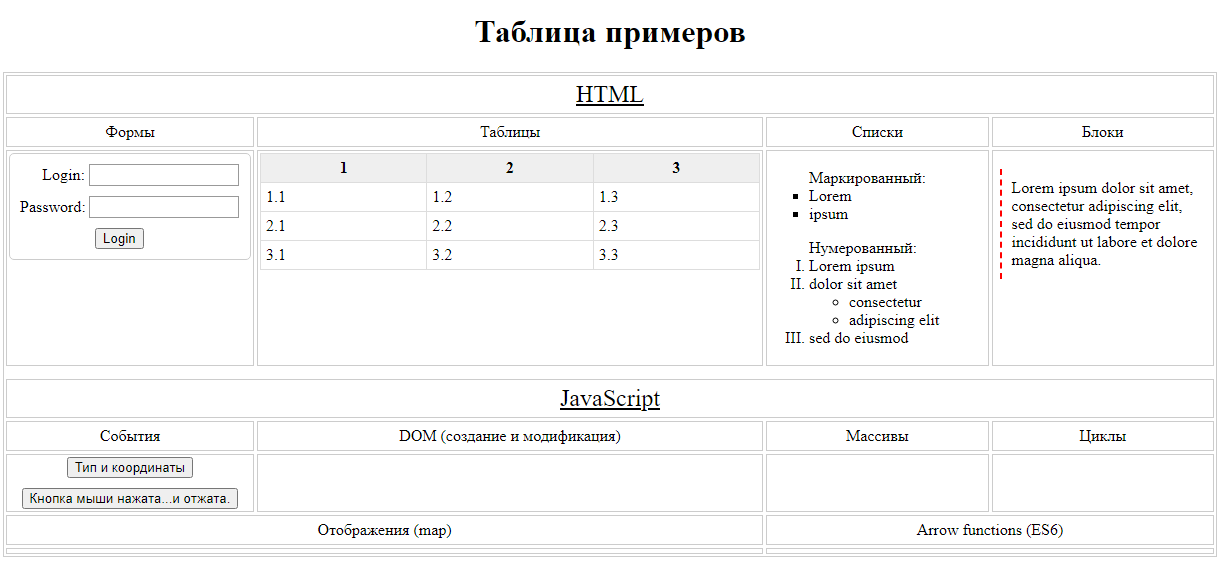
            }

            div {

                padding: 2px; /\* контур формы \*/

                border: 1px solid #CCCCCC;

            }

Отображение в браузере:

# **JavaScript**

## **События**

Виды событий:

* Click
* Contextmenu
* Mouseover/mouseout
* Mousedown/mouseup
* Mousemove
* Submit на элементе button
* Focus на элементах input
* Keydown/keyup
* DOMContentLoaded
* Transitionend

**Вывод сообщения:**

Листинг:

<button id="eventb" style="margin: 0 auto; display: block;">

                    Тип и координаты

</button>

<script>

eventb.onclick = function(event) {

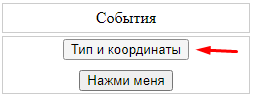
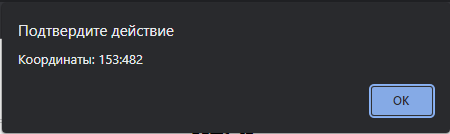
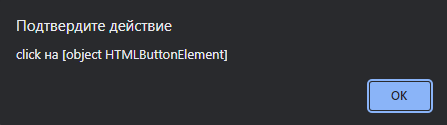
    // вывести тип события, элемент и координаты клика

    alert(event.type + " на " + event.currentTarget);

    alert("Координаты: " + event.clientX + ":" + event.clientY);

};

</script>

Отображение:

**Событие по нажатию:**

Листинг:

<button id="mousedown" style="margin: 0 auto; display: block; margin-top: 10px">

     Нажми меня

</button>

<script>

class Menu {

    handleEvent(event) {

    // mousedown -> onMousedown

    let method = 'on' + event.type[0].toUpperCase() + event.type.slice(1);

        this[method](event);

