

# Team Budget Planner Project

## GitHub Link:

<https://github.com/Karthiga-web/TeamBudgetPlanner.git> is my GitHub link for the Restaurant Website Bootstrap Project.

## Algorithm:

1. This project has 2 pages, Finance team page and Program manager's page.
2. In Program Manager's page, the program manager adds new deals that he does and adds the cost of those deals and saves the form.
3. The HTML and JavaScript are used to handle these data and local storage is used to save the data in local storage.
4. Whenever the manager adds new data, it is also added to the local storage.
5. The Finance team view has a table and Get Budget button.
6. The Table shows all the deals that are added by the managers.
7. The Get Budget button when clicked shows the whole total budget of all deals.
8. It also shows individual budgets of each program managers in the form of a table.
9. The functionalities are handled by JavaScript and alignment is done by CSS and components are built using HTML.

## Code:

```

JS create_table.js X  <> index.html  JS finance.js  <> finance.html
JS create_table.js > AddNewDeal
2  var myData = [
3      {"Id" : 0, "ClientName" : "Microsoft", "DealName" : "Apollo Project", "ProgramManager" : "Mary", "Cost" : 1000},
4      {"Id" : 1, "ClientName" : "Intel", "DealName" : "Hermes Project", "ProgramManager" : "Bob", "Cost" : 10000},
5      {"Id" : 2, "ClientName" : "Apple", "DealName" : "Zeus Project", "ProgramManager" : "Jane", "Cost" : 100000}
6  ]
7  var currentDealId = myData.length;
8  localStorage.setItem("myData", JSON.stringify(myData))
9  var myDataTest = localStorage.getItem("myData")
10 AddNewDeal = function(e) {
11     var clientName = document.getElementById("clientNameInput").value;
12     var projectName = document.getElementById("projectNameInput").value;
13     var projectManager = document.getElementById("projectManagerInput").value;
14     var projectCost = document.getElementById("projectCostInput").value;
15     document.getElementById("clientNameInput").value = "";
16     document.getElementById("projectNameInput").value = "";
17     document.getElementById("projectManagerInput").value = "";
18     document.getElementById("projectCostInput").value = "";
19     InsertRow(currentDealId, clientName, projectName, projectManager, parseInt(projectCost));
20 };
21 function InsertRow(dealId, clientName, projectName, projectManager, projectCost) {
22     myData.push({"Id": dealId, "ClientName" : clientName, "DealName" : projectName, "ProgramManager" : projectManager, "Cost" : projectCost})
23     currentDealId++;
24     localStorage.clear();
25     localStorage.setItem("myData", JSON.stringify(myData))
26     CreateTableFromJSON();
27 }
28

```

JS create\_table.js

&lt;&gt; index.html

JS finance.js

&lt;&gt; finance.html X

&lt;&gt; finance.html &gt; html

```
1  <!DOCTYPE html>
2  <html>
3
4  <head>
5      <title>Finance Team View</title>
6      <style>
7          th,
8          td,
9          p,
10         input {
11             font: 14px Verdana;
12         }
13
14         table,
15         th,
16         td {
17             border: solid 1px #DDD;
18             border-collapse: collapse;
19             padding: 2px 3px;
20             text-align: center;
21         }
22
23         th {
24             font-weight: bold;
25         }
26     </style>
27     <script src="finance.js"></script>
28 </head>
29
30 <body onload="CreateTableFromJSON()">
31     <p id="showData"></p>
32     <br>
33     <input type="button" onclick="somemethod()" id="method" value = "Get Budget"/><br>
34     <p><h4 id="budget"></h4></p>
35     <br>
36     <p id="individualData"></p>
37 </body>
```

```
1  var myDataTest = JSON.parse(localStorage.getItem("myData"))
2  function CreateTableFromJSON() {
3      let col = [];
4      for (let i = 0; i < myDataTest.length; i++) {
5          for (let key in myDataTest[i]) {
6              if (col.indexOf(key) === -1) {
7                  col.push(key);
8              }
9          }
10     }
11     let table = document.createElement("table");
12     let tr = table.insertRow(-1);
13     for (let i = 0; i < col.length; i++) {
14         let th = document.createElement("th");
15         th.innerHTML = col[i];
16         tr.appendChild(th);
17     }
18     for (let i = 0; i < myDataTest.length; i++) {
19         tr = table.insertRow(-1);
20         for (let j = 0; j < col.length; j++) {
21             let tabCell = tr.insertCell(-1);
22             tabCell.innerHTML = myDataTest[i][col[j]];
23         }
24     }
25     let divContainer = document.getElementById("showData");
26     divContainer.innerHTML = "";
27     divContainer.appendChild(table);
28 }
29 function somemethod() {
30     let a = 0;
31     for (let i = 0; i < myDataTest.length; i++) {
32         a += parseInt(myDataTest[i].Cost);
33     }
34     let para = document.createElement("h3");
35     para.innerHTML = "Total Budget: " + a;
36     document.getElementById("budget").appendChild(para);
37     function onlyUnique(value, index, self) {
38         return self.indexOf(value) === index;
39     }
40     let array = new Array(myDataTest.length);
41     for (let j = 0; j < myDataTest.length; j++) {
42         array[j] = myDataTest[j];
43     }
44     let array1 = new Array(myDataTest.length);
45     for (let j = 0; j < myDataTest.length; j++) {
46         array1[j] = myDataTest[j].ProgramManager;
47     }
```

