ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет програмування та комп’ютерних

і телекомунікаційних систем

Кафедра інженерії програмного забезпечення

**Лабораторна робота №** 7

з дисципліни «Веб програмування»

на тему:

# «JAVASCRIPT»

**Виконав:**

студент 3 курсу, групи ІПЗ-17-1 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ І. С. Мисик

(підпис) (Ініціали, прізвище)

**Перевірив:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Яшина О. Н.

(підпис)

Хмельницький – 2020 р.

**Завдання 1**: Розробити Інтернет сторінку де наряду із звичайним html текстом буде міститись прізвище, ім’я, по батькові студента, надрукованого засобами Java Script. Розмістити на сторінці кнопку, за натиском якої користувачеві на екран з’являтиметься повідомлення про дату та місце народження студента.

**Завдання 2:** Добавити сторінку на якій знаходиться лише одна кнопка. На подію натиснення призначити виконання функції розрахунку арифметичної операції вибраної за номером студента в списку:

|  |  |
| --- | --- |
| 12 | https://studfile.net/html/2706/1145/html_HoZJ7_UJWj.yTQi/img-WR4rab.png |

**Дати детальну відповідь на запитання із наведенням прикладів.**

1. Функції, синтаксис, створення та виклик функції.

2.Параметри функції. Область видимості змінних.

3. Інструкція return.

4.Логічні операції у функціях.

5.Рекурсивні функції.

6.Допоміжні функції.

Хід роботи

**Завдання 1**

Програмний код сторінки.

<!DOCTYPE HTML>

<html>

<head>

<title>

Заголовок документа

</title>

<meta http-equiv="content-type" content="text/HTML; charset=utf-8"/>

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

<link href="https://fonts.googleapis.com/css2?family=Roboto&family=Rosario:wght@400;700&display=swap" rel="stylesheet">

<script>

function a()

{

alert('02.03.2000 смт. Гриців');

}

</script>

</head>

<body>

<header>

<h1>Резюме</h1>

<nav>

<div class="topnav">

<a href="#">Досвід роботи</a>

<a href="#">Освіта</a>

<a href="#">Навички</a>

<a href="#">Знання мов</a>

<a href="#">Зворотній зв'язок</a>

</div>

</nav>

</header>

<div class="about">

<script>

document.write("Мисик Іван Сергійович");

</script>

<input class="dateB" type="button" value="Дата народження" OnClick="a();">

</div>

<main>

<h3>Досвід роботи</h3>

<div class="content">

<p>З жовтня 2019 - не визначено</p>

<p>Devloop, Хмельницький — Junior Java розробник</p>

<p><strong>Обов'язки:</strong></p>

<ul>

<li>Написання коду на Java</li>

<li>Робота з Postgre SQL</li>

<li>Створення і впровадженн нових функцій у проект</li>

<li>Рефакторинг та оптимізація наявного коду</li>

<li>Розробка API для наявного сервісу</li>

</ul>

</div>

<h3>Освіта</h3>

<div class="content">

<ol>

<li>

<p>Незакінчена вища <em>(2017 - \_\_\_\_)</em> . <b>Хмельницький національний університет.</b>.</p>

<p>Факультет: Програмування та телекомунікаційних систем.<br/>Кафедра: Інженерія програмного забезпечення.</p>

</li>

<li>

<p>Середня <em>(2006 - 2017)</em>. <b>Грицівська ЗОШ</b>.</p>

</li>

</ol>

</div>

<h3>Навички</h3>

<div class="content">

<ol>

<li>Чудове розуміння <strong>Windows</strong> на рівні досвідченого користувача</li>

<li>Вільно працюю з пакетом <strong>MS Office</strong></li>

<li>Хороші знання в <strong>Java 8</strong>. Основні знання в <strong>Kotlin</strong>.</li>

<li>Знання <strong>ООП</strong>, патерни проектування</li>

<li>Розуміння таких фреймворків, як <strong>Spring, Hibernate, JAX-RS та його реалізація Jersey</strong></li>