    }

    onMousedown() {mousedown.innerHTML = "Кнопка мыши нажата";}

    onMouseup() {mousedown.innerHTML += "...и отжата.";}

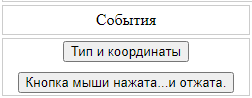
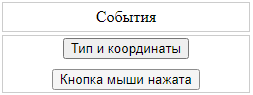
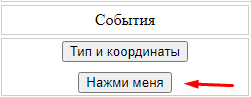
}

let menu = new Menu();

mousedown.addEventListener('mousedown', menu);

mousedown.addEventListener('mouseup', menu);

</script>

Отображение:

## **DOM**

**Вывод обновляющихся часов:**

Листинг:

<script>

    function update() {

        let clock = document.getElementById('clock');

        let date = new Date(); // (\*)

        let hours = date.getHours();

        if (hours < 10) hours = '0' + hours;

        clock.children[0].innerHTML = hours;

        let minutes = date.getMinutes();

        if (minutes < 10) minutes = '0' + minutes;

        clock.children[1].innerHTML = minutes;

        let seconds = date.getSeconds();

        if (seconds < 10) seconds = '0' + seconds;

        clock.children[2].innerHTML = seconds;

    }

    let timerId;

    function clockStart() { // запустить часы

        timerId = setInterval(update, 1000);

        update(); // (\*)

    }

    function clockStop() {

        clearInterval(timerId);

        timerId = null;

    }

</script>

<div id="clock" style="text-align: center;">

<span class="hour">hh</span>:<span class="min">mm</span>:<span

class="sec">ss</span>

</div>

<div style="text-align: center;">

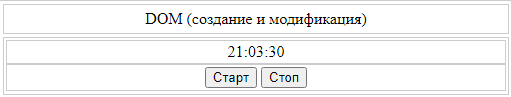
    <!-- клик по этой кнопке вызовет clockStart() -->

    <button onclick="clockStart()">Старт</button>

    <!-- клик по этой кнопке вызовет clockStop() -->

    <button onclick="clockStop()">Стоп</button>

</div>

Отображение:

**Вывод сообщений:**

<script>

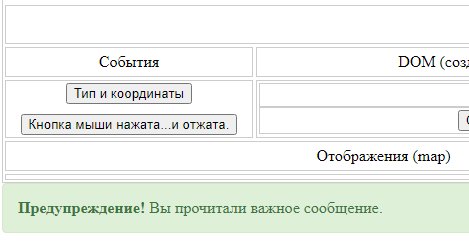
let div = document.createElement('div');

div.className = "alert";

div.innerHTML = "<strong>Предупреждение!</strong> Вы прочитали важное сообщение.";

document.body.append(div);

</script>

Отображение:

## **Массивы**

Запись значений в массив и вывод в виде списка.

Листинг:

<script>

var array = [];

for (let i = 0; i < 5; i++) {

    array.push("Элемент " + i)

}

let out\_arr = document.getElementById('out\_arr');

let str = ' ';

for (let i = 0; i< array.length; i++  ) {

    if (array[i]!==undefined) {

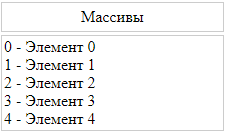
        str +=i+' - '+array[i]+'<br>';

    }

}

out\_arr.innerHTML = str;

</script>

Отображение:

## **Циклы**

for (let i = 0; i < 5; i++) {

    array.push("Элемент " + i)

}

## **Отображения**

myArray.map( (ad) => console.log(ad));

## **Arrow functions**

myArray.map( (ad) => console.log(ad));

# **JQuery**

## **DOM**

function(event){

    $(this).siblings().css("background-color", "yellow");

    $(this).css("background-color", "red");

    $(this).next().css("background-color", "blue");

    $(this).prev().css("background-color", "blue");}

## **Плагины**

Листиг:

