

Bootstrap jQuery Plugin

A simple light weight jquery plugin library to make working with Bootstrap css.

Download jquery.bootstrap.min.js (<https://raw.githubusercontent.com/newghost/bootstrap-jquery->

Fork 52

Star 58

Follow @newghost 38

Download VPN for China



Unblock any Site. Try it Risk Free. 256-Bit SSL. High Speed Guaranteed!



string.format.js

Provide a simple string.format method

```
1.      "{0} is old, {1} is new, {0} long lives".format("ASP", "ASP.net")
2.      //console
3.      "ASP is old, ASP.net is new, ASP long lives"
4.
```

jquery.dialog.js

(<https://raw.githubusercontent.com/newghost/bootstrap-jquery-plugin/master/src/jquery.dialog.js>)

jquery dialog plugin based on bootstrap css

DEMO: This is login form

```
1.      <div id="loginwrap">
2.      <form id="loginform" role="form">
3.          <div class="form-group">
4.              <label for="username">Username</label>
5.              <input type="username" class="form-control"
6.                  id="username">
7.          </div>
8.          <div class="form-group">
9.              <label for="password">Password</label>
10.             <input type="password" class="form-control"
11.                 id="password">
12.          </div>
13.      </form>
14.  </div>
15.
```

create

Create a dialog

Run

```
1.      $("#loginwrap").dialog({title:"Login"});
2.
```

create (with backdrop or keyboard)

Create a dialog which doesn't close on click on the background.

Run

```
1.      $("#loginwrap2").dialog({title:"Login", backdrop: "static"});
2.
```

create without auto open

Create a dialog with (autoOpen or show)

Run

```
1.      $("#loginwrap").dialog({title:"Login", autoOpen: false});
2.
```

open

Open a dialog

Run

```
1.      $("#loginwrap").dialog("open");
2.
```

destroy

Rut the element to the old place, remove the overlay from DOM.

```
1.      $("#loginwrap").dialog("destroy");
2.
```

onClose

Override onClose event

```
1.      $("#loginwrap").dialog({
2.          title:    "DEMO"
3.          , onClose: function() {
4.              // your handler
5.              $(this).dialog("destroy");
6.          }
7.      });
```

option buttons

Redefine the buttons

Run

```
1.      $( "#loginwrap" ).dialog(
2.          "option"
3.          , "buttons"
4.          , [{
5.              text: "Ok"
6.              , click: function() {
7.                  $( this ).dialog( "close" );
8.              }
9.          }]
10.     );
11.
```

buttons: Array; class: Custom Class

Open dialog with buttons

Run

```

1.      $("#loginwrap").dialog({
2.          title:      "Login"
3.          , 'class':  "mydialog" /*add custom class for this dialog*/
4.          , onClose: function() { $(this).dialog("close"); }
5.          , buttons: [
6.              {
7.                  text: "Close"
8.                  , 'class': "btn-primary"
9.                  , click: function() {
10.                      $(this).dialog("close");
11.                  }
12.              },
13.              {
14.                  text: "Login"
15.                  , 'class': "btn-success"
16.                  , click: function() {
17.                      /*your login handler*/
18.
19.                      $(this).dialog("close");
20.                  }
21.              },
22.              {
23.                  text: "Destroy"
24.                  , classed: "btn-error" /*classed equal to 'class'*/
25.                  , click: function() {
26.                      /*your login handler*/
27.
28.                      $(this).dialog("destroy");
29.                  }
30.              }
31.          ]
32.      });
33.

```

buttons: Object

Open dialog with buttons: Object

Run

```

1.      $("#loginwrap").dialog({
2.          title:      "Login"
3.          , 'class':  "mydialog" /*add custom class for this dialog*/
4.          , onClose: function() { $(this).dialog("close"); }
5.          , buttons: {
6.              "Close": function() {
7.                  $(this).dialog("close");
8.              }
9.              , "Login": function() {
10.                  /*your login handler*/
11.
12.                  $(this).dialog("close");
13.              }
14.          }
15.      });
16.

```

dialogClass: Optional sizes

Smaller

Run

```

1.      $("#loginwrap").dialog({
2.          title      : "Login"
3.          , dialogClass : "modal-sm"
4.      });
5.

```

Bigger

Run

```
1.      $("#loginwrap").dialog({
2.          title      : "Login"
3.          , dialogClass : "modal-lg"
4.      });
5.
```

jquery.messenger.js

jquery messenger(message box) plugin based on bootstrap css

alert

popup alert message

simple message

Run

```
1.      $.messenger.alert("This is message!");
2.
```

message with title

Run

```
1.      $.messenger.alert("Title", "This is message!");
2.
```

confirm

confirm message

with callback

Run

```
1.      $.messenger.confirm("title", "This is message!", function() {
2.          alert("you closed it")
3.      });
4.
```

localization, override message

Run

```
1.      $.messenger.model = {
2.          ok:{ text: "关闭", classed: 'btn-default' },
3.          cancel: { text: "取消", classed: 'btn-error' }
4.      };
5.      $.messenger.confirm("title", "This is message!", function() {
6.          alert("you closed it")
7.      });
8.
```

popup

popup message

Run

```
1.      $.messenger.popup("This is message!");
2.
```

custom css, apply and run again

Apply

```
<style>
.modal-open {
  overflow: auto;
  padding: 0 !important;
}

.modal.msg-popup {
  top: auto;
}

.modal.msg-popup .modal-dialog {
  width: 300px !important;
  margin: 30px auto 60px;
}

.modal.msg-popup .modal-body {
  background-color: #000;
  color: #fff;
}

.modal.msg-popup .modal-content {
  border: none;
}
</style>
<script>
$.messenger.popup("Custom layout");
</script>
```

jquery.datagrid.js

jquery datagrid plugin based on bootstrap css

load

load data from json object

html

```
1.      <table id="tablewrap1" class="table table-striped"></table>
2.
```

