Documentation for Financial Transactions HTML Page

Jason N.

April 26, 2020

Contents

1 HTML

1.1 Preamble and head

```
1 <!DOCTYPE html>
```

This line declares that the document is an HTML5 document.

```
2 <head>
3     <meta charset = "UTF-8"/>
4     <link rel="stylesheet" type="text/css" href="./style.css"/>
5     <script src="./script.js"></script>
6 </head>
```

`<head>` tags are used to contain meta information about the document. Within the `head` element:

- The first line defines the character set of the document.
- The second line defines the source of an external CSS document.
- The third line defines the source of an external Javascript document.

1.2 Input Fields

1.2.1 Date

```
11 <section>
12     <label for="date">Date:</label><br/>
13     <input id="date" name="date" type="date" required/>
14 </section>
```

1.2.2 Text

```
16 <section>
17     <label for="account">Account Number:</label><br/>
18     <input id="account" name="account" type="text" placeholder="Account
           ↪ Number" required/>
19 </section>
```

1.2.3 List

```
21 <section>
22     <label for="type">Transaction Type:</label><br/>
23     <select id="type" name="type">
24         <option value=""></option>
25         <option value="BUY">BUY</option>
26         <option value="SELL">SELL</option>
27         <option value="DIVIDEND">DIVIDEND</option>
28         <option value="INTEREST">INTEREST</option>
29         <option value="WITHDRAW">WITHDRAW</option>
30         <option value="DEPOSIT">DEPOSIT</option>
31     </select>
32 </section>
```

1.3 Table

2 Javascript

2.1 getData()

2.2 validate()

2.3 generateId()

2.4 calculateCostBasis()

2.5 addTransaction()

2.6 deleteRow()

2.7 editRow()

2.8 saveChanges()

2.9 discardChanges()

2.10 sortTable()

3 CSS

3.1 Vertical Scrolling Table

3.2 Horizontal Scrolling on Overflow

3.3 Miscellaneous

A HTML Source

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset = "UTF-8"/>
        <link rel="stylesheet" type="text/css" href="./style.css"/>
        <script src="./script.js"></script>
    </head>
    <body>
        <article id="inputFields">
            <form onsubmit="return false" autocomplete="off">
                <section>
                    <label for="date">Date:</label><br/>
                    <input id="date" name="date" type="date" required/>
                </section>

                <section>
                    <label for="account">Account Number:</label><br/>
                    <input id="account" name="account" type="text"
                        ↪ placeholder="Account Number" required/>
                </section>

                <section>
                    <label for="type">Transaction Type:</label><br/>
                    <select id="type" name="type">
                        <option value=""></option>
                        <option value="BUY">BUY</option>
                        <option value="SELL">SELL</option>
                        <option value="DIVIDEND">DIVIDEND</option>
                        <option value="INTEREST">INTEREST</option>
                        <option value="WITHDRAW">WITHDRAW</option>
                        <option value="DEPOSIT">DEPOSIT</option>
                    </select>
                </section>

                <section>
                    <label for="security">Security:</label><br/>
                    <input id="security" name="security" type="text"
                        ↪ placeholder="Security" required/>
                </section>

                <section>
                    <label for="amount">Amount:</label><br/>
                    <input id="amount" name="amount" type="text"
                        ↪ placeholder="Unit Amount" required/>
                </section>

                <section>
                    <label for="dAmount">$ Amount:</label><br/>
                    <input id="dAmount" name="dAmount" type="text"
                        ↪ placeholder="$ Amount" required/>
                </section>

```

