

```

// Navigation Script
const hamburger = document.getElementById('hamburger');
const navMenu = document.getElementById('navMenu');

hamburger.addEventListener('click', function() {
  navMenu.classList.toggle('active');
});

function showApp(appName) {
  // Hide all app sections
  const allSections = document.querySelectorAll('.app-section');
  allSections.forEach(section => section.classList.remove('active'));

  // Show selected app
  document.getElementById(appName + '-app').classList.add('active');

  // Close mobile menu
  navMenu.classList.remove('active');
}

// Temp-convereter functions
function updateFormula() {

  const conversionType = document.getElementById("conversion-type").value;
  const formulaElement = document.getElementById("formula");

  if (conversionType === "ftoc") {
    formulaElement.textContent = " $^{\circ}\text{C} = (^{\circ}\text{F} - 32) \times 5/9$ ";
  } else {
    formulaElement.textContent = " $^{\circ}\text{F} = (^{\circ}\text{C} \times 9/5) + 32$ ";
  }
}

function assessTemperature(temp, scale) {
  const tempElement = document.getElementById("temp-assessment");

```

```
let assessment = "";
let color = "";

if (scale === "celsius") {
  if (temp <= 0) {
    assessment = "Very Cold";
    color = "#3498db"; // Blue
  } else if (temp < 10) {
    assessment = "Cold";
    color = "#7fb3d5"; // Light blue
  } else if (temp < 20) {
    assessment = "Cool";
    color = "#a9cce3"; // Very light blue
  } else if (temp < 30) {
    assessment = "Moderate";
    color = "#2ecc71"; // Green
  } else if (temp < 40) {
    assessment = "Warm";
    color = "#f39c12"; // Orange
  } else {
    assessment = "Hot";
    color = "#e74c3c"; // Red
  }
} else if (scale === "fahrenheit") {
  if (temp <= 32) {
    assessment = "Very Cold";
    color = "#3498db"; // Blue
  } else if (temp < 50) {
    assessment = "Cold";
    color = "#7fb3d5"; // Light blue
  } else if (temp < 68) {
    assessment = "Cool";
    color = "#a9cce3"; // Very light blue
  } else if (temp < 86) {
    assessment = "Moderate";
    color = "#2ecc71"; // Green
  }
}
```

```

    } else if (temp < 104) {
        assessment = "Warm";
        color = "#f39c12"; // Orange
    } else {
        assessment = "Hot";
        color = "#e74c3c"; // Red
    }
}
tempElement.textContent = `Temperature Assessment: ${assessment}`;
tempElement.style.color = color;
tempElement.style.fontWeight = "bold";
}

function convertTemperature() {
    const tempInput = document.getElementById("temperature");
    const tempValue = tempInput.value;
    const temperatureValue = parseFloat(tempValue);

    const conversionType = document.getElementById("conversion-type").value;
    const resultElement = document.getElementById("conversion-result");
    if (isNaN(temperatureValue)) {
        resultElement.textContent = "Invalid input. Please enter a number.";
        document.getElementById("temp-assessment").textContent = "";
        return;
    }
    let result;
    if (conversionType === "ftoc") {
        result = (temperatureValue - 32) * 5 / 9;
        resultElement.textContent = `${temperatureValue}°F = ${result.toFixed(2)}°C`;
        assessTemperature(result, "celsius");
    } else if (conversionType === "ctof") {
        result = (temperatureValue * 9 / 5) + 32;
        resultElement.textContent = `${temperatureValue}°C = ${result.toFixed(2)}°F`;
        assessTemperature(result, "fahrenheit");
    }
}

```

```
function clearConverter() {  
    document.getElementById("temperature").value = "";  
    document.getElementById("conversion-result").textContent = "";  
    document.getElementById("temp-assessment").textContent = "";  
}
```

```
// Magic 8 Ball Functions
```

```
const answers = [  
    "It is certain",  
    "It is decidedly so",  
    "Without a doubt",  
    "Yes definitely",  
    "You may rely on it",  
    "As I see it, yes",  
    "Most likely",  
    "Outlook good",  
    "Yes",  
    "Signs point to yes",  
    "Reply hazy, try again",  
    "Ask again later",  
    "Better not tell you now",  
    "Cannot predict now",  
    "Concentrate and ask again",  
    "Don't count on it",  
    "My reply is no",  
    "My sources say no",  
    "Outlook not so good",  
    "Very doubtful"  
];
```

```
let historyItems = [];
```

```
function getRandomAnswer() {  
    const randomIndex = Math.floor(Math.random() * answers.length);  
    return answers[randomIndex];  
}
```

```
}
```

```
function shakeBall() {  
  const question = document.getElementById('question').value.trim();  
  const ball = document.getElementById('ball');  
  const answerElement = document.getElementById('answer');  
  const questionDisplay = document.getElementById('question-display');  
  const questionInput = document.getElementById('question');  
  
  if (question === "") {  
    alert('Please ask a question first!');  
    return;  
  }  
  
  answerElement.textContent = '8';  
  
  ball.style.transform = 'translateX(-5px)';  
  setTimeout(() => { ball.style.transform = 'translateX(5px)'; }, 100);  
  setTimeout(() => { ball.style.transform = 'translateX(-5px)'; }, 200);  
  setTimeout(() => { ball.style.transform = 'translateX(5px)'; }, 300);  
  setTimeout(() => { ball.style.transform = 'translateX(0)'; }, 400);  
  
  setTimeout(() => {  
    const randomAnswer = getRandomAnswer();  
    answerElement.textContent = randomAnswer;  
    questionDisplay.textContent = `${question}`;  
    questionDisplay.style.opacity = 1;  
    addToHistory(question);  
  }, 500);  
  
  questionInput.value = "";  
}
```

```
function resetBall() {  
  document.getElementById('answer').textContent = '8';  
  document.getElementById('question-display').textContent = "";
```

```
document.getElementById('question-display').style.opacity = 0;
document.getElementById('question').value = "";
}
```

```
function addToHistory(question) {
  historyItems.unshift(question);
  updateHistoryDisplay();
}
```

```
function updateHistoryDisplay() {
  const questionHistory = document.getElementById('question-history');
  questionHistory.innerHTML = "";

  historyItems.forEach((question, index) => {
    const listItem = document.createElement('li');
    listItem.className = 'history-item';
    listItem.textContent = question;
    questionHistory.appendChild(listItem);
  });
}
```

```
function clearHistory() {
  historyItems = [];
  updateHistoryDisplay();
}
```

// Task List Functions

// 2D Array to store tasks - each task is [taskText, priority]

let tasks = [];

// New 2D array to store date and time - each item is [dateString, timeString]

let taskDueDates = [];

// Array of random tasks for the random task feature - focused on health and student wellness

```
const randomTasks = [
  "Take a short walk",
```

"Drink a glass of water",  
"Stretch for 5 minutes",  
"Practice deep breathing for 2 minutes",  
"Stand up and move around for 5 minutes",  
"Do a quick meditation session",  
"Write in a gratitude journal",  
"Have a healthy snack",