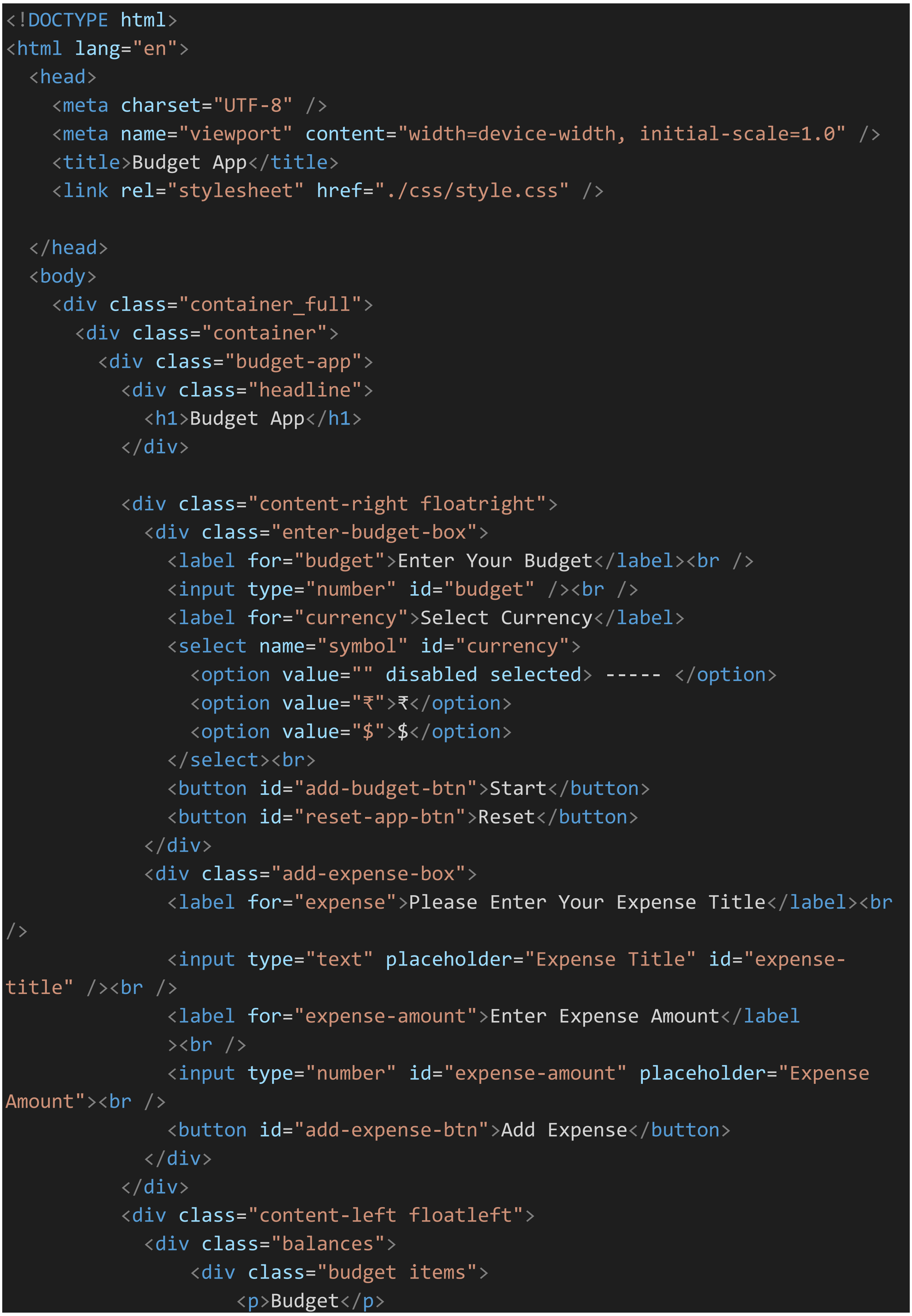
Team Budget Planner (Source Code)

Index.html:



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Budget App</title>
    <link rel="stylesheet" href="./css/style.css" />

  </head>
  <body>
    <div class="container_full">
      <div class="container">
        <div class="budget-app">
          <div class="headline">
            <h1>Budget App</h1>
          </div>

          <div class="content-right floatright">
            <div class="enter-budget-box">
              <label for="budget">Enter Your Budget</label><br />
              <input type="number" id="budget" /><br />
              <label for="currency">Select Currency</label>
              <select name="symbol" id="currency">
                <option value="" disabled selected> ----- </option>
                <option value="₹">₹</option>
                <option value="$">$</option>
              </select><br>
              <button id="add-budget-btn">Start</button>
              <button id="reset-app-btn">Reset</button>
            </div>
            <div class="add-expense-box">
              <label for="expense">Please Enter Your Expense Title</label><br
/>
              <input type="text" placeholder="Expense Title" id="expense-
title" /><br />
              <label for="expense-amount">Enter Expense Amount</label
              ><br />
              <input type="number" id="expense-amount" placeholder="Expense
Amount"><br />
              <button id="add-expense-btn">Add Expense</button>
            </div>
          </div>
          <div class="content-left floatleft">
            <div class="balances">
                <div class="budget items">
                    <p>Budget</p>
```

Team Budget Planner (Source Code)

