<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Scriptural Birth Date Calculator - 364YHWH</title>

<script src="https://cdn.jsdelivr.net/npm/chart.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/jspdf/2.5.1/jspdf.umd.min.js"></script>

<style>

body { font-family: Arial, sans-serif; text-align: center; padding: 20px; background: #f4f4f4; }

#calculator { margin: 20px auto; max-width: 600px; background: white; padding: 20px; border-radius: 10px; box-shadow: 0 0 10px rgba(0,0,0,0.1); }

input, button { padding: 10px; margin: 10px; font-size: 16px; }

button { background: #1e90ff; color: white; border: none; border-radius: 5px; cursor: pointer; }

button:hover { background: #1565c0; }

#result { margin-top: 20px; font-size: 18px; }

canvas { max-width: 600px; margin: 20px auto; background: white; padding: 10px; border-radius: 10px; }

#download { margin-top: 20px; }

</style>

</head>

<body>

<div id="calculator">

<h1>Scriptural Birth Date Calculator</h1>

<label for="birthdate">Enter your birth date (YYYY-MM-DD):</label>

<input type="date" id="birthdate" required>

<button onclick="calculateScripturalDate()">Calculate</button>

<div id="result"></div>

<button id="download" onclick="downloadPDF()">Download as PDF</button>

</div>

<canvas id="calendarChart"></canvas>

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

</body>

</html>

let currentChart = null;

function calculateScripturalDate() {

const birthdate = new Date(document.getElementById('birthdate').value);

if (!birthdate || isNaN(birthdate) || birthdate > new Date()) {

document.getElementById('result').innerHTML = "Please enter a valid past date.";

return;

}

// Creation: March 17, 4060 BCE

const creationDate = new Date(-4060, 2, 17);

const daysSinceCreation = Math.floor((birthdate - creationDate) / (1000 \* 60 \* 60 \* 24));

// YHWH Calendar (364 days)

const yhwhYear = Math.floor(daysSinceCreation / 364) + 1;

let daysInYhwhYear = daysSinceCreation % 364;

const isLastSabbath = daysInYhwhYear === 363;

let zeroDaysNote = isLastSabbath ? ' (Last Sabbath, prolonged by 1–2 zero days)' : '';

if (isLastSabbath) daysInYhwhYear = 364;

const week = Math.floor(daysInYhwhYear / 7) + 1;

const dayOfWeek = (daysInYhwhYear % 7) + 1;

// Custom Months from PDF

const customMonths = ["abiyah", "davison", "chadasha", "leonard", "sharl", "gavanah", "ansel", "trish", "angie", "elmarie", "desmond", "nehemyah"];

const monthDays = [30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 34];

let daysRemaining = daysInYhwhYear;

let yhwhMonthIndex = 0, yhwhDay = 0;

for (let i = 0; i < monthDays.length; i++) {

if (daysRemaining <= monthDays[i]) {

yhwhMonthIndex = i;

yhwhDay = daysRemaining + 1;

break;

}

daysRemaining -= monthDays[i];

}

const yhwhMonth = customMonths[yhwhMonthIndex];

// Solar Calendar

const solarYear = Math.floor(daysSinceCreation / 365.25) + 1;

const isLeapYear = (solarYear % 4 === 0);

let daysInSolarYear = daysSinceCreation % (isLeapYear ? 366 : 365);

const monthLengths = isLeapYear ? [31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31] : [31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31];

let solarMonth = 1, solarDay = daysInSolarYear + 1;

for (let i = 0; i < monthLengths.length; i++) {

if (solarDay <= monthLengths[i]) break;

solarDay -= monthLengths[i];

solarMonth++;

}

// Lunar Calendar

const lunarYear = Math.floor(daysSinceCreation / 354) + 1;

let daysInLunarYear = daysSinceCreation % 354;

let lunarMonth = Math.floor((daysInLunarYear - 1) / 29.5) + 1;

let lunarDay = (daysInLunarYear - 1) % 29.5 + 1;

let lunarPhase = lunarDay <= 2 ? 'Full Moon' : lunarDay <= 8 ? 'Waning Gibbous' :

lunarDay <= 14 ? 'Last Quarter' : lunarDay <= 16 ? 'New Moon' :

lunarDay <= 22 ? 'Waxing Crescent' : lunarDay <= 28 ? 'First Quarter' : 'Waxing Gibbous';

