

TEAM BUDGET PLANNER SOURCE CODE

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 Planner</title>

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

</head>

<body>

  <h2>Budget Planner</h2>

  <div class="container">

    <h4>your balance</h4>

    <h1 id="balance">$0.00</h1>

    <div class="inc-exp-container">

      <div>

        <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>

<h3>History</h3>

<ul id="list" class="list">

<li class="minus">

Cash -\$400<button class="delete-btn">x</button>

<h3>Add New Transition</h3>

<form id="form">

<div class="form-control">

<label for="Text">Text</label>

<input type="text" id="text" placeholder="Enter Text...">

</div>

<div class="form-control">

<label for="amount">Amount</label>

<input type="number" id="amount" placeholder="Enter Amount...">

</div>

<button class="btn">All Transactions</button>

</form>

</div>

<!--JavaScript File-->

<script src="script.js"></script>

</body>

</html>

CSS CODE

```
:root{
  --box-shadow:0 1px 3px rgba(0, 0, 0, 0.12),
  0 1px 2px rgba(0, 0, 0, 0.24);
}

*{
  box-sizing: border-box;
}

body{
  background-color: #f7f7f7;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  min-height: 100vh;
  margin: 0;
  font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto, Oxygen, Ubuntu, Cantarell,
'Open Sans', 'Helvetica Neue', sans-serif;
}

.container{
  margin: 30px auto;
  width: 350px;
}

h1{
```

```
    letter-spacing: 1px;  
    margin: 0;  
}
```

```
h3{  
    border-bottom: 1px solid #bbb;  
    padding-bottom: 10px;  
    margin: 40px 0 10px;  
}
```

```
h4{  
    margin: 0;  
    text-transform: uppercase;  
}
```

```
.inc-exp-container{  
    background-color: #fff;  
    box-shadow: var(--box-shadow);  
    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: 1px solid #dedede;  
}
```

```
.money{  
    font-size: 2;  
    letter-spacing: 1px;  
    margin: 5px;  
}
```

```
.money-plus{  
    color:#2ecc71;  
}
```

```
.money-minus{  
    color: #c0392b;  
}
```

```
label{  
    display:inline-block;  
    margin: 10px 0;  
}
```

```
input[type="text"],input[type="number"]{  
    border: 1px solid #dedede;  
    border-radius: 2px;  
    display: block;  
    font-size: 16px;  
    padding: 10px;  
    width: 100%;
```

```
}
```

```
.btn{  
  cursor: pointer;  
  background-color: #9c88ff;  
  box-shadow: var(--box-shadow);  
  color: #fff;  
  border: 0;  
  display: block;  
  font-size: 16px;  
  margin: 10px 0 30px;  
  padding: 10px;  
  width: 100%;  
}
```

```
.btn:focus,.delete-btn:focus{  
  outline: 0;  
}
```

```
.list{  
  list-style-type: none;  
  padding: 0;  
  margin-bottom: 40px;  
}
```

```
.list li{  
  background-color: #fff;  
  box-shadow: var(--box-shadow);  
  color: #333;
```

```
display: flex;
justify-content: space-between;
position: relative;
padding: 10px;
margin: 10px 0;
}
```

```
.list li.plus{
border-right: 5px solid #2ecc71;
}
```

```
.list li.minus{
border-right: 5px solid #c0392b;
}
```

```
.delete-btn{
cursor: pointer;
background-color: #e74c3c;
border: 0;
color: #fff;
font-size: 20px;
line-height: 20px;
padding: 2px 5px;
position: absolute;
top: 50;
left: 0;
transform: translate(-100%, -50%);
opacity: 0;
transition: opacity 0.3s ease;
```

```
}
```

```
.list li:hover .delete-btn{  
  opacity: 1;  
}
```

JAVA SCRIPT CODE

```
const balance = document.getElementById(  
  "balance"  
);
```

```
const money_plus = document.getElementById('money-plus');  
const money_minus = document.getElementById('money-minus');  
const list = document.getElementById('list');  
const form = document.getElementById('form');  
const text = document.getElementById('text');  
const amount = document.getElementById('amount');
```

```
const localStorageTransaction=JSON.parse(localStorage.getItem("transactions"));  
let transactions = localStorage.getItem("transactions")!= null ? localStorageTransaction: [];
```

```
//Add Transactions
```

```
function addTransaction(e){  
  e.preventDefault();  
  if(  
    text.value.trim()!=="" || amount.value.trim() == ""
```



```

    ){
        alert("Please enter text and value");
    }else{
        const transaction = {
            id:generateID(),
            text:text.value,
            amount:+amount.value,
        };

        transactions.push(transaction);
        addTransactionDOM(transaction);
        updateLocalStorage();
        updateValues();
        text.value="";
        amount.value="";
    }
}

//Generate ID
function generateID(){
    return Math.floor(Math.random()*100000000);
}

function addTransactionDOM(transaction){
    const sign=transaction.amount<0?" ":"+";
    const item=document.createElement("li");

    item.classList.add(
        transaction.amount < 0 ? "minus" : "plus"
    )
}

```

```
)
```

```
item.innerHTML = `
```

```
  ${transaction.text}<span>${sign}${Math.abs(transaction.amount)}</span>
```

```
  <button class="delete-btn" onclick="removetransaction(${transaction.id})">x</button>
```

```
`;
```

```
list.appendChild(item);
```

```
}
```

```
//removetransaction
```

```
function removetransaction(id){
```

```
  transactions=transactions.filter(transaction=>transaction.id !==id);
```

```
  updateLocalStorage();
```

```
  Init();
```

```
}
```

```
//update value
```

```
function updateValues(){
```

```
  const amounts=transactions.map(transaction=>transaction.amount);
```

```
  const total = amounts.reduce((acc, item) => (acc += item),0).toFixed(2);
```

```
  const income = amounts.filter(item => item > 0).reduce((acc, item)=> (acc += item), 0).toFixed(2);
```

```
  const expense = (
```

```
    amounts.filter(item => item < 0).reduce((acc, item) =>(acc += item),0) * -1
```

```
  ).toFixed(2);
```

```
  balance.innerText=`$${total}`;
```

```
  money_plus.innerText=`$${income}`;
```

```
    money_minus.innerText=`${expense}`;
}

//update local storage
function updateLocalStorage(){
    localStorage.setItem(
        "transactions",JSON.stringify(transactions)
    );
}

//Init App
function Init(){
    list.innerHTML = "";
    transactions.forEach(addTransactionDOM);
    updateValues();
}

Init();

form.addEventListener("submit", addTransaction);
```