<span><b class="list">About</b></span>

<ul>

    <p class="line">

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

      </p>

</ul>

$(".list").click(function () {

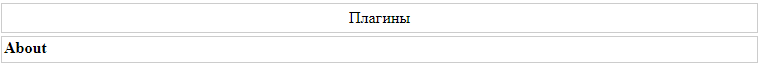
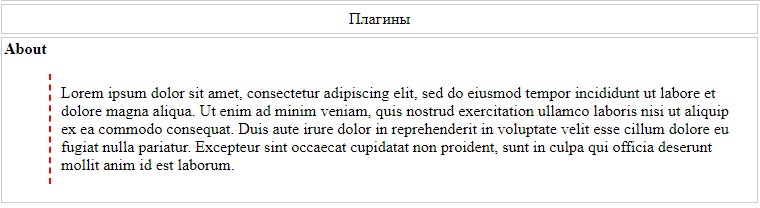
    $("ul").addClass("hide").slideToggle("fast");

});

$("#link").click(function () {

    window.location = $(this).find("a").attr("href");

});

Отображение:

## **Ajax**

Листинг:

<div style="grid-column-end: span 2"> <!-- Ajax -->

    <button id="aj" style="margin: 0 auto; display: block;">

        Узнать погоду

    </button>

</div>

const appendModelsFromApi = () => {

    $.ajax({

        url:'http://api.worldweatheronline.com/premium/v1/weather.ashx?key=&q=&format=json',

        method:'get',

        data: { key: 'd18fb8888484486e859173748220404', q: 'Пермь' },

        success: function (data) {

            alert("Температура в Перми " + data["data"]["current\_condition"][0]["temp\_C"] + "℃");

        },

        error: function (error) {

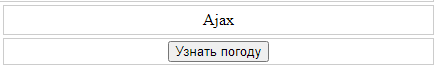
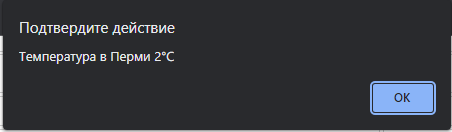
            alert("Не удалось получить температуру");

        }

    });

}

$('#aj').click(appendModelsFromApi)

Отображение:

# **AngularJS**

## **Связывание**

Листинг:

<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.14/angular.min.js"></script>

<div ng-app> <!-- Связывание -->

                <table>

                    <tr>

                        <td>Имя:</td>

                        <td><input ng-model="firstName" type="text"></td>

                    </tr>

                    <tr>

                        <td>Фамилия:</td>

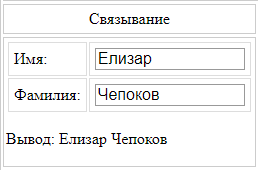
                        <td><input ng-model="lastName" type="text"></td>

                    </tr>

                </table>

                <p>Вывод: {{firstName}} {{lastName}}</p>

            </div>

Отображение:

## **Вывод «Таблицы»**

Листинг:

<div ng-app="App" ng-controller="Controller"> <!-- Вывод "таблицы" -->

                <table>

                    <tr ng-repeat="item in items">

                        <td>{{item.Name}}</td>

                        <td>{{item.Description}}</td>

                    </tr>

                </table>

            </div>

<script type="text/javascript">

            var app = angular.module('App',[]);

            app.controller('Controller',function($scope) {

                $scope.items = [

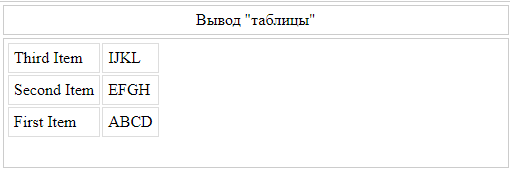
                    {Name:"First Item",Description :"IJKL"},

                    {Name:"Second Item",Description :"EFGH"},

                    {Name:"Third Item",Description :"ABCD"}

                ]});

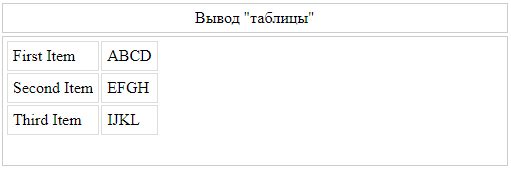
        </script>

Отображение:

## **Фильтры по «Таблице»**

Листинг:

<tr ng-repeat="item in items | orderBy: 'Name'">

Отображение:

# **React**

## **Состояние**

Листинг:

<script>

function tick() {

    const element = (

        <div>

<p>Время в  перми: {new Date().toLocaleTimeString()}.</p>

        </div>

        );

    ReactDOM.render(

        element,

        document.getElementById('root')

