

# Liquid UI: Using Insertion to Sort Items

In this example, the Insertion algorithm will be used to sort items in a table. This article is based off a previous article, "Creating a Class". A link is provided below:

<http://www.guixt.com/forum/index.php?topic=123.0>


Sorting is a common process used in many different scenarios. Insertion is an easy to implement algorithm however is not the most efficient when the data set is very large. The Big O is  $n^2$ .

## User Interface

Log in to SAP and navigate to the transaction VA03.

The screenshot shows the SAP VA03 transaction interface. At the top, there is a menu bar with 'Sales document', 'Edit', 'Goto', 'Environment', 'System', and 'Help'. Below the menu is a toolbar with various icons for navigation and actions. The main title of the screen is 'Display Sales Order: Initial Screen'. Below the title, there are tabs for 'Sales', 'Item overview', 'Ordering party', and 'Orders'. The 'Sales' tab is currently selected. In the center of the screen, there is a text field labeled 'Order' containing the value '6666'. To the right of this field is a 'Display items' button. Below the 'Order' field, there is a 'Search Criteria' section with several input fields: 'Purchase Order No.', 'Sold-to party', 'Delivery', 'Billing Document', and 'WBS Element'. At the bottom of this section is a 'Search' button.

Enter and order number and then click the pushbutton “Display items”. A function will run, fetching the data from the table of the order you entered. Then the data will be displayed to the user on the initial screen along with three buttons. Each button will sort the table according to the selection.

 Sort By Item

Sort By Material

Sort By Quantity

Line Items


Item	Material	Quantity	SU	
10	R100000	21	PC	⋮
20	R100001	2	PC	
30	R100003	3	PC	
40	R100022	3	PC	

< >

⋮

< >

Sort By Item

 Sort By Material

Sort By Quantity

Line Items

Item	Material	Quantity	SU	
10	R100000	21	PC	⋮
20	R100001	2	PC	
30	R100003	3	PC	
40	R100022	3	PC	

< >

⋮

< >

Sort By Item

Sort By Material

✓ Sort By Quantity

Line Items

Item	Material	Quantity	SU	
20	R100001	2	PC	⋮
30	R100003	3	PC	
40	R100022	3	PC	
10	R100000	21	PC	
				^
				v

< > ⋮ < >

## Liquid UI Code [Script]

```

1  //////////////////////////////////////////////////////////////////////////////////////////////////////////////////
2  // Author: Synactive, Inc. [1065 E. Hillsdale Blvd, Foster City, CA, 94404, USA]
3  // Email: support@guixt.com; sales@guixt.com;
4  // Contact: 650.341.3310
5  //////////////////////////////////////////////////////////////////////////////////////////////////////////////////
6
7  // This function creates a class for Item
8  function Item(itm,mat,qty,su){
9      // Attributes of the class
10     this.item = itm;
11     this.material = mat;
12     this.quantity = qty;
13     this.sales_unit = su;
14     // Function used to retrieve information on Item
15     this.getInfo = function(){
16         return "Item:"+this.item+", Material:"+this.material+", Quantity:"+this.quantity+", Sales Unit:"+this.sales_unit;
17     }
18 }
19
20
21
22 // Only execute the following for VA03
23 if(_transaction == "VA03"){
24     // Pushbutton that will trigger the function
25     pushbutton("F[Order]+[0,62]", "Display items", "?", {"process":fetchItems});
26
27     // If item_array is undefined, initialize it
28     if(!item_array){
29         item_array = [];
30     }
31
32     // If the array has data in it
33     if(item_array.length>0){
34
35         pushbutton([15,0], "&V[item_sel]Sort By Item", "?", {"process":changeSort,"using":{"sort_type":"item"}});
36         pushbutton([15,30], "&V[mat_sel]Sort By Material", "?", {"process":changeSort,"using":{"sort_type":"material"}});
37         pushbutton([15,60], "&V[qty_sel]Sort By Quantity", "?", {"process":changeSort,"using":{"sort_type":"quantity"}});
38
39         // Create a table and columns
40         table([17,0], [26,50], {"name":"z_table", "title":"Line Items", "rows":item_array.length});
41         column("Item", {"size":6,"name":"z_item", "table":"z_table"});
42         column("Material", {"size":18,"name":"z_mat", "table":"z_table"});
43         column("Quantity", {"size":15,"name":"z_qty", "table":"z_table"});
44         column("SU", {"size":3,"name":"z_su", "table":"z_table"});