```
                    <div class="budget-amount"><span id="currency-
symbol">$</span> <span id="budget-amount">0</span></div>
                </div>
                <div class="expenses items">
                    <p>Expenses</p>
                    <div class="expenses-amount"><span id="currency-
symbol">$</span> <span id="expenses-amount">0</span></div>
                </div>
                <div class="balance items">
                    <p>Balance</p>
                    <div class="balance-amount"><span id="currency-
symbol">$</span> <span id="balance-amount">0</span></div>
                </div>
            </div>
            <div class="expenses">
                    <table>
                        <thead>
                            <tr>
                                <td>Expense Title</td>
                                <td>Expense
Amount</td>
                                <td>Delete</td>
                            </tr>
                        </thead>
                    </table>
                    <div id="expense-list"></div>
                </div>
            </div>
          </div>
        </div>
      </div>
    </div>

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

Style.css:

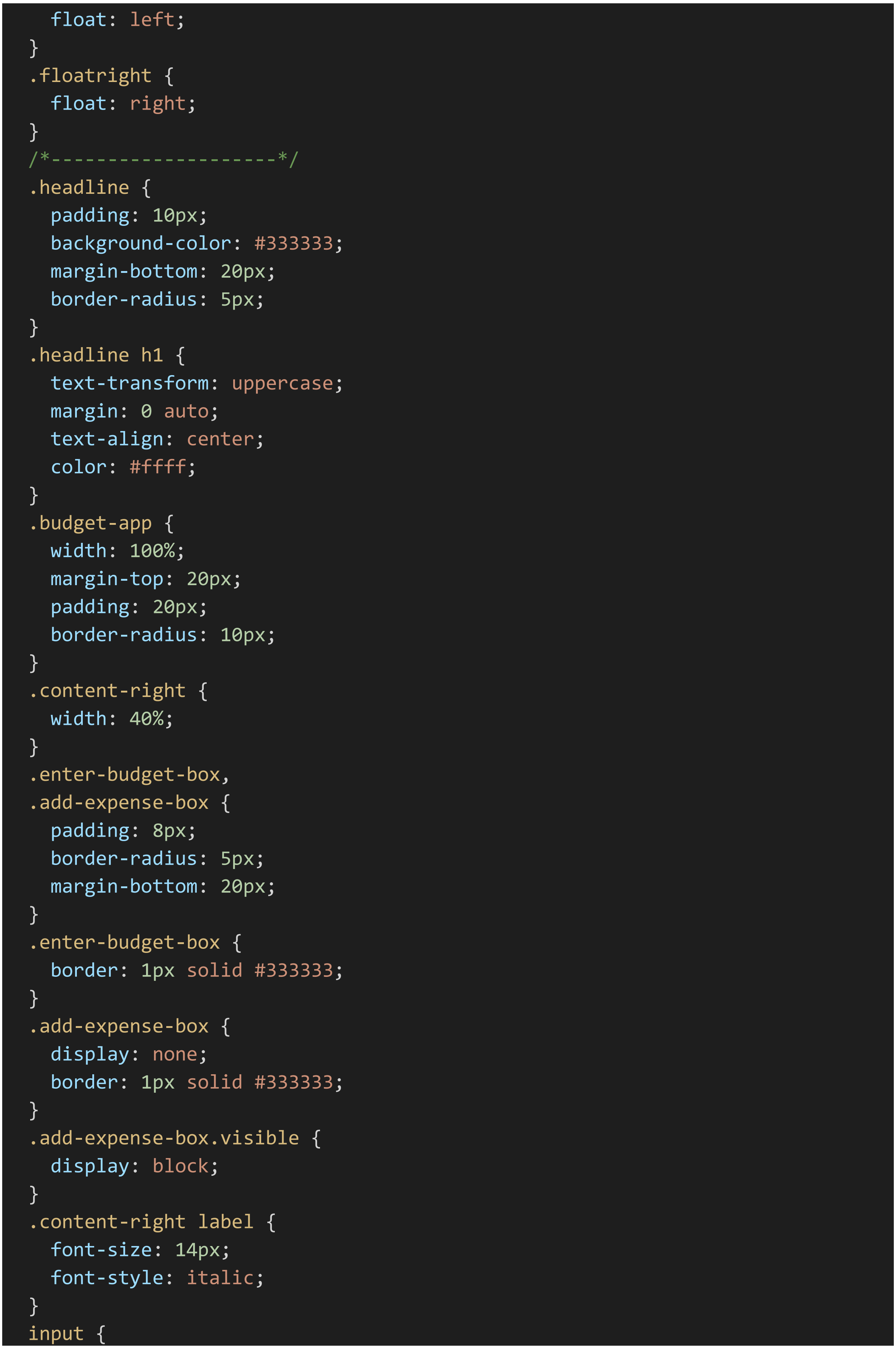
```
* {
    margin: 0;
    padding: 0;
  }
  body {
    font-family: Sans-Serif, Arial !important;
    transition: all 0.3s ease;
    -webkit-transition: all 0.3s ease;
    -moz-transition: all 0.3s ease;
```