Name	Type	Number
item12	Free	143400

item13	Free	3426919
item14	Free	3343312
item15	Free	120330
item16	Not Free	319492
item31	Not Free	321338
item13	Unknow	342219
item54	Unknow	231000
item36	Unknow	259451
item34	Unknow	449563

codes

```
1.         var $table = $('#tablewrap1');
2.
3.         var rows =[
4.             {"name":"item12","type":"Free","sum":143400},
5.             {"name":"item13","type":"Free","sum":3426919},
6.             {"name":"item14","type":"Free","sum":3343312},
7.             {"name":"item15","type":"Free","sum":120330},
8.             {"name":"item16","type":"Not Free","sum":319492},
9.             {"name":"item31","type":"Not Free","sum":321338},
10.            {"name":"item13","type":"Unknow","sum":342219},
11.            {"name":"item54","type":"Unknow","sum":231000},
12.            {"name":"item36","type":"Unknow","sum":259451},
13.            {"name":"item34","type":"Unknow","sum":449563}
14.        ];
15.
16.        $table.datagrid({
17.            columns:[[
18.                {title: "Name", field: "name"}
19.                , {title: "Type", field: "type"}
20.                , {title: "Number", field: "sum"}
21.            ]]
22.        }).datagrid("loadData", {rows: rows});
23.
```

edit

edit json data

html

```
1.         <table id="tablewrap2" class="table table-hover"></table>
2.
```

Name	Type	Number
item12	<input type="text" value="Free"/>	<input type="text" value="143400"/>
item13	<input type="text" value="Free"/>	<input type="text" value="3426919"/>
item14	<input type="text" value="Free"/>	<input type="text" value="3343312"/>
item15	<input type="text" value="Free"/>	<input type="text" value="120330"/>
item16	<input type="text" value="Not Free"/>	<input type="text" value="319492"/>

item31	Not Free	321338
item13	Unknow	342219
item54	Unknow	231000
item36	Unknow	259451
item34	Unknow	449563

codes

```

1.      var $editTable = $('#tablewrap2');
2.
3.      $editTable.datagrid({
4.          columns:[[
5.              {title: "Name", field: "name", formatter: function(val) {
6.                  //override default input field
7.                  return val;
8.              }}
9.          , {title: "Type", field: "type", readonly: true}
10.         , {title: "Number", field: "sum"}
11.     ]]}
12.     , edit: true
13.     , singleSelect: true //false allow multi select
14.     , selectedClass: 'danger' //default: 'success'
15.     , selectChange: function(selected, rowIndex, rowData, $row) {
16.         //allow multi-select
17.         //console.log(selected, rowIndex, rowData, $row);
18.     }
19.     }).datagrid("loadData", {rows: rows});
20.

```

updateRow(index) Run

```

1.      $('#tablewrap2').datagrid("updateRow", {index: 0, row: {name: "Good luck", sum: new Date() }});

```

updateRow Run

```

1.      $('#tablewrap2').datagrid("updateRow", {row: {name: "Good luck", sum: new Date() }});

```

selectRow(index) Run

```

1.      $('#tablewrap2').datagrid("selectRow", 1);

```

selectRow Run

```

1.      $('#tablewrap2').datagrid("selectRow");

```

unselectRow(index) Run

```

1.      $('#tablewrap2').datagrid("unselectRow", 1);

```

unselectRow Run

```

1.      $('#tablewrap2').datagrid("unselectRow");

```

getSelections Run


```
1. $('#tablewrap2').datagrid("getSelections");
```

getSelectedIndex Run

```
1. $('#tablewrap2').datagrid("getSelectedIndex");
```

getData Run

```
1. $('#tablewrap2').datagrid("getData");
```

getConfig Run

```
1. $('#tablewrap2').datagrid("getConfig");
```

getColumns Run

```
1. $('#tablewrap2').datagrid("getColumns");
```

insertRow(index) Run

```
1. $('#tablewrap2').datagrid("insertRow", {index: 3, row: {"name":"new" + (Math.random() * 100 | 0)}});
```

insertRow() Run

```
1. $('#tablewrap2').datagrid("insertRow", {row: {"name":"new" + (Math.random() * 100 | 0)}});
```

deleteRow(index) Run

```
1. $('#tablewrap2').datagrid("deleteRow", 0);
```






deleteRow() Run

```
1. $('#tablewrap2').datagrid("deleteRow");
```

jquery.tree.js

html

```
1. <ul id="treewrap"></ul>
2.
```

-  Root
 -  Tool
 -  Users
-  Guest
-  Admin

codes

```

1.      var $tree = $('#treewrap');
2.
3.      var treeDataArr = [
4.          {
5.              id:      "root"
6.              , text:   "Root"
7.              , attributes: { yourfield: "your value" }
8.              , children: [
9.                  {
10.                     id:      "tool"
11.                     , text:   "Tool"
12.                 }
13.                , {
14.                    id:      "users"
15.                    , text:   "Users"
16.                    , state:   "close"
17.                    , children: [
18.                        {text: "Kris"}
19.                        , {text: "Tom"}
20.                        , {text: "Jerry"}
21.                        , {text: "Dna"}
22.                    ]
23.                }
24.            ]
25.        }
26.        , {
27.            text:      "Guest"
28.        }
29.        , {
30.            id:      "admin"
31.            , text:   "Admin"
32.        }
33.    ];
34.
35.    $tree.tree({data: treeDataArr});
36.

```

onClick

Run

```

1.      $('#treewrap').tree({
2.          data:      treeDataArr
3.          , onClick:  function(node, $link) {
4.              console.log($(this), node, $link);
5.          }
6.      });
7.

```