**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ**

**ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ П. О. СУХОГО**

Факультет автоматизированных и информационных систем

Кафедра «Информатика»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 10

по дисциплине «Разработка приложений для Интернет»

на тему: «Сериализация объектов. Библиотека jQuery»

Выполнил: студент гр. ИП-32

Коледа А.С.

Принял: преподаватель

Процкая М.А.

Гомель 2021

**Цель работы**: ознакомиться с нотацией JSON, научиться использовать библиотеку jQuery.

**Задание 1:** В соответствии с вариантом задания 1 лабораторной работы №6 сформировать массив объектов. Сериализовать массив в формат JSON с последующим сохранением его в объекте localStorage. Используя библиотеку jQuery, написать сценарий, считывающий информацию из локального хранилища и формирующий таблицу из полученных данных. Предусмотреть сортировку отображаемых данных при щелчке по заголовкам таблицы.

Код файла lab\_10\_1.html:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link rel="shortcut icon" href="favicon.png" type="image/png">

<title>Задание 10.1 Коледа ИП-32</title>

<style>

table {

border: 1px solid gray;

background-color: gray;

}

th {

height: 20px;

border: 1px solid gray;

background-color: rgb(223, 223, 223);

padding: 5px;

font-family: 'Courier New', Courier, monospace;

font-weight: bold;

}

td {

height: 20px;

border: 1px solid gray;

background-color: rgb(255, 255, 255);

padding: 5px;

font-family: 'Courier New', Courier, monospace;

font-weight: bold;

}

</style>

<script src="https://code.jquery.com/jquery-3.6.0.js"

integrity="sha256-H+K7U5CnXl1h5ywQfKtSj8PCmoN9aaq30gDh27Xc0jk=" crossorigin="anonymous"></script>

</head>

<body>

<script src='class.js'></script>

<script>

"use strict"

let arr = [];

arr.push(new Kingston('SSD', 512, 'PCI', 0, 2007, 145));

arr.push(new Kingston('Flesh', 1000, 'PCI', 0, 2018, 998));

arr.push(new Kingston('SSD', 512, 'PCI', 0, 2010, 723));

arr.push(new Kingston('HDD', 256, 'M.2', 10000, 2008, 234));

let str = JSON.stringify(arr);

localStorage.setItem('array', str);

let prop\_name\_array = ['тип', 'ёмкость', 'интерфейс', 'скорость', 'год релиза', 'цена'];

let property\_str = JSON.stringify(prop\_name\_array);

localStorage.setItem('property\_array', property\_str);

createTableFromLocalStorage('array', 'property\_array');

function createTableFromLocalStorage(item\_name, property\_item\_name) {

let array = JSON.parse(localStorage.getItem(item\_name));

let prop\_name = JSON.parse(localStorage.getItem(property\_item\_name));

$('body').append('<table></table>');

$('table').append(`<tr></tr>`);

for (const name of prop\_name) {

$('tr:last').append(`<th>${name}</th>`);

}

for (const obj of array) {

$('table').append(`<tr></tr>`);

for (const prop in obj) {

if (Object.hasOwnProperty.call(obj, prop)) {

const element = obj[prop];

$('tr:last').append(`<td>${element}</td>`);

}

}

}

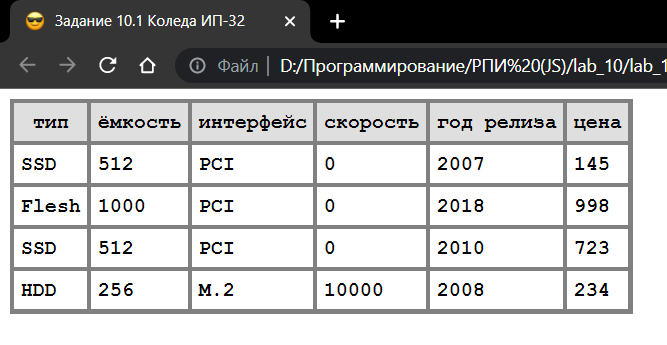
}

</script>

</body>

</html>

Результат выполнения 1-го задания:



**Задание 2:** Создать HTML-документ с реализацией стандартного калькулятора, выполняющего арифметические операции над введенными данными и имеющего возможность сохранения результатов в памяти. При написании сценария использовать. Предусмотреть ввод данных и выбор операции как с помощью мыши, так и с помощью клавиатуры.

