

ArtEditableGrid

EditableGrid is an addon for Microsoft Dynamics CRM platform providing user comfort of work with mass information. Static lists of read-only records in views or static lists of read-only records in data, those are attached to detailed form, changes and completes by in-line data editing. Allows simple and data adding and changes, but immediate conditional formatting, validations and calculations too. Functionality can be extended by user wizards and in connection with server components can handle difficult application and business logics. An advantage is possible consolidated update of information from multiple mutually connected entities.

Basic configuration

To make addon functional you need to have these scripts on your form:

- ArtExt.js
- ArtEditableGrid11.js
- Your custom script

Keep scripts in this order.

In Form properties has to be set the OnLoad function. For example:

```
Object.Common.OnLoad
```

Object is name of object in your custom script.

Configuration example (in your custom script)

```
OnLoad: function () {
    formType = Xrm.Page.ui.getFormType();
    if (formType != FORM_TYPE_CREATE) {
        currGUID = Xrm.Page.data.entity.getId();
        var myPreload = new EditableGridPreload(); //Creation of new EditableGridPreload object
        myPreload.onloaded = function () { //Definition of function, that will be held at eGrid
            load
            eGrid = new EditableGrid(); //Creation of new EditableGrid object
                eGrid.viewId = 'egView'; //Data in eGrid will be loaded from view named
                'egView'
            eGrid.foreignKey = "parent_entity"; //Sets the foreign key of master entity
                eGrid.gridPlace = "gridSection"; //Places the eGrid in a specific section on
                form
            eGrid.EditableGridInit();//The EditableGrid initialization
        }
        myPreload.init(); //The EditableGridPreload initialization
    }
}
```

Feature list and other configurations

- Any other configurations have to be placed in body of myPreload.onloaded function
- Data are built by standard CRM view (ID in view note)

```
eGrid.viewId = 'egView';
```

- Making fields read-only

```
eGrid.readonlyFields = []; //Write array of fields to make read-only in []
eGrid.readonlyAll = []; //Makes read-only all fields
```

- Conditional making field read-only (relative to value in any field)

```
eGrid.CellBeforeEdit = function (e, editor) { //Parameter e describes row
    if (e.field == "field") {
        if (e.record.data.name == "AAAA") {
            return false; //If name has not value AAAA, field is read-only
        }
    }
};
```

- Setting hidden fields (fields are in data, but not visible)

```
eGrid.hiddenFields = []; //Do [] vepsat array (pole) skrytých polí
```

- Copying one or more records

Název	Jméno	Příjmení	Datum narození	Prodejce
<input checked="" type="checkbox"/> DrozdJan	Jan	Drozd	01.07.1964	<input checked="" type="checkbox"/> NedorostStanis
<input checked="" type="checkbox"/> SuchýTomáš	Tomáš	Suchý	18.11.1984	<input checked="" type="checkbox"/> NedorostStanis
<input type="checkbox"/> UličnýVáclav	Václav	Uličný	21.05.1974	<input checked="" type="checkbox"/> NedorostStanis

- Renaming fields

```
eGrid.renamefield["field_id"] = "newName";
```

- Filtration, moving columns, width changing and following changes saving to user view – this view can be set as default
- Saving edited view: click “Views” button -> “Add record” -> fill “View name” (check “Default view” if you want to use it as default) -> “Save records”

- Selecting the saved view

Název	Jméno	Příjmení	Datum narození	Prodejce
<input type="checkbox"/> DrozdJan	Jan	Drozd	01.07.1964	<input checked="" type="checkbox"/> NedorostStanis
<input type="checkbox"/> SuchýTomáš	Tomáš	Suchý	18.11.1984	<input checked="" type="checkbox"/> NedorostStanis
<input type="checkbox"/> UličnýVáclav	Václav	Uličný	21.05.1974	<input checked="" type="checkbox"/> NedorostStanis

- Adding record – including default values defined in advance

Název	Jméno	Příjmení	Datum narození	Prodejce
DrozdJan	Jan	Drozd	01.07.1964	NedorostStanislav
SuchýTomáš	Tomáš	Suchý	18.11.1984	NedorostStanislav
UličnýVáclav	Václav	Uličný	21.05.1974	NedorostStanislav

```
eGrid.defaultValues = {}; //Sets default values
```

- Placing eGrid to section on form or to new window

```
eGrid.gridDestination = "section"; //"window" for draw eGrid to new window
```

- All standard buttons can be hidden

```
eGrid.addButton = false; //Hides button for adding records
eGrid.saveButton = false; //Hides button for saving eGrid
eGrid.deleteButton = false; //Hides button for deleting records
eGrid.copyButton = false; //Hides button for copying records
eGrid.saveViewButton = false; //Hides button for saving views
eGrid.filterButton = false; //Hides button for filters
eGrid.viewSelector = false; //Hides button for selecting view
```