Team Budget Planner (Source Code)

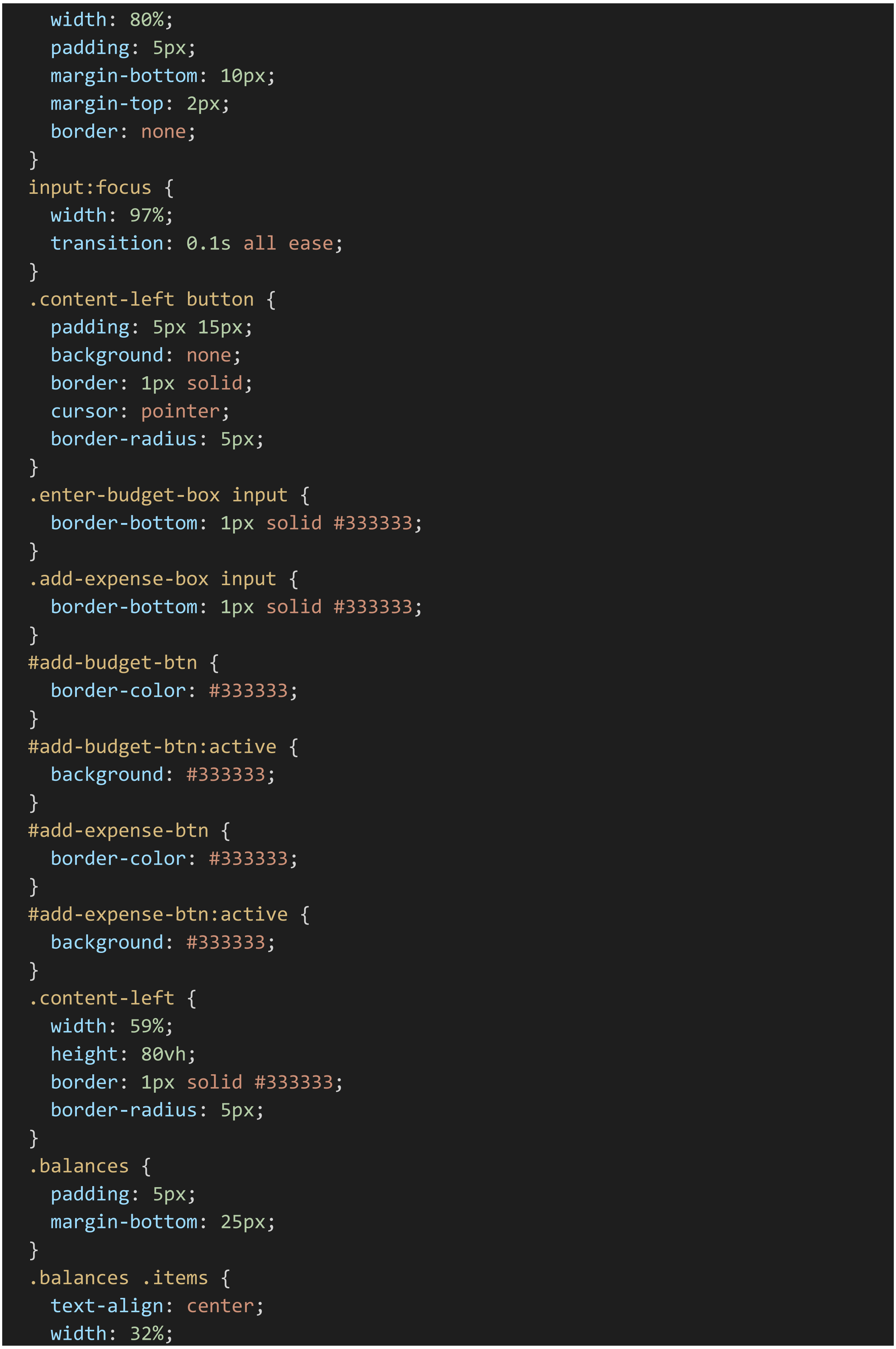
```
}
body,
html {
  height: 100%;
}
img {
  max-width: 100%;
}
li {
  list-style-type: none;
}
a {
  text-decoration: none;
}
a:hover {
  text-decoration: underline;
}
h1,
h2,
h3,
h4,
h5,
h6 {
  font-weight: normal;
  margin-top: 10px;
  margin-bottom: 10px;
}
p {
  line-height: 150%;
  /* margin-bottom: 2em; */
}

.container_full {
  width: 100%;
  height: auto;
}
.container {
  width: 75%;
  margin: 0 auto;
}
.fix {
  overflow: hidden;
}
.aligncenter {
  display: block;
  margin: 0 auto;
}
.floatleft {
```