```
49                  <section>
50                      <button id="add" type="submit" onclick="
                            ↪ addTransactionButton();">Add Transaction</button
                            ↪ >
51                      <button id="save" type="submit" hidden="true" onclick
                            ↪ ="saveChanges();">Save</button>
52                      <button id="discard" type="button" hidden="true"
                            ↪ onclick="discardChanges();">Discard</button>
53                  </section>
54              <form>
55          </article>
56
57          <article id="table">
58              <table>
59                  <thead>
60                      <tr>
61                          <th>
62                              <section>
63                                  Transaction ID
64                              </section>
65                              <section class="sort">
66                                  <button type=button onclick="sortTable(0,
                                        ↪ true)">^</button>
67                                  <button type=button onclick="sortTable(0,
                                        ↪ false)">v</button>
68                              </section>
69                          </th>
70                          <th>
71                              <section>
72                                  Date
73                              </section>
74                              <section class="sort">
75                                  <button type=button onclick="sortTable(1,
                                        ↪ true)">^</button>
76                                  <button type=button onclick="sortTable(1,
                                        ↪ false)">v</button>
77                              </section>
78                          </th>
79                          <th>
80                              <section>
81                                  Account Number
82                              </section>
83                              <section class="sort">
84                                  <button type=button onclick="sortTable(2,
                                        ↪ true)">^</button>
85                                  <button type=button onclick="sortTable(2,
                                        ↪ false)">v</button>
86                              </section>
87                          </th>
88                          <th>
89                              <section>
90                                  Transaction Type
91                              </section>
92                              <section class="sort">
```

```
93                                  <button type=button onclick="sortTable(3,
                                        ↪ true)">^</button>
94                                  <button type=button onclick="sortTable(3,
                                        ↪ false)">v</button>
95                              </section>
96                          </th>
97                          <th>
98                              <section>
99                                  Security
100                             </section>
101                             <section class="sort">
102                                 <button type=button onclick="sortTable(4,
                                        ↪ true)">^</button>
103                                 <button type=button onclick="sortTable(4,
                                        ↪ false)">v</button>
104                             </section>
105                         </th>
106                         <th>
107                             <section>
108                                 Amount
109                             </section>
110                             <section class="sort">
111                                 <button type=button onclick="sortTable(5,
                                        ↪ true)">^</button>
112                                 <button type=button onclick="sortTable(5,
                                        ↪ false)">v</button>
113                             </section>
114                         </th>
115                         <th>
116                             <section>
117                                 $ Amount
118                             </section>
119                             <section class="sort">
120                                 <button type=button onclick="sortTable(6,
                                        ↪ true)">^</button>
121                                 <button type=button onclick="sortTable(6,
                                        ↪ false)">v</button>
122                             </section>
123                         </th>
124                         <th>
125                             <section>
126                                 Cost Basis
127                             </section>
128                             <section class="sort">
129                                 <button type=button onclick="sortTable(7,
                                        ↪ true)">^</button>
130                                 <button type=button onclick="sortTable(7,
                                        ↪ false)">v</button>
131                             </section>
132                         </th>
133                         <th>Actions</th>
134                     </tr>
135                 </thead>
136                 <tbody id="tableBody">
```

```
                </tbody>
            </table>
        </article>
    </body>
</html>
```

B Javascript Source

```
function getData() {
    var date = document.getElementById("date").value;
    var account = document.getElementById("account").value;
    var type = document.getElementById("type").value;
    var security = document.getElementById("security").value;
    var amount = document.getElementById("amount").value;
    var dAmount = document.getElementById("dAmount").value;

    amount = Number(amount);

    if(dAmount[0] == '$') {
        dAmount = dAmount.substr(1);
    }
    dAmount = Number(dAmount);

    if(validate(date, account, type, security, amount, dAmount)) {
        var costBasis = calculateCostBasis(amount, dAmount);
        dAmount = '$' + dAmount.toFixed(2);

        return [ date, account, type, security, amount, dAmount, costBasis
            ↪ ];
    }
    else return false;
}

function validate(date, account, type, security, amount, dAmount) {
    if(!validateDate(date)) return false;
    if(!validateAccount(account)) return false;
    if(!validateType(type)) return false;
    if(!validateSecurity(security)) return false;
    if(!validateAmount(amount)) return false;
    if(!validateDAmount(dAmount)) return false;