    );

}

setInterval(tick, 1000);

</script>

Отображение:

## **Рендер**

Листинг:

<script>

class LoginControl extends React.Component {

  constructor(props) {

    super(props);

    this.handleLoginClick = this.handleLoginClick.bind(this);

    this.handleLogoutClick = this.handleLogoutClick.bind(this);

    this.state = {isLoggedIn: false};

  }

  handleLoginClick() {

    this.setState({isLoggedIn: true});

  }

  handleLogoutClick() {

    this.setState({isLoggedIn: false});

  }

  render() {

    const isLoggedIn = this.state.isLoggedIn;

    let button;

    if (isLoggedIn) {

      button = <LogoutButton onClick={this.handleLogoutClick} />;

    } else {

      button = <LoginButton onClick={this.handleLoginClick} />;

    }

    return (

      <div>

        <Greeting isLoggedIn={isLoggedIn} />

        {button}

      </div>

    );

  }

}

function UserGreeting(props) {

  return <h1>Welcome back!</h1>;

}

function GuestGreeting(props) {

  return <h1>Please sign up.</h1>;

}

function Greeting(props) {

  const isLoggedIn = props.isLoggedIn;

  if (isLoggedIn) {

    return <UserGreeting />;

  }

  return <GuestGreeting />;

}

function LoginButton(props) {

  return (

    <button onClick={props.onClick}>

      Login

    </button>

  );

}

function LogoutButton(props) {

  return (

    <button onClick={props.onClick}>

      Logout

    </button>

  );

}

ReactDOM.render(

  <LoginControl />,

  document.getElementById('root')

);

</script>

Отображение:

# **Bootstrap**

## **Сетка**

Листинг:

        <div class="container">

            <div class="row justify-content-md-center">

                <div class="col col-lg-2">

                    1 of 3

                </div>

                <div class="col-md-auto">

                    Variable width content

                </div>

                <div class="col col-lg-2">

                    3 of 3

                </div>

            </div>

            <div class="row">

                <div class="col">

                    1 of 3

                </div>

                <div class="col-md-auto">

                    Variable width content

                </div>

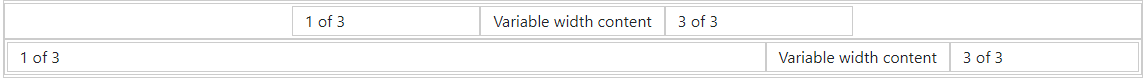
                <div class="col col-lg-2">

                    3 of 3

                </div>

            </div>

        </div>

Отображение: 

## **Адаптивная сетка**

Листинг:

        <div class="row">

            <div class="col-sm-8">col-sm-8</div>

            <div class="col-sm-4">col-sm-4</div>

        </div>

        <div class="row">

            <div class="col-sm">col-sm</div>

            <div class="col-sm">col-sm</div>

            <div class="col-sm">col-sm</div>

        </div>

Отображение:

## **Глифы**

Листинг:

                <p>Search icon: <span class="glyphicon glyphicon-search"></span></p>

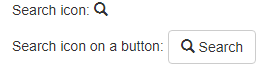
                <p>Search icon on a button:

                  <button type="button" class="btn btn-default">

                    <span class="glyphicon glyphicon-search"></span> Search

                  </button>

                </p>

Отображение:

# **Листинг сайта**

<!DOCTYPE HTML>

<html>

    <head>

        <meta charset="utf-8">

        <title>Веб разработка ЧепоковЕС (ПИ-18-2)</title>

        <!-- #region Styles -->

        <style>

            .wrapper {

                display: grid;

                grid-template-columns: repeat(4, 1fr);

                grid-auto-rows: auto;

                padding: 2px; /\* контур формы \*/

                border: 1px solid #CCCCCC;

                gap: 3px; /\* зазоры по вертикали \*/

            }

            .container1 {

                grid-column-start: 1;

                grid-column-end: span 4;

            }

            div {

                padding: 2px; /\* контур формы \*/

                border: 1px solid #CCCCCC;

            }

            .table {

                width: 500px;

                margin-bottom: 20px;

                border: 1px solid #dddddd;

                border-collapse: collapse;

                margin: 0 auto;