Код файла lab\_10\_2.html:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link rel="shortcut icon" href="favicon.png" type="image/png">

<title>Задание 10.2 Коледа ИП-32</title>

<script src="jQuery.js"></script>

<!--<script src="https://code.jquery.com/jquery-3.6.0.js"

integrity="sha256-H+K7U5CnXl1h5ywQfKtSj8PCmoN9aaq30gDh27Xc0jk=" crossorigin="anonymous"></script>-->

<style>

body {

background-color: lightgray;

}

main {

background-color: lightskyblue;

display: flex;

flex-direction: column;

width: 500px;

border: 2px solid gray;

padding: 10px;

margin: 10px;

}

#button\_box {

display: flex;

flex-direction: row;

padding: 0px;

margin: 10px;

}

div {

/\*border: 2px dashed red;\*/

padding: 0px;

margin: 0px;

}

#input\_field\_box {

padding: 0px;

margin: 10px;

height: 100px;

border: 2px solid gray;

}

.in\_out\_field {

background-color: lightgreen;

border: none;

width: 100%;

height: 50%;

margin: 0px;

padding: 0px 0px;

font-size: 20pt;

font-family: 'Courier New', Courier, monospace;

text-align: right;

}

#actions\_box {

width: 60%;

padding: 0px;

margin: 0px;

margin-right: 10px;

}

#number\_box {

width: 40%;

padding: 0px;

margin: 0px;

margin-left: 10px;

}

table {

background-color: rgb(255, 0, 0);

font-size: 25pt;

font-family: 'Courier New', Courier, monospace;

}

td {

border: 1px solid gray;

background-color: gray;

width: 70px;

height: 50px;

text-align: center;

}

td:hover {

border: 1px solid gray;

background-color: lightgray;

font-weight: bold;

cursor: pointer;

}

</style>

</head>

<body>

<main>

<div id='input\_field\_box'>

<p class='in\_out\_field' id='input\_field'></p>

<p class='in\_out\_field' id='output\_field'></p>

</div>

<div id='button\_box'>

<div id='actions\_box'>

<table>

<tr>

<td>\*</td>

<td>/</td>

<td>+</td>

<td>-</td>

</tr>

<tr>

<td>(</td>

<td>)</td>

<td>^</td>

<td></td>

</tr>

<tr>

<td>clr</td>

<td>del</td>

<td colspan="2">Enter</td>

</tr>

<tr>

<td colspan="2">save</td>

<td colspan="2">load</td>

</tr>

</table>

</div>

<div id='number\_box'>

<table>

<tr>

<td>7</td>

<td>8</td>

<td>9</td>

</tr>

<tr>

<td>4</td>

<td>5</td>

<td>6</td>

</tr>

<tr>

<td>1</td>

<td>2</td>

<td>3</td>

</tr>

<tr>

<td>0</td>

<td>.</td>

<td></td>

</tr>

</table>

</div>

</div>

</main>

<script>

"use strict"

let input\_string = '';

$('td').click(add\_to\_input);

//$('body').keydown((event) => { console.log(event.which, event.key) });

$('body').keydown((event) => { add\_to\_input(event) });

function add\_to\_input() {

console.log(event.type);

let val = '';

let key = '';

if (event.type == 'keydown') {

// Если нажатие клавиши на клавиатуре.

console.log(event.key);

val = event.key;

key = val;

} else if (event.type == 'click') {

// Если тык в кнопку на экране.

console.log(event.target.innerText);

val = event.target.innerText;

}

if (val == 'clr' || key == 'c') {

input\_string = '';

error\_off();

} else if (val == 'del' || key == 'd') {

input\_string = input\_string.slice(0, -1);

error\_off();

} else if (val == 'save' || key == 's') {

localStorage.setItem('result', output\_field.innerText);

} else if (val == 'load' || key == 'l') {

let res = localStorage.getItem('result');

if (res) {

output\_field.innerText = res;

}

} else if (val == 'Enter' || key == 'Enter') {

// Защита от ввода пустых строк.

if (input\_string.length) {

let result = solve(input\_string);

if (result == 'error') {

// Если ошибка вычисления.

error\_on();

} else {

// Если всё чётко.

output\_field.innerText = result;

}

}

} else {

if (val == '0' || val == '1' || val == '2' || val == '3' || val == '4' || val == '5' || val == '6' || val == '7' || val == '8' || val == '9' || val == '.' || val == '\*' || val == '/' || val == '+' || val == '-' || val == '(' || val == ')' || val == '^') {

input\_string += val;

