**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 № 8

по дисциплине «Разработка приложений для Интернет»

на тему: «Обработка и валидация форм с использованием JavaScript»

Выполнил: студент гр. ИП-32

Бородина Н.Н.

Принял: преподаватель

Свинтицкий П.В.

Гомель 2023

**Цель работы**: научиться обрабатывать значения, введенные пользователем в элементы форм, применять валидацию формы на стороне клиента..

**Задание 1:** Написать скрипт, который находит среднее арифметическое введенных значений и выводит результат в поле формы при нажатии на кнопку «Вычислить». В программе предусмотреть обработку исключительной ситуации, когда пользователь не ввел значений (хотя бы одного) или введенные значения не являются числами, с выводом сообщения в модальном окне. Модальное окно реализовать самостоятельно, не использую стандартные диалоговые окна объекта window.

Код файла lab\_8\_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="preconnect" href="https://fonts.googleapis.com">

<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

<link href="https://fonts.googleapis.com/css2?family=UnifrakturCook:wght@700&display=swap" rel="stylesheet">

<title>lab 8.1 Бородина Наталья ИП-32</title>

<style>

\* {

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

}

form {

display: flex;

flex-direction: column;

row-gap: 10px;

}

label {

margin-right: 10px;

}

button {

width: fit-content;

}

textarea {

width: 400px;

height: 100px;

}

div p {

width: 90%;

padding: 5px;

margin: 10px;

}

#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: 'UnifrakturCook', cursive;

font-weight: 300;

font-size: 13pt;

color: black;

border: 2px solid grey;

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;

}

#modal\_window\_entry p {

font-family: 'UnifrakturCook', cursive;

}

#modal\_window\_entry button {

font-family: 'UnifrakturCook', cursive;

}

</style>

</head>

<body>

<div id='modal\_window' hidden>

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

<p>Ich wollte zur Wolga kommen, und zwar an einer bestimmten Stelle, an einer bestimmten Stadt.

Zufaelligerweise traegt sie den Namen von Stalin selber!</p>

<p id='achtung\_text'></p>

<div>

<button onclick='achtung\_off()'>Ok</button>

</div>

</div>

</div>

<form onsubmit='calculate(event)'>

<textarea>Написать скрипт, который находит среднее арифметическое введенных значений и выводит результат в поле формы при нажатии на кнопку «Вычислить». В программе предусмотреть обработку исключительной ситуации, когда пользователь не ввел значений (хотя бы одного) или введенные значения не являются числами, с выводом сообщения в модальном окне. Модальное окно реализовать самостоятельно, не используя стандартные диалоговые окна объекта window.

</textarea>

<span>

<label for='input\_a'>Введите число a:</label>

<input type='text' id='input\_a' value='1'>

</span>

<span>

<label for='input\_b'>Введите число b:</label>

<input type='text' id='input\_b' value=''>

</span>

<span>

<label for='input\_c'>Введите число c:</label>

<input type='text' id='input\_c' value='6'>

</span>

<span>

<label for='result\_field'>Результат:</label>

<input type='text' id='result\_field' required readonly>

</span>

<button id="submit\_button" tabindex='1'>Вычислить</button>

</form>

<script>

"use strict"

function calculate(event) {

// Предотвращает перезагрузку страницы.

event.preventDefault();

// Очищает поле результата.

result\_field.value = '';

let str\_a = input\_a.value;

let str\_b = input\_b.value;

let str\_c = input\_c.value;

let a = parseInt(str\_a);

let b = parseInt(str\_b);

let c = parseInt(str\_c);

if (!a || isNaN(str\_a)) {

achtung\_on('a');

} else if (!b || isNaN(str\_b)) {

achtung\_on('b');

} else if (!c || isNaN(str\_c)) {

achtung\_on('c');

} else {

result\_field.value = (a + b + c) / 3;

}

}

function achtung\_on(number) {

achtung\_text.innerHTML = 'Korrigieren Sie den Wert der Variablen: ' + number;

modal\_window.removeAttribute('hidden');