            }

            .table th {

                font-weight: bold;

                padding: 5px;

                background: #efefef;

                border: 1px solid #dddddd;

            }

            table td {

                border: 1px solid #dddddd;

                padding: 5px;

}

            p.title {

                margin: 0 auto;

                text-align: center;

                padding: 3px; /\* контур формы \*/

            }

            P.line {

                margin-left: 5px;

                text-align: left;

                border-left: 2px dashed red;

                padding: 10px;

            }

            form {

                margin: 0 auto; /\* Расположим форму по центру страницы \*/

                width: 220px;

                padding: 10px; /\* контур формы \*/

                border: 1px solid #CCCCCC;

                border-radius: 5px;

            }

            .ul {

                list-style: none;

                padding: 0;

                margin: 0;

            }

            form li + li {

                margin-top: 10px;

            }

            label {

                display: inline-block;

                width: 65px;

                text-align: right;

            }

            input,

            textarea {

                font: 1em sans-serif;

                width: 150px; /\* размер полей \*/

                box-sizing: border-box;

                border: 1px solid #999999; /\* границы полей \*/

            }

            input:focus,

            textarea:focus { /\* подсветка для элементов в фокусе \*/

                border-color: #000000;

            }

            .button {

                padding-left: 65px;

                margin-left: 10px;

            }

            .alert {

                padding: 15px;

                border: 1px solid #d6e9c6;

                border-radius: 4px;

                color: #3c763d;

                background-color: #dff0d8;

            }

        </style>

        <!-- #endregion -->

        <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC" crossorigin="anonymous">

        <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrapicons@1.7.2/font/bootstrap-icons.css">

        <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM" crossorigin="anonymous"></script>

    </head>

    <body>

        <header>

            <h1 style="text-align: center">Таблица примеров</h1>

        </header>

        <div class="wrapper">

            <!-- #region HTML -->

            <div class="container1">

                <p class="title" style="font-size: 150%; text-decoration: underline">HTML</p>

            </div>

            <div>

                <p class="title">Формы</p>

            </div>

            <div>

                <p class="title">Таблицы</p>

            </div>

            <div>

                <p class="title">Списки</p>

            </div>

            <div>

                <p class="title">Блоки</p>

            </div>

            <div> <!-- Формы -->

                <!-- <form action="/my-handling-form-page" method="post"> -->

                <form>

                    <ul class="ul">

                        <li>

                            <label for="login">Login:</label>

                            <input type="login" id="login" name="user\_login">

                        </li>

                        <li>

                            <label for="password">Password:</label>

                            <input type="password" id="password" name="user\_password">

                        </li>

                        <li class="button">

                            <button type="submit">Login</button>

                        </li>

                    </ul>

                </form>

            </div>

            <div> <!-- Таблицы -->

                <table class="table">

                    <tr>

                        <th class="table th">1</th>

                        <th class="table th">2</th>

                        <th class="table th">3</th>

                    </tr>

                    <tr>

                        <td>1.1</td>

                        <td>1.2</td>

                        <td>1.3</td>

                    </tr>

                    <tr>

                        <td>2.1</td>

                        <td>2.2</td>

                        <td>2.3</td>

                    </tr>

                    <tr>

                        <td>3.1</td>

                        <td>3.2</td>

                        <td>3.3</td>

                    </tr>

                </table>

            </div>

            <div> <!-- Списки -->

                <ul type="square">Маркированный:

                    <li>Lorem</li>

                    <li>ipsum</li>

                </ul>

                <ol type="I">Нумерованный:

                    <li>Lorem ipsum</li>

                    <li>dolor sit amet</li>

                    <ul type="circle">

                        <li>consectetur</li>

                        <li>adipiscing elit</li>

                    </ul>

                    <li>sed do eiusmod</li>

                </ol>

            </div>

            <div> <!-- Блоки -->

                <p class="line">

                    Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

                </p>

            </div>

            <!-- #endregion -->

            <!-- #region JavaScript -->

            <div class="container1" style="margin-top: 10px">

                <p class="title" style="font-size: 150%; text-decoration: underline">JavaScript</p>