} else if (key == '0' || key == '1' || key == '2' || key == '3' || key == '4' || key == '5' || key == '6' || key == '7' || key == '8' || key == '9' || key == '.' || key == '\*' || key == '/' || key == '+' || key == '-' || key == '(' || key == ')' || key == '^') {

input\_string += key;

}

}

// Update input string.

input\_field.innerText = input\_string;

}

function solve(str) {

str = parse\_input\_string(str);

let res = 0;

try {

res = eval(str);

} catch (error) {

return 'error';

}

return res;

}

function parse\_input\_string(str) {

let arr = str.split('');

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

if (arr[i] == '^') {

// Выделяем число или скобки.

let temp\_arr = ['\*\*'];

let left = arr.slice(0, i);

let right = arr.slice(i + 1, arr.length);

arr = left.concat(temp\_arr);

arr = arr.concat(right);

}

}

return arr.join('');

}

function error\_on() {

$('#output\_field').css('background-color', 'lightcoral');

output\_field.innerText = 'error';

}

function error\_off() {

$('#output\_field').css('background-color', 'lightgreen');

output\_field.innerText = '';

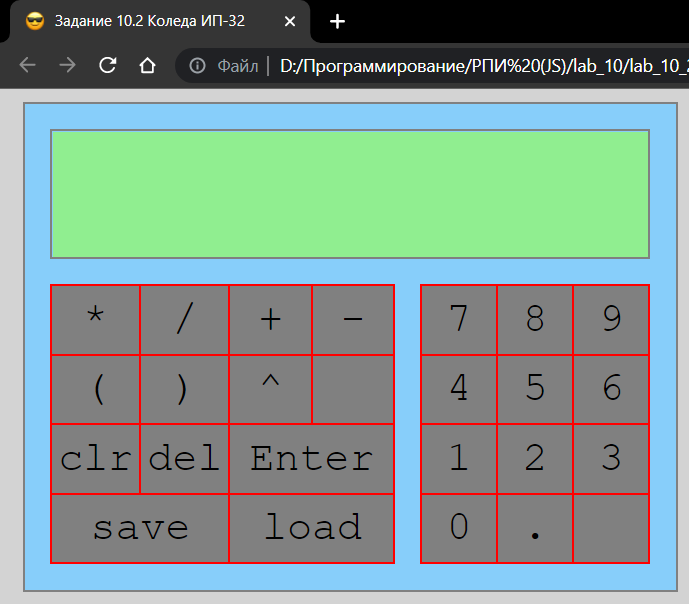
}

</script>

</body>

</html>

Результат выполнения 2-го задания:



**Задание 3:** Создать HTML-документ с полем ввода даты. Используя библиотеку jQuery, сформировать календарь для текущего месяца введенной даты. Календарь оформить в виде таблицы, неделя начинается с понедельника, в ячейки таблицы выводится день месяца.

Код файла lab\_10\_3.html:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link rel="shortcut icon" href="favicon.png" type="image/png">

<title>Задание 10.3 Коледа ИП-32</title>

<script src="https://code.jquery.com/jquery-3.6.0.js"

integrity="sha256-H+K7U5CnXl1h5ywQfKtSj8PCmoN9aaq30gDh27Xc0jk=" crossorigin="anonymous"></script>

<style>

body {

background-color: lightgray;

font-size: 12pt;

padding: 10px;

}

div {

border: 2px dashed red;

padding: 0px;

margin: 0px;

}

table {

border: 1px solid gray;

background-color: gray;

}

th {

height: 20px;

width: 20px;

border: 3px solid gray;

background-color: rgb(223, 223, 223);

padding: 5px;

font-family: 'Courier New', Courier, monospace;

font-weight: bold;

}

td {

height: 20px;

border: 3px solid gray;

background-color: rgb(184, 184, 184);

color: rgba(0, 0, 0, 0.5);

padding: 5px;

font-family: 'Courier New', Courier, monospace;

font-weight: bold;

}

.active {

background-color: rgb(255, 255, 255);

color: rgba(0, 0, 0, 1);

}

.current {

color: rgb(255, 0, 0);

border: 3px solid red;

}

input {

margin-right: 35px;

}

</style>

</head>

<body>

<form onsubmit="start(event)">

<input id='input\_date' type="date" value="2021-10-26" required>

<input id='click' type="submit" value="НЕ ЖМИ!!!">

</form>

<br>

<script>

"use strict"

// ДЛЯ ОТЛАДКИ.

