TEAM BUDGET PLANNER

SOURCE CODE:

Index.html:

```html
<!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</title>
 <link rel="stylesheet" href="style.css" />
 <link rel="stylesheet" href="./Icon/style.css" />
 <link
 rel="stylesheet"
href="https://fonts.googleapis.com/css2?family=Material+Symbols+Outlin
ed:opsz,wght,FIL
L,GRAD@20..48,100..700,0..1,-50..200"
 />
 <link
 href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap
.min.css"
 rel="stylesheet"
 integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"
 crossorigin="anonymous"
 />
 </head>
 <body>
 <nav class="navbar navbar-dark bg-dark">
 <div class="container">
 <a class="navbar-brand" href="#">BUDGET TRACKER</a>
 </div>
 </nav>
 <div class="container-body">
 <div class="container mt-3 w-75 p-3">
 <div class="balance">
 <h4>Your Balance</h4>
 <h3 id="balance">$0.00</h3>
 </div>
 <div class="inc-exp-container">
 <div>
```

```html
<h4>Income</h4>
<p id="money-plus" class="money-plus">+$0.00</p>
</div>
<div>
<h4>Expense</h4>
<p id="money-minus" class="money-minus">-$0.00</p>
</div>
</div>
<div class="container-flex">
<div class="form-flex">
<!--Name-->
<div class="mb-3 row mt-5">
<label for="staticEmail" class="col-sm-2 col-form-label"
>Name</label
>
<div class="col-sm-10">
<input
type="text"
class="form-control"
id="Name"
placeholder="Enter text..."
/>
<span id="inputerrName"></span>
</div>
</div>
<!--Amount-->
<div class="mb-3 row">
<label class="col-sm-2 col-form-label">Amount</label>
<div class="col-sm-10">
<input
type="number"
class="form-control"
placeholder="Enter amount... "
id="Amount"
/>
<div class="message">
Note : ( negative : expense , positive : income )
</div>
<span id="inputerrAmount"></span>
</div>
<!--button-->
<div class="col-auto mt-4">
<button type="submit" class="btn btn-primary mb-3">
Add transaction
</button>
</div>
```

```html
    </div>
    </div>
    <div class="transaction-container">
    <h1 class=" ">Transaction History</h1>
    <ul id="transaction"></ul>
    </div>
    </div>
    </div>
    </div>
    <script type="text/javascript" src="script.js"></script>
    </body>
</html>
```

Style.css:

```css
* {
    padding: 0;
    margin: 0;
    box-sizing: border-box;
}
.material-symbols-outlined {
    color: red;
    font-variation-settings: "FILL" 0, "wght" 400, "GRAD" 0, "opsz"
48;
}
#transaction {
    list-style: none;
    margin: 0;
    padding: 0;
    margin-top: 30px;
}
#transaction li {
    display: flex;
    flex-direction: row;
    justify-content: space-between;
    padding: 10px 25px;
    margin-bottom: 10px;
    width: 500px;
    box-shadow: 0px 1px 1px 1px rgb(177, 177, 177);
}
#transaction li .income {
    display: inline-block;
    background-color: green;
    color: white;
    padding: 6px;
```

```css
  margin-top: 5px;
  border-radius: 3px;
}
#transaction li .expense {
  display: inline-block;
  background-color: red;
  color: white;
  padding: 5px;
  border-radius: 3px;
}
.inc-exp-container {
  background-color: #fff;
  padding: 20px;
  display: flex;
  justify-content: center;
  margin: 20px 0;
}
.inc-exp-container > div {
  flex: 1;
  text-align: center;
}
.inc-exp-container > div:first-of-type {
  border-right: 3px solid #dedede;
}
.money-plus {
  color: #2ecc71;
}
.money-minus {
  color: #c0392b;
}
#inputerrAmount,
#inputerrName {
  color: #c0392b;
  font-size: 12px;
}
.message {
  margin-top: 5px;
  font-size: 12px;
  color: rgba(104, 104, 104, 0.842);
}
```