            </div>

            <div>

                <p class="title">События</p>

            </div>

            <div>

                <p class="title">DOM (создание и модификация)</p>

            </div>

            <div>

                <p class="title">Массивы</p>

            </div>

            <div>

                <p class="title">Циклы</p>

            </div>

            <div> <!-- События -->

                <button id="eventb" style="margin: 0 auto; display: block;">

                    Тип и координаты

                </button>

                <button id="mousedown" style="margin: 0 auto; display: block; margin-top: 10px">

                    Нажми меня

                </button>

            </div>

            <div> <!-- DOM -->

                <div id="clock" style="text-align: center;">

                    <span class="hour">hh</span>:<span class="min">mm</span>:<span class="sec">ss</span>

                </div>

                <div style="text-align: center;">

                    <!-- клик по этой кнопке вызовет clockStart() -->

                    <button onclick="clockStart()">Старт</button>

                    <!-- клик по этой кнопке вызовет clockStop() -->

                    <button onclick="clockStop()">Стоп</button>

                </div>

            </div>

            <div id="out\_arr" style="grid-column-end: span 2"> <!-- Массивы --> </div>

            <!-- #endregion -->

            <!-- #region JQuery -->

            <div class="container1" style="margin-top: 10px">

                <p class="title" style="font-size: 150%; text-decoration: underline">JQuery</p>

            </div>

            <div style="grid-column-end: span 2">

                <p class="title">Плагины</p>

            </div>

            <div style="grid-column-end: span 2">

                <p class="title">Ajax</p>

            </div>

            <div style="grid-column-end: span 2"> <!-- Плагины -->

                <span><b class="list">About</b></span>

                <ul>

                    <p class="line">

                        Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

                    </p>

                </ul>

            </div>

            <div style="grid-column-end: span 2"> <!-- Ajax -->

                <button id="aj" style="margin: 0 auto; display: block;">

                    Узнать погоду

                </button>

            </div>

            <!-- #endregion -->

            <!-- #region AngularJS -->

            <div class="container1" style="margin-top: 10px">

                <p class="title" style="font-size: 150%; text-decoration: underline">AngularJS</p>

            </div>

            <div>

                <p class="title">Связывание</p>

            </div>

            <div>

                <p class="title">Вывод "таблицы"</p>

            </div>

            <div style="grid-column-end: span 2">

                <p class="title">Фильтры по "таблице"</p>

            </div>

            <!-- <div ng-app> -->

            <div> <!-- Связывание -->

                <table>

                    <tr>

                        <td>Имя:</td>

                        <td><input ng-model="firstName" type="text"></td>

                    </tr>

                    <tr>

                        <td>Фамилия:</td>

                        <td><input ng-model="lastName" type="text"></td>

                    </tr>

                </table>

                <p>Вывод: {{firstName}} {{lastName}}</p>

            </div>

            <!-- <div ng-app="App" ng-controller="Controller"> -->

            <div> <!-- Вывод "таблицы" -->

                <table>

                    <tr ng-repeat="item in items">

                        <td>{{item.Name}}</td>

                        <td>{{item.Description}}</td>

                    </tr>

                </table>

            </div>

            <div style="grid-column-end: span 2" ng-app="App" ng-controller="Controller"> <!-- Фильтры по "таблице" -->

                <table>

                    <tr ng-repeat="item in items | orderBy: 'Name'">

                        <td>{{item.Name}}</td>

                        <td>{{item.Description}}</td>

                    </tr>

                </table>

            </div>

            <!-- #endregion -->

            <!-- #region React -->

            <div class="container1" style="margin-top: 10px">

                <p class="title" style="font-size: 150%; text-decoration: underline">React</p>

            </div>

            <div>

                <p class="title">Состояние</p>

            </div>

            <div>

                <p class="title">Изменение состояния</p>

            </div>

            <div style="grid-column-end: span 2">

                <p class="title">Рендер</p>

            </div>

            <div id="root"> <!-- Состояние -->

            </div>

            <div> <!-- Изменение состояния -->

            </div>

            <div  id="login-root" style="grid-column-end: span 2"> <!-- Рендер -->

            </div>

            <!-- #endregion -->

            <!-- #region Bootstrap -->

            <div class="container1" style="margin-top: 10px">

                <p class="title" style="font-size: 150%; text-decoration: underline">Bootstrap</p>