}

function achtung\_off() {

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

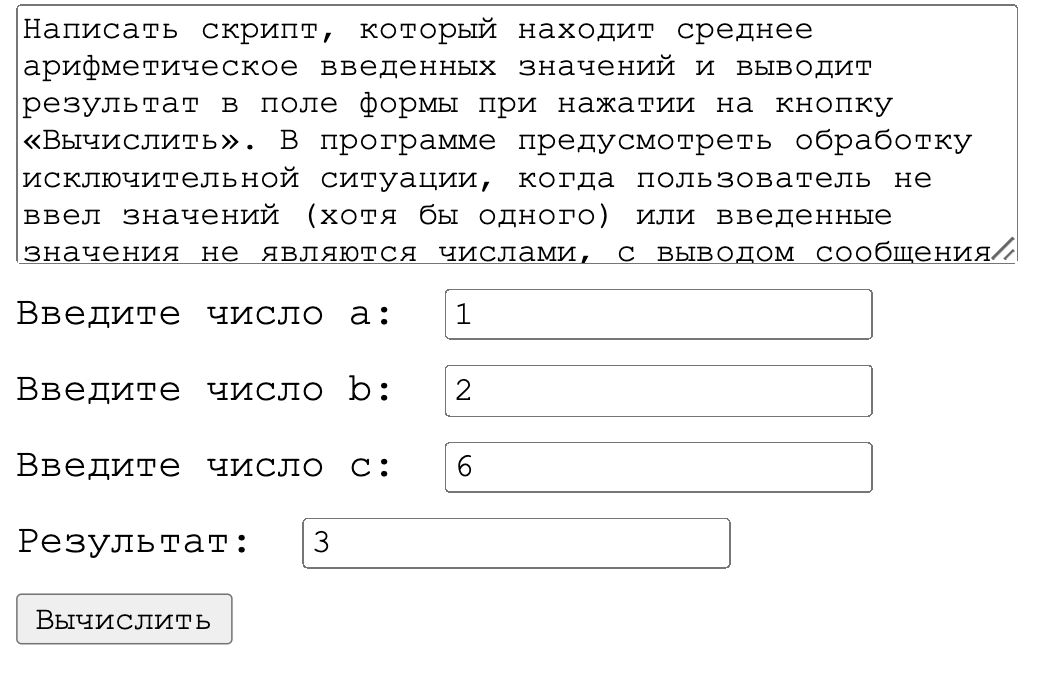
}

</script>

</body>

</html>

Результат выполнения 1-го задания:



**Задание 2:** Необходимо реализовать форму с валидацией. Валидация осуществляется без использования атрибутов элементов ввода HTML. В поле паспорт могут вводится только две латинские буквы, в поле номер паспорта – только 7 цифр, номер бланка ответа – 7 цифр, предмет предполагает выбор языка (белорусский или русский), математика, физика. Код на картинке – 6 цифр. При не заполнении какого-либо поля должна выводится соответствующая подсказка.

Код файла lab\_8\_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">

<title>lab 8.2 Бородина Наталья ИП-32</title>

<style>

\* {

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

}

.prompt {

color: red;

display: none;

}

div {

width: 90%;

padding: 5px;

margin: 10px;

}

.form\_part label,

input,

p {

display: inline;

}

</style>

</head>

<body>

<form onsubmit='send\_data(event)'>

<div class='form\_part'>

<label for='input\_passportSeries'>Серия паспорта</label>

<input type='text' id='input\_passportSeries'>

<p id='alert\_passportSeries' class="prompt">Введите серию паспорта</p>

</div>

<div class='form\_part'>

<label for='input\_passportNumber'>Номер паспорта</label>

<input type='text' id='input\_passportNumber'>

<p id='alert\_passportNumber' class="prompt">Введите номер паспорта</p>

</div>

<div class='form\_part'>

<label for='input\_blankNumber'>Номер бланка</label>

<input type='text' id='input\_blankNumber'>

<p id='alert\_blankNumber' class="prompt">Введите номер паспорта</p>

</div>

<div class='form\_part'>

<label for='input\_subject'>Предмет</label>

<select id='input\_subject'>

<option>Белорусский</option>

<option>Русский</option>

<option>Математика</option>

<option>Физика</option>

<option value='' selected disabled hidden>Выберите предмет</option>

</select>

<p id='alert\_subject' class="prompt">Выберите предмет</p>

</div>

<div class='form\_part'>

<img src="image.png"><br>

<label for='input\_imageCode'>Код на картинке</label>

<input type='text' id='input\_imageCode'>

<p id='alert\_imageCode' class="prompt">Введите код</p>

</div>

<button id="submit\_button" tabindex='1'>Найти</button>

</form>

<script>

"use strict"

function send\_data(event) {

// Предотвращает перезагрузку страницы.

