**ajaxCRUD**

constructor for instantiating the class  
new ajaxCrud ( String buttonText, String dbTableName, String dbPrimaryKey, [ String ajaxcrudDir = ""] )

This constuctor is used to instantiate the PHP object. You must pass in the three required values and this will create your ajaxCRUD object!

buttonText

The name of the item your database table is defining (eg for a table of people this value would be "Person")

dbTable

This is the name of your database table

dbPrimaryKey

This is the primary key of the database table

ajaxcrudDir(optional)

Defaults to "". If you want to put your ajaxcrud file(s) elsewhere in your dir structure, change this variable (eg "../ajaxcrud")

**Details**

|  |  |
| --- | --- |
| **Type** | constructor |
| **Returns** | object |

**Example**

$tblDemo = new ajaxCRUD("Apple", "tblApples", "pkAppleID");  
  
Example2: ajaxCRUD("Item", "ItemTable", "id") will yield an ADD button that says "Add Item"  
Example3: ajaxCRUD("PERSON", "tblPerson", "pkPersonID") will yield an ADD button that says "Add PERSON"  
Example4: ajaxCRUD("EMPLOYEE", "tblEmployee", "pkEmployeeID", "../ajaxcrud") // the php file that uses this must have all ajaxCRUD files in the ../ajaxcrud dir

**See also**

[showTable](http://www.ajaxcrud.com/api/index.php?id=showTable).

**addAjaxFilterBox**

adds a search box at the top of the page

void addAjaxFilterBox( String fieldName, [ int textBoxSize = 10 ])

Use this function when you want to be able to search your table by a field. This function allows only ajax-based searching (which is best) and searches "as you type"

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |

**Parameters**

fieldName

name of the field you want to search by

*[optional]* textBoxSize

size/length of the textbox

**Example**

$tblDemo->addAjaxFilterBox('fldFamilyName');  
  
$tblDemo->addAjaxFilterBox('fldFamilyName', 15);

**See also**

[setAjaxFilterBoxSize](http://www.ajaxcrud.com/api/index.php?id=setAjaxFilterBoxSize), [addAjaxFilterBoxAllFields](http://www.ajaxcrud.com/api/index.php?id=addAjaxFilterBoxAllFields).

**addAjaxFilterBoxAllFields**

adds searchboxes for all fields in your table

void addButtontoRow( )

Use this function when you want to be able to search by all (displayable) fields in your table

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | all fields searchable |

**Example**

$tblDemo->addAjaxFilterBoxAllFields()

**See also**

[addAjaxFilterBox](http://www.ajaxcrud.com/api/index.php?id=addAjaxFilterBox).

**addButton**

adds a button to the bottom of the page

void addButton ( String buttonValue, String URL, [ String tags = "" ] )

Use this function when you want to provide easy navigation to another page (for easy processing). It inserts a button at the bottom of the page right next to the "add item" button.

You can use the optional *tags* parameter to attach events to a button.

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |

**Parameters**

Button value

The value of the button (ie the text displaying on the button).

URL

The page to go to (ie .php, html, asp page...).

*[optional]* tags

javascript tags (eg onClick, onMouseOver, etc)

**Example**

$tblDemo->addButton("Home", "index.php");

**See also**

[addButtonToRow](http://www.ajaxcrud.com/api/index.php?id=addButtonToRow).

**addButtonToRow**

adds a button to every row and sends the id (pk) of that item to any url

void addButtontoRow( String buttonValue, String URL, [ String $attach\_params ], [ String $javascript\_tags ], [ String $window\_to\_open ] )

Use this function to pass the primary key to another page (url) to do some other process. This function is very useful when your database has many-to-many relationships and you need to "attach" multiple items to an item.

**By default** the GET parameter for the primary key is passed to the page with the id being the primary key value. However, you can use the word "**ALL**" as an optional third parameter to have ALL fields and their values passed!

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |
| **Default** | no button |

**Parameters**

buttonValue

Value for the button

URL

a web script (eg a php, .asp, .jsp file)

attach\_param (optional)

the field you want to pass to the URL. Can be "all" which means all fields

javascript\_tags (optional)

any javascript you want added to the button

window\_to\_open (optional)

open this page in a "new" window (default is "same")

**Example**

$tblDemo->addButtonToRow("View", "myOwnDisplayPage.php");   
OR   
//this second example will pass ALL fields to the update form page  
$tblDemo->addButtonToRow("Edit", "myOwnUpdateForm.php", "all");   
OR   
//this third example will pass only the "myDBField" variable (and value) to the update page  
$tblDemo->addButtonToRow("Edit", "myOwnUpdateForm.php", "myDBField");   
OR   
//this fourth example will pass in the name of a user-defined JavaScript function which will allow the page to open up in specific window  
<?php  
$mytable->addButtonToRow("View", "", "", "myJavascriptFunctionToOpenInNewWindow");  
?>  
.  
.  
.  
<script> function myJavascriptFunctionToOpenInNewWindow(id){ alert('opening new window'); window.open('test.php?courseid=' + id'); } </script>

**See also**

[addButton](http://www.ajaxcrud.com/api/index.php?id=addButton), [addButtonToRowWindowOpen](http://www.ajaxcrud.com/api/index.php?id=addButtonToRowWindowOpen).

**addOrderBy**

orders the table by a specified field

void addOrderBy ( String orderBySQL )

This function merely injects the SQL with the "ORDER BY" expression and you can specify what field(s) you want to order the table by.

