Angel Daniel Vásquez Leal

Metodología de resolución de problemas

Metodología resolución problema



**Introducción a los sistemas de computo**

Contenido

[Descripción del problema 1](#_Toc103416605)

[Autor 1](#_Toc103416606)

[Algoritmo 1](#_Toc103416607)

[Diccionario de datos 2](#_Toc103416608)

[Código fuente 2](#_Toc103416609)

[Manual de usuario y paso a paso 5](#_Toc103416610)

# Descripción del problema

Crear un programa que recopile los datos de nombre, apellido, fecha de nacimiento y mes de nacimiento así mismo que este represente y determine si el año es bisiesto y el mes es par o impar.

# Participantes

Angel Daniel Vásquez Leal

# Código fuente

<!DOCTYPE html>

<html lang="es">

<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 href="https://fonts.googleapis.com/css2?family=Montserrat:wght@600&display=swap" rel="stylesheet">

    <title>Registro</title>

    <script src="https://cdn.jsdelivr.net/npm/dexie@3.2.2/dist/dexie.min.js"></script>

</head>

<body>

    <h1>Angel Daniel Vasquez Leal</h1>

    <h2>Usuario GitHub: avasque25</h2>

    <a href="https://github.com/avasque25" target="\_blank">Mi Usuario de GitHub</a>

    <section>

        <div class="encabezado">

            <div class="hijo-encabezado">

                <div class="texto-encabezado">

                    PersonaDB ¡Base de Datos!

                </div>

            </div>

        </div>

        <div id="caja-padre">

            <div class="cont-form">

                <form id="formulario">

                    <input type="text" id="nombre" placeholder="NOMBRE" required>

                    <input type="text" id="apellido" placeholder="APELLIDO" required>

                    <input type="number" id="anioNacimiento" placeholder="AÑO DE NACIMIENTO" required>

                    <input type="number" id="mesNacimiento" placeholder="MES DE NACIMIENTO (1-12)" required>

                </form>

            </div>

            <div class="cont-boton">

                <div class="cont-hijo">

                    <div class="padre-boton">

                        <input type="button" id="btn-guardar" value="Guardar">

                    </div>

                </div>

            </div>

        </div>

        <div id="lista-personas" style="width: 1000px; color: white; margin-top: 20px;"></div>

    </section>

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

</body>

</html>

Java script

let db;

function IniciarBaseDatos() {

    CrearAlmacen();

}

function CrearAlmacen() {

    db = new Dexie("PersonasDB");

    db.version(1).stores({

        personas: "++id, nombre, apellido, anioNacimiento, mesNacimiento, esBisiesto, mesPar"

    });

    Comenzar();

}

function Comenzar() {

    document.getElementById("btn-guardar").addEventListener("click", GuardarPersona);

    MostrarPersonas();

}

function MostrarError(error) {

    alert("Tenemos un ERROR: " + error.name + " / " + error.message);

}

function esBisiesto(anio) {

    return (anio % 4 === 0 && anio % 100 !== 0) || (anio % 400 === 0);

}

function esMesPar(mes) {

    return mes % 2 === 0;

}

async function GuardarPersona() {

    const nombre = document.getElementById("nombre").value.trim();

    const apellido = document.getElementById("apellido").value.trim();

    const anioNacimiento = parseInt(document.getElementById("anioNacimiento").value.trim());

    const mesNacimiento = parseInt(document.getElementById("mesNacimiento").value.trim());

    if (!nombre || !apellido || isNaN(anioNacimiento) || isNaN(mesNacimiento)) {

        alert("Por favor completa todos los campos correctamente.");

        return;

    }

    const bisiesto = esBisiesto(anioNacimiento);

    const parMes = esMesPar(mesNacimiento);

    try {

        await db.personas.add({ nombre, apellido, anioNacimiento, mesNacimiento, esBisiesto: bisiesto, mesPar: parMes });

        // Limpiar campos

        document.getElementById("formulario").reset();

        MostrarPersonas();

    } catch (e) {

        MostrarError(e);

    }

}

async function MostrarPersonas() {

    try {

        const lista = document.getElementById("lista-personas");

        const personas = await db.personas.toArray();

        lista.innerHTML = "<h2>Personas registradas</h2><ul style='list-style: none; padding: 0;'>";

        personas.forEach(p => {

            const edad = new Date().getFullYear() - p.anioNacimiento;

            lista.innerHTML += `

                <li style="margin: 10px 0; background-color: #341968; padding: 10px 20px; border-radius: 15px;">

                    ${p.nombre} ${p.apellido} — Edad: ${edad} años<br>

                    Año ${p.anioNacimiento} es ${p.esBisiesto ? "bisiesto" : "no bisiesto"}<br>

                    Mes ${p.mesNacimiento} es ${p.mesPar ? "par" : "impar"}

                </li>`;

        });

        lista.innerHTML += "</ul>";

    } catch (e) {

        MostrarError(e);

    }

}

window.addEventListener("load", IniciarBaseDatos);

5