event.preventDefault();

let passportSeries = check\_passportSeries();

let passportNumber = check\_passportNumber();

let blankNumber = check\_blankNumber();

let subject = check\_subject();

let imageCode = check\_imageCode();

if (passportSeries && passportNumber && blankNumber && subject && imageCode) {

alert('успешно');

} else {

alert('AHTUNG');

}

}

// Функция проверяет первое поле формы и скрывает или отображает предупреждение.

function check\_passportSeries() {

let str = document.getElementById("input\_passportSeries").value;

const alert = document.getElementById("alert\_passportSeries");

if (str.length === 2) {

let reg = new RegExp('[a-zA-Z][a-zA-Z]');

if (reg.test(str)) {

alertOff(alert);

return true;

} else {

alertOn(alert);

return false;

}

} else {

alertOn(alert);

return false;

}

}

function check\_passportNumber() {

let str = document.getElementById("input\_passportNumber").value;

const alert = document.getElementById("alert\_passportNumber")

if (str.length === 7) {

let reg = new RegExp('[0-9]{7}');

if (reg.test(str)) {

alertOff(alert);

return true;

} else {

alertOn(alert);

return false;

}

} else {

alertOn(alert);

return false;

}

}

function check\_blankNumber() {

let str = document.getElementById("input\_blankNumber").value;

const alert = document.getElementById("alert\_blankNumber")

if (str.length === 7) {

let reg = new RegExp('[0-9]{7}');

if (reg.test(str)) {

alertOff(alert);

return true;

} else {

alertOn(alert);

return false;

}

} else {

alertOn(alert);

return false;

}

}

function check\_subject() {

let str = document.getElementById("input\_subject").value;

const alert = document.getElementById("alert\_subject");

if (str === '') {

alertOn(alert);

return false;

} else {

alertOff(alert);

return true;

}

}

function check\_imageCode() {

let str = document.getElementById("input\_imageCode").value;

const alert = document.getElementById("alert\_imageCode");

if (str.length === 6) {

let reg = new RegExp('[0-9]{6}');

if (reg.test(str)) {

alertOff(alert);

return true;

} else {

alertOn(alert);

return false;

}

} else {

alertOn(alert);

return false;

}

}

function alertOff(element) {

element.style.display = 'none';

}

function alertOn(element) {

element.style.display = 'initial';

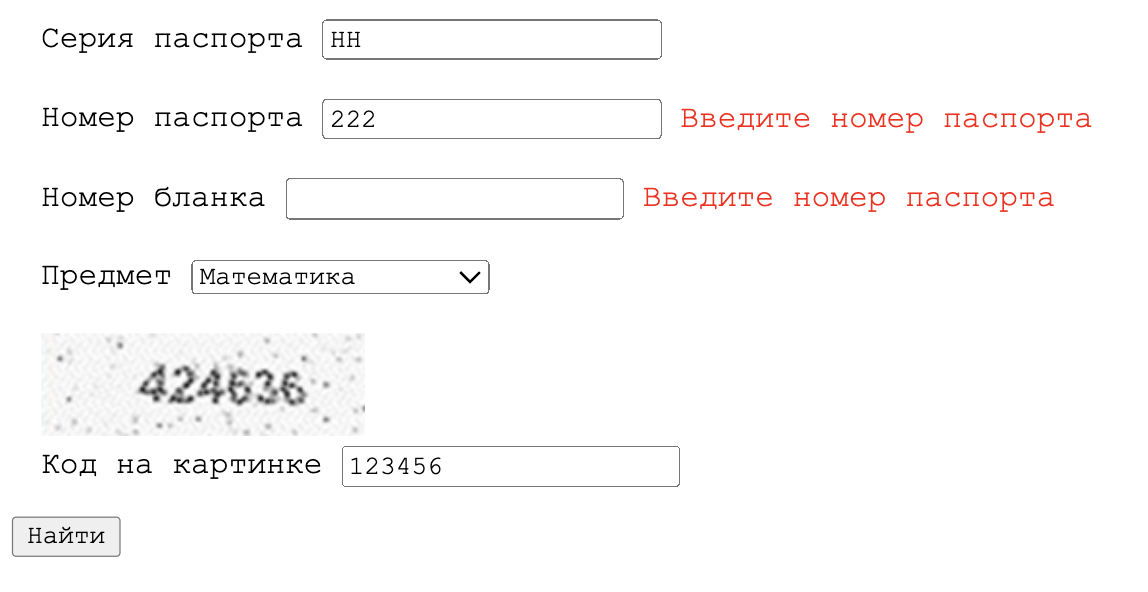
}

</script>

</body>

</html>

Результат выполнения 2-го задания:



**Задание 3:** Создайте файл с HTML-формой. Разработайте код JavaScript, в котором анализируются введенные пользователем данные, позволяющие изменить размер, положение на странице и цвет горизонтальной линии:

Вместо линии можно использовать блок и свойства CSS – width, height, backgroundColor. Для выравнивания блока используйте свойство margin.

