МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федерально автономное образовательное учреждение высшего образования

«Севастопольский государственный университет»

кафедра Информационных систем

Куркчи Ариф Эрнестович

Институт информационных технологий и управления в технических системах

курс 3 группа ИС/б-31-о

09.03.02 Информационные системы и технологии (уровень бакалавриата)

ОТЧЕТ

по лабораторной работе №3

по дисциплине «Веб-технологии»

на тему «Исследование объектной модели документа (DOM)

и системы событий Java Script.»

Отметка о зачете \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_

(дата)

Руководитель практикума

ст. преподаватель   А. Л. Овчинников

(должность) (подпись) (инициалы, фамилия)

Севастополь 2017

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.

1. ПОСТАНОВКА ЗАДАЧИ

Реализовать интерактивное графическое меню сайта, в соответствии со следующими требованиями:

- при наведении мыши на соответствующий пункт меню изображение, соответствующее этому пункту меняется на другое (в том случае если этот пункт не активен, т.е. не загружена соответствующая ему страница);

- сделать пункт меню “Мои интересы” в виде выпадающего меню, каждый элемент которого ведет на соответствующую ему подсекцию данной страницы. Выпадающее меню реализовать с помощью тегов <ul> и <li>. Данный список должен раскрываться в зависимости от варианта задания:

|  |  |
| --- | --- |
| 1 | Выпадающее меню раскрывается по наведению |

Реализовать отображение в области меню сайта текущих даты и времени (обновление времени 1 раз в секунду). Формат даты определяется как остаток от деления последней цифры зачетной книжки на 4:

|  |  |
| --- | --- |
| 3 | ЧЧ Месяц ГГГГ |

(«День недели» – полная запись дня недели на русском языке, «Месяц» - полная запись названия месяца на русском языке)

3.3. На странице «Контакт» добавить поле «Дата рождения», для которого реализовать всплывающий снизу элемент «календарь».

При этом элемент должен давать возможность менять дату, месяц и код. Верстка элемента должна быть выполнена с помощью тегов <div> и <select> (для выбора месяца и года). Варианты задания элемента «календарь» приведены ниже:

|  |  |  |
| --- | --- | --- |
| 3 | English | день.месяц.год |

3.4. 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 корректно, оно должно быть подсвечено зеленым цветом, иначе оно должно быть подсвечено красным, а после данного поля должна появиться надпись, поясняющая характер ошибки. После исправления пользователем ошибки, надпись должна исчезнуть. Если все поля формы заполнены корректно должна стать активной кнопка «Отправить».

3.5. Реализовать открытие в динамически формируемом новом окне (блоке DIV) соответствующих больших фото при щелчке мыши по маленьким фото на странице «Фотоальбом».

3.6. 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).

Для реализации страницы «История просмотра» необходимо добавить на каждую страницу вызов JavaScript функции, которая будет сохранять информацию о просмотре страницы в Local Storage и Cookies.