//$("#click").click();

function start(event) {

event.preventDefault();

Date.prototype.daysInMonth = function () {

return 32 - new Date(this.getFullYear(), this.getMonth(), 32).getDate();

};

Date.prototype.getWeekDay = function () {

let days = ['ВС', 'ПН', 'ВТ', 'СР', 'ЧТ', 'ПТ', 'СБ'];

return days[this.getDay()];

};

let date = new Date(input\_date.value);

let start\_date = new Date(date.getFullYear(), date.getMonth(), 1);

let prev\_date = new Date(date.getFullYear(), date.getMonth() - 1, 1);

let diff = prev\_date.daysInMonth() - start\_date.getDay() + 1;

let array = [];

let date\_start = diff;

let type = false;

let isCurrent = false;

for (let i = 0; i < 42; i++) {

if (date\_start >= prev\_date.daysInMonth() + 1 && type == false) {

date\_start = 1;

type = true;

} else if (date\_start >= date.daysInMonth() + 1 && type == true) {

date\_start = 1;

type = false;

}

if (date\_start == date.getDate() && type == true) {

isCurrent = true;

}

array.push({ number: date\_start, active: type, current: isCurrent });

isCurrent = false;

date\_start += 1;

}

//console.log(array);

createTableFromArray(array);

}

function createTableFromArray(array) {

let days = ['ВС', 'ПН', 'ВТ', 'СР', 'ЧТ', 'ПТ', 'СБ'];

if ($('table').length > 0) {

$('table').remove();

}

$('body').append('<table></table>');

$('table').append(`<tr></tr>`);

for (const name of days) {

$('tr:last').append(`<th>${name}</th>`);

}

let count = 0;

for (let i = 0; i < 6; i++) {

$('table').append(`<tr></tr>`);

for (let j = 0; j < 7; j++) {

$('tr:last').append(`<td>${array[count].number}</td>`);

if (array[count].active) {

$('td:last').addClass('active');

}

if (array[count].current) {

$('td:last').addClass('current');

}

count++;

}

}

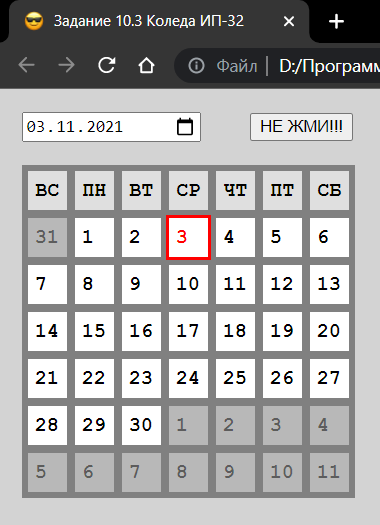
}

</script>

</body>

</html>

Результат выполнения 3-го задания:



**Задание 4:** Воспользоваться созданным в предыдущем задании календарем. Сформировать строку JSON с информацией о мероприятиях на текущий месяц. Строка хранится в Local Storage. Предусмотреть выделения цветом дней месяца, для которых запланировано мероприятие. При щелчке на такой день, должна выдаваться полная информация о мероприятии. В скрипте предусмотреть возможность добавления мероприятия на выбранный день с сохранением информации в Local Storage. Для добавления мероприятия и просмотра информации использовать модальное окно.

Код файла lab\_10\_4.html:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link rel="shortcut icon" href="favicon.png" type="image/png">

<title>Задание 10.4 Коледа ИП-32</title>

<script src="jQuery.js"></script>

<!--<script src="https://code.jquery.com/jquery-3.6.0.js"

integrity="sha256-H+K7U5CnXl1h5ywQfKtSj8PCmoN9aaq30gDh27Xc0jk=" crossorigin="anonymous"></script>-->

<style>

body {

background-color: lightgray;

font-size: 12pt;

padding: 10px;

}

div {

padding: 0px;

margin: 0px;

}

table {

border: 1px solid gray;

background-color: gray;

}

th {

height: 20px;

width: 20px;

border: 3px solid gray;

background-color: rgb(223, 223, 223);

padding: 5px;

font-family: 'Courier New', Courier, monospace;

font-weight: bold;

}

td {

height: 20px;

border: 3px solid gray;

background-color: rgb(184, 184, 184);

color: rgba(0, 0, 0, 0.5);

padding: 5px;

font-family: 'Courier New', Courier, monospace;

font-weight: bold;