Team Budget Planner (Source Code)



```css
  float: left;
}
.floatright {
  float: right;
}
/*--------------------*/
.headline {
  padding: 10px;
  background-color: #333333;
  margin-bottom: 20px;
  border-radius: 5px;
}
.headline h1 {
  text-transform: uppercase;
  margin: 0 auto;
  text-align: center;
  color: #ffff;
}
.budget-app {
  width: 100%;
  margin-top: 20px;
  padding: 20px;
  border-radius: 10px;
}
.content-right {
  width: 40%;
}
.enter-budget-box,
.add-expense-box {
  padding: 8px;
  border-radius: 5px;
  margin-bottom: 20px;
}
.enter-budget-box {
  border: 1px solid #333333;
}
.add-expense-box {
  display: none;
  border: 1px solid #333333;
}
.add-expense-box.visible {
  display: block;
}
.content-right label {
  font-size: 14px;
  font-style: italic;
}
input {
```

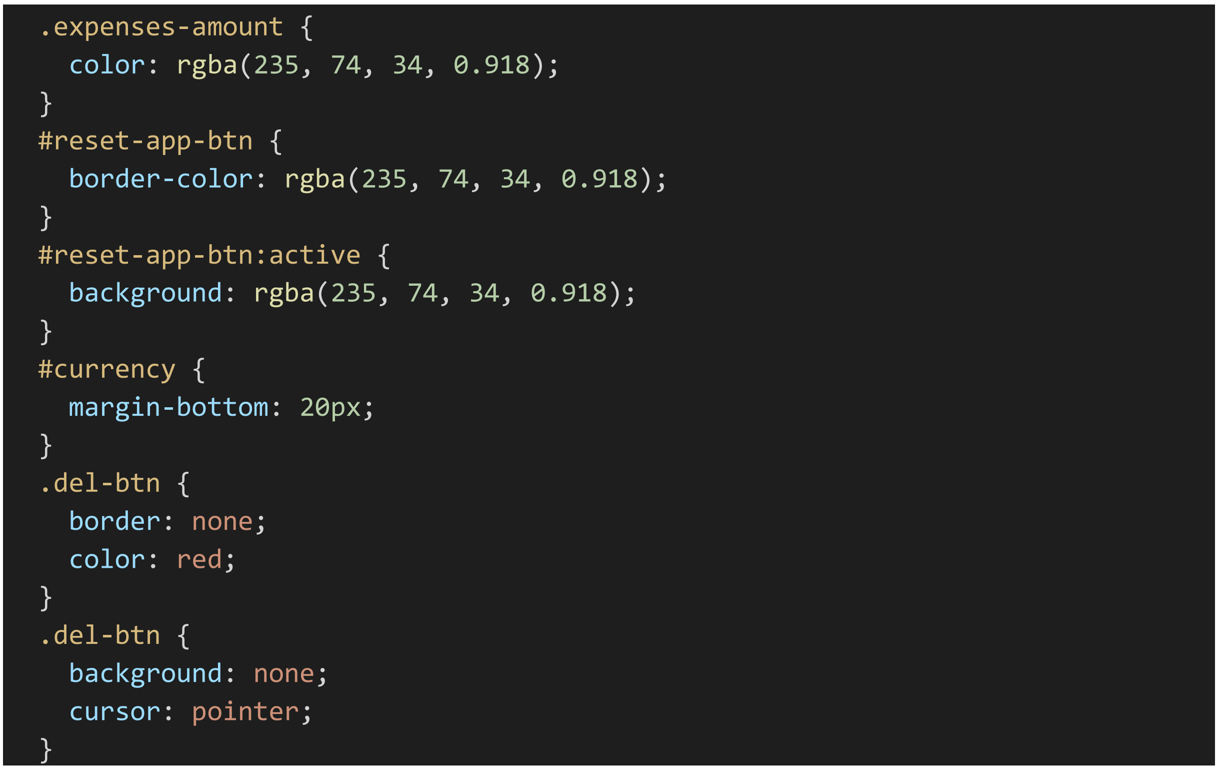


```
    width: 80%;
    padding: 5px;
    margin-bottom: 10px;
    margin-top: 2px;
    border: none;
  }
  input:focus {
    width: 97%;
    transition: 0.1s all ease;
  }
  .content-left button {
    padding: 5px 15px;
    background: none;
    border: 1px solid;
    cursor: pointer;
    border-radius: 5px;
  }
  .enter-budget-box input {
    border-bottom: 1px solid #333333;
  }
  .add-expense-box input {
    border-bottom: 1px solid #333333;
  }
  #add-budget-btn {
    border-color: #333333;
  }
  #add-budget-btn:active {
    background: #333333;
  }
  #add-expense-btn {
    border-color: #333333;
  }
  #add-expense-btn:active {
    background: #333333;
  }
  .content-left {
    width: 59%;
    height: 80vh;
    border: 1px solid #333333;
    border-radius: 5px;
  }
  .balances {
    padding: 5px;
    margin-bottom: 25px;
  }
  .balances .items {
    text-align: center;
    width: 32%;
```