1. ИСХОДНЫЙ КОД

calendar.js

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | 1 | *function* calendar\_parse(*value*, *not\_now*) { | | 2 | if (value) { | | 3 | value = value.split('.'); | | 4 | return new *Date*(value[2], value[1] - 1, value[0]); | | 5 | } | | 6 | return not\_now ? undefined : new *Date*(); | | 7 | } | | 8 |  | | 9 | *function* calendar\_month\_length(*date*) { | | 10 | *var* tmp = new *Date*(date.getFullYear(), date.getMonth() + 1, -1); | | 11 | return tmp.getDate() + 1; | | 12 | } | | 13 |  | | 14 | *function* calendar\_week\_offset(*date*) { | | 15 | *var* tmp = new *Date*(date.getFullYear(), date.getMonth(), 1); | | 16 | return tmp.getDay() == 0 ? 7 : tmp.getDay() - 1; | | 17 | } | | 18 |  | | 19 | *function* calendar\_day\_click() { | | 20 | *var* cal = *document*.getElementById('calendar'), | | 21 | field = cal.parentElement.getElementsByTagName('input')[0], | | 22 | tds = cal.getElementsByTagName('td'); | | 23 |  | | 24 | for (*var* i = 0; i < tds.length; i++) { | | 25 | tds[i].classList.remove('selected'); | | 26 | } | | 27 | this.classList.add('selected'); | | 28 | field.value = leftPad(this.innerHTML, 2, '0') + '.' + leftPad(*document*.getElementById('calendar-month').value, 2, '0') + '.' + *document*.getElementById('calendar-year').value; | | 29 | fireEvent(field, 'change'); | | 30 | cal.remove(); | | 31 | } | | 32 |  | | 33 | *function* calendar\_select\_change() { | | 34 | *var* cal = *document*.getElementById('calendar'), | | 35 | field = cal.parentElement.getElementsByTagName('input')[0], | | 36 | tbody = *document*.getElementById('calendar-body'), | | 37 | month = *document*.getElementById('calendar-month'), | | 38 | year = *document*.getElementById('calendar-year'); | | 39 | tbody.innerHTML = ''; | | 40 | calendar\_generate\_body(tbody, new *Date*(year.value, month.value - 1, 1), calendar\_parse(field.value)); | | 41 | } | | 42 |  | | 43 | *function* calendar\_generate\_months(*date*) { | | 44 | *var* month = *document*.createElement('select'), | | 45 | months = ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August', 'September', 'October', 'November', 'December']; | | 46 | for (*var* i = 0; i < months.length; i++) { | | 47 | *var* option = *document*.createElement('option'); | | 48 | option.value = i + 1; | | 49 | option.innerHTML = months[i]; | | 50 | if (i == date.getMonth()) { | | 51 | option.setAttribute('selected', 'selected'); | | 52 | } | | 53 | month.appendChild(option); | | 54 | } | | 55 | month.id = 'calendar-month'; | | 56 | month.addEventListener('change', calendar\_select\_change); | | 57 | return month; | | 58 | } | | 59 |  | | 60 | *function* calendar\_generate\_years(*date*) { | | 61 | *var* year = *document*.createElement('select'), | | 62 | now = new *Date*(); | | 63 | for (*var* i = now.getFullYear() - 100; i <= now.getFullYear(); i++) { | | 64 | *var* option = *document*.createElement('option'); | | 65 | option.innerHTML = i; | | 66 | if (i == date.getFullYear()) { | | 67 | option.setAttribute('selected', 'selected'); | | 68 | } | | 69 | year.appendChild(option); | | 70 | } | | 71 | year.id = 'calendar-year'; | | 72 | year.addEventListener('change', calendar\_select\_change); | | 73 | return year; | | 74 | } | | 75 |  | | 76 | *function* calendar\_generate\_week() { | | 77 | *var* tr = *document*.createElement('tr'); | | 78 | *var* week\_days = ['Mo', 'Tu', 'We', 'Th', 'Fr', 'Sa', 'Su']; | | 79 | for (*var* i = 0; i < week\_days.length; i++) { | | 80 | *var* th = *document*.createElement('th'); | | 81 | th.innerHTML = week\_days[i]; | | 82 | tr.appendChild(th); | | 83 | } | | 84 | return tr; | | 85 | } | | 86 |  | | 87 | *function* calendar\_generate\_body(*tbody*, *date*, *selected*) { | | 88 | *var* month\_length = calendar\_month\_length(date), | | 89 | week\_offset = calendar\_week\_offset(date), | | 90 | tr = *document*.createElement('tr'); | | 91 | for (*var* day = 1, pos = 1; day <= month\_length || pos % 7 != 1; pos++) { | | 92 |  | | 93 | *var* td = *document*.createElement('td'); | | 94 | td.className = 'disabled'; | | 95 | if (week\_offset > 0) { | | 96 | week\_offset--; | | 97 | } else if (day <= month\_length) { | | 98 | td.innerHTML = day + ''; | | 99 | td.className = selected && | | 100 | selected.getFullYear() == date.getFullYear() && | | 101 | selected.getMonth() == date.getMonth() && | | 102 | selected.getDate() == day ? 'selected' : ''; | | 103 | td.addEventListener('click', calendar\_day\_click); | | 104 | day++; | | 105 | } | | 106 | tr.appendChild(td); | | 107 | if (pos % 7 == 0) { | | 108 | tbody.appendChild(tr); | | 109 | tr = *document*.createElement('tr'); | | 110 | } | | 111 | } | | 112 | } | | 113 |  | | 114 | *function* create\_calendar(*value*) { | | 115 | *var* el = *document*.createElement('div'), | | 116 | header = *document*.createElement('div'), | | 117 | body = *document*.createElement('div'), | | 118 | table = *document*.createElement('table'), | | 119 | thead = *document*.createElement('thead'), | | 120 | tbody = *document*.createElement('tbody'), | | 121 | date = calendar\_parse(value); | | 122 |  | | 123 | header.appendChild(calendar\_generate\_months(date)); | | 124 | header.appendChild(calendar\_generate\_years(date)); | | 125 | header.className = 'calendar-header'; | | 126 |  | | 127 | thead.appendChild(calendar\_generate\_week()); | | 128 |  | | 129 | calendar\_generate\_body(tbody, date, calendar\_parse(value, true)); | | 130 | tbody.id = 'calendar-body'; | | 131 |  | | 132 | table.appendChild(thead); | | 133 | table.appendChild(tbody); | | 134 |  | | 135 | body.appendChild(table); | | 136 | body.className = 'calendar-body'; | | 137 |  | | 138 | el.appendChild(header); | | 139 | el.appendChild(body); | | 140 | el.id = 'calendar'; | | 141 |  | | 142 | el.addEventListener('mouseleave', *function*() { | | 143 | el.remove(); | | 144 | }); | | 145 |  | | 146 | return el; | | 147 | } | | 148 |  | | 149 | *function* open\_calendar() { | | 150 | *var* calendar = *document*.getElementById('calendar'); | | 151 | if (calendar) { | | 152 | calendar.remove(); | | 153 | } | | 154 | fireEvent(this, 'change'); | | 155 | this.parentNode.appendChild(create\_calendar(this.value)); | | 156 | } | | 157 |  | | 158 | *var* once\_calendar = true; | | 159 | *document*.body.addEventListener('load', *function* () { | | 160 | if (!once\_calendar) { | | 161 | return; | | 162 | } | | 163 | once\_calendar = false; | | 164 |  | | 165 | *var* calendars = *document*.getElementsByClassName('js-calendar'); | | 166 | for(*var* i=0;i<calendars.length;i++) { | | 167 | calendars[i].addEventListener('click', open\_calendar); | | 168 | } | | 169 | }, true); | |  |