JS finance.js &gt; CreateTableFromJSON

```
34 let para = document.createElement("h3");
35 para.innerHTML = "Total Budget: " + a;
36 document.getElementById("budget").appendChild(para);
37 function onlyUnique(value, index, self) {
38     return self.indexOf(value) === index;
39 }
40 let array = new Array(myDataTest.length);
41 for (let j = 0; j < myDataTest.length; j++) {
42     array[j] = myDataTest[j];
43 }
44 let array1 = new Array(myDataTest.length);
45 for (let j = 0; j < myDataTest.length; j++) {
46     array1[j] = myDataTest[j].ProgramManager;
47 }
48 let uniquearray = array1.filter(onlyUnique);
49 let newarray = new Array(uniquearray.length);
50 for (let k = 0; k < uniquearray.length; k++) {
51     for (let l = 0; l < array.length; l++) {
52         if (array[l].ProgramManager == uniquearray[k]) {
53             if (newarray[k] == null) {
54                 newarray[k] = array[l].Cost;
55             } else {
56                 newarray[k] += array[l].Cost
57             }
58         }
59     }
60 }
61 let col = [];
62 col.push("Team");
63 col.push("Budget");
64 let table = document.createElement("table");
65 let tr = table.insertRow(-1);
66 for (let i = 0; i < col.length; i++) {
67     let th = document.createElement("th");
68     th.innerHTML = col[i];
69     tr.appendChild(th);
70 }
71 for (let i = 0; i < uniquearray.length; i++) {
72     tr = table.insertRow(-1);
73     var tabCell = tr.insertCell(-1);
74     tabCell.innerHTML = uniquearray[i];
75     var tabCell = tr.insertCell(-1);
76     tabCell.innerHTML = newarray[i];
77 }
78 let divContainer = document.getElementById("individualData");
79 divContainer.innerHTML = "";
80 divContainer.appendChild(table);
81 }
```



JS create\_table.js

&lt;&gt; index.html X

JS finance.js

&lt;&gt; finance.html

&lt;&gt; index.html &gt; html &gt; head &gt; style &gt; table

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Program Manager View</title>
5      <style>
6          th,
7          td,
8          p,
9          input {
10             font: 14px Verdana;
11         }
12
13         table,
14         th,
15         td {
16             border: solid 1px #DDD;
17             border-collapse: collapse;
18             padding: 2px 3px;
19             text-align: center;
20         }
21
22         th {
23             font-weight: bold;
24         }
25     </style>
26     <script src="create_table.js"></script>
27 </head>
28 <body>
29     <H3>Add New Deal</H3>
30     <form>
31         <table style="width: 50%">
32             <tr>
33                 <th><label>Client Name</label></th>
34                 <td><input type="text" id="clientNameInput" required></td>
35             </tr>
36             <tr>
37                 <th><label>Project Name</label></th>
38                 <td><input type="text" id="projectNameInput" required="true" /></td>
39             </tr>
40             <tr>
41                 <th><label>Project Manager</label></th>
42                 <td><input type="text" id="projectManagerInput" required="true" /></td>
43             </tr>
44             <tr>
45                 <th><label>Project Cost</label></th>
46                 <td><input type="number" id="projectCostInput" required="true" /></td>
47             </tr>
```

