Министерство науки и высшего образования РФ

Севастопольский государственный университет

Кафедра информационных систем

ОТЧЕТ

по лабораторной работе №3

Исследование объектной модели документа (DOM) и системы событий JavaScript

по дисциплине «Веб-технологии»

Выполнил:

Студент группы ИС/б 17-2-о

Черняев Н.Г.

Проверил:

Овчинников А.Л.

г. Севастополь 2019

1.Цель работы

Изучить динамическую объектную модель документа, предоставляемую стандартом DOM и систему событий языка JavaScript, возможность хранения данных на стороне клиента. Приобрести практические навыки работы с событиями JavaScript, деревом документа, Local Storage и Cookies.

2.Постановка задачи

Вариант №18

* Реализовать отображение в области меню сайта текущих даты и времени (обновление времени 1 раз в секунду).
* сделать пункт меню «Мои интересы» в виде выпадающего меню.
* На странице «Контакт» добавить поле «Дата рождения», для которого реализовать всплывающий снизу элемент «календарь».
* Реализовать динамическую проверку корректности заполнения пользователем формы на странице «Контакт» таким образом, чтобы при потере фокуса заполняемого поля осуществлялась проверка корректности его заполнения. В случае если поле заполнено некорректно, должна появиться надпись, поясняющая характер ошибки. После исправления пользователем ошибки, надпись должна исчезнуть. Если все поля формы заполнены корректно должна стать активной кнопка «Отправить».
* Реализовать открытие в динамически формируемом новом окне (блоке DIV) соответствующих больших фото при щелчке мыши по маленьким фото на странице «Фотоальбом».
* Добавить страницу «История просмотра». На данной странице реализовать отображение двух таблиц:
  + «История текущего сеанса» – в данной таблице отображается количество посещений каждой страницы за время текущего сеанса. Реализовать хранение этих данных в Local Storage.
  + «История за все время» – в данной таблице отображается количество посещений каждой страницы за все время. Реализовать хранение этих данных в хранилище Cookies. Для выполнения этого задания необходимо создать два JavaScript метода: getCookie (name) и setCookie(name, value, exlpiration\_days).

3.Ход работы

Файл history.js

var localCont;

var sessionCont;

function getLocalHistory(){

localCont = document.getElementById("history-local");

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

switch (i) {

case 0:

localCont.innerHTML += "<tr><td>Интересы</td><td>"+localStorage.getItem("Интересы")+"</td></tr>";

break;

case 1:

localCont.innerHTML += "<tr><td>Учеба</td><td>"+localStorage.getItem("Учеба")+"</td></tr>";

break;

case 2:

localCont.innerHTML += "<tr><td>Контакт</td><td>"+localStorage.getItem("Контакт")+"</td></tr>";

break;

case 3:

localCont.innerHTML += "<tr><td>Фотоальбом</td><td>"+localStorage.getItem("Фотоальбом")+"</td></tr>";

break;

case 4:

localCont.innerHTML += "<tr><td>Тест</td><td>"+localStorage.getItem("Тест")+"</td></tr>";

break;

case 5:

localCont.innerHTML += "<tr><td>История</td><td>"+localStorage.getItem("История")+"</td></tr>";

break;

}

}

}

function getSessionHistory(){

sessionCont = document.getElementById("history-cookies");

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

switch (i) {

case 0:

sessionCont.innerHTML += "<tr><td>Интересы</td><td>"+sessionStorage.getItem("Интересы")+"</td></tr>";

break;

case 1:

sessionCont.innerHTML += "<tr><td>Учеба</td><td>"+sessionStorage.getItem("Учеба")+"</td></tr>";

break;

case 2:

sessionCont.innerHTML += "<tr><td>Контакт</td><td>"+sessionStorage.getItem("Контакт")+"</td></tr>";

break;

case 3:

sessionCont.innerHTML += "<tr><td>Фотоальбом</td><td>"+sessionStorage.getItem("Фотоальбом")+"</td></tr>";

break;

case 4:

sessionCont.innerHTML += "<tr><td>Тест</td><td>"+sessionStorage.getItem("Тест")+"</td></tr>";

break;

case 5:

sessionCont.innerHTML += "<tr><td>История</td><td>"+sessionStorage.getItem("История")+"</td></tr>";

break;

}

}

}

getLocalHistory();

getSessionHistory();

