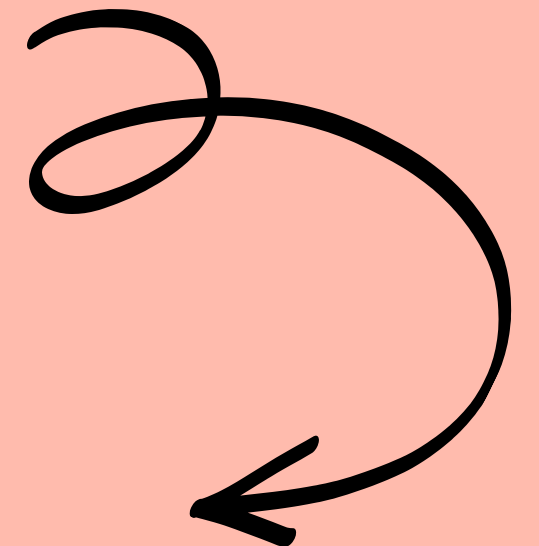




# HEAL HUB

MAKE YOUR LIFE EASIER





# INTRODUCTION

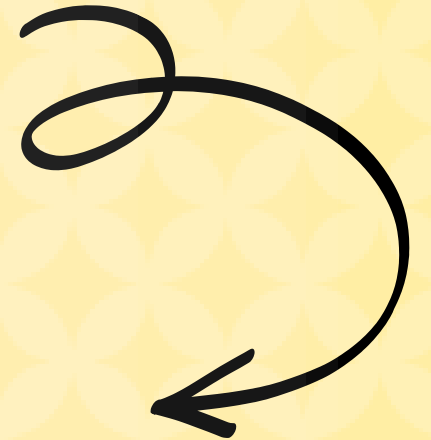
**"In the daily hustle of student life, we encounter challenges that often go unnoticed. Let's take a moment to understand the real struggles faced by students."**

- "Balancing academic demands"
- "Navigating financial stress"
- "Prioritizing mental and physical health"

# "A Student's Wish: Heal Hub"

**"Now, let's bring this wish to life with Heal Hub – the ultimate solution designed to meet every student's needs. Join us on a journey to simplify, organize, and thrive!"**

- "Reliable To-Do Reminder: Keeping track of assignments and deadlines seamlessly."
- "Financial Tracking: Budgeting made easy for a stress-free financial life."
- "Sleep and Health Monitoring: Prioritizing well-being amidst a hectic schedule."



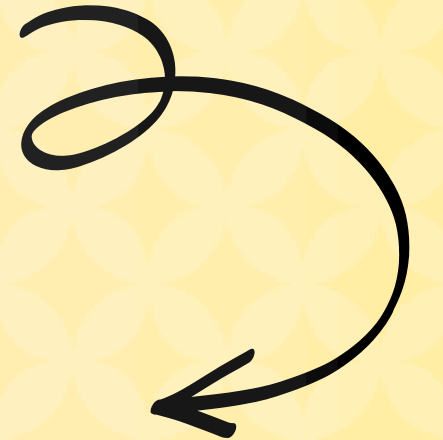


# FEATURES

1. Comprehensive wellness
2. time management
3. Financial Empowerment
4. Health and well being
5. user friendly Interface

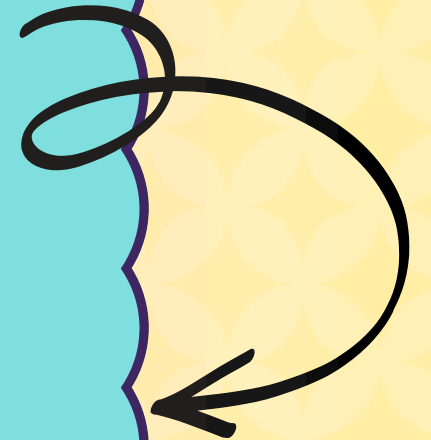
# "What's inside Heal Hub"