Team Budget Planner (Source Code)

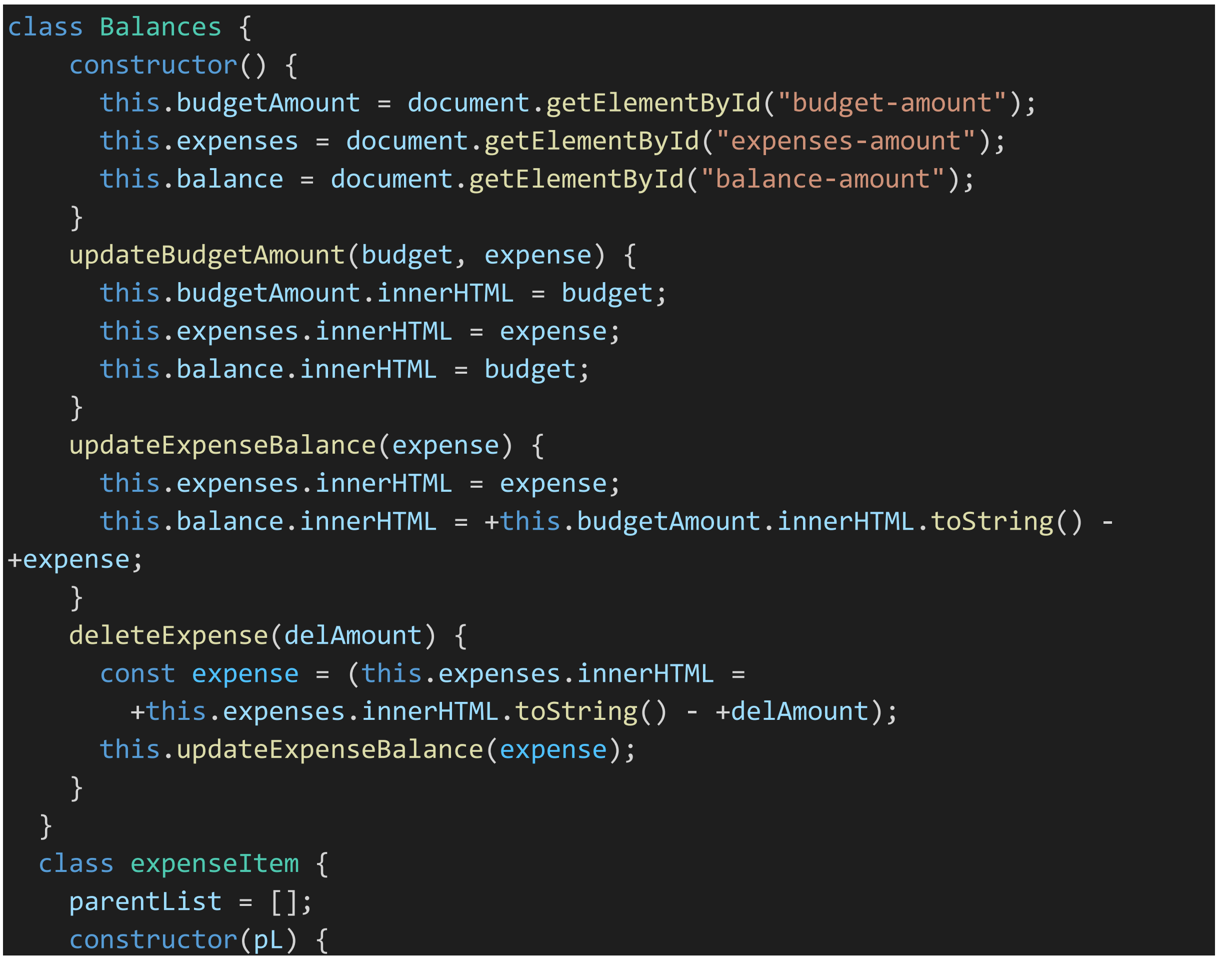
```
  display: inline-block;
}
.balances .items p {
  font-size: 20px;
  text-transform: uppercase;
}
.balances .items p{
  margin-bottom: 10px;
}
.expenses {
  padding: 5px;
  font-size: 14px;
}
.expenses table {
  text-align: center;
  border: 1px solid #333333;
  padding: 5px 0;
}
.expenses table td {
  width: 1%;
}
#expense-list {
  margin-top: 20px;
  width: 100%;
  text-align: center;
  height: 210px;
  overflow-y: scroll;
}
#expense-list td {
  height: 30px;
  width: 1%;
}
.expenses table thead {
  margin-bottom: 20px;
}

.budget-amount,
.balance-amount,
.expenses-amount {
  font-size: 30px;
  font-weight: bold;
}
.budget-amount {
  color: rgb(75, 197, 75);
}
.balance-amount {
  color: rgb(35, 147, 226);
}
```



```
 .expenses-amount {
  color: rgba(235, 74, 34, 0.918);
 }
 #reset-app-btn {
  border-color: rgba(235, 74, 34, 0.918);
 }
 #reset-app-btn:active {
  background: rgba(235, 74, 34, 0.918);
 }
 #currency {
  margin-bottom: 20px;
 }
 .del-btn {
  border: none;
  color: red;
 }
 .del-btn {
  background: none;
  cursor: pointer;
 }
```