Код файла lab\_8\_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">

<title>lab 8.3 Бородина Наталья ИП-32</title>

<style>

\* {

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

}

#line\_box {

width: 500px;

height: 250px;

border: 2px dashed black;

}

#line {

width: 300px;

height: 25px;

background-color: #ff0000;

}

</style>

</head>

<body>

<div id='line\_box'>

<div id='line'></div>

</div>

<form onsubmit="update\_style(event)">

<div class="form\_part">

<p>Выравнивание</p>

<input type="radio" value="left" name='margin\_select' id='margin\_left' checked>

<label for='margin\_left'>LEFT</label>

<br>

<input type="radio" value="center" name='margin\_select' id='margin\_center'>

<label for='margin\_center'>CENTER</label>

<br>

<input type="radio" value="right" name='margin\_select' id='margin\_right'>

<label for='margin\_right'>RIGHT</label>

</div>

<div class="form\_part">

<p>Высота</p>

<input type="number" id='input\_height' max="">

<label for='input\_height'>Толщина (px)</label>

</div>

<div class="form\_part">

<p>Ширина</p>

<input type="number" id='input\_width'>

<label for='input\_width'>Ширина (px)</label>

</div>

<div class="form\_part">

<p>Цвет</p>

<input type="color" id='input\_color' value="#ff0000">

<label for='input\_color'>Цвет</label>

</div>

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

</form>

<script>

"use strict"

// Начальные рассчитанные значения до изменений.

let line\_box\_width = parseFloat(window.getComputedStyle(line\_box).width);

let line\_box\_height = parseFloat(window.getComputedStyle(line\_box).height);

let line\_width = parseFloat(window.getComputedStyle(line).width);

let line\_height = parseFloat(window.getComputedStyle(line).height);

// Установка максимальных значений высоты и ширины для формы ввода.

input\_height.max = line\_box\_height;

input\_width.max = line\_box\_width;

// Установка актуальных значений высоты и ширины для формы ввода.

input\_height.value = line\_height;

input\_width.value = line\_width;

function update\_style(event) {

// Применение введённых значений ширины и высоты.

line.style.height = parseInt(input\_height.value) + 'px';

line.style.width = parseInt(input\_width.value) + 'px';

line\_width = parseFloat(window.getComputedStyle(line).width);

line\_height = parseFloat(window.getComputedStyle(line).height);

console.log(line\_width);

console.log(line\_height);

event.preventDefault();

// Установка выравнивания.

let margins = document.getElementsByName('margin\_select');

for (const m of margins) {

if (m.checked) {

//console.log(m.value);

if (m.value === 'left') {

line.style.marginLeft = '0px';

line.style.marginRight = (line\_box\_width - line\_width) + 'px';

} else if (m.value === 'right') {

line.style.marginRight = '0px';

line.style.marginLeft = (line\_box\_width - line\_width) + 'px';

} else if (m.value === 'center') {

line.style.marginRight = (line\_box\_width - line\_width) / 2 + 'px';

line.style.marginLeft = (line\_box\_width - line\_width) / 2 + 'px';

}

break;

}

}

// Применения введёного цвета.

line.style.backgroundColor = input\_color.value;