Файл calendar.js

var d0 = new Date(1910, 00, 01);

var d1 = new Date();

var dS = d1;

var year;

var month;

var text;

var calendar;

window.onload = function setDefault(){

year = document.getElementById("calendar-year");

month = document.getElementById("calendar-month");

calendar = document.getElementById("calendar-table");

text = document.getElementById("calendar-text");

for(var i = d1.getFullYear(); i>=d0.getFullYear(); i--){

year.innerHTML += "<option value =\""+i+"\" >"+i+"</option>";

}

for(var i = 1; i<=12; i++){

month.innerHTML += "<option value =\""+i+"\" >"+i+"</option>";

}

month.value = d1.getMonth()+1

year.value = d1.getFullYear()

updateDate()

}

function updateDate(){

dS.setFullYear(year.value);

dS.setMonth(month.value-1);

text.value = dS.getDate()+" "+(dS.getMonth()+1)+" "+dS.getFullYear();

var days = [];

calendar.innerHTML = "<tr><th>Пн</th><th>Вт</th><th>Ср</th><th>Чт</th><th>Пт</th><th>Сб</th><th>Вс</th></tr>"

var firstDay = new Date(dS.getFullYear(),dS.getMonth(),1).getDay()

for(var i=1; i<=daysInMonth(dS.getFullYear(), dS.getMonth()+1); i++) days.push(i);

if(firstDay==0) firstDay = 7;

var row = 1;

var isNullDays = true;

while(days.length>0){

calendar.innerHTML += "<tr id=\"row"+row+"\"></tr>";

for(var i=0; i<7; i++){

if(i==firstDay-1) isNullDays = false;

if(!isNullDays){

if(days.length==0){

document.getElementById("row"+row).innerHTML += "<td></td>";

} else{

if(days[0]==dS.getDate()) document.getElementById("row"+row).innerHTML += "<td class=\"day selected\" id=\"day"+days[0]+"\" onclick=\"selectDate(this)\"><a>"+days[0]+"</a></td>";

else document.getElementById("row"+row).innerHTML += "<td class=\"day\" id=\"day"+days[0]+"\" onclick=\"selectDate(this)\"><a>"+days[0]+"</a></td>";

days.splice(0,1);

}

} else document.getElementById("row"+row).innerHTML += "<td></td>";

}

row++;

}

t = setTimeout(function() {

updateDate()

}, 500);

}

function daysInMonth(year, month) {

return new Date(year, month, 0).getDate();

}

function selectDate(element){

dS.setDate(element.children[0].innerHTML)

updateDate()

}

Файл contact.js

const FIO = document.getElementById('fullname');

const email = document.getElementById('email');

const phone = document.getElementById('phone');

const form = document.getElementById('form');

const errorElementFIO = document.getElementById('errorFIO');

const errorElementEmail = document.getElementById('errorEmail');

const errorElementPhone = document.getElementById('errorPhone');

const errorElementText = document.getElementById('errorText');

const textArea = document.getElementById('message');

function is\_checked() {

var choice1\_checked = document.getElementById('Choice1').checked;

var choice2\_checked = document.getElementById('Choice2').checked;

return choice1\_checked || choice2\_checked;

}

FIO.addEventListener('blur', (e) => {

let messages = [];

let FIORegEng = /[A-Za-z]+ [A-Za-z]+ [A-Za-z]+/;

let FIORegRus = /[А-ЯЁа-яё]+ [А-ЯЁа-яё]+ [А-ЯЁа-яё]+/;

if (FIO.value === '' || FIO.value == null) {

FIO.focus();

messages.push('Поле ФИО не заполнено');

} else if (FIORegEng.test(FIO.value) == false && FIORegRus.test(FIO.value) == false) {

FIO.focus();

messages.push('Поле ФИО заполнено не верно');

}

if (messages.length > 0) {

e.preventDefault();

errorElementFIO.innerText = messages.join(', ');

}

if (messages.length == 0) {

errorElementFIO.innerText = "";

}

})

email.addEventListener('blur', (e) => {

let messages = [];

if (email.value === '' || email.value == null) {

email.focus();

messages.push('Поле Емаил не заполнено');

}

if (messages.length > 0) {

e.preventDefault();

errorElementEmail.innerText = messages.join(', ');

}

if (messages.length == 0) {

errorElementEmail.innerText = "";

}

})

phone.addEventListener('blur', (e) => {

let messages = [];

let phoneReg = /^[+][7|3]\d{8,10}$/g;

if (phone.value === '' || phone.value == null) {

phone.focus();

messages.push('Поле телефон не заполнено');

