Team Budget Planner

Source Code

home.html

```html
<!DOCTYPE html>

<html lang="en">



<head>

    <meta charset="UTF-8" />

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

    <title>Budget Planner</title>



    <link rel="preconnect" href="https://fonts.googleapis.com">

    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

    <link
href="https://fonts.googleapis.com/css2?family=Open+Sans&display=swap"
rel="stylesheet">

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

</head>



<body>
```

```html
    <ul>
        <li><a style="font-size: 20px;" class="active"
href="home.html"><b>Budget Planner</b></a></li>
        <li><a style="padding-top: 20px;" class="active"
href="home.html">Home</a></li>
    </ul>


    <div class="container">


        <div class="header">
            <h1><u>Budget Planner App</u></h1>
        </div>


        <div class="budget-section">


            <div class="budget col col-md col-sm">
                <h2>budget</h2>
                <img src="image/money-bag.svg" width="40" alt="" />
                <p class="amount">Rs. <span id="budgetAmount">0</span></p>
            </div>


            <div class="expenses col col-md col-sm">
                <h2>expenses</h2>
                <img src="image/accounting.svg" width="40" alt="" />
                <p class="exp-amount">Rs. <span
id="expensesAmount">0</span></p>
            </div>
```

```html
        <div class="balance col col-md col-sm">

            <h2>balance</h2>

            <img src="image/law.svg" width="40" alt="" />

            <p class="amount bala">Rs. <span
id="balanceAmount">0</span></p>

        </div>


    </div>


    <div id="displayExpenses">


        <div class="budget-list">


            <div class="col">

                <h4>expense title</h4>

            </div>


            <div class="col">

                <h4>expense value</h4>

            </div>


            <div class="col"></div>

        </div>


        <div id="expValue"></div>

    </div>

</div>
```

```html
<div class="toggle">

    <button id="myBtn" type="button">+</button>

</div>


<!-- The Modal -->

<div id="myModal" class="modal">

    <!-- Modal content -->

    <div class="modal-content">

        <div class="modal-header">

            <span class="close">&times;</span>

            <h2>Budget Form</h2>

        </div>

        <div class="modal-body">

            <div class="budget-form" id="budgetform">

                <form id="addForm">

                    <label for=""> Make a budget</label> <br />

                    <input type="number" id="number" /> <br />

                    <button type="submit">Add Budget</button>

                </form>

            </div>


            <div class="expense-form" id="expense-form">

                <form action="" id="expForm">

                    <div class="">
```

```html
                        <label for="">please enter your expense</label>
<br />
                        <input type="text" id="expName" />
                    </div>
                    <div class="">
                        <label for="">please enter expense amount</label>
<br />
                        <input type="number" id="expNumber" />
                    </div>
                    <button type="submit" id="submitExpen">Add
expense</button>
                </form>
                <button onclick="callBudget()" id="bug">
                    Back to add budget ->
                </button>
            </div>


        </div>

    </div>

</div>


<div class="footer">

    <footer>

        <p>All Rights Reserved. Copyright © 2022 || Budget Planner by
Bikki Shaw </p>

    </footer>

</div>
```

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

</body>


</html>
```

style.CSS

```css
body {

  font-family: 'Open Sans', sans-serif;

  margin: 0;

  padding: 0;

  width: 100%;

  height: 100vh;

  background-color: #f5f5f5;

}


.container {

  min-height: 80vh;

}


.header {

  height: 60px;

  width: 100%;

}


h1 {

  margin: 0;
```

```css
    text-align: center;

    padding-top: 10px;

}


.budget-section {

    display: flex;

    justify-content: space-around;

    align-items: flex-start;

    width: 100%;

    min-height: 160px;

    font-weight: 900;

}


.budget-list {

    display: flex;

    justify-content: space-around;

    align-items: flex-start;

    width: 100%;

    height: 70px;

    font-size: 20px;

}


.exp-amount {

    color: #4d4daf;

}


.amount {

    color: rgb(55, 159, 146);
```

```css
}

.col {

  width: 40%;

  text-align: center;

  text-transform: capitalize;