- 1.To do list
- 2.Expense tracker
- 3.Sleep tracker
- 4.Period cycle tracker



## TO DO LIST

The to-do tracker is a simple tool for listing tasks and tracking progress. It includes sections for tasks to complete and a list of completed tasks



# ToDoList

☐

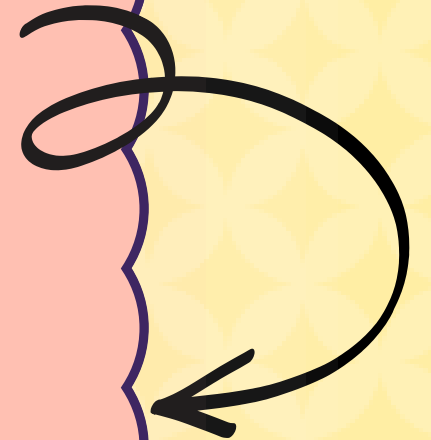
presentation

Add new task

Add

## Expense Tracker

An expense tracker is a comprehensive solution for organizing and monitoring financial transactions. users can systematically categorize their expenses, such as housing, transportation, groceries, dining out, entertainment, health & wellness, and miscellaneous items.





# Expense Tracker

Category:

Bills

Description:

Amount:

Add Expense

## Bills Expenses

Description	Amount
vbn	3.00
bjsnhkj	8789.00

bjsnhkj	8789.00
asdfvg	781.00
asdfvg	781.00
vbn vv	34.00

## Food&travel Expenses

Description	Amount
res	6000.00

## Health Expenses

Description	Amount
surgery	10000.00

## Entertainment Expenses

Description	Amount
movie	300.00

## Entertainment Expenses

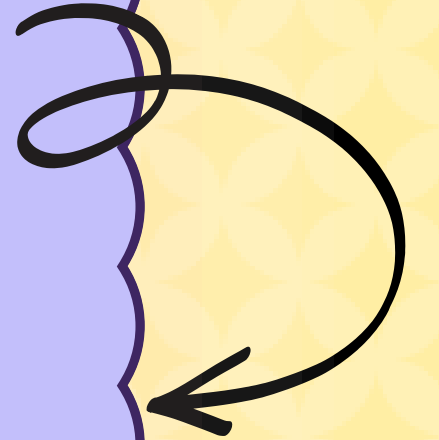
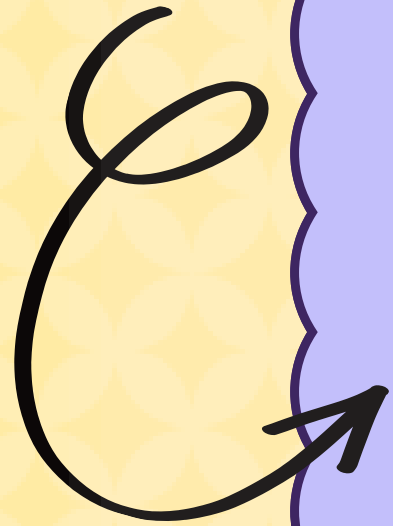
Description	Amount
movie	300.00

## Others Expenses

Description	Amount
asdfvg	42255.00

# Sleep tracker

The Sleep Tracker feature allows you to monitor and analyze your sleep patterns, providing insights into the duration and quality of your sleep each night.