} else if (phoneReg.test(phone.value) == false) {

phone.focus();

messages.push('Поле телефон заполнено не верно');

}

if (messages.length > 0) {

e.preventDefault();

errorElementPhone.innerText = messages.join(', ');

}

if (messages.length == 0) {

errorElementPhone.innerText = "";

}

})

textArea.addEventListener('blur', (e) => {

let messages = [];

if (textArea.value === '' || textArea.value == null) {

textArea.focus();

messages.push('Поле текста не заполнено');

}

if (messages.length > 0) {

e.preventDefault();

errorElementText.innerText = messages.join(', ');

}

if (messages.length == 0) {

errorElementText.innerText = "";

}

})

form.addEventListener('submit', (e) => {

let messages = [];

if (is\_checked() == false) {

messages.push('Пол не выбран');

}

if (messages.length > 0) {

e.preventDefault();

errorElement.innerText = messages.join(', ');

}

})

Файл time.js

function checkTime(i) {

if (i < 10) {

i = "0" + i;

}

return i;

}

function startTime() {

var today = new Date();

var h = today.getHours();

var m = today.getMinutes();

var s = today.getSeconds();

var today = new Date();

var dd = String(today.getDate()).padStart(2, '0');

var mm = String(today.getMonth() + 1).padStart(2, '0'); //January is 0!

var yyyy = today.getFullYear();

// add a zero in front of numbers<10

m = checkTime(m);

s = checkTime(s);

document.getElementById('time').innerHTML = h + ":" + m + ":" + s;

document.getElementById('date').innerHTML = dd + "." + mm + "." + yyyy;

t = setTimeout(function() {

startTime()

}, 500);

}

startTime();