    return true;
}

function validateDate(date) {
    realDate = new Date();
    inputDate = document.getElementById('date').valueAsNumber;

    if(isNaN(inputDate)) {
        alert('Error: No date specified');
        return false;
    }

    if(realDate.valueOf() < inputDate) {
        alert('Error: Date is in the future');
        return false;
    }

    return true;
}
```

```
52
53  function validateAccount(account) {
54      if(account == '') {
55          alert('Error: Missing Account Number');
56          return false;
57      }
58
59      return true;
60  }
61
62  function validateType(type) {
63      if(type == '') {
64          alert('Error: Missing Transaction Type');
65          return false;
66      }
67
68      return true;
69  }
70
71  function validateSecurity(security) {
72      if(security == '') {
73          alert('Error: Missing Security');
74          return false;
75      }
76
77      return true;
78  }
79
80  function validateAmount(amount) {
81      if(amount == '') {
82          alert('Error: Missing Amount');
83          return false;
84      }
85
86      if(isNaN(amount)) {
87          alert('Error: Invalid Amount');
88          return false;
89      }
90
91      return true;
92  }
93
94  function validateDAmount(dAmount) {
95      if(dAmount == '') {
96          alert('Error: Missing $ Amount');
97          return false;
98      }
99
100     if(isNaN(dAmount)) {
101         alert('Error: Invalid $ Amount');
102         return false;
103     }
104
105     return true;
```

```
}

function generateId() {
    var id = '';
    var idLength = 6;

    var characters = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789';
    var charactersLength = characters.length;

    var unique = false;

    while(!unique) {
        for(var i = 0; i < idLength; i++) {
            id += characters.charAt(Math.floor(Math.random() *
               ↪ charactersLength));
        }

        unique = true;
        for(var i = 0; i < document.getElementsByClassName('idCell').
           ↪ length; i++) {
            if(document.getElementsByClassName('idCell')[i].innerText ==
               ↪ id) {
                unique = false;
                break;
            }
        }
    }
    return id;
}

function calculateCostBasis(amount, dAmount) {
    costBasis = '$' + (dAmount / amount).toFixed(2);
    return costBasis;
}

function addTransaction(id, date, account, type, security, amount, dAmount
   ↪ , costBasis) {
    var tableBody = document.getElementById('tableBody');
    var newRow = tableBody.insertRow(0);
    newRow.classList += "bodyRow";

    var actionsContent = "<button type='button' onclick='editRow(this)'>
       ↪ Edit</button> <button type='button' onclick='deleteRow(this)'>
       ↪ Delete</button>";
    var rowContents = [id, date, account, type, security, amount, dAmount,
       ↪  costBasis, actionsContent];

    for(var i = 0; i < rowContents.length; i++) {
        var newCell = newRow.insertCell(i);
        newCell.innerHTML = rowContents[i];
        if(i == 0) {
            newCell.classList += "idCell";
        }
    }
```