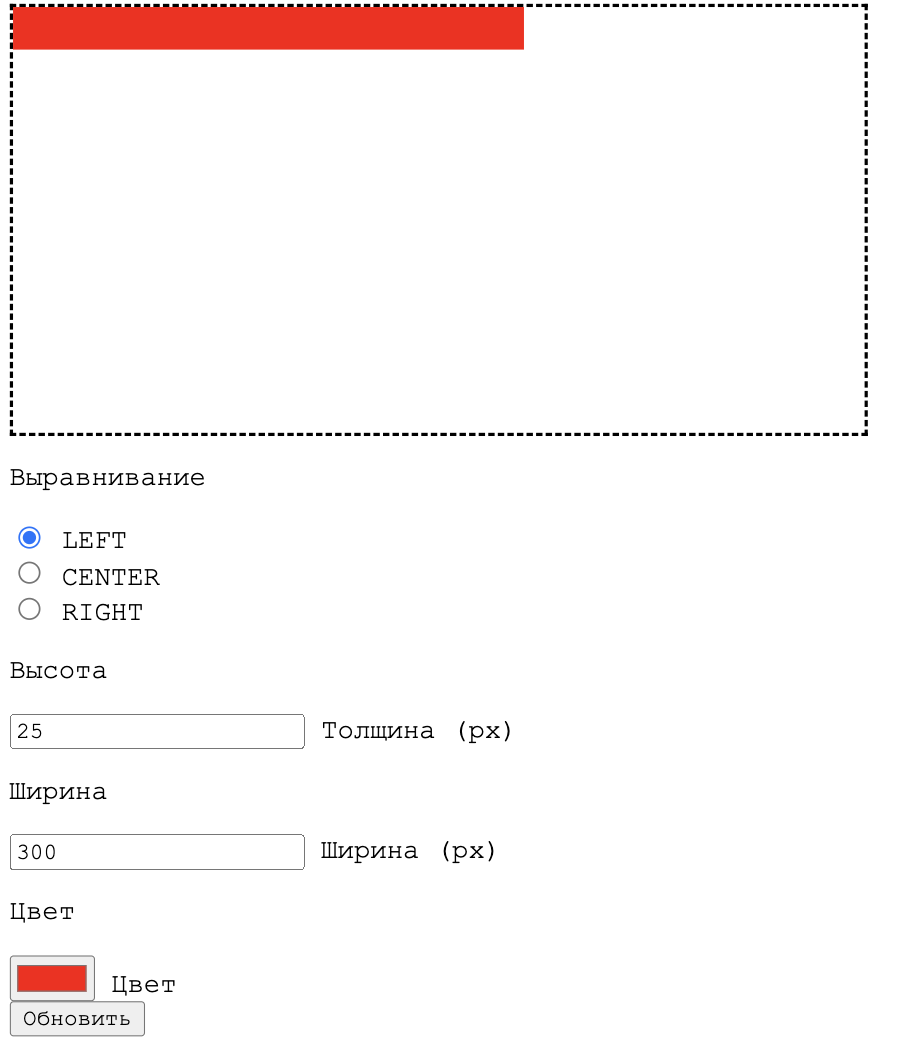
}

</script>

</body>

</html>

Результат выполнения 3-го задания:



**Задание 4:** В соответствии со своим вариантом (задание 1 лабораторной работы №6), создать форму для ввода свойств объекта. Сформировать массив объектов, организовав цикл, в котором вызывается форма для ввода как модальное окно. Введенный массив должен быть отображен на странице в виде таблицы.

Код файла lab\_8\_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">

<title>lab 8.4 Бородина Наталья ИП-32</title>

<style>

\* {

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

}

input {

float: right;

}

#input\_form {

display: flex;

flex-direction: column;

}

.form\_part {

display: inline;

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: 0;

}

#first\_button {

float: left;

}

#second\_button {

float: right;

}

#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 grey;

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;

}

table {

border: 1px solid grey;

background-color: grey;

}

th {

height: 20px;

border: 1px solid grey;

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

padding: 5px;

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

font-weight: bold;

}

td {

height: 20px;

border: 1px solid grey;

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

padding: 5px;

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

font-weight: bold;

}

</style>

</head>

<body>

<div id='modal\_window' hidden>

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

<form id='input\_form' onsubmit='process(event)'>

<div class='form\_part'>

<label for='input\_clockSpeed'>Частота:</label>

<input type='number' id='input\_clockSpeed' required value="400000000">

</div>

<div class='form\_part'>

<label for='input\_cores'>Ядра:</label>

<input type='number' id='input\_cores' required value="4">

</div>

<div class='form\_part'>

<label for='input\_capacity'>Разрядность:</label>

<input type='text' id='input\_capacity' required value="32">

</div>

<div class='form\_part'>

<label for='input\_technicalProcess'>Техпроцесс:</label>

<input type='number' id='input\_technicalProcess' required value="20">

</div>

<div class='form\_part'>

<label for='input\_release'>год выпуска:</label>

<input type='number' id='input\_release' required value="2018">

</div>

<div class='form\_part'>

<label for='input\_price'>цена:</label>

<input type='number' id='input\_price' required value="23000">

</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>

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

<script>

"use strict"

let array = [];

show\_form();

function process(event) {

// Предотвращает перезагрузку страницы.