Style 1.css:

```css
@font-face {
    font-family: "icomoon";
    src: url("fonts/icomoon.eot?y61vgu");
```

```css
    src: url("fonts/icomoon.eot?y61vgu#iefix") format("embedded-
opentype"),
    url("fonts/icomoon.ttf?y61vgu") format("truetype"),
    url("fonts/icomoon.woff?y61vgu") format("woff"),
    url("fonts/icomoon.svg?y61vgu#icomoon") format("svg");
    font-weight: normal;
    font-style: normal;
    font-display: block;
}
[class^="icon-"],
[class*=" icon-"] {
    /* use !important to prevent issues with browser extensions that
change fonts */
    font-family: "icomoon" !important;
    speak-as: never;
    font-style: normal;
    font-weight: normal;
    font-variant: normal;
    text-transform: none;
    line-height: 1;
    /* Better Font Rendering =========== */
    -webkit-font-smoothing: antialiased;
    -moz-osx-font-smoothing: grayscale;
}
.icon {
 cursor: pointer;
 position: relative;
 top: 5px;
 margin-left: 10px;
 font-size: 24px;
 color: red;
}
.icon-bin:before {
 content: "\e9ac";
}
.icon-cancel-circle:before {
 content: "\ea0d";
}
```

Script.js:

```javascript
const table = document.getElementById("transaction");
const Name = document.getElementById("Name");
const Total = document.getElementById("balance");
const Amount = document.getElementById("Amount");
```

```javascript
const AddExpense = document.querySelector("Button");
const moneyPlus = document.getElementById("money-plus");
const moneyMinus = document.getElementById("money-minus");
const inputerrName = document.getElementById("inputerrName");
const inputerrAmount = document.getElementById("inputerrAmount");
window.onload = () => {
 Name.value = "";
 Amount.value = "";
};
let transactions = [];
let removeTransaction = (id) => {
 transactions = transactions.filter((item) => {
 return item["id"] != id;
 });
 yourBalance();
 displayTransaction();
};
AddExpense.addEventListener("click", () => {
 addTransaction();
});
let addTransaction = () => {
 if (Name.value.trim() === "" || Amount.value.trim() === "") {
 alert("please add text and amount");
 } else {
 const transaction = {
 id: generateID(),
 name: Name.value,
 date: currentDate(),
 amount: parseInt(Amount.value),
 };
 transactions.push(transaction);
 yourBalance();
 displayTransaction();
 Name.value = "";
 Amount.value = "";
 }
};
// Generate random ID
let generateID = () => {
 return Math.floor(Math.random() * 100000000);
};
// current DATE
let currentDate = () => {
 const date = new Date();
 let currentDate = `${date.getDate()}-${
 date.getMonth() + 1
```

```javascript
    }-${date.getFullYear()}`;
  return currentDate;
};
let displayTransaction = () => {
  let Data = "";
  const result = transactions.map((item) => {
  Data += `<li>
  ${item["name"]}
  <div>
  <button type="button" class="${
  Number(item["amount"]) > 0 ? "btn btn-success" : "btn btn-danger"
  } ">$${Math.abs(Number(item["amount"]))} </button>
  <span id="remove" onclick="removeTransaction( ${
  item["id"]
  } ) " class="icon-bin icon" > </span>
  </div>
  </li> `;
  });
  table.innerHTML = Data;
};
let yourBalance = () => {
  const amounts = transactions.map((amount) => {
  return amount["amount"];
  });
  const total = amounts.reduce((acc, item) => (acc += item),
0).toFixed(2);
  const income = amounts
  .filter((item) => {
  return item > 0;
  })
  .reduce((acc, item) => (acc += item), 0)
  .toFixed(2);
  const expense = amounts
  .filter((item) => {
  return item < 0;
  })
  .reduce((acc, item) => (acc += item), 0)
  .toFixed(2);
  moneyPlus.innerText = `+${income}`;
  moneyMinus.innerText = `${expense}`;
  Total.innerText = `$${total}`;
};
```