```

```
C:\GuiXT\Tutorials\Forum\SAPMV45A.E0102.sjs - Notepad++
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins Window ?
SAPMV45A.E0102.sjs
46 switch(sorted){
47     // Use insertion sort to sort the array
48     case "material":
49
50         // Sort By Material
51         for(i=1; i<item_array.length; i++){
52
53             while(i>0 && item_array[i-1].material>item_array[i].material){
54                 temp = item_array[i-1];
55                 item_array[i-1] = item_array[i];
56                 item_array[i] = temp;
57                 i--;
58             }
59
60         }
61         break;
62
63     case "quantity":
64
65         // Sort By Quantity
66         for(i=1; i<item_array.length; i++){
67
68             while(i>0 && parseInt(item_array[i-1].quantity)>parseInt(item_array[i].quantity)){
69                 temp = item_array[i-1];
70                 item_array[i-1] = item_array[i];
71                 item_array[i] = temp;
72                 i--;
73             }
74
75         }
76         break;
77
78     case "item":
79
80         // Sort By Item
81         for(i=1; i<item_array.length; i++){
82
83             while(i>0 && parseInt(item_array[i-1].item)>parseInt(item_array[i].item)){
84                 temp = item_array[i-1];
85                 item_array[i-1] = item_array[i];
86                 item_array[i] = temp;
87                 i--;
88             }
89
90         }
91     }
```

```

C:\GuiXT\Tutorials\Forum\SAPMV45A.E0102.sjs - Notepad++
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins Window ?
SAPMV45A.E0102.sjs
77
78     case "item":
79
80         // Sort By Item
81         for(i=1; i<item_array.length; i++){
82
83             while(i>0 && parseInt(item_array[i-1].item)>parseInt(item_array[i].item)){
84                 temp = item_array[i-1];
85                 item_array[i-1] = item_array[i];
86                 item_array[i] = temp;
87                 i--;
88             }
89
90         }
91         break;
92
93     default:
94         // Do nothing, the array is already sorted by Item
95     }
96
97     // Fill out the table
98     for(i=0; i<item_array.length; i++){
99         z_table.z_item[i] = item_array[i].item;
100         z_table.z_mat[i] = item_array[i].material;
101         z_table.z_qty[i] = item_array[i].quantity;
102         z_table.z_su[i] = item_array[i].sales_unit;
103     }
104 }
105
106
107
108
109
110 // This function will table scroll through the VA03 transaction and fetch the data from the table
111 function fetchItems(){
112     onscreen 'SAPMV45A.0102'
113     enter();
114     onerror
115         message(_message);
116     enter("?");
117     goto FUNC_END;
118     onscreen 'SAPMV45A.4001'
119     // Clear the value of item_array
120     item_array = [];
121     absrow = 1;

```

```

C:\GuiXT\Tutorials\Forum\SAPMV45A.E0102.sjs - Notepad++
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins Window ?
SAPMV45A.E0102.sjs
119 // Clear the value of item_array
120 item_array = [];
121 absrow = 1;
122 enter("/ScrollToLine=&V[absrow]", {"table": "T[All items]"});
123
124 NEW_SCREEN:;
125 onscreen 'SAPMV45A.4001'
126 gettableattribute("T[All items]", {"firstvisiblerow": "FVR", "lastvisiblerow": "LVR", "lastrow": "LR"});
127 relrow = 1;
128 NEW_ROW:;
129 println("absrow:"+absrow+", LVR:"+LVR+", LR:"+LR);
130 // end of table?
131 if(absrow>LR){
132     goto END_OF_TABLE;
133 }
134 // end of screen?
135 if(absrow>LVR) {
136     goto NEW_SCREEN;
137 }
138
139 set("V[z_temp_item]", "&cell[All items,Item,&V[relrow]]");
140 set("V[z_temp_mat]", "&cell[All items,Material,&V[relrow]]");
141 set("V[z_temp_qty]", "&cell[All items,Order Quantity,&V[relrow]]");
142 set("V[z_temp_su]", "&cell[All items,SU,&V[relrow]]");
143
144 // Push a new Item to the array
145 item_array.push(new Item(z_temp_item,z_temp_mat,z_temp_qty,z_temp_su));
146
147 absrow++;
148 relrow++;
149 goto NEW_ROW;
150
151 END_OF_TABLE:;
152
153 enter('/ScrollToLine=1', {"table": "T[All items]"});
154
155 onscreen 'SAPMV45A.4001'
156 enter("/3")
157
158 FUNC_END:;
159 }
160
161

```

```
163 function changeSort(param){
164     onscreen 'SAPMV45A.0102'
165     sorted = param.sort_type;
166     if(sorted == "material"){
167         item_sel = "";
168         mat_sel = "@01@";
169         qty_sel = "";
170     }
171     else if(sorted == "quantity"){
172         item_sel = "";
173         mat_sel = "";
174         qty_sel = "@01@";
175     }
176     else if(sorted == "item"){
177         item_sel = "@01@";
178         mat_sel = "";
179         qty_sel = "";
180     }
181     enter("?");
182 }
```