event.preventDefault();

let type = input\_clockSpeed.value;

let capacity = parseInt(input\_cores.value);

let tire = input\_capacity.value;

let speed = parseInt(input\_technicalProcess.value);

let release = parseInt(input\_release.value);

let price = parseInt(input\_price.value);

array.push(new Intel(type, capacity, tire, speed, release, price));

// Очищаем форму

input\_form.reset();

generateTable(array);

}

function show\_form() {

modal\_window.removeAttribute('hidden');

}

function close\_form() {

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

}

function generateTable(array) {

if (document.getElementById('array\_table')) {

// Таблица уже есть.

let table = document.getElementById('array\_table');

let row = document.createElement("tr");

let properties = Object.entries(array.at(-1));

properties.forEach(key => {

let cell = document.createElement("td");

cell.innerText = key[1];

row.appendChild(cell);

});

table.appendChild(row);

} else {

// Таблицы ещё нет.

let prop\_name = ['частота', 'ядра', 'разрядность', 'техпроцесс', 'выход', 'цена'];

let table = document.createElement("table");

table.id = 'array\_table';

let row = document.createElement("tr");

prop\_name.forEach(a => {

let cell = document.createElement("th");

cell.innerText = a;

row.appendChild(cell);

});

table.appendChild(row);

array.forEach(element => {

let row = document.createElement("tr");

let properties = Object.entries(element);

properties.forEach(key => {

let cell = document.createElement("td");

cell.innerText = key[1];

row.appendChild(cell);

});

table.appendChild(row);

});

document.body.appendChild(table);

}

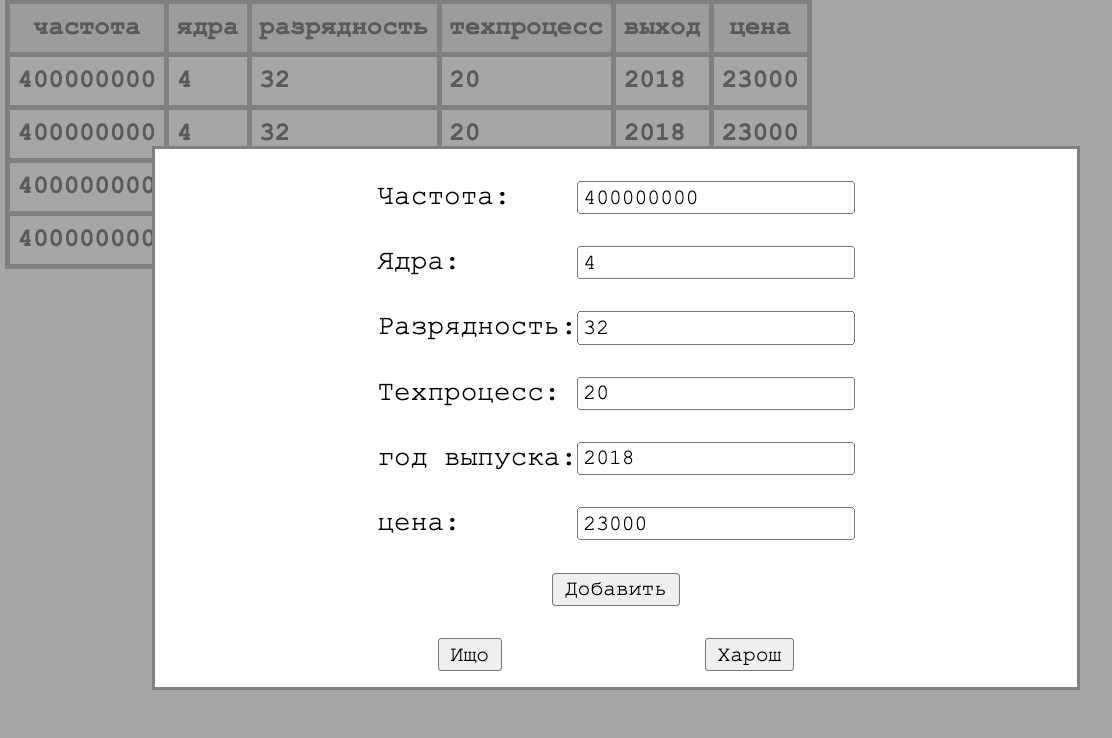
}

</script>

</body>

</html>

Результат выполнения 4-го задания:



**Выводы:** В процессе выполнения лабораторной работы научился обрабатывать значения введённые в форму. Научился использовать валидацию форм на стороне клиента.