            </div>

            <div>

                <p class="title">Сетка</p>

            </div>

            <div>

                <p class="title">Адаптивная верстка</p>

            </div>

            <div style="grid-column-end: span 2">

                <p class="title">Глифы</p>

            </div>

            <div> <!-- Сетка -->

                <div class="container">

                    <div class="row justify-content-md-center">

                        <div class="col col-lg-2">

                            1 of 3

                          </div>

                          <div class="col-md-auto">

                            Variable width content

                          </div>

                          <div class="col col-lg-2">

                            3 of 3

                        </div>

                    </div>

                    <div class="row">

                        <div class="col">

                            1 of 3

                        </div>

                        <div class="col-md-auto">

                            Variable width content

                        </div>

                        <div class="col col-lg-2">

                            3 of 3

                        </div>

                    </div>

                </div>

            </div>

            <div> <!-- Адаптивная верстка -->

                <div class="row">

                    <div class="col-sm-8">col-sm-8</div>

                    <div class="col-sm-4">col-sm-4</div>

                </div>

                <div class="row">

                    <div class="col-sm">col-sm</div>

                    <div class="col-sm">col-sm</div>

                    <div class="col-sm">col-sm</div>

                </div>

            </div>

            <div style="grid-column-end: span 2"> <!-- Глифы -->

                <p>Search icon: <span class="glyphicon glyphicon-search"></span></p>

                <p>Search icon on a button:

                  <button type="button" class="btn btn-default">

                    <span class="glyphicon glyphicon-search"></span> Search

                  </button>

                </p>

            </div>

            <!-- #endregion -->

        </div>

        <span class="glyphicon glyphicon-search"></span>

        <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>

        <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.14/angular.min.js"></script>

        <script src="https://cdnjs.cloudflare.com/ajax/libs/angular.js/1.3.15/angular.min.js"></script>

        <script src="https://code.angularjs.org/1.6.9/angular-route.js"></script>

        <script src="https://code.angularjs.org/1.6.9/angular.min.js"></script>

        <script src="https://code.angularjs.org/1.6.9/angular.js"></script>

        <script src="/lib/bootstrap.js"></script>

        <script src="/lib/bootstrap.css"></script>

        <script src="https://unpkg.com/react@17/umd/react.production.min.js" crossorigin></script>

        <script src="https://unpkg.com/react-dom@17/umd/react-dom.production.min.js" crossorigin></script>

        <!-- #region Оповещение тип и координаты -->

        <script>

        eventb.onclick = function(event) {

            // вывести тип события, элемент и координаты клика

            alert(event.type + " на " + event.currentTarget);

            alert("Координаты: " + event.clientX + ":" + event.clientY);

        };

        </script>

        <!-- #endregion -->

        <!-- #region Событие нажатия на кнопку -->

        <script>

        class Menu {

            handleEvent(event) {

                // mousedown -> onMousedown

                let method = 'on' + event.type[0].toUpperCase() + event.type.slice(1);

                this[method](event);

            }

            onMousedown() {mousedown.innerHTML = "Кнопка мыши нажата";}

            onMouseup() {mousedown.innerHTML += "...и отжата.";}

        }

        let menu = new Menu();

        mousedown.addEventListener('mousedown', menu);

        mousedown.addEventListener('mouseup', menu);

        </script>

        <!-- #endregion -->

        <!-- #region Часы -->

        <script>

            function update() {

                let clock = document.getElementById('clock');

                let date = new Date(); // (\*)

                let hours = date.getHours();

                if (hours < 10) hours = '0' + hours;

                clock.children[0].innerHTML = hours;

                let minutes = date.getMinutes();

                if (minutes < 10) minutes = '0' + minutes;

                clock.children[1].innerHTML = minutes;

                let seconds = date.getSeconds();

                if (seconds < 10) seconds = '0' + seconds;

                clock.children[2].innerHTML = seconds;

            }

            let timerId;

            function clockStart() { // запустить часы

                timerId = setInterval(update, 1000);

                update(); // (\*)

            }

            function clockStop() {

                clearInterval(timerId);

                timerId = null;