```
153 }
154
155 function addTransactionButton() {
156     var data = getData();
157     if(data) {
158         var date = data[0];
159         var account = data[1];
160         var type = data[2];
161         var security = data[3];
162         var amount = data[4];
163         var dAmount = data[5];
164         var costBasis = data[6];
165
166         var id = generateId();
167
168         addTransaction(id, date, account, type, security, amount, dAmount,
                ↪  costBasis);
169     }
170 }
171
172 function deleteRow(button) {
173     var row = button.parentElement.parentElement;
174     document.getElementById("tableBody").removeChild(row);
175
176     if(document.getElementsByClassName('editing').length == 0) {
177         document.getElementById('add').removeAttribute('hidden');
178         document.getElementById('save').setAttribute('hidden', true);
179         document.getElementById('discard').setAttribute('hidden', true);
180     }
181 }
182
183 function editRow(button) {
184     if(document.getElementsByClassName('editing').length > 0)
185         document.getElementsByClassName('editing')[0].classList = "bodyRow
                ↪ ";
186
187     var row = button.parentElement.parentElement;
188     var rowContent = row.getElementsByTagName('td');
189     row.classList = "bodyRow editing";
190
191     document.getElementById('date').value = rowContent[1].innerText;
192     document.getElementById('account').value = rowContent[2].innerText;
193     document.getElementById('type').value = rowContent[3].innerText;
194     document.getElementById('security').value = rowContent[4].innerText;
195     document.getElementById('amount').value = rowContent[5].innerText;
196     document.getElementById('dAmount').value = rowContent[6].innerText;
197
198     document.getElementById('add').setAttribute('hidden', true);
199     document.getElementById('save').removeAttribute('hidden');
200     document.getElementById('discard').removeAttribute('hidden');
201 }
202
203 function saveChanges() {
204     data = getData();
```

```
205     if(data) {
206         rowToEdit = document.getElementsByClassName('editing')[0];
207         cellsToEdit = rowToEdit.getElementsByTagName('td');
208 
209         for(var i = 0; i < data.length; i++) {
210             cellsToEdit[i + 1].innerHTML = data[i];
211         }
212         rowToEdit.classList = "bodyRow";
213     }
214 
215     document.getElementById('add').removeAttribute('hidden');
216     document.getElementById('save').setAttribute('hidden', true);
217     document.getElementById('discard').setAttribute('hidden', true);
218 }
219 
220 function discardChanges() {
221     document.getElementsByClassName('editing')[0].classList = "bodyRow";
222 
223     document.getElementById('add').removeAttribute('hidden');
224     document.getElementById('save').setAttribute('hidden', true);
225     document.getElementById('discard').setAttribute('hidden', true);
226 }
227 
228 function sortTable(column, ascending) {
229     var tableBody = document.getElementById('tableBody');
230     var rows = document.getElementsByClassName('bodyRow');
231 
232     var sorting = true;
233     while(sorting) {
234         sorting = false;
235         for(var i = 0; i < (rows.length - 1); i++) {
236             rowA = rows[i].getElementsByTagName('td')[column];
237             rowB = rows[i + 1].getElementsByTagName('td')[column];
238 
239             var swap = false;
240 
241             if(ascending && rowA.innerHTML.toLowerCase() > rowB.innerHTML.
                    ↪ toLowerCase()) swap = true;
242             else if(!ascending && rowA.innerHTML.toLowerCase() < rowB.
                    ↪ innerHTML.toLowerCase()) swap = true;
243 
244             if(swap) {
245                 sorting = true;
246                 rows[i].parentNode.insertBefore(rows[i + 1], rows[i]);
247             }
248         }
249     }
250 }
```

C CSS Source

```
body {
    font-size: 14px;
}

#inputFields {
    padding: 10px 0;
    overflow-x: auto;
}

form {
    min-width: 1900px;
}

    form > section {
        width: 14%;
        display: inline-block;
    }

    input,
    select {
        min-width: 100px;
        width: 80%;
    }

    button {
        width: 40%;
    }

#table {
    max-height: 80vh;
    overflow: auto;
}

table {
    width: 100%;
    margin: auto;
    border-collapse: collapse;
}
    th {
        min-width: 200px;
        width: 10%;
        position: sticky;
        background: white;
        top: 0;
    }

    th > section {
        width: 80%;
        display: inline-block;
        padding: 0;
        margin: 0;
    }
```

```
53
54      .sort {
55          width: 10%;
56      }
57
58      .sort > button {
59          padding: 0;
60          border: 0;
61          display: block;
62          width: 100%;
63      }
64
65      .editing {
66          background-color: yellow;
67      }
68
69      #table,
70      table,
71      td,
72      th {
73          box-shadow: 1px 1px black, inset 1px 1px black;
74      }
```