# Sleep Tracker

Start Time:

dd-mm-yyyy -- : --

Wake-up Time:

dd-mm-yyyy -- : --

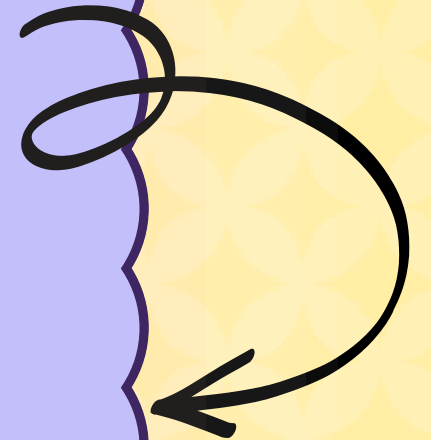
Add Sleep Entry

## Last 10 Days Sleep Quality

Start Time	Wake-up Time	Duration (hours)	Sleep Quality
2024-02-22 23:07	2024-02-22 00:08	-22.98	Very Bad
2024-02-08 01:08	2024-02-09 05:12	28.07	Very Good

# Period tracker

- Allowing you to plan and prepare accordingly. By utilizing the Period Tracker, you can effectively monitor your menstrual cycle, receive accurate predictions for upcoming periods, and stay informed about your reproductive health.



# Period Tracker

## Period Tracker

Last Period Date:

16 - 03 - 2024



Period Cycle Time (in days):

28

Submit

April  
2024

Mon	Tue	Wed	Thu	Fri	Sat	Sun
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					



# Calorie Tracker

By utilizing the Calorie Tracker, you can easily monitor your daily calorie intake and make informed decisions about your dietary choices. Whether you're looking to maintain, gain, or lose weight, this tool empowers you to track your food consumption and stay on track with your health and fitness goals.

# Calorie Tracker

Food:

Hyderabadi Biryani (400 calories) ▼

Meal:

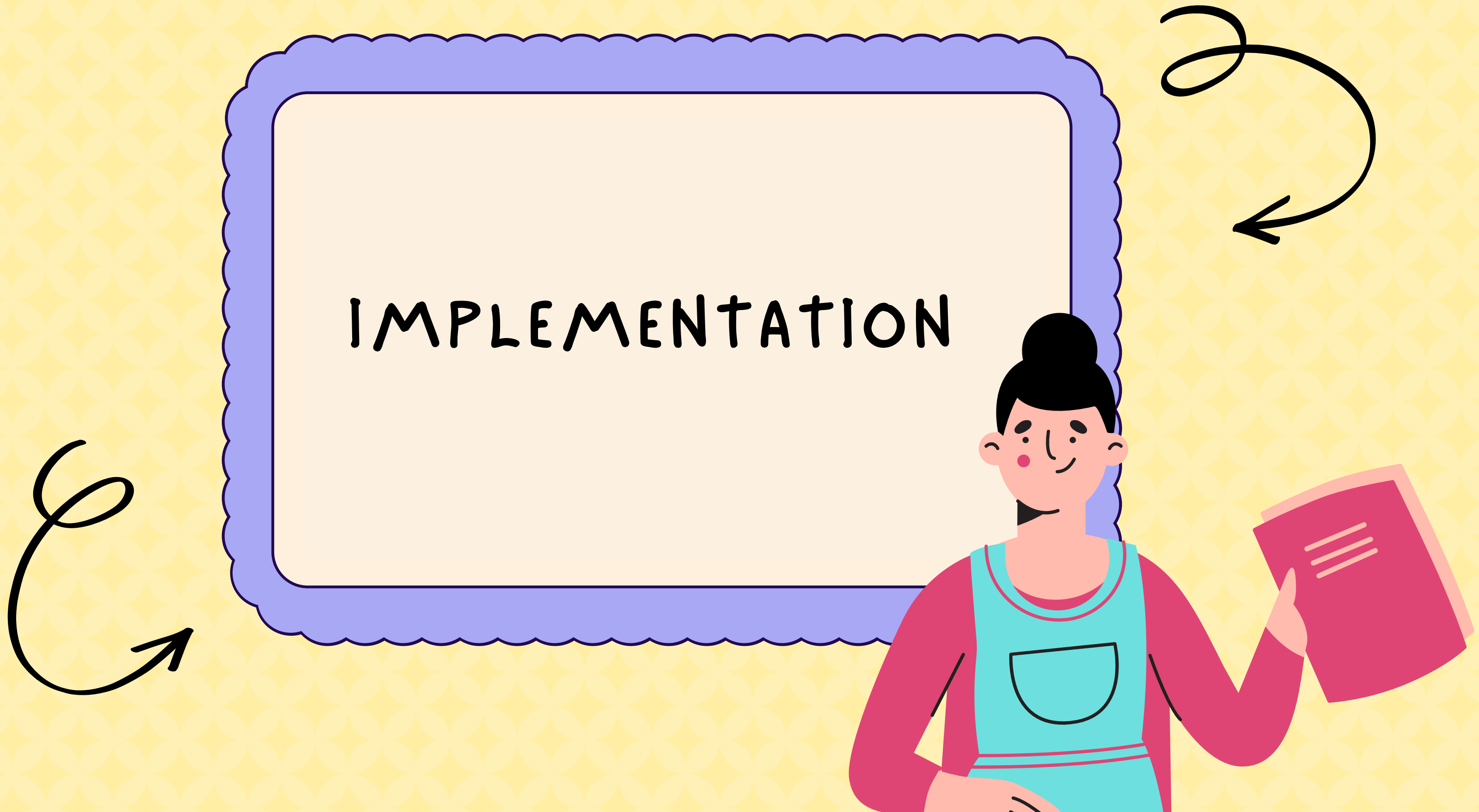
Breakfast ▼

Add Calories Consumed

## Calories Consumed Today



IMPLEMENTATION







**front end**

```

1  <?php
2  // Database connection
3  $servername = "localhost";
4  $username = "root";
5  $password = "";
6  $dbname = "heal app";
7
8  $conn = new mysqli($servername, $username, $password, $dbname);
9
10 if ($conn->connect_error) {
11     die("Connection failed: " . $conn->connect_error);
12 }
13
14 // Add task
15 if (isset($_POST['add_task'])) {
16     $task_name = $_POST['task_name'];
17     $sql = "INSERT INTO tasks (task_name) VALUES ('$task_name')";
18     $conn->query($sql);
19 }
20
21 // Get tasks
22 $sql = "SELECT * FROM tasks WHERE completed = 0 ORDER BY id DESC";
23 $result = $conn->query($sql);
24
25 ?>
26
27 <!DOCTYPE html>
28 <html lang="en">
29 <head>
30     <meta charset="UTF-8">
31     <meta name="viewport" content="width=device-width, initial-scale=1.0">
32     <title>To-Do List</title>
33     <style>

```

```

<style>
    body {
        font-family: 'Arial', sans-serif;
        margin: 0;
        padding: 0;
        background-color: #fff8d8;
        text-align: center;
    }

    header {
        background-color: #080808;
        color: #fff;
        padding: 20px 0;
    }

    h1 span:nth-child(1) { color: #ff719a; }
    h1 span:nth-child(2) { color: #ffebcc; }
    h1 span:nth-child(3) { color: #c2f0c2; }
    h1 span:nth-child(4) { color: #c2e0ff; }
    h1 span:nth-child(5) { color: #e0ccff; }
    h1 span:nth-child(6) { color: #ffcce5; }
    h1 span:nth-child(7) { color: #ff719a; }
    h1 span:nth-child(8) { color: #c2f0c2; }
    h1 span:nth-child(9) { color: #ffcce5; }

    #todo-container {
        width: 300px;
        margin: 20px auto;
        background-color: #fff;
        box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
        border-radius: 10px;
        overflow: hidden;
    }

```

```

    #todo-list {
        list-style: none;
        padding: 0;
    }

    #todo-list li {
        display: flex;
        justify-content: space-between;
        align-items: center;
        padding: 10px;
        border-bottom: 1px solid #eee;
    }

    #todo-list li:last-child {
        border-bottom: none;
    }

    #todo-list input[type="checkbox"] {
        margin-right: 10px;
    }

    #add-todo {
        padding: 10px;
        border-top: 1px solid #eee;
        display: flex;
        justify-content: space-between;
        align-items: center;
    }