Script.js:

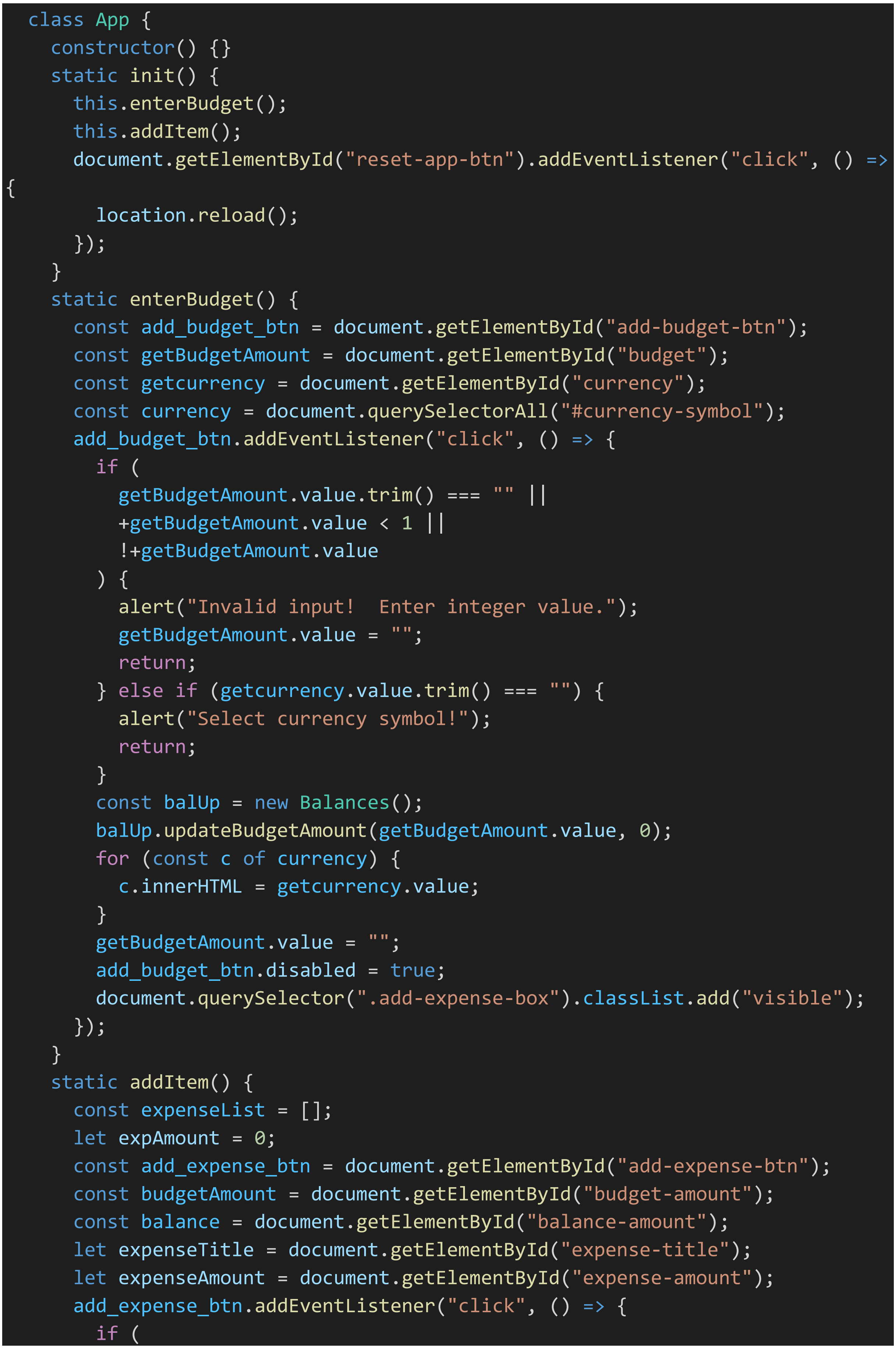


```
class Balances {
  constructor() {
   this.budgetAmount = document.getElementById("budget-amount");
   this.expenses = document.getElementById("expenses-amount");
   this.balance = document.getElementById("balance-amount");
  }
  updateBudgetAmount(budget, expense) {
   this.budgetAmount.innerHTML = budget;
   this.expenses.innerHTML = expense;
   this.balance.innerHTML = budget;
  }
  updateExpenseBalance(expense) {
   this.expenses.innerHTML = expense;
   this.balance.innerHTML = +this.budgetAmount.innerHTML.toString() -
+expense;
  }
  deleteExpense(delAmount) {
   const expense = (this.expenses.innerHTML =
    +this.expenses.innerHTML.toString() - +delAmount);
   this.updateExpenseBalance(expense);
  }
 }
 class expenseItem {
  parentList = [];
  constructor(pL) {
```

