

Index.html

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

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

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

  <title>Money Tracker WebApp</title>

</head>

<body>

  <h1> Money Tracker Web App</h1>

  <div class="input-section">

    <label for="category-select">Category:</label>

    <select id="category-select">

      <option value="College Fee">College Fee</option>

      <option value="Rent">Rent</option>

      <option value="Transport">Transport</option>

      <option value="Food">Food</option>

      <option value="Shopping">Shopping</option>

      <option value="Cool Drinks">Cool Drinks</option>

    </select>

    <label for="amount-input">Amount:</label>

    <input type="number" id="amount-input">

    <label for="date-input">Date:</label>

    <input type="date" id="date-input">

    <button id="add-btn">Add</button>

  </div>

  <div class="expenses-list">

    <h2>Expenses List</h2>
```

```
<table>

  <thead>

    <tr>

      <th>Category</th>

      <th>Amount</th>

      <th>Date</th>

      <th>Delete</th>

    </tr>

  </thead>

  <tbody id="expense-table-body">

  </tbody>

  <tfoot>

    <td>Total:</td>

    <td id="total-amount"></td>

    <td></td>

    <td></td>

  </tfoot>

</table>

</div>

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

</body>

</html>
```

Script.js

```
Let expenses =[];
```

```
Let totalAmount = 0;
```

```
Const categorySelect = document.getElementById('category-select')
```

```
Const amountInput = document.getElementById('amount-input')
```

```
Const dateInput = document.getElementById('date-input')
```

```
Const addBtn = document.getElementById('add-btn')
```

```
Const expenseTableBody = document.getElementById('expense-table-body')
```

```
Const totalAmountCell = document.getElementById('total-amount')
```

```
addBtn.addEventListener('click', function(){
```

```
    const category = categorySelect.value;
```

```
    const amount = Number(amountInput.value);
```

```
    const date= dateInput.value;
```

```
    if(category === ""){
```

```
        alert('please select a category');
```

```
        return;
```

```
    }
```

```
    If(isNaN(amount) || amount <= 0){
```

```
        Alert('please enter a valid amount');
```

```
        Return;
```

```
    }
```

```
    If(date === ""){
```

```
        Alert('please select a date');
```

```
        Return;
```

```
    }
```

```
    Expenses.push({category,amount,date})
```

```
    totalAmount += amount;
```

```

totalAmountCell.textContent=totalAmount;

const newRow = expenseTableBody.insertRow();

const categoryCell = newRow.insertCell();
const AmountCell = newRow.insertCell();
const dateCell = newRow.insertCell();
const deleteCell = newRow.insertCell();

const deleteBtn = document.createElement('button');

deleteBtn.textContent = 'Delete';
deleteBtn.classList.add('delete-btn');
deleteBtn.addEventListener('click', function(){
    expenses.splice(expenses.indexOf(expense),1);

    totalAmount -= expense.amount;
    totalAmountCell.textContent = totalAmount;

    expenseTableBody.removeChild(newRow);
})

Const expense= expenses[expenses.length - 1];
categoryCell.textContent=expense.category;
AmountCell.textContent=expense.amount;
dateCell.textContent=expense.date;
deleteCell.appendChild(deleteBtn);

});

For(const exexpense of expenses){

```

```

totalAmount += amount;
totalAmountCell.textContent=totalAmount;

const newRow = expenseTableBody.insertRow();

const categoryCell = newRow.insertCell();
const AmountCell = newRow.insertCell();
const dateCell = newRow.insertCell();
const deleteCell = newRow.insertCell();

const deleteBtn = document.createElement('button');

deleteBtn.textContent = 'Delete';
deleteBtn.classList.add('delete-btn');
deleteBtn.addEventListener('click', function(){
    expenses.splice(expenses.indexOf(expense),1);

    totalAmount -= expense.amount;
    totalAmountCell.textContent = totalAmount;

    expenseTableBody.removeChild(newRow);
})
Const expense= expenses[expenses.length - 1];
categoryCell.textContent=expense.category;
AmountCell.textContent=expense.amount;
dateCell.textContent=expense.date;
deleteCell.appendChild(deleteBtn);
}

```

Style.css

Body{

Font-family: Arial, Helvetica, sans-serif;

Margin:0;

}

H1,h2{

Text-align: center;

}

.input-section{

Display: flex;

Flex-direction: row;

Justify-content: space-between;

Align-items: center;

Padding: 10px;

}

.input-section label{

Font-weight: bold;

Margin-right: 10px;

}

.input-section input[type="number"], .input-section input[type="date"]{

Padding: 5px;

Margin-right: 10px;

}

.input-section button{

Padding: 5px 10px;

Background-color: rgb(238, 9, 9);

Color:white;

Border: none;

Border-radius: 4px;

```
    Cursor: pointer;
}
```

```
.expenses-list{
    Margin: 20px;
}
```

```
Table{
    Width: 100%;
    Border-collapse: collapse;
}
```

```
Th,td{
    Border: 1px solid #ddd;
    Padding: 8px;
    Text-align: left;
}
```

```
Th{
    Background-color: rgb(215, 222, 38);
    Color: rgb(12, 1, 1);
}
```

```
Tfoot td{
    Font-weight: bold;
}
```

```
.delete-btn{
    Padding: 5px 10px;
    Background-color: rgb(238, 9, 9);
    Color:white;
    Border: none;
    Border-radius: 4px;
```

```
    Cursor: pointer;  
}
```

Index.js

```
Var express = require("express")
```

```
Var bodyParser = require("body-parser")
```

```
Var mongoose = require("mongoose")
```

```
Const app= express()
```

```
App.use(bodyParser.json())
```

```
App.use(express.static('public'))
```

```
App.use(bodyParser.urlencoded({
```

```
    Extended: true
```

```
}))
```

```
Mongoose.connect('mongodb://localhost:27017/MoneyList')
```

```
Var db = mongoose.connection
```

```
Db.on('error', ()=> console.log("Error in connecting to the Database"))
```

```
Db.once('open', () => console.log("Connected to Database"))
```

```
App.post("/add", (req,res) =>{
```

```
    Var category_select = req.body.category_select
```

```
    Var amount_input= req.body.amount_input
```

```
    Var info = req.body.info
```

```
    Var date_input = req.body.date_input
```

```
    Var data={
```

```
        "Category": category_select,
```

```
        "Amount" : amount_input,
```



```
    "Info": info,
    "Date": date_input
  }
  Db.collection('users').insertOne(data, (err, collection) => {
    if(err){
      Throw err;
    }
    Console.log("Record Inserted Successfully")

  })
})
App.get("/", (req, res) =>{
  Res.set({
    "Allow-access-Allow-Origin": '*'
  })
  Return res.redirect('index.html')
}).listen(5000)

Console.log("Listening on port 5000")
```