            }

        </script>

        <!-- #endregion -->

        <!-- #region Вывод сообщения -->

        <script>

        let div = document.createElement('div');

        div.className = "alert";

        div.innerHTML = "<strong>Предупреждение!</strong> Вы прочитали важное сообщение.";

        document.body.append(div);

        </script>

        <!-- #endregion -->

        <!-- #region Заполнение массива -->

        <script>

        var array = [];

        for (let i = 0; i < 5; i++) {

            array.push("Элемент " + i)

        }

        let out\_arr = document.getElementById('out\_arr');

        let str = ' ';

        for (let i = 0; i< array.length; i++  ) {

            if (array[i]!==undefined) {

                str +=i+' - '+array[i]+'<br>';

            }

        }

        out\_arr.innerHTML = str;

        </script>

        <!-- #endregion -->

        <!-- #region Ajax -->

        <script>

        //use plugin

        //animation for list

        $(".list").click(function () {

            $("ul").addClass("hide").slideToggle("fast");

        });

        $("#link").click(function () {

            window.location = $(this).find("a").attr("href");

        });

        //get temperature by ajax

        const appendModelsFromApi = () => {

            $.ajax({

                url:'http://api.worldweatheronline.com/premium/v1/weather.ashx?key=&q=&format=json',

                method:'get',

                data: { key: 'd18fb8888484486e859173748220404', q: 'Пермь' },

                success: function (data) {

                    alert("Температура в Перми " + data["data"]["current\_condition"][0]["temp\_C"] + "℃");

                },

                error: function (error) {

                    alert("Не удалось получить температуру");

                }

            });

        }

        $('#aj').click(appendModelsFromApi)

        </script>

        <!-- #endregion -->

        <!-- #region Вывод таблицы React -->

        <script type="text/javascript">

            var app = angular.module('App',[]);

            app.controller('Controller',function($scope) {

                $scope.items = [

                    {Name:"Third Item",Description :"IJKL"},

                    {Name:"Second Item",Description :"EFGH"},

                    {Name:"First Item",Description :"ABCD"}

                ]});

        </script>

            <!-- #endregion -->

        <!-- #region Clocks React-->

        <script>

            function tick() {

                const element = (

                    <div>

                        <h1>Hello, world!</h1>

                        <h2>It is {new Date().toLocaleTimeString()}.</h2>

                    </div>

                    );

                ReactDOM.render(

                    element,

                    document.getElementById('root')

                );

            }

            setInterval(tick, 1000);

        </script>

        <!-- #endregion -->

        <!-- #region Login update-->

        <script>

        class LoginControl extends React.Component {

        constructor(props) {

            super(props);

            this.handleLoginClick = this.handleLoginClick.bind(this);

            this.handleLogoutClick = this.handleLogoutClick.bind(this);

            this.state = {isLoggedIn: false};

        }

        handleLoginClick() {

            this.setState({isLoggedIn: true});

        }

        handleLogoutClick() {

            this.setState({isLoggedIn: false});

        }

        render() {

            const isLoggedIn = this.state.isLoggedIn;

            let button;

            if (isLoggedIn) {

            button = <LogoutButton onClick={this.handleLogoutClick} />;

            } else {

            button = <LoginButton onClick={this.handleLoginClick} />;

            }

            return (

            <div>

                <Greeting isLoggedIn={isLoggedIn} />

                {button}

            </div>

            );

        }

        }

        function UserGreeting(props) {

        return <h1>Welcome back!</h1>;

        }

        function GuestGreeting(props) {

        return <h1>Please sign up.</h1>;

        }

        function Greeting(props) {

        const isLoggedIn = props.isLoggedIn;

        if (isLoggedIn) {

            return <UserGreeting />;

        }

        return <GuestGreeting />;

        }

        function LoginButton(props) {

        return (

            <button onClick={props.onClick}>

            Login

            </button>

        );

        }

        function LogoutButton(props) {

        return (

            <button onClick={props.onClick}>

            Logout

            </button>

        );

        }

        ReactDOM.render(

        <LoginControl />,

        document.getElementById('login-root')

        );

        </script>

        <!-- #endregion -->

    </body>

</html>