contact.js

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | 1 | *function* validate() { | | 2 | *var* valid\_global = validate\_form(*document*.form\_\_contact), | | 3 | valid\_specific = validate\_contacts(*document*.form\_\_contact); | | 4 | *document*.form\_\_contact.submit.disabled = valid\_global || valid\_specific; | | 5 | } | | 6 |  | | 7 | *function* validate\_contacts(*form*) { | | 8 | *var* validates = [ | | 9 | validate\_field(form.name, /^[\wа-яё]+\s+[\wа-яё]+\s\*[\wа-яё]\*$/i, 'Введите в формате: Фамилия Имя Отчество'), | | 10 | validate\_field(form.tel, /^\+?(?:380(\d){9}|7\s\*\(?\d{3}\)?\s\*(?:-?\d){7})$/i, 'Допустимы действительные номера +7 и +380'), | | 11 | validate\_field(form.email, /^.+@.+\..+$/i, 'Введите действительную почту') | | 12 | ]; | | 13 | for (*var* i = 0; i < validates.length; i++) { | | 14 | if (validates[i]) { | | 15 | return true; | | 16 | } | | 17 | } | | 18 | return false; | | 19 | } | | 20 |  | | 21 | *function* initValidation() { | | 22 | *var* form = *document*.form\_\_contact, | | 23 | inputs = form.getElementsByTagName('input'), | | 24 | selects = form.getElementsByTagName('select'), | | 25 | i; | | 26 | for (i = 0; i < inputs.length; i++) { | | 27 | inputs[i].addEventListener('change', validate); | | 28 | if (inputs[i].type != 'checkbox' && inputs[i].type != 'radio') { | | 29 | inputs[i].addEventListener('keyup', validate); | | 30 | } | | 31 | } | | 32 | for (i = 0; i < selects.length; i++) { | | 33 | selects[i].addEventListener('change', validate); | | 34 | } | | 35 | } | | 36 |  | | 37 | *var* once\_contact = true; | | 38 | *document*.body.addEventListener('load', *function* () { | | 39 | if (!once\_contact) { | | 40 | return; | | 41 | } | | 42 | once\_contact = false; | | 43 |  | | 44 | initValidation(); | | 45 | }, true); | |  |