- Row actions for selected rows – mass changes

```
eGrid.GetGrid().view.selModel.selected.items //Gets all selected rows
```

- Setting number of decimals for whole eGrid

```
eGrid1.globalPrecision = 2;
```

- Editing all kinds of CRM fields

Editing lookup field

Editing date field

<input type="checkbox"/>	Název	Jméno	Příjmení	Datum narození	Prodejce
<input checked="" type="checkbox"/>	DrozdJan	Jan	Drozd	01.07.1964	NedorostStanis
<input type="checkbox"/>	SuchýTomáš	Tomáš	Suchý	18.11.1984	NedorostStanis
<input type="checkbox"/>	UličnýVáclav	Václav	Uličný	21.05.1974	NedorostStanis

Strana 1 z 1

Editing text field

- At lookup field can be redefined string conditions for searched records

```
eGrid.lookupConfig = {};
```

- Adding any buttons with custom actions

```
eGrid.addTopBarButtonsLeftArray = [ //Adds array of buttons to top bar from left
(addTopBarButtonsRigthArray adds button array from right)
{
    text: "Button text", //Text viewed on the button
    handler: function () {} //Definition of function, that will be held after click on
the button
}
];
```

- Paging large lists and records on page count configuration

```
eGrid.pageSize = 10; //Configures max rows count on page
```

- Setting colors for new and/or copied row

```
eGrid.addCss = "green"; //Sets color of new rows
eGrid.copyCss = "blue"; //Sets color of copied rows
```

- Possibility of adding filters to view filters

- Setting fix eGrid height (in pixels)

```
eGrid.gridHeight = 200; //Sets the eGrid height to 200px
```

- Setting first day in week

```
eGrid.startDay = 2; //Number 2 is Monday
```

- Possibility of hiding whole top bar

```
eGrid.isTopBar = false; //true - top bar stays visible
```

- Possibility of hiding whole bottom bar

```
eGrid.isBottomBar = false; //true - bottom bar stays visible
```

- Turning loading eGrid progress bar on/off

```
eGrid.useProgressBar = true; //true - turns the bar on, false - turns it off
```

- Setting checkboxes hidden

```
eGrid.checkBoxSelection = false //true - checkboxes stay visible
```

- Setting color of invalid rows

```
eGrid.noValidColor = 'redDark';
```

- Possibility of silent save (without prompt)

```
eGrid.silentSave = true; //false - with prompt
```

- Configuration of window where is eGrid placed

```
eGrid.destinationWindow = {
  title: 'Titulek okna',
  icon: '../adresar/icon.png', //Icon path
  modal: true,
  maximized: true,
  resizable: false,
  layout: 'fit',
  draggable: false,
  listeners: {
    close: function () { //Definition of function that is held at closing window
    },
    show: function () { //Definition of function that is held at showing window
      localMask = new Ext.LoadMask(eGrid1.GetDestinationWindow(),
        {
          msg: "Wait please...",
          listeners: {
            hide: function (m) {
              m.msg = " Wait please...";
            }
          }
        });
      radkyOkno = true;
    }
  },
  closeAction: "hide"
};
```

- Adding columns, that are not in view neither in data

```
eGrid.customColumns = [{
  name: "myCustomField",
  label: 'Custom field',
  visible: true,
  linked: true,
  width: 100,
  order: 3
}];
```

Methods usable for EditableGrid

- Reloading eGrid data

```
eGrid.ReloadData();
```

- Getting sum, average, count, maximum and minimum in specific column – defines at parameter

```
eGrid.GetSum(column); //Sum
eGrid.GetAvg(column); //Average
eGrid.GetCount(column); //Count
eGrid.GetMax(column); //Maximum
eGrid.GetMin(column); //Minimum
```

- Definition of function, that is held at row double click

```
eGrid.OnRowDoubleClick = function (view, record, item, index) { }; //Definition
eGrid.OnRowDoubleClick(view, record, item, index); //Usage
```

- Saving eGrid data

```
eGrid.SaveGrid();
```

- Definition of function held on eGrid data editing

```
eGrid.OnEdit = function (e) { }; //Definition, parameter e - row  
eGrid.OnEdit (e); //Usage
```

- Setting row style

```
eGrid.SetRowStyle = function (record) { //Definition  
  
    if (record.data.ac_duplicita > 0) {  
        if (record.data.ac_duplicita % 2 == 0) {  
            record.set('artColorField', 'blueLight');  
        } else {  
            record.set('artColorField', 'greenLight');  
        }  
    }  
};  
eGrid.SetRowStyle(record); //Usage
```