

Budget Tracker

Your Balance

£ 0

Add Transaction

Name

Type

☒ Income

☐ Expense

Amount

Submit

Transaction Details

No transactions found!

© Designed & Developed by Anupam Pammi

Budget Tracker

Your Balance

£ 250

Add Transaction

Name

Type





☐ Income

☒ Expense

Amount

Submit

Transaction Details

SALARY	BILLS
£ 450	£ 200
 	 

BUDGET TRACKER APPLICATION USING HTML, CSS & JS

Designed & Developed by **Anupam Pammi**

HTML CODE

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
    <title>Budget Tracker Application</title>

    <!-- CSS LINK -->
    <link rel="stylesheet" href="./main.css" />

    <!-- Bootstrap CSS -->
    <link

href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bo
otstrap.min.css"
    rel="stylesheet"
    integrity="sha384-
Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeu0xjzrPF/et3URy9Bv1WT
Ri"
    crossorigin="anonymous"
  />

    <!-- FONT AWESOME CDN -->
    <script
      src="https://kit.fontawesome.com/0581082bd4.js"
      crossorigin="anonymous"
    ></script>
  </head>

  <body>
    <h1 class="head">Budget Tracker</h1>

    <div class="main">
      <div class="container">
        <div class="topbar">
```

```

    <p class="bal_head">Your Balance</p>
    <p class="bal_body">
        <span id="currency">f</span>
        <span id="balance">100</span>
    </p>
</div>
<div class="form">
    <p class="trans_head">Add Transaction</p>
    <div class="form-c">
        <label class="label" for="name">Name</label>
        <input type="text" name="name" id="name"
placeholder="Details" />
    </div>
    <div class="form-c">
        <label class="label" for="type">Type</label>
        <div class="radio">
            <input
                class="form-check-input"
                type="radio"
                name="type"
                id="income"
                checked
            />
            <label for="radio">Income</label>
        </div>
        <div class="radio">
            <input
                class="form-check-input"
                type="radio"
                name="type"
                id="expense"
            />
            <label for="radio">Expense</label>
        </div>
    </div>
    <div class="form-c">
        <label class="label" for="amount">Amount</label>
        <input

```



```
        <i class="fa-solid fa-pen-to-square fa-  
lg"></i>  
        <i class="fa-solid fa-trash fa-lg"></i>  
    </div>  
    </li>  
    </ul>  
    </div>  
    </div>  
    </div>  
    <div class="container">  
        <span>&copy; Designed & Developed by Anupam  
Pammi</span>  
    </div>  
    </div>  
    <script src="./index.js"></script>  
    </body>  
</html>
```

CSS CODE

```
@import
url("https://fonts.googleapis.com/css2?family=Montserrat:wght@
100;200;300;400;500;600;700;800;900&family=Poppins:wght@100;20
0;300;400;500;600;700;800;900&family=Roboto:wght@100;300;400;5
00;700;900&family=Syne+Mono&display=swap");

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

.head {
  text-align: center;
  background-color: #0d4c92;
  padding: 15px;
  font-family: "Syne Mono", monospace;
  font-weight: bold;
  color: white;
}

body {
  min-height: 100vh;
  display: flex;
  flex-direction: column;
}

main {
  flex-grow: 1;
  display: flex;
  justify-content: center;
  align-items: center;
}

.topbar {
  margin-top: 20px;
  width: 100%;
```

```
background-color: #282a3a;
padding: 20px;
text-align: center;
}

.bal_head {
  font-family: "Roboto", sans-serif;
  font-size: 15px;
  font-weight: bold;
  color: white;
}

.bal_body {
  font-family: "Poppins", sans-serif;
  color: white;
  font-size: 45px;
}

.form {
  margin-top: 10px;
  text-align: center;
  padding: 15px;
  background-color: #ccc;
}

.trans_head {
  font-family: "Roboto", sans-serif;
  font-size: 30px;
  font-weight: bold;
}

.form-c {
  margin: 10px;
}

[type="number"],
[type="text"] {
  height: 40px;
```

```
    width: 50%;
}

.label {
    font-weight: bold;
}

.radio {
    margin: 5px;
    align-items: center;
}

input::placeholder {
    padding: 5px;
}

#cancel {
    display: none;
    float: right;
}

.trans_list {
    background-color: #e1e5ea;
    margin-top: 10px;
    text-align: center;
    padding: 15px;
}