<li>Розуміння баз даних. <strong>SQL, MySQL, PostgreSQL</strong></li>

<li>Вміння використовувати <strong>Maven</strong></li>

</ol>

</div>

<h3>Знання мов</h3>

<div class="content">

<ul>

<li>Українська - рідна</li>

<li>Російська – друга</li>

<li>Англійська - базові знання</li>

</ul>

<h4>Ключові навички</h4>

<p>Комунікабельність, стресостійкість, відкритий до вивчення нового, усидчивість, відповідальність, пунктуальність, уважність.</p>

</div>

<h3>Зворотній зв'язок</h3>

</main>

<form class="form" name="contact" action="#" method="post">

<div>

<label>Ім'я:</label>

<input type="text" name="firstname"/>

</div>

<div>

<label>Електронна пошта:</label>

<input type="email" name="email"/>

</div>

<div>

<label>Телефон:</label>

<input type="tel" name="number"/>

</div>

<div>

<label>Повідомлення:</label>

<textarea name="massege">Введіть повідомлення</textarea>

</div>

<div>

<input type="submit" name="submit" value="Відправити"/>

</div>

</form>

<div class="contact">

<p>Номер телефону: +380970760418<br/>

E-mail: ivan.mysyk04@gmail.com</p>

<div>

<a target="\_blank" href="https://t.me/fraGGou\_NPE">

<img src="images/telegram.png" alt="Telegram"/>

</a>

<a target="\_blank" href="https://instagram.com/mrfragou?igshid=hzqvzwsigwn5">

<img src="images/instagram.png" alt="instagram"/>

</a>

</div>

</div>

<footer class="footer">

<hr>

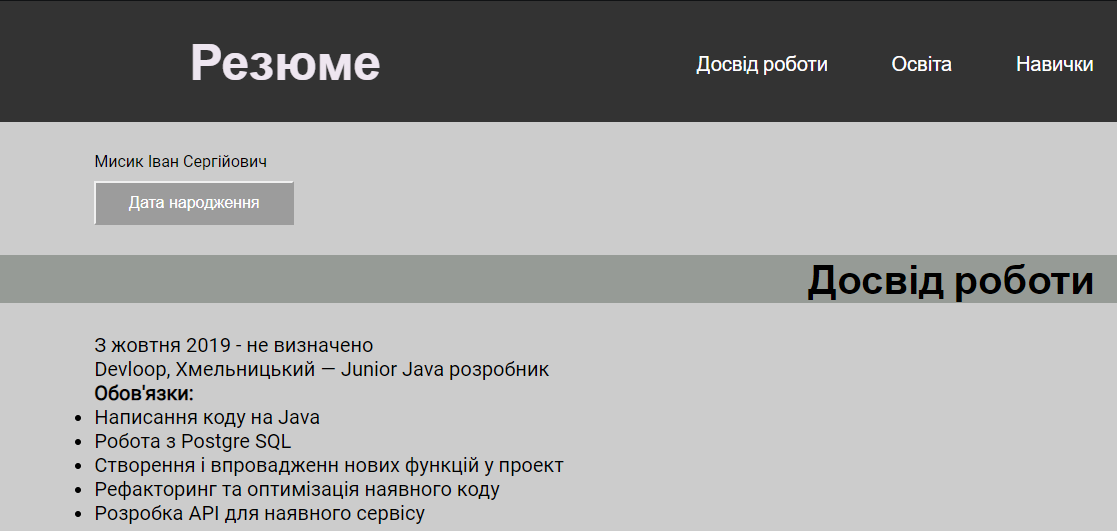
<a href="table.html">Я вмію створювати таблиці</a>

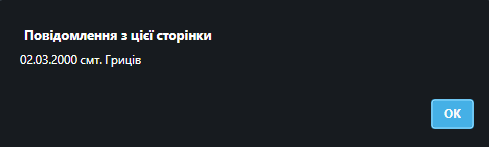
</footer>

</body>

</html>

Результат роботи.

При завантаженні сторінки виводиться ім’я та кнопка при натисканні якої спливає вікно з датою народження та місцем проживання.



**Завдання 2**

Програмний код сторінки.

