



SKILLSPROUT
COMMIT TO CODE WORKSHOP
BATCH - 2
SOURCE CODES

PROJECT 1: SkillSprout LifeOS PRO

INDEX.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>SkillSprout LifeOS PRO</title>
<link rel="stylesheet" href="style.css">
</head>
<body>
```

```
<header>

<h1>  SkillSprout LifeOS PRO</h1>

<p>Your Personal Digital Twin Dashboard</p>

<button onclick="toggleTheme()">  Toggle Theme</button>

<button onclick="downloadPDF()">  Export as PDF</button>

</header>

<div class="container">

<div class="box">

<h2>  Life Goals</h2>

<input id="goalInput" placeholder="Enter a goal">

<button onclick="addItem('goal')">Add</button>

<ul id="goalList"></ul>

</div>

<div class="box">

<h2>  Skills Tracker</h2>

<input id="skillInput" placeholder="Enter a skill">

<button onclick="addItem('skill')">Add</button>

<ul id="skillList"></ul>

</div>

<div class="box">

<h2>  Achievements</h2>

<input id="achieveInput" placeholder="Enter an achievement">

<button onclick="addItem('achieve')">Add</button>

<ul id="achieveList"></ul>

</div>

</div>

<div class="progress-box">

<h2>  Growth Progress</h2>

<div class="progress-bar">
```

```
<div id="progress"></div>  
</div>  
  
<p id="badge">🌟 Starter</p>  
</div>  
  
<script src="script.js"></script>  
</body>  
</html>
```

STYLE.CSS

```
body {  
    margin: 0;  
    font-family: Arial, sans-serif;  
    background: linear-gradient(to right, #1a2a6c, #b21f1f, #fdbb2d);  
    color: white;  
    text-align: center;  
    transition: 0.3s;  
}  
  
.dark{  
    background: linear-gradient(to right, #0f2027, #203a43, #2c5364);  
    color: white;  
}  
  
header {  
    padding: 30px;  
    background: rgba(0,0,0,0.3);  
}  
  
button {
```

```
margin: 5px;  
background: #1a2a6c;  
color: white;  
border: none;  
padding: 8px;  
border-radius: 5px;  
cursor: pointer;  
}
```

```
.container {  
display: flex;  
flex-wrap: wrap;  
justify-content: center;  
margin: 20px;  
}
```

```
.box {  
background: white;  
color: black;  
width: 260px;  
margin: 10px;  
padding: 15px;  
border-radius: 10px;  
}  
.dark .box,  
.dark .progress-box {  
background: #1c1c1c;  
color: white;  
}  
input {
```

```
width: 80%;  
padding: 8px;  
margin-bottom: 5px;  
}  
  
.progress-box{  
background: white;  
color: black;  
width: 80%;  
margin: 20px auto;  
padding: 15px;  
border-radius: 10px;  
}  
  
.progress-bar{  
background: #ddd;  
width: 100%;  
height: 20px;  
border-radius: 10px;  
overflow: hidden;  
}  
  
#progress{  
background: linear-gradient(to right, #00f260, #0575e6);  
height: 100%;  
width: 0%;  
transition: 0.3s;  
}
```

SCRIPT.JS

```
let count = localStorage.getItem("progress") || 0;

let theme = localStorage.getItem("theme") || "light";

document.getElementById("progress").style.width = count + "%";

setBadge();

loadData();

if (theme === "dark") {

    document.body.classList.add("dark");

}

// Add Item Function

function addItem(type) {

    let input, list;

    if (type === "goal") {

        input = goalInput;

        list = goalList;

    } else if (type === "skill") {

        input = skillInput;

        list = skillList;

    } else {

        input = achieveInput;

        list = achieveList;

    }

    if (!input.value) return;

    const li = document.createElement("li");

    li.innerText = input.value;

    list.appendChild(li);

    saveData(type, input.value);
```

```
input.value = "";
updateProgress();
}

// Progress & Badge

function updateProgress() {
    count = parseInt(count) + 5;
    if (count > 100) count = 100;
    document.getElementById("progress").style.width = count + "%";
    localStorage.setItem("progress", count);
    setBadge();
}

function setBadge() {
    const badge = document.getElementById("badge");
    if (count < 25) badge.innerText = "🌟 Starter";
    else if (count < 50) badge.innerText = "🚀 Builder";
    else if (count < 75) badge.innerText = "🔥 Achiever";
    else badge.innerText = "👑 Life Architect";
}

