Pomodoro timer

Html code

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Pomodoro Timer</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <div class="container">
        <h1>🍅 Pomodoro Timer</h1>
        <p class="subtitle">Stay focused and productive</p>
        <div class="status" id="status">Ready to focus</div>
        <div class="timer" id="timer">25:00</div>
        <div class="progress"><div class="fill" id="fill"></div></div>

        <div class="session-info">
            <div><span>Sessions</span><b id="sessions">0</b></div>
            <div><span>Total Time</span><b id="totalTime">0m</b></div>
        </div>

        <label>Work Duration (minutes)</label>
        <input type="text" id="duration" value="25" placeholder="Enter minutes">

        <div class="presets">
            <button onclick="setTime(15)">15 min</button>
            <button onclick="setTime(25)">25 min</button>
            <button onclick="setTime(45)">45 min</button>
        </div>

        <div class="controls">
            <button id="start" onclick="start()">▶️ Start</button>
            <button id="pause" onclick="pause()">⏸️ Pause</button>
            <button onclick="reset()">⟲ Reset</button>
        </div>


    </div>
    <script src="java.js"></script>
</body>
```

```
</html>
```

Css code

```
* { margin: 0; padding: 0; box-sizing: border-box; }

body {
   font-family: 'Segoe UI', sans-serif;
   background: linear-gradient(135deg, #667eea, #764ba2);
   min-height: 100vh;
   display: flex;
   justify-content: center;
   align-items: center;
   padding: 20px;
}

.container {
   background: white;
   border-radius: 25px;
   padding: 40px;
   box-shadow: 0 25px 70px rgba(0,0,0,0.4);
   max-width: 500px;
   width: 100%;
   text-align: center;
}

h1 { color: #333; margin-bottom: 10px; font-size: 2.5em; }
.subtitle { color: #888; font-size: 1.1em; margin-bottom: 25px; }
.status { color: #666; margin-bottom: 20px; font-size: 1.3em; font-weight: 500; }

.timer {
   font-size: 5em;
   font-weight: bold;
   color: #667eea;
   margin: 25px 0;
   font-family: monospace;
}

.progress {
   width: 100%;
   height: 12px;
   background: #e0e0e0;
   border-radius: 6px;
   margin: 20px 0;
   overflow: hidden;
```

```css
}

.fill {
   height: 100%;
   background: linear-gradient(90deg, #667eea, #764ba2);
   width: 100%;
   transition: width 1s linear;
}

.session-info {
   display: flex;
   gap: 15px;
   margin: 20px 0;
}

.session-info div {
   flex: 1;
   background: linear-gradient(135deg, #f5f7fa, #c3cfe2);
   padding: 15px;
   border-radius: 10px;
   display: flex;
   flex-direction: column;
   gap: 5px;
}

.session-info span { font-size: 0.9em; color: #666; }
.session-info b { font-size: 1.8em; color: #667eea; }

label {
   display: block;
   margin: 15px 0 8px;
   color: #555;
   font-weight: 600;
}

input[type="text"] {
   width: 100%;
   padding: 12px;
   border: 2px solid #e0e0e0;
   border-radius: 10px;
   font-size: 1em;
   transition: all 0.3s;
}

input:focus {
   outline: none;
   border-color: #667eea;
   box-shadow: 0 0 10px rgba(102, 126, 234, 0.3);
}

.presets, .controls { display: flex; gap: 12px; margin: 20px 0; }

.presets button {
   flex: 1;
   padding: 10px;
   background: linear-gradient(135deg, #f5f7fa, #c3cfe2);
   border: 2px solid #e0e0e0;
   border-radius: 10px;
   cursor: pointer;
   font-weight: 600;
   transition: all 0.3s;
}

.presets button:hover {
   background: linear-gradient(135deg, #667eea, #764ba2);
   color: white;
   transform: translateY(-2px);
}

button {
   flex: 1;
   padding: 16px;
   border: none;
   border-radius: 10px;
   font-size: 1.1em;
   font-weight: 700;
   cursor: pointer;
   transition: all 0.3s;
}

button:hover { transform: translateY(-2px); }

#start { background: linear-gradient(135deg, #667eea, #764ba2); color: white; }
#pause { background: linear-gradient(135deg, #f39c12, #e67e22); color: white;
display: none; }
#pause + button { background: linear-gradient(135deg, #e74c3c, #c0392b); color:
white; }