}

.active {

background-color: rgb(255, 255, 255);

color: rgba(0, 0, 0, 1);

}

.current {

color: rgb(255, 0, 0);

background-color: #ffd3d3;

border: 3px solid red;

}

input {

margin-right: 35px;

}

#modal\_window {

position: fixed;

top: 0;

bottom: 0;

right: 0;

left: 0;

background-color: rgba(128, 128, 128, 0.7);

}

#modal\_window\_entry {

font-family: 'Courier New', Courier, monospace;

font-weight: 300;

font-size: 13pt;

color: black;

border: 2px solid gray;

background-color: white;

margin: 100px 100px 100px 100px;

padding: 10px 10px 10px 10px;

display: flex;

flex-flow: column nowrap;

justify-content: center;

align-items: center;

}

#events\_description\_box {

padding: 10px;

}

#input\_form {

display: flex;

flex-direction: column;

}

.form\_part {

margin: 10px;

}

#form\_bottom {

width: 40%;

margin-top: 10px;

}

#submit\_button\_box input {

float: none;

}

#submit\_button\_box {

text-align: center;

}

.button\_box {

display: inline-block;

margin: 0px 0px;

}

#first\_button {

float: left;

}

#second\_button {

float: right;

}

</style>

</head>

<body>

<div id='modal\_window' hidden>

<div id='modal\_window\_entry'>

<div id='events\_description\_box'>

<h7>Запланированные события на этот день:</h7>

</div>

<form id='input\_form' onsubmit='add\_new\_event(event)'>

<div class='form\_part' hidden>

<label for='input\_date\_hidden'>Число мероприятия:</label>

<input type='number' id='input\_date\_hidden' value="0">

</div>

<div class='form\_part'>

<label for='input\_event\_name'>Название мероприятия:</label>

<input type='text' id='input\_event\_name' required value="днюха у собаки">

</div>

<div class='form\_part'>

<label for='input\_event\_color'>цвет маркера:</label>

<input type='color' id='input\_event\_color' required value="#59c062">

</div>

<div class='form\_part' id='submit\_button\_box'>

<input type="submit" value="Добавить">

</div>

</form>

<div id='form\_bottom'>

<div class="button\_box" id='first\_button'>

<button onclick=''>Ищо</button>

</div>

<div class="button\_box" id='second\_button'>

<button onclick='close\_form()'>Харош</button>

</div>

</div>

</div>

</div>

<form onsubmit="start(event)">

<input id='input\_date' type="date" value="2021-10-26" required>

<input id='click' type="submit" value="НЕ ЖМИ!!!">

</form>

<br>

<script>

"use strict"

add\_info\_to\_localStorage();

// ДЛЯ ОТЛАДКИ.

//$("#click").click();

//$("#20").click();

//$('#submit\_button\_box input').click();

function add\_info\_to\_localStorage() {

if (localStorage.getItem('events') == null) {

let event\_array = [

{ date: '2021-10-7', name: 'день защиты кошек', color: 'black' },

{ date: '2021-10-10', name: 'вернуть сотку Сане', color: 'coral' },

{ date: '2021-10-30', name: 'годовщина изобретения стола', color: 'blue' },

{ date: '2021-11-05', name: 'Днюха Императора Александра', color: 'pink' },

{ date: '2021-10-21', name: 'годовщина похищения инопланетянами', color: 'yellow' },]

localStorage.setItem('events', JSON.stringify(event\_array));

}

}

function start(event) {

console.log('start');

Date.prototype.daysInMonth = function () {

return 32 - new Date(this.getFullYear(), this.getMonth(), 32).getDate();

};

Date.prototype.getWeekDay = function () {

let days = ['ВС', 'ПН', 'ВТ', 'СР', 'ЧТ', 'ПТ', 'СБ'];

return days[this.getDay()];

};

Date.prototype.getMonthName = function () {

let months = ['Январь', 'Февраль', 'Март', 'Апрель', 'Май', 'Июнь', 'Июль', 'Август', 'Сентябрь', 'Октябрь', 'Ноябрь', 'Декабрь'];

return months[this.getMonth()];