}


.col p {

  font-size: 40px;

  padding-top: 5px;

  margin: 0;

}


.toggle {

  position: absolute;

  padding-bottom: 100px;

  right: 0;

  bottom: 18px;

}


.toggle button {

  border: none;

  border-radius: 600px;

  background: #4d4daf;

  padding: 14px 20px;

  color: #ffffff;

  font-size: 30px;
```

```css
    margin: 10px 50px 0 0;

    cursor: pointer;

}


/* modal form */

.budget-form {

    width: 100%;

    min-height: 200px;

}


form {

    margin: 10px 0 20px 10px;

    min-height: 100px;

}


input {

    width: 60%;

    padding: 8px 0;

    border-radius: 6px;

    border: 1px solid gray;

    margin: 20px 0;

    padding-left: 5px;

    color: #495057;

}


.budget-list .exp p {

    color: #4d4daf;

    font-size: 18px;
```

```css
}


#displayExpenses {

  width: 100%;

  display: none;

}


.expValue {

  display: flex;

  justify-content: space-around;

  height: 40px;

  font-weight: 900;

  text-align: center;

}


#expTitleName {

  width: 40%;

}


#expValueAmount {

  width: 40%;

}


#edite_delete {

  width: 40%;

}



#edite_delete button {
```

```css
    margin-left: 1px;

    background: none;

    border: none;

}


.budget-form button {

    background: rgb(55, 159, 146);

    border: none;

    border-radius: 6px;

    color: #ffffff;

    padding: 10px 40px;

    cursor: pointer;

    text-transform: capitalize;

    font-size: 16px;

    font-weight: 400;

}


#bug {

    font-size: 14px;

    background: none;

    color: #4d4daf;

    margin: 0px 0px 15px 10px;

    text-align: justify;

    padding: 0;

}


#expense-form {

    width: 100%;
```

```css
    display: none;
}



.expense-form button {

  background: #4d4daf;

  border: none;

  border-radius: 6px;

  color: #ffffff;

  padding: 10px 30px;

  cursor: pointer;

  text-transform: capitalize;

  font-size: 16px;

  font-weight: 400;
}


/* The Modal (background) */

.modal {

  display: block;

  position: fixed;

  z-index: 1;

  left: 0;

  top: 0;

  width: 100%;

  height: 100%;

  overflow: auto;

  background-color: rgba(0, 0, 0, 0.4);
```

```css
  /* animation-name: fadeIn;

  animation-duration: 0.4s; */

  text-transform: capitalize;

  font-weight: 600;

}


/* Modal Content */
.modal-content {

  position: fixed;

  top: 20%;

  left: 25%;

  background-color: #fefefe;

  width: 50%;

  min-height: 40px;

  animation-name: slideIn;

  animation-duration: 0.4s;

}


/* The Close Button */
.close {

  color: white;

  float: right;

  font-size: 28px;

  font-weight: bold;

  cursor: pointer;

  color: #000;

}
```

```css
.modal-header {

  padding: 2px 16px;

  border-bottom: 2px solid gray;

}


.modal-body {

  padding: 2px 16px;

}


/* navbar */
ul {

  list-style-type: none;

  margin: 0;

  padding: 0;

  overflow: hidden;

  background-color: #333;

}


li {

  float: left;

}


li a {

  display: block;

  color: white;

  text-align: center;

  padding: 14px 16px;

  padding-bottom: 16px;
```

```css
      text-decoration: none;

}


li a:hover {

  background-color: #111;

}


/* footer */

.footer {

  text-align: center;

  padding: 3px;

  background-color: #333;

  color: white;

  position: absolute;

  bottom: 0;

  width: 99.5%

}


/* Add Animation */

@keyframes slideIn {

  from {

    top: -300px;

    opacity: 0;