```
JS create_table.js X index.html X JS finance.js finance.html
index.html > html > head > style > table
9      input {
10         font: 14px Verdana;
11     }
12
13     table,
14     th,
15     td {
16         border: solid 1px #DDD;
17         border-collapse: collapse;
18         padding: 2px 3px;
19         text-align: center;
20     }
21
22     th {
23         font-weight: bold;
24     }
25 </style>
26 <script src="create_table.js"></script>
27 </head>
28 <body>
29     <H3>Add New Deal</H3>
30     <form>
31         <table style="width: 50%">
32             <tr>
33                 <th><label>Client Name</label></th>
34                 <td><input type="text" id="clientNameInput" required></td>
35             </tr>
36             <tr>
37                 <th><label>Project Name</label></th>
38                 <td><input type="text" id="projectNameInput" required="true" /></td>
39             </tr>
40             <tr>
41                 <th><label>Project Manager</label></th>
42                 <td><input type="text" id="projectManagerInput" required="true" /></td>
43             </tr>
44             <tr>
45                 <th><label>Project Cost</label></th>
46                 <td><input type="number" id="projectCostInput" required="true" /></td>
47             </tr>
48         </table>
49         <br>
50         <input type="submit" onclick="AddNewDeal()" value="Save" />
51     </form>
52 </body>
53 </html>
```

Outputs:

Program Manager View

Finance Team View

+

←

→

↺

🔍 File | C:/Users/grkar/OneDrive/Documents/TeamBudgetPlanner/team-budget-planner/finance.html

🔖

🔍

🔗

👤

⋮

Id	ClientName	DealName	ProgramManager	Cost
0	Microsoft	Apollo Project	Mary	1000
1	Intel	Hermes Project	Bob	10000
2	Apple	Zeus Project	Jane	100000
3	Apple	B Project	Mary	3000
4	A	B	C	1000
5	1	2	3	1000
6	1	2	3	2000
7	sc	sd	sdv	234

Get Budget

Program Manager View

Finance Team View

+

←

→

↺

🔍 File | C:/Users/grkar/OneDrive/Documents/TeamBudgetPlanner/team-budget-planner/finance.html

🔖

🔍

🔗

👤

⋮

Id	ClientName	DealName	ProgramManager	Cost
0	Microsoft	Apollo Project	Mary	1000
1	Intel	Hermes Project	Bob	10000
2	Apple	Zeus Project	Jane	100000
3	Apple	B Project	Mary	3000
4	A	B	C	1000
5	1	2	3	1000
6	1	2	3	2000
7	sc	sd	sdv	234

Get Budget

Total Budget: 118234

Team	Budget
Mary	4000
Bob	10000
Jane	100000
C	1000
3	3000
sdv	234



Add New Deal

Client Name		
Project Name		
Project Manager		
Project Cost		

Save

Add New Deal

Client Name		Apple
Project Name		B Project
Project Manager		Mary
Project Cost		3000

Save

Program Manager View

File

C:/Users/grkar/OneDrive/Documents/TeamBudgetPlanner/team-budget-planner/index.html?

Save

Print

Share

Profile

More

Add New Deal

Client Name		A	
Project Name		B	
Project Manager		C	
Project Cost		1000	

Save

Program Manager View

File

C:/Users/grkar/OneDrive/Documents/TeamBudgetPlanner/team-budget-planner/index.html?

Save

Print

Share

Profile

More

Add New Deal

Client Name		1	
Project Name		2	
Project Manager		3	
Project Cost		1000	

Save

Add New Deal

Client Name		1	
Project Name		2	
Project Manager		3	
Project Cost		2000	

Save

Add New Deal

Client Name		sc	
Project Name		sd	
Project Manager		sdv	
Project Cost		234	

Save