// Sabbath and Jubilee

const sabbathCycle = Math.floor(solarYear / 7) + 1;

const yearInSabbath = solarYear % 7 || 7;

const jubileeCycle = Math.floor(solarYear / 49) + 1;

const yearInJubilee = solarYear % 49 || 49;

// PDF Data Lookup

const calendarData = [

{ gregorian: '2021-05-06', portal: 5, nightLength: 7.4, yhwhMonth: 'davison', yhwhDay: 18 },

{ gregorian: '2003-01-02', portal: 1, nightLength: 11.45, yhwhMonth: 'nehemyah', yhwhDay: 13 },

{ gregorian: '2008-03-27', portal: 4, nightLength: 6.75, yhwhMonth: 'abiyah', yhwhDay: 6 }

];

const match = calendarData.find(entry => new Date(entry.gregorian).toDateString() === birthdate.toDateString()) ||

{ portal: 4, nightLength: 6.81, yhwhMonth, yhwhDay };

// Output

document.getElementById('result').innerHTML = `

<h2>Scriptural Birth Date</h2>

<p><b>YHWH's Calendar</b>: Year ${yhwhYear}, Month ${match.yhwhMonth}, Day ${match.yhwhDay}, Week ${week}, Day ${dayOfWeek}${zeroDaysNote}</p>

<p><b>Solar/Stellar Calendar</b>: Year ${solarYear}, Month ${solarMonth}, Day ${solarDay}</p>

<p><b>Lunar Calendar</b>: Year ${lunarYear}, Month ${lunarMonth}, Day ${lunarDay} (${lunarPhase})</p>

<p><b>Sabbath Cycle</b>: Cycle ${sabbathCycle}, Year ${yearInSabbath}${yearInSabbath === 7 ? ' (Sabbath Year)' : ''}</p>

<p><b>Jubilee Cycle</b>: Cycle ${jubileeCycle}, Year ${yearInJubilee}${yearInJubilee === 49 ? ' (Jubilee Year)' : ''}</p>

<p><b>Portal</b>: ${match.portal}</p>

<p><b>Night Length</b>: ${match.nightLength} hours</p>

`;

// Update Chart

updateChart(solarYear, lunarYear, yhwhYear, sabbathCycle, yearInSabbath, jubileeCycle, yearInJubilee);

}

function updateChart(solarYear, lunarYear, yhwhYear, sabbathCycle, yearInSabbath, jubileeCycle, yearInJubilee) {

const ctx = document.getElementById('calendarChart').getContext('2d');

if (currentChart) currentChart.destroy();

currentChart = new Chart(ctx, {

type: 'line',

data: {

labels: ['Creation', 'Your Birth'],

datasets: [

{ label: 'YHWH’s Years (364 days)', data: [0, yhwhYear], borderColor: '#1e90ff', backgroundColor: '#1e90ff', fill: false },

{ label: 'Solar/Stellar Years (365/366 days)', data: [0, solarYear], borderColor: '#ff4500', backgroundColor: '#ff4500', fill: false },

{ label: 'Lunar Years (354 days)', data: [0, lunarYear], borderColor: '#32cd32', backgroundColor: '#32cd32', fill: false }

]

},

options: {

responsive: true,

scales: { y: { title: { display: true, text: 'Years Since Creation' } } },

plugins: { title: { display: true, text: `Sabbath Cycle ${sabbathCycle} (Year ${yearInSabbath}), Jubilee Cycle ${jubileeCycle} (Year ${yearInJubilee})` } }

}

});

}

function downloadPDF() {

const { jsPDF } = window.jspdf;

const doc = new jsPDF();

const resultText = document.getElementById('result').innerText;

doc.setFontSize(12);

doc.text(resultText, 10, 10);

doc.save('scriptural\_birth\_date.pdf');

}