Team Budget Planner (Source Code)

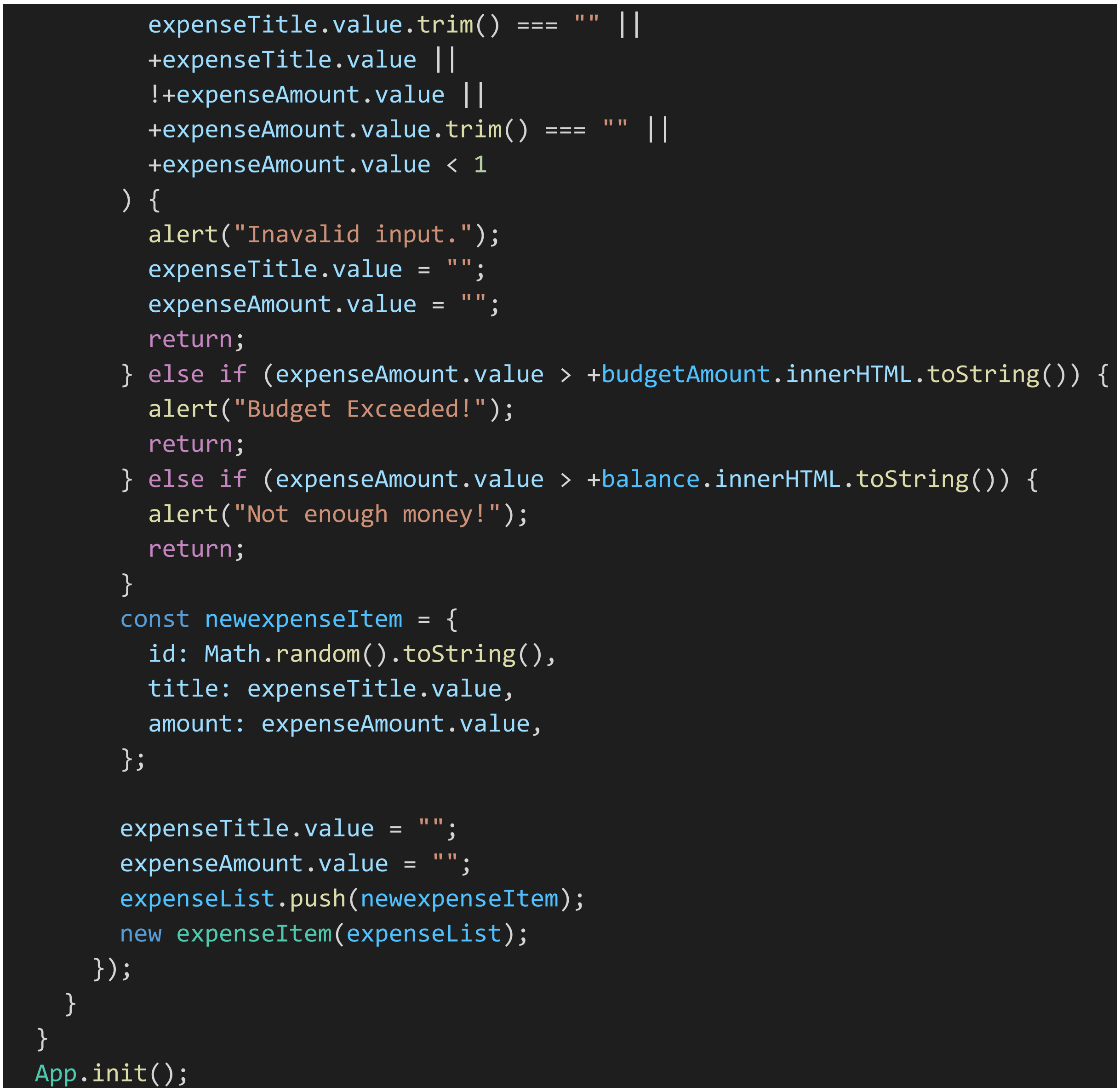
```
      this.parentList = pL;
      this.updateBalances();
      this.renderExpensesItem(this.parentList);
    }
    updateBalances() {
      let expenseAmount = 0;
      for (const item of this.parentList) {
        expenseAmount += +item.amount;
      }
      const balObj = new Balances();
      balObj.updateExpenseBalance(expenseAmount);
    }
    deleteExpenseItem(eid) {
      let itemIndex = 0;
      const balObj = new Balances();
      for (const item of this.parentList) {
        if (item.id == eid) {
          balObj.deleteExpense(item.amount);
          break;
        }
        itemIndex++;
      }
      this.parentList.splice(itemIndex, 1);
      this.renderExpensesItem(this.parentList);
    }
    renderExpensesItem(expenseList) {
      const expense_list = document.getElementById("expense-list");
      expense_list.innerHTML = "";
      const getcurrency = document.getElementById("currency");
      if (expenseList && expenseList.length > 0) {
        for (const expenseItem of expenseList) {
          const ItemRow = document.createElement("tr");
          ItemRow.innerHTML = `
        <td>${expenseItem.title}</td>
        <td><span id="">${getcurrency.value}</span>
${expenseItem.amount}</td>
        <td><button class="del-btn">&times;</button></td>
        `;
          const delBtn = ItemRow.querySelector(".del-btn");
          delBtn.addEventListener(
            "click",
            this.deleteExpenseItem.bind(this, expenseItem.id)
          );
          expense_list.append(ItemRow);
        }
      }
    }
  }
```