};

event.preventDefault();

let date = new Date(input\_date.value);

let start\_date = new Date(date.getFullYear(), date.getMonth(), 1);

let prev\_date = new Date(date.getFullYear(), date.getMonth() - 1, 1);

let diff = prev\_date.daysInMonth() - start\_date.getDay() + 1;

let event\_array = localStorage.getItem('events');

let array = [];

let date\_start = diff;

let type = false;

let isCurrent = false;

for (let i = 0; i < 42; i++) {

if (date\_start >= prev\_date.daysInMonth() + 1 && type == false) {

date\_start = 1;

type = true;

} else if (date\_start >= date.daysInMonth() + 1 && type == true) {

date\_start = 1;

type = false;

}

if (date\_start == date.getDate() && type == true) {

isCurrent = true;

}

array.push({ number: date\_start, active: type, current: isCurrent });

isCurrent = false;

date\_start += 1;

}

if (createTableFromArray(array))

modify\_table();

}

function createTableFromArray(array) {

console.log('create table');

let days = ['ВС', 'ПН', 'ВТ', 'СР', 'ЧТ', 'ПТ', 'СБ'];

// Удаляем старую таблицу если она есть.

if ($('table').length > 0) {

$('table').remove();

}

$('body').append('<table></table>');

$('table').append(`<tr></tr>`);

for (const name of days) {

$('tr:last').append(`<th>${name}</th>`);

}

let count = 0;

for (let i = 0; i < 6; i++) {

$('table').append(`<tr></tr>`);

for (let j = 0; j < 7; j++) {

$('tr:last').append(`<td>${array[count].number}</td>`);

// День их текущего месяца.

if (array[count].active) {

$('td:last').addClass('active');

$('td:last').attr('id', array[count].number);

$('td:last').click(show\_form);

}

if (array[count].current) {

$('td:last').addClass('current');

}

count++;

}

}

return true;

}

function modify\_table() {

console.log('modify table');

let current\_month = new Date(input\_date.value).getMonth();

let event\_array = JSON.parse(localStorage.getItem('events'));

for (let i = 0; i < event\_array.length; i++) {

let event\_date = new Date(event\_array[i].date);

// Выбираем все события текущего месяца.

if (event\_date.getMonth() == current\_month) {

$(`#${event\_date.getDate()}`).css('border-color', event\_array[i].color);

}

}

}

function show\_form(event) {

modal\_window.removeAttribute('hidden');

let current\_date = new Date(input\_date.value);

let event\_array = JSON.parse(localStorage.getItem('events'));

for (let i = 0; i < event\_array.length; i++) {

let event\_date = new Date(event\_array[i].date);

// Выбираем все события текущего месяца.

if (event\_date.getMonth() == current\_date.getMonth() && event\_date.getDate() == event.target.id) {

$('#events\_description\_box').append(`<p>${event\_array[i].name}</p>`);

// Добавляем новому элементу класс, чтобы можно было по классу все эти элеенты удалять.

$('p:last').addClass('event\_description\_paragraph');

}

}

// Устанавливает значение числа для того чтобы можно было это число потом считать из формы.

$('#input\_date\_hidden').val(event.target.id);

}

function close\_form() {

modal\_window.setAttribute('hidden', 'true');

$('.event\_description\_paragraph').remove();

$('#input\_event\_name').val('');

$('#input\_event\_color').val('');

}

function add\_new\_event(event) {

event.preventDefault();

let current\_date = new Date(input\_date.value);

let event\_array = JSON.parse(localStorage.getItem('events'));

event\_array.push({ date: current\_date.getFullYear() + '-' + (current\_date.getMonth() + 1) + '-' + $('#input\_date\_hidden').val(), name: $('#input\_event\_name').val(), color: $('#input\_event\_color').val() })

console.log(event\_array);

localStorage.setItem('events', JSON.stringify(event\_array));

modify\_table();

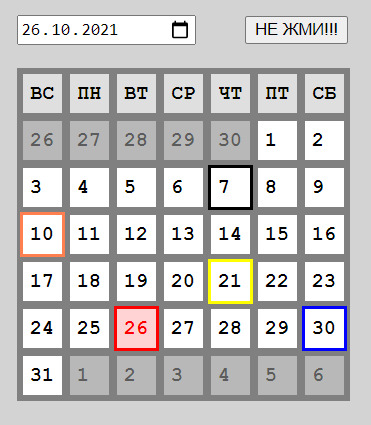
}

</script>

</body>

</html>

Результат выполнения 4-го задания:



**Выводы:** В процессе выполнения лабораторной работы ознакомился с нотацией JSON. Получил практику в использовании библиотеки jQuery.