main.js

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | 1 | *function* validate\_form(*form*) { | | 2 | *var* hasError = false; | | 3 | for (*var* i = 0, len = form.elements.length; i < len; i++) { | | 4 | *var* element = form.elements[i], | | 5 | p = parent(element, 'form-group'); | | 6 | if (p) { | | 7 | *var* messages = p.getElementsByClassName('validation-message'); | | 8 | for (*var* j = 0; j < messages.length; j++) { | | 9 | messages[j].remove(); | | 10 | } | | 11 | p.classList.remove('has-success', 'has-error'); | | 12 | if (element.required && element.value.trim() == '') { | | 13 | p.classList.add('has-error'); | | 14 | hasError = true; | | 15 | } else { | | 16 | p.classList.add('has-success'); | | 17 | } | | 18 | } | | 19 | } | | 20 | return hasError; | | 21 | } | | 22 |  | | 23 | *function* parent(*el*, *cls*) { | | 24 | while ((el = el.parentElement) && !el.classList.contains(cls)); | | 25 | return el; | | 26 | } | | 27 |  | | 28 | *function* validate\_field(*field*, *regex*, *message*) { | | 29 | *var* hasError = false; | | 30 | if (field) { | | 31 | *var* value = field.value, | | 32 | p = parent(field, 'form-group'); | | 33 | if (!regex.test(value)) { | | 34 | p.classList.remove('has-success'); | | 35 | p.classList.add('has-error'); | | 36 | hasError = true; | | 37 | if (message) { | | 38 | *var* msg = *document*.createElement('div'); | | 39 | msg.classList.add('validation-message'); | | 40 | msg.innerHTML = message; | | 41 | p.appendChild(msg); | | 42 | field.setCustomValidity(message); | | 43 | } else { | | 44 | field.setCustomValidity('Поле заполнено некорректно') | | 45 | } | | 46 | } else { | | 47 | p.classList.add('has-success'); | | 48 | p.classList.remove('has-error'); | | 49 | field.setCustomValidity(''); | | 50 | } | | 51 | } | | 52 | return hasError; | | 53 | } | | 54 |  | | 55 | *function* navbar\_hover() { | | 56 | *var* navbar = *document*.getElementById('navbar'), | | 57 | lis = navbar.getElementsByTagName('li'); | | 58 | for (*var* i = 0; i < lis.length; i++) { | | 59 | lis[i].addEventListener('mouseover', *function* () { | | 60 | this.classList.add('hover'); | | 61 | }, true); | | 62 | lis[i].addEventListener('mouseout', *function* () { | | 63 | this.classList.remove('hover'); | | 64 | }, true); | | 65 | } | | 66 | } | | 67 |  | | 68 | *function* initDropdown(*cont*) { | | 69 | *var* drops = cont.getElementsByClassName('dropdown'); | | 70 | for (*var* i = 0; i < drops.length; i++) { | | 71 | *var* drop = drops.item(i); | | 72 | drop.addEventListener('mouseenter', *function* (*e*) { | | 73 | this.classList.add('open'); | | 74 | }); | | 75 | drop.addEventListener('mouseleave', *function* (*e*) { | | 76 | this.classList.remove('open'); | | 77 | }); | | 78 | } | | 79 | } | | 80 | // Storage functions | | 81 | *var* storage = typeof(Storage) !== undefined ? localStorage : undefined; | | 82 | *function* localStore(*action*, *key*, *val*) { | | 83 | *var* ret\_val = undefined; | | 84 | if (storage !== undefined) { | | 85 | *var* data; | | 86 | try { | | 87 | data = JSON.parse(storage[key]); | | 88 | } catch (e) { | | 89 | data = {}; | | 90 | } | | 91 | ret\_val = data; | | 92 | if (action === 'set' && val != undefined) { | | 93 | data = val; | | 94 | } | | 95 | storage[key] = JSON.stringify(data); | | 96 | } | | 97 | return ret\_val; | | 98 | } | | 99 | *function* localSet(*key*, *val*) { | | 100 | return localStore('set', key, val); | | 101 | } | | 102 | *function* localGet(*key*, *def*) { | | 103 | return localStore('get', key, def); | | 104 | } | | 105 | *function* setCookie(*name*, *value*, *days*) { | | 106 | *var* expires = ""; | | 107 | if (days) { | | 108 | *var* date = new *Date*(); | | 109 | date.setTime(date.getTime() + (days \* 24 \* 60 \* 60 \* 1000)); | | 110 | expires = "; expires=" + date.toUTCString(); | | 111 | } | | 112 | *document*.cookie = name + "=" + JSON.stringify(value); | | 113 | } | | 114 | *function* getCookie(*name*) { | | 115 | *var* matches = *document*.cookie.match(new *RegExp*( | | 116 | "(?:^|; )" + name.replace(/([\.