Team Budget Planner (Source Code)



```
  class App {
    constructor() {}
    static init() {
      this.enterBudget();
      this.addItem();
      document.getElementById("reset-app-btn").addEventListener("click", () =>
{
        location.reload();
      });
    }
    static enterBudget() {
      const add_budget_btn = document.getElementById("add-budget-btn");
      const getBudgetAmount = document.getElementById("budget");
      const getcurrency = document.getElementById("currency");
      const currency = document.querySelectorAll("#currency-symbol");
      add_budget_btn.addEventListener("click", () => {
        if (
          getBudgetAmount.value.trim() === "" ||
          +getBudgetAmount.value < 1 ||
          !+getBudgetAmount.value
        ) {
          alert("Invalid input!  Enter integer value.");
          getBudgetAmount.value = "";
          return;
        } else if (getcurrency.value.trim() === "") {
          alert("Select currency symbol!");
          return;
        }
        const balUp = new Balances();
        balUp.updateBudgetAmount(getBudgetAmount.value, 0);
        for (const c of currency) {
          c.innerHTML = getcurrency.value;
        }
        getBudgetAmount.value = "";
        add_budget_btn.disabled = true;
        document.querySelector(".add-expense-box").classList.add("visible");
      });
    }
    static addItem() {
      const expenseList = [];
      let expAmount = 0;
      const add_expense_btn = document.getElementById("add-expense-btn");
      const budgetAmount = document.getElementById("budget-amount");
      const balance = document.getElementById("balance-amount");
      let expenseTitle = document.getElementById("expense-title");
      let expenseAmount = document.getElementById("expense-amount");
      add_expense_btn.addEventListener("click", () => {
        if (
```

Team Budget Planner (Source Code)



```
        expenseTitle.value.trim() === "" ||
        +expenseTitle.value ||
        !+expenseAmount.value ||
        +expenseAmount.value.trim() === "" ||
        +expenseAmount.value < 1
      ) {
        alert("Inavalid input.");
        expenseTitle.value = "";
        expenseAmount.value = "";
        return;
      } else if (expenseAmount.value > +budgetAmount.innerHTML.toString()) {
        alert("Budget Exceeded!");
        return;
      } else if (expenseAmount.value > +balance.innerHTML.toString()) {
        alert("Not enough money!");
        return;
      }
      const newexpenseItem = {
        id: Math.random().toString(),
        title: expenseTitle.value,
        amount: expenseAmount.value,
      };

      expenseTitle.value = "";
      expenseAmount.value = "";
      expenseList.push(newexpenseItem);
      new expenseItem(expenseList);
    });
  }
}
App.init();
```