// Theme Toggle

function toggleTheme() {
    document.body.classList.toggle("dark");
    theme = document.body.classList.contains("dark") ? "dark" : "light";
    localStorage.setItem("theme", theme);
}

// Save & Load

function saveData(type, value) {
    let data = JSON.parse(localStorage.getItem(type) || "[]");
    data.push(value);
}
```

```
localStorage.setItem(type, JSON.stringify(data));

}

function loadData() {
  ["goal", "skill", "achieve"].forEach(type => {
    let data = JSON.parse(localStorage.getItem(type) || "[]");
    const list = document.getElementById(type + "List");
    data.forEach(item => {
      const li = document.createElement("li");
      li.innerText = item;
      list.appendChild(li);
    });
  });
}

// Export as PDF

function downloadPDF() {
  window.print();
}
```

PROJECT: 2 - SkillSprout MoodScape

INDEX.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>SkillSprout MoodScape</title>
```

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

</head>

<body>

<header>

<h1>  SkillSprout MoodScape</h1>

<p>Your Emotion-Based Experience Generator</p>

</header>

<div class="card">

<h2>How are you feeling today?</h2>

<select id="mood">

<option value="">Select Mood</option>

<option value="happy"> 😊 Happy</option>

<option value="sad"> 😞 Sad</option>

<option value="focus"> 🚀 Focused</option>

<option value="stress"> 😰 Stressed</option>

<option value="excited"> 😃 Excited</option>

</select>

<button onclick="generate()">Generate My Experience</button>

<div id="result">

<h3 id="message"></h3>

<p id="music"></p>

<p id="challenge"></p>

</div>

</div>
```

```
<script src="script.js"></script>  
</body>  
</html>
```

STYLE.CSS

```
body {  
    margin: 0;  
    font-family: Arial, sans-serif;  
    background: linear-gradient(to right, #141e30, #243b55);  
    color: white;  
    text-align: center;  
    transition: 0.5s;  
}  
  
header {  
    padding: 30px;  
    background: rgba(0,0,0,0.3);  
}
```

```
.card {  
    background: white;  
    color: black;  
    width: 300px;  
    margin: 40px auto;  
    padding: 20px;  
    border-radius: 15px;  
}
```

```
select, button {  
    width: 90%;  
    padding: 10px;  
    margin: 10px 0;  
}  
  
button {  
    background: #243b55;  
    color: white;  
    border: none;  
    cursor: pointer;  
    border-radius: 5px;  
}
```

SCRIPT.JS

```
function generate() {  
    const mood = document.getElementById("mood").value;  
    const message = document.getElementById("message");  
    const music = document.getElementById("music");  
    const challenge = document.getElementById("challenge");  
  
    if (!mood) {  
        alert("Select a mood!");  
        return;  
    }  
  
    let bg, msg, song, task;
```

```
if (mood === "happy") {  
  bg = "linear-gradient(to right, #f7971e, #ffd200)";  
  msg = "Keep smiling — you're spreading positive energy! 😊";  
  song = "🎵 Music: Upbeat pop or dance playlist";  
  task = "🎯 Challenge: Compliment 3 people today";  
}  
  
if (mood === "sad") {  
  bg = "linear-gradient(to right, #2c3e50, #4ca1af)";  
  msg = "It's okay to slow down. You're stronger than you think 💙";  
  song = "🎵 Music: Calm instrumental or lo-fi";  
  task = "🎯 Challenge: Write 3 things you're grateful for";  
}  
  
if (mood === "focus") {  
  bg = "linear-gradient(to right, #00c6ff, #0072ff)";  
  msg = "Deep work mode activated. Let's build greatness 🔥";  
  song = "🎵 Music: Coding or concentration beats";  
  task = "🎯 Challenge: Work for 25 minutes without distractions";  
}  
  
if (mood === "stress") {  
  bg = "linear-gradient(to right, #c31432, #240b36)";  
  msg = "Pause. Breathe. Reset. You've got this 🌱";  
  song = "🎵 Music: Nature or meditation sounds";  
  task = "🎯 Challenge: Take 5 deep breaths";  
}  
  
if (mood === "excited") {
```

```
bg = "linear-gradient(to right, #fc4a1a, #f7b733);  
msg = "Your energy is contagious — go create something amazing 🚀";  
song = "🎵 Music: High-energy motivation playlist";  
task = "🎯 Challenge: Start a new mini project today";  
}  
  
document.body.style.background = bg;  
message.innerText = msg;  
music.innerText = song;  
challenge.innerText = task;
```

```
}
```