5.Вывод



Рисунок 4.1 - Отображение в области меню сайта текущих даты и времени

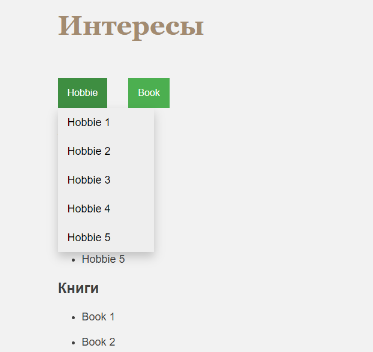


Рисунок 4.2 - Пункт меню «Мои интересы» в виде выпадающего меню

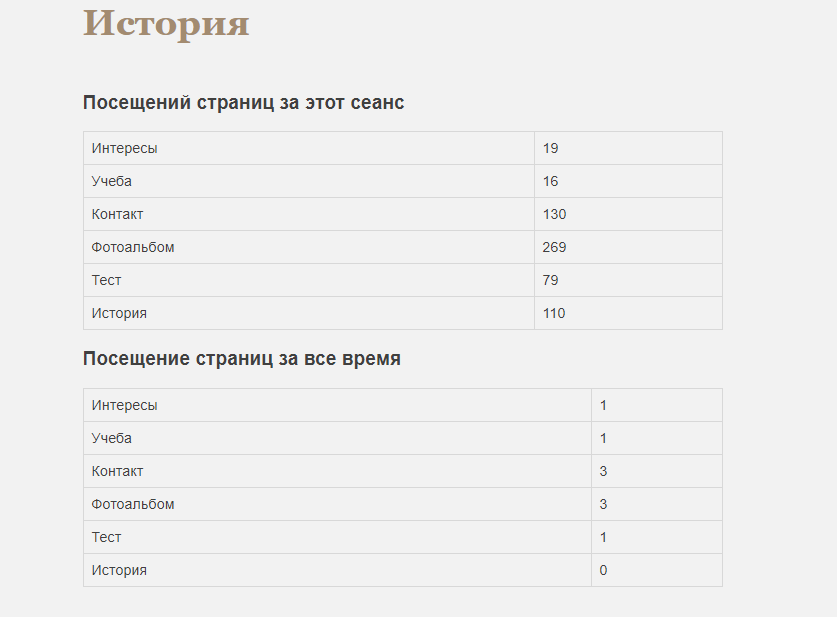


Рисунок 4.3 - Страница история

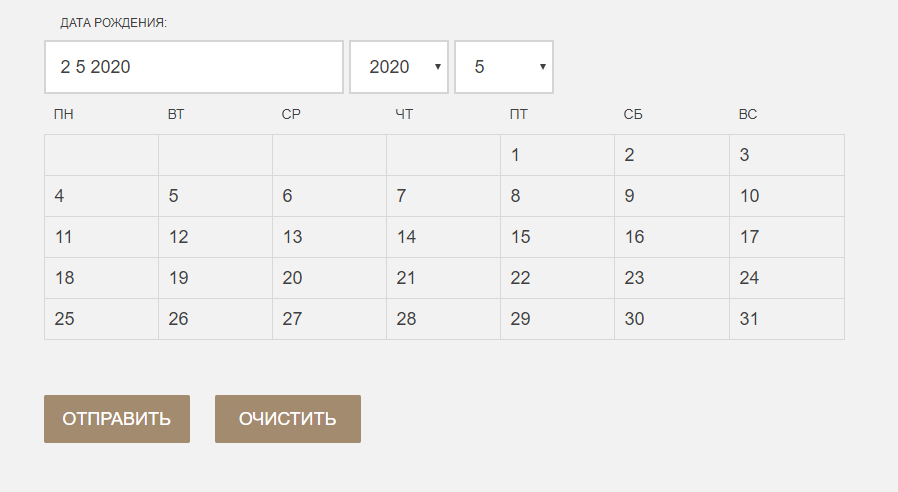


Рисунок 4.4 - Поле «Дата рождения», для которого реализован элемент «календарь»

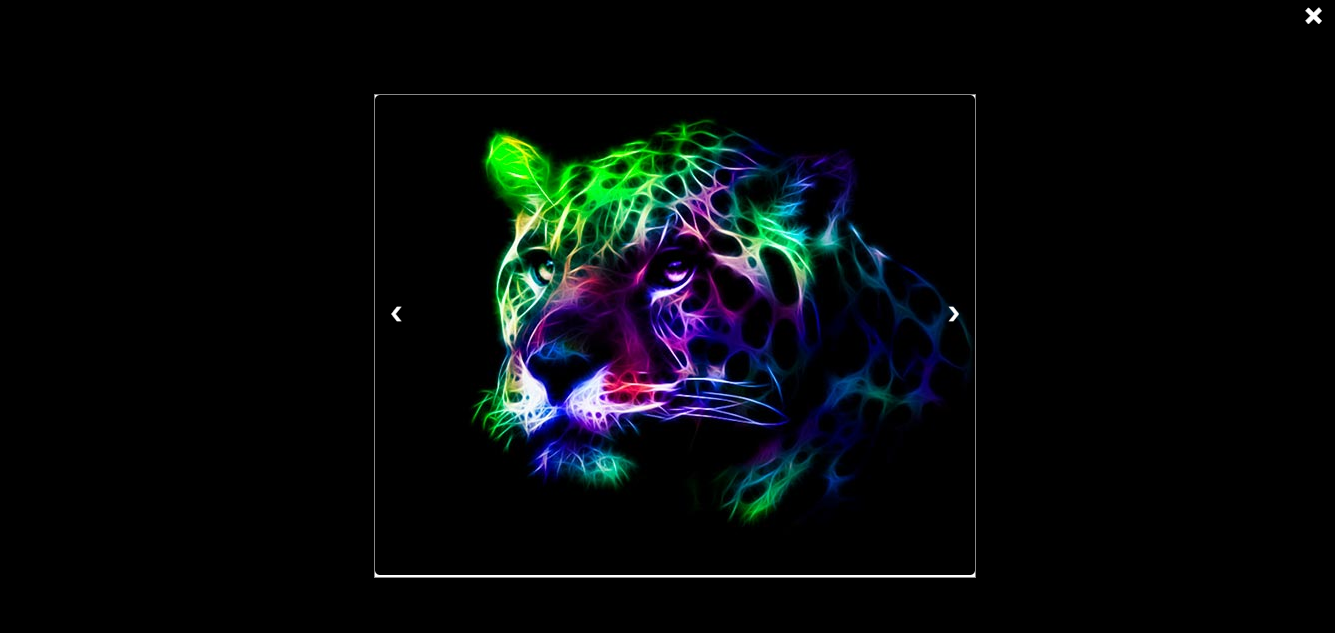


Рисунок 4.5 - Вывод больших фото при щелчке мыши по маленьким фото на странице «Фотоальбом»

5.Вывод

В данной лабораторной работе была изучена динамическая объектная модель документа, предоставляемую стандартом DOM и система событий языка JavaScript и возможность хранения данных на стороне клиента, а также были приобретены практические навыки работы с событиями JavaScript, деревом документа, Local Storage и Cookies.