$?\*|{}\(\)\[\]\\\/\+^])/g, '\\$1') + "=([^;]\*)" | | 117 | )); | | 118 | return matches ? JSON.parse(decodeURIComponent(matches[1])) : undefined; | | 119 | } | | 120 |  | | 121 | *function* visit() { | | 122 | *var* local = localGet('visits', {}), | | 123 | cookie = getCookie('visits'), | | 124 | loc = *window*.location.pathname; | | 125 | if (cookie === undefined) { | | 126 | cookie = {}; | | 127 | } | | 128 | if (!local[loc]) { | | 129 | local[loc] = 0; | | 130 | } | | 131 | if (!cookie[loc]) { | | 132 | cookie[loc] = 0; | | 133 | } | | 134 | local[loc]++; | | 135 | cookie[loc]++; | | 136 | localSet('visits', local); | | 137 | setCookie('visits', local); | | 138 | } | | 139 |  | | 140 | *var* clock = *document*.getElementById('clock'), | | 141 | months = [ | | 142 | 'Января', 'Февраля', ' Марта', 'Апреля', 'Майя', 'Июня', 'Июля', 'Августа', 'Сентября', 'Октября', 'Ноября', 'Декабря' | | 143 | ]; | | 144 |  | | 145 | *function* leftPad(*str*, *len*, *ch*) { | | 146 | str = str + ''; | | 147 | len = len - str.length; | | 148 | if (len <= 0) return str; | | 149 | if (!ch && ch !== 0) ch = ' '; | | 150 | ch = ch + ''; | | 151 | *var* pad = ''; | | 152 | while (true) { | | 153 | if (len & 1) pad += ch; | | 154 | len >>= 1; | | 155 | if (len) ch += ch; | | 156 | else break; | | 157 | } | | 158 | return pad + str; | | 159 | } | | 160 |  | | 161 | *function* fireEvent(*element*, *event\_type*) { | | 162 | *var* event; | | 163 | if (*document*.createEvent) { | | 164 | event = *document*.createEvent('HTMLEvents'); | | 165 | event.initEvent(event\_type, true, true); | | 166 | } else { | | 167 | event = *document*.createEventObject(); | | 168 | event.eventType = event\_type; | | 169 | } | | 170 |  | | 171 | event.eventName = event\_type; | | 172 |  | | 173 | if (*document*.createEvent) { | | 174 | element.dispatchEvent(event); | | 175 | } else { | | 176 | element.fireEvent('on'.event.eventType, event); | | 177 | } | | 178 | } | | 179 |  | | 180 | *function* initClock() { | | 181 | *var* current = new *Date*(); | | 182 |  | | 183 | clock.innerHTML = leftPad(current.getDay(), 2, '0') + ' ' + | | 184 | months[current.getMonth() - 1] + ' ' + | | 185 | current.getFullYear() + ' ' + | | 186 | leftPad(current.getHours(), 2, 0) + ':' + | | 187 | leftPad(current.getMinutes(), 2, '0') + ':' + | | 188 | leftPad(current.getSeconds(), 2, '0'); | | 189 |  | | 190 | setTimeout(initClock, 100); | | 191 | } | | 192 |  | | 193 | *var* once = true; | | 194 | *document*.body.addEventListener('load', *function* () { | | 195 | if (!once) { | | 196 | return; | | 197 | } | | 198 | once = false; | | 199 | navbar\_hover(); | | 200 | initDropdown(this); | | 201 | initClock(); | | 202 |  | | 203 | visit(); | | 204 | }, true); | |  |

history.js

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | 1 | *function* fillHistory(*cont*, *history*) { | | 2 | *var* html = ''; | | 3 | for(*var* page in history) { | | 4 | html += '<tr><td><a href="'+page+'">'+page+'</a></td><td>'+history[page]+'</td></tr>'; | | 5 | } | | 6 | cont.innerHTML = html; | | 7 | } | | 8 |  | | 9 | *var* onceHistory = true; | | 10 | *document*.body.addEventListener('load', *function*() { | | 11 | if(!onceHistory) { | | 12 | return; | | 13 | } | | 14 | onceHistory = false; | | 15 | *var* localCont = *document*.getElementById('history-local'), | | 16 | cookieCont = *document*.getElementById('history-cookies'); | | 17 | fillHistory(localCont, localGet('visits', {})); | | 18 | fillHistory(cookieCont, getCookie('visits')); | | 19 | }, true); | |  |

ВЫВОДЫ

В ходе лабораторной работы были повторены основы работы с динамической объектной моделью документа, представляемой стандартов DOM и систему событий языка JavaScript. Написаны функции взаимодействия с деревом документа, событиями и хранилищами Local Storage и Cookies.