.settings {
   background: #f8f9fa;
```

```
    padding: 15px;
    border-radius: 10px;
    margin: 20px 0;
    display: flex;
    gap: 20px;
    justify-content: center;
}

.settings label {
    display: flex;
    align-items: center;
    gap: 8px;
    margin: 0;
    cursor: pointer;
    font-size: 0.95em;
    font-weight: normal;
}

.settings input { width: auto; cursor: pointer; }
```

Javascript code

```
let interval, remaining, total, running = false, sessions = 0, totalMinutes = 0;

//Page Load Setup
window.onload = function() {
    if (Notification.permission === 'default') Notification.requestPermission();
    updateDisplay(1500);
    let saved = localStorage.getItem('stats');
    if (saved) {
        saved = JSON.parse(saved);
        sessions = saved.sessions;
        totalMinutes = saved.totalMinutes;
        document.getElementById('sessions').textContent = sessions;
        document.getElementById('totalTime').textContent = totalMinutes + 'm';
    }
};

//Time Set
function setTime(mins) {
    if (!running) {
        document.getElementById('duration').value = mins;
        updateDisplay(mins * 60);
    }
}
```

```
//Timer Start
function start() {
   if (running) return;
   let mins = parseInt(document.getElementById('duration').value);
   if (!mins || mins < 1) return alert('Please enter valid minutes');

   if (!interval) { total = mins * 60; remaining = total; }
   running = true;
   document.getElementById('status').textContent = '⏳ Working...';
   document.getElementById('start').style.display = 'none';
   document.getElementById('pause').style.display = 'block';
   document.getElementById('duration').disabled = true;

   interval = setInterval(function() {
       remaining--;
       updateDisplay(remaining);
       document.getElementById('fill').style.width = (remaining / total * 100) +
'%';
       if (remaining <= 0) complete();
   }, 1000);
}

//Timer Pause
function pause() {
   clearInterval(interval);
   interval = null;
   running = false;
   document.getElementById('status').textContent = '⏸️ Paused';
   document.getElementById('start').style.display = 'block';
   document.getElementById('pause').style.display = 'none';
}

//Timer Reset
function reset() {
   clearInterval(interval);
   interval = null;
   running = false;
   let input = document.getElementById('duration').value;
   let mins = parseInt(input);

   // Check if already at the input value, then reset to 0
   if (remaining === mins * 60 && total === mins * 60) {
       remaining = 0;
       total = 0;
```

```
        document.getElementById('duration').value = '';
    } else if (!input || !mins || mins < 1) {
        // If no valid input, reset to 0
        remaining = 0;
        total = 0;
        document.getElementById('duration').value = '';
    } else {
        // Reset to the input value
        remaining = mins * 60;
        total = remaining;
    }

    updateDisplay(remaining);
    document.getElementById('fill').style.width = (total > 0 ? (remaining / total * 100) : 0) + '%';
    document.getElementById('status').textContent = 'Ready to focus';
    document.getElementById('start').style.display = 'block';
    document.getElementById('pause').style.display = 'none';
    document.getElementById('duration').disabled = false;
}


//Timer Complete
function complete() {
    clearInterval(interval);
    interval = null;
    running = false;

    sessions++;
    totalMinutes += Math.floor(total / 60);
    document.getElementById('sessions').textContent = sessions;
    document.getElementById('totalTime').textContent = totalMinutes + 'm';
    localStorage.setItem('stats', JSON.stringify({sessions: sessions, totalMinutes: totalMinutes}));

    document.getElementById('status').textContent = '✅ Time for a break!';
    document.getElementById('start').style.display = 'block';
    document.getElementById('pause').style.display = 'none';
    document.getElementById('duration').disabled = false;

    if (Notification.permission === 'granted') {
        new Notification('Pomodoro Timer', { body: 'Great work! Time for a break! 🎉' });
    }
    if (document.getElementById('autoBreak').checked) {
        setTimeout(function() {
            if (confirm('Start 5 minute break?')) {
                document.getElementById('duration').value = 5;
                updateDisplay(300);
                start();
            }
        }, 2000);
    }
}


//Display Update
function updateDisplay(secs) {
    let m = Math.floor(secs / 60);
    let s = secs % 60;
    document.getElementById('timer').textContent = (m < 10 ? '0' + m : m) + ':' + (s < 10 ? '0' + s : s);
    if (running) document.title = (m < 10 ? '0' + m : m) + ':' + (s < 10 ? '0' + s : s) + ' - Pomodoro';
}
```