.trans_lihead {
    font-family: "Roboto", sans-serif;
    font-size: 30px;
    font-weight: bold;
}

ul {
    list-style: none;
    margin: 0 40px;
    line-height: 30px;
```



```
}

.transaction {
  align-items: center;
  display: flex;
  justify-content: space-between;
  font-weight: bold;
  color: white;
  border-radius: 10px;
  text-transform: uppercase;
}

.income {
  background-color: #1a8d72;
  margin: 5px;
  padding: 10px;
}

.expense {
  background-color: #ff6464;
  margin: 5px;
  padding: 10px;
}

.right {
  display: inline-block;
  gap: 10px;
}

.fa-solid {
  cursor: pointer;
  vertical-align: middle;
  transition: color 0.3s ease;
  margin: 10px;
}

.fa-solid:hover {
  color: black;
}
```

```
}

.amount_display {
  font-size: 25px;
}

.footer {
  margin-top: 20px;
  text-align: center;
  background-color: #0d4c92;
  color: white;
}

@media screen and (min-width: 1024px) {
  [type="number"],
  [type="text"] {
    height: 40px;
    width: 30%;
  }
  ul {
    margin: 0 150px;
  }
  .right {
    align-items: center;
  }
  .transaction {
    display: inline-block;
    vertical-align: baseline;
    width: 120px;
  }
}
```

JAVASCRIPT CODE

```
const currencyHolder = document.getElementById("currency");
const balanceHolder = document.getElementById("balance");
const transactionNameHolder = document.getElementById("name");
const transactionAmountHolder =
document.getElementById("amount");
const income = document.getElementById("income");
const expense = document.getElementById("expense");
const saveButton = document.getElementById("submit");
const cancelButton = document.getElementById("cancel");
const displayList =
document.getElementById("list_of_transactions");

let symbol = "f";
let listOfTransactions = [];
let currentBalance = 0;

let editIndex = -1;
//Edit Function
function edit(i) {
  editIndex = i;
  transactionNameHolder.value = listOfTransactions[i].name;
  transactionAmountHolder.value =
listOfTransactions[i].amount;
  if (listOfTransactions[i].type == "income") {
    income.checked = true;
  } else {
    expense.checked = true;
  }
  cancelButton.style.display = "block";
}

//Delete Function
function del(i) {
  listOfTransactions = listOfTransactions.filter((e, index) =>
i !== index);
  render();
}
```

```
//Cancel Function
cancelButton.addEventListener("click", () => {
  editIndex = -1;
  transactionNameHolder.value = "";
  transactionAmountHolder.value = "";
  cancelButton.style.display = "none";
});

//Save Function
function saveData() {
  localStorage.setItem("symbol", symbol);
  localStorage.setItem("balance", currentBalance);
  localStorage.setItem("list",
JSON.stringify(listOfTransactions));
}

//Load Data from Local Storage
function loadData() {
  symbol = localStorage.getItem("symbol");
  listOfTransactions =
JSON.parse(localStorage.getItem("list"));
  currentBalance = Number(localStorage.getItem("balance"));
}

//Save Function - Onsubmit
saveButton.addEventListener("click", () => {
  if (
    transactionNameHolder.value == "" ||
    Number(transactionAmountHolder.value) <= 0
  ) {
    alert("Sorry, input fields can't be empty or negative");
    return;
  }

  let transaction = {
    name: transactionNameHolder.value,
    amount: Number(transactionAmountHolder.value),
```

```

    type: income.checked ? "income" : "expense",
  };

  if (editIndex == -1) listOfTransactions.push(transaction);
  else listOfTransactions[editIndex] = transaction;

  editIndex = -1;
  transactionNameHolder.value = "";
  transactionAmountHolder.value = "";
  render();
  cancelButton.style.display = "none";
});

//Main Function
function render() {
  currentBalance = listOfTransactions.reduce((total, value) =>
  {
    return value.type == "expense"
      ? total - value.amount
      : total + value.amount;
  }, 0);

  displayList.innerHTML = "";

  if (listOfTransactions == 0) {
    displayList.innerHTML += "No transactions found!";
  } else {
    listOfTransactions.forEach((e, i) => {
      displayList.innerHTML += `
        <li class="transaction ${e.type}">
          <p>${e.name}</p>
          <div class="right">
            <p class="amount_display">${symbol}
${e.amount}</p>
            <i onclick="edit(${i})" class="fa-solid fa-
pen-to-square fa-lg"></i>
            <i onclick="del(${i})" class="fa-solid fa-
trash fa-lg"></i>

```

```
        </div>
      </li>
    `;
  });
}

currencyHolder.innerHTML = symbol;
balanceHolder.innerHTML = currentBalance;
saveData();
}

loadData();

render();
```