<!doctype html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport"

content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">

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

<title>Document</title>

<script>

function calcNumber(){

let x = prompt("Введіть х", '');

let y = prompt("Введіть у", '');

let z = prompt("Введіть z", '');

let a = 1 - parseFloat(x)/parseFloat(y);

alert(Math.pow(a, 2) + parseFloat(z));

}

</script>

</head>

<body>

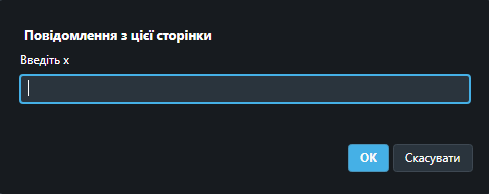
<input type="button" value="кнопка" onclick="calcNumber()"/>

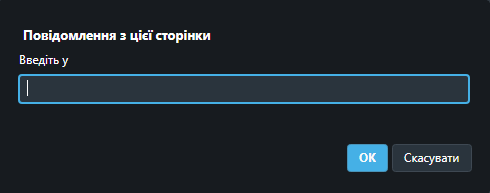
</body>

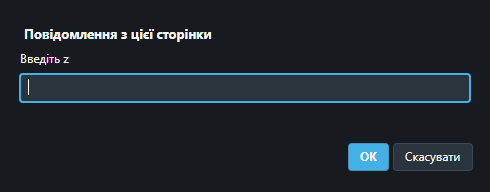
</html>

Результат роботи.

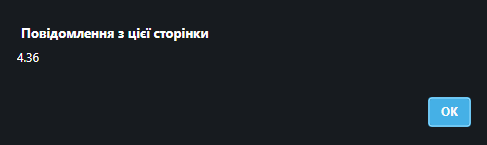
При натисканні кнопки спливають по черзі три вікна з запитом на введення змінних.







В результаті спливає вікно з результатом обрахунку.



**Відповіді на запитання**

1. **Функції, синтаксис, створення та виклик функції.**

Функції є одним з фундаментальних блоків у JavaScript. Функція є процедурою JavaScript, це набір команд, які виконують ту чи іншу задачу, або розраховують значення.

Визначення функції:

function square(number)

{

return number \* number;

}

Виклик функції

let a = square(2) //a = 4

1. **Параметри функції. Область видимості змінних.**

Параметри функції відіграють роль локальних змінних в тілі функції. При створенні параметрів функції ключове слово var або let використовувати не потрібно, JavaScript оголошує їх в якості локальних змінні автоматично. Параметрів може бути зазначено будь-яку кількість.

Змінні, визначені всередині функції, недоступні ззовні цієї функції, бо змінна визначається тільки у області видимості функції. Проте, функція може звертатись до усіх змінних та функцій, визначених у області видимості, де вона оголошена. Іншими словами, функція, оголошена у глобальній області видимості, може звертатись до усіх змінних, оголошених у глобальній області видимості. Функція, оголошена всередині іншої функції, має доступ до усіх змінних, оголошених у батьківській функції, а також до будь-якої змінної, до якої має доступ батьківська функція.

1. **Інструкція return.**

Коли у тілі функції використовується оператор return, виконання функції зупиняється. За наявності, надане значення повертається у інструкцію, що викликала функцію. Наприклад, наступна функція повертає квадрат свого аргументу x, де x є числом.

function square(number)

{

return number \* number;

}

let a = square(2) //a = 4

1. **Логічні операції у функціях.**

Логічні оператори зазвичай застосовуються до [булевих](https://developer.mozilla.org/uk/docs/Web/JavaScript/Reference/Global_Objects/Boolean) (логічних) значень. В цьому випадку вони повертають значення типу Boolean. Однак, оператори && та || насправді повертають значення одного з заданих операндів, тому, якщо ці оператори використовуються зі значеннями не булевого типу, вони повернуть значення не булевого типу.



1. **Рекурсивні функції.**

Їх суть полягає в тому, що функція викликає саму себе.

Наприклад, розглянемо функцію, визначальну факторіал числа:

function getFactorial(n){

if (n === 1){

return 1;

}

else{

return n \* getFactorial(n - 1);

}

}

var result = getFactorial(4);

console.log(result); // 24

Функція getFactorial () повертає значення 1, якщо параметр n дорівнює 1, або повертає результат знову ж функції getFactorial, то в неї передається значення n-1. Наприклад, при передачі числа 4, у нас утворюється наступний ланцюжок викликів:

var result = 4 \* getFactorial(3);

var result = 4 \* 3 \* getFactorial(2);

var result = 4 \* 3 \* 2 \* getFactorial(1);

var result = 4 \* 3 \* 2 \* 1; // 24