```

```

    #new-task {
        flex-grow: 1;
        margin-right: 10px;
        padding: 8px;
        border: 1px solid #ddd;
        border-radius: 5px;
    }

    #add-button, .delete-button {
        background-color: #fbd59; /* Pink button */
        color: #fff;
        padding: 8px 12px;
        cursor: pointer;
        border: none;
        border-radius: 5px;
        transition: background-color 0.3s;
    }

    #add-button:hover, .delete-button:hover {
        background-color: #ffbb33; /* Darker pink on hover */
    }
</style>
</head>
<body>
    <header>
        <h1><span>T</span><span>o</span><span>D</span><span>o</span><span>L</span><span>i</span><span>s</span></h1>
    </header>

```

```

<div id="todo-container">
    <ul id="todo-list">
        <?php while ($row = $result->fetch_assoc()) : ?>
            <li>
                <input type="checkbox" data-id="<?php echo $row['id']; ?>">
                <label><?php echo $row['task_name']; ?></label>
            </li>
        <?php endwhile; ?>
    </ul>
    <form id="add-todo" method="POST" action="">
        <input type="text" id="new-task" name="task_name" placeholder="Add new task" required>
        <button type="submit" id="add-button" name="add_task">Add</button>
    </form>
</div>

<script>
    // JavaScript for handling checkbox click event
    const checkboxes = document.querySelectorAll('#todo-list input[type="checkbox"]');
    checkboxes.forEach(checkbox => {
        checkbox.addEventListener('change', function() {
            const taskId = this.getAttribute('data-id');
            const completed = this.checked ? 1 : 0;
            fetch('update_task.php', {
                method: 'POST',
                headers: {
                    'Content-Type': 'application/json',
                },
                body: JSON.stringify({ id: taskId, completed: completed }),
            })
                .then(response => response.json())
                .then(data => {

```

```

                // JavaScript for handling checkbox click event
                const checkboxes = document.querySelectorAll('#todo-list input[type="checkbox"]');
                checkboxes.forEach(checkbox => {
                    checkbox.addEventListener('change', function() {
                        const taskId = this.getAttribute('data-id');
                        const completed = this.checked ? 1 : 0;
                        fetch('update_task.php', {
                            method: 'POST',
                            headers: {
                                'Content-Type': 'application/json',
                            },
                            body: JSON.stringify({ id: taskId, completed: completed }),
                        })
                            .then(response => response.json())
                            .then(data => {
                                if (data.success) {
                                    if (completed) {
                                        this.parentElement.classList.add('completed');
                                        this.checked = true;
                                    } else {
                                        this.parentElement.classList.remove('completed');
                                        this.checked = false;
                                    }
                                }
                            })
                            .catch(error => console.error('Error:', error));
                    });
                });
            </script>
</body>
</html>

```

## **Tools and languages used:**

1. HTML
2. CSS
3. JS(for front end)
4. php (for connection)
5. SQL(for data base)
6. Xampp server for SQL
7. Vs Code for php and front end



# Data Base

	Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/>	calorie1	Browse  Structure  Search  Insert  Empty  Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
<input type="checkbox"/>	calorie2	Browse  Structure  Search  Insert  Empty  Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
<input type="checkbox"/>	expense	Browse  Structure  Search  Insert  Empty  Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
<input type="checkbox"/>	period	Browse  Structure  Search  Insert  Empty  Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
<input type="checkbox"/>	sleep	Browse  Structure  Search  Insert  Empty  Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
<input type="checkbox"/>	todo	Browse  Structure  Search  Insert  Empty  Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
6 tables Sum			0	InnoDB	utf8mb4_general_ci	96.0 KiB	0 B

☐ Check all
 

With selected: ▼

thank you!