  }


  to {

    top: 0;

    opacity: 1;
```

```css
    }
}


@keyframes fadeIn {
  from {
    opacity: 0;
  }


  to {
    opacity: 1;
  }
}
```

main.js

```js
const modal = document.getElementById("myModal");
const btn = document.getElementById("myBtn");
const span = document.getElementsByClassName("close")[0];
btn.onclick = function () {
  expName.value = "";
  expNumber.value = "";
  expenseForm.style.display = "block";
  modal.style.display = "block";
};
span.onclick = function () {
  modal.style.display = "none";
};
```

```javascript
window.onclick = function (event) {

  if (event.target == modal) {

    modal.style.display = "none";

  }

};


const amountInput = document.getElementById("number");

const addForm = document.getElementById("addForm");

const budgetAmount = document.getElementById("budgetAmount");

const balanceAmount = document.getElementById("balanceAmount");


const expForm = document.getElementById("expForm");

const expensesAmount = document.getElementById("expensesAmount");

const expValue = document.getElementById("expValue");

const displayExpenses = document.getElementById("displayExpenses");

const expenseForm = document.getElementById("expense-form");

const budgetform = document.getElementById("budgetform");


let expName = document.getElementById("expName");

let expNumber = document.getElementById("expNumber");

let id = 0;

let details = [];


function getBudgetAmount(amount) {

  if (!amount) {

    amountInput.style.border = "1px solid #b80c09";

    amountInput.placeholder = "input can not be empty";
```

```javascript
      amountInput.style.color = "#b80c09";

      setTimeout(() => {

        amountInput.style.color = "#495057";

        amountInput.style.border = "1px solid gray";

      }, 3000);

    } else {

      budgetAmount.innerText = amount;

      balanceAmount.innerText = amount;

      expenseForm.style.display = "block";

      budgetform.style.display = "none";

      amountInput.value = "";

    }

}


function addExpenses(name, number) {

  if (!name.length || !number.length) {

    expName.style.border = "1px solid #b80c09";

    expName.placeholder = "input can not be empty";

    expName.style.color = "#b80c09";


    expNumber.style.border = "1px solid #b80c09";

    expNumber.placeholder = "input can not be empty";

    expNumber.style.color = "#b80c09";


    setTimeout(() => {

      expName.style.color = "#495057";

      expName.style.border = "1px solid gray";

      expName.placeholder = "input can not be empty";
```

```javascript
                expNumber.placeholder = "input can not be empty";

                expNumber.style.border = "1px solid gray";

                expNumber.style.color = "#495057";

            }, 3000);

        } else {

            const userExp = {

                id: id,

                name: name,

                number: parseInt(number),

            };

            details.push(userExp);

            displayExp(details);

            id++;

            expName.value = "";

            expNumber.value = "";

    }

}


function displayExp(details) {

    expValue.innerHTML = null;

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

        expValue.innerHTML += `

        <div class="expValue" id="${details[i].id}">

            <div id="expTitleName" class="exp"><p>${details[i].name}</p></div>

            <div id="expValueAmount" class="exp"><p> <span>Rs.</span>

${details[i].number}</p></div>

            <div id="edite_delete">
```

```
        <p>
          <button id="${details[i].id}"
onclick="delExpenseDetails(${details[i].id})"></button>
        </p>
      </div>
    </div>
  `;
  }
  calcExpenses();
  displayExpenses.style.display = "block";
}


function calcExpenses() {
  let totalExp = 0;
  for (i = 0; i < details.length; i++) {
    totalExp = details[i].number + totalExp;
  }
  expensesAmount.innerText = totalExp;
  updateBalance();
}


function updateBalance() {
  balanceAmount.innerText =
    parseInt(budgetAmount.innerText) - parseInt(expensesAmount.innerText);
}


function delExpenseDetails(id) {
  let index = details.findIndex((item) => item.id === id);
```

```javascript
    details.splice(index, 1);

    displayExp(details);

}




function callBudget() {

  budgetform.style.display = "block";

  expenseForm.style.display = "none";

}




expForm.addEventListener("submit", (e) => {

  e.preventDefault();

  addExpenses(expName.value, expNumber.value);

});



addForm.addEventListener("submit", (e) => {

  e.preventDefault();

  getBudgetAmount(amountInput.value);

});
```