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |

**Parameters**

orderBySQL

The SQL "order by" string (must begin with "ORDER BY")

**Example**

$tblDemo->addOrderBy("ORDER BY fldSort asc");

**See also**

[addWhereClause](http://www.ajaxcrud.com/api/index.php?id=addWhereClause).

**addTableBorder**

adds a border to the html table

***this function is deprecated!*** *-- please use* [*setCSSFile*](http://www.ajaxcrud.com/api/index.php?id=setCSSFile) *to control style/layout*

void addTableBorder ()

Use this function if you want your table to have a border around it

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |
| **Default** | off |

**Parameters**

none

**Example**

$tblDemo->addTableBorder();

**See also**

[showTable](http://www.ajaxcrud.com/api/index.php?id=showTable), [setCSSFile](http://www.ajaxcrud.com/api/index.php?id=setCSSFile), [cellspacing](http://www.ajaxcrud.com/api/index.php?id=cellspacing).

**addValueOnInsert**

sets a value to a field on row insert

void addValueOnInsert( String fieldName, String value )

This function should be used when certain fields should be immediately populated with pre-defined data. A good use of this function is when your table has a "date created" field for an entity. This field can be automatically determined (by mySQL) and then set when adding the row.

Keep in mind: this function will overwrite any value input by the user for this field. Therefore it should always be used in conjuctiton with [omitAddField](http://www.ajaxcrud.com/api/index.php?id=omitAddField) so that an input box for the field does not even display when adding an entry.

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |

**Parameters**

fieldName

The database field name you want to be able to assign a specific data input to on row insert

value

Value to overwrite the field with - if the value is "NOW()" it is parsed not as a string, but as a mySQL expression

**Example**

$tblDemo->addValueOnInsert("fldCreationDate", "NOW()");   
  
  
**NOTE:** you MUST use omitAddField with the field for it to be overwritten; reason being - if a value is sent for that field as part of the "add" form then that takes priority over addValueOnInsert

**See also**

[omitAddField](http://www.ajaxcrud.com/api/index.php?id=omitAddField) [defineAllowableValues](http://www.ajaxcrud.com/api/index.php?id=defineAllowableValues) [onAddExecuteCallBackFunction](http://www.ajaxcrud.com/api/index.php?id=onAddExecuteCallBackFunction).

**addWhereClause**

adds a SQL "where clause" to the query

void addWhereClause ( String whereClauseSQL )

Use this function when you do not want to display ALL rows in the table but only ones that meet certain conditions.

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |
| **Default** | null |

**Parameters**

whereClauseSQL

SQL where clause (must begin with "WHERE"...)

**Example**

$tblDemo->addWhereClause("WHERE fldCreateDate > '1984-10-11'");

**See also**

[addOrderBy](http://www.ajaxcrud.com/api/index.php?id=addOrderBy).

**appendUploadFilename**

append the value of the row to the beginning of the filename

void appendUploadFilename ( String fieldName )

Use this function to prevent easily "overwriting" filenames on upload. You can have each filename easily be unique by attaching the primary key (or some other field) to the filename.

**Details**

|  |  |
| --- | --- |
| **Requires** | [setFileUpload](http://www.ajaxcrud.com/api/index.php?id=setFileUpload) |
| **Type** | function |
| **Returns** |  |
| **Default** | no value appended |

**Parameters**

fieldName

The name of the field which VALUE will be appended

**Example**

$tblDemo->setFileUpload("fldFilename", "uploads/", "uploads/");  
$tblDemo->appendUploadFilename("pkRowID");

**See also**

[setFileUpload](http://www.ajaxcrud.com/api/index.php?id=setFileUpload).

## defineAllowableValues

define "allowable" values for a field.

void defineAllowableValues ( String fieldName, String[] validValues, [ onedit\_textbox = FALSE ] )

These allowable fields will display in a dropdown of the values you specify. You can have the values for the dropdown be the same as the text or separate. Both examples are provided below.

In addition, you can use the third parameter (optional) to allow fields (after being inserted) to NOT have only the allowed values permissable. If set to TRUE then a regular textbox will appear when editing this field--allowing any values.

### Details

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |

### Parameters

***fieldname***: The database field name to define allowable values for

***validValues***: The allowed values for that field. Can be an *array of strings* (for value = text) or an *array of arrays* (where [0] = value and [1] = text)

*(optional)* ***onedit\_textbox*** (default: false) : If set to true this value allows the field to not open a dropdown box on EDIT of the field. It will be a textbox allowing any input

### Example

#### With id and value the same

$valid\_values = array("DropdownValue 1", "Dropdown Value2");

$tblDemo->defineAllowableValues("fldCertainFields", $valid\_values);

#### With id and value different

$valid\_values = array( array(0 => "ddv1", 1 => "Dropdown Value 1"),

array(0 => "ddv2", 1 => "Dropdown Value2") );

$tblDemo->defineAllowableValues("fldCertainFields", $valid\_values);

### See also

[defineRelationship](http://www.ajaxcrud.com/api/index.php?id=defineRelationship).

**defineCheckbox**

Sets the field to be of type 'checkbox'.

void defineCheckbox ( String fieldName, [ String value\_on], [String value\_off] )

Use this function if you have an **on/off** or **yes/no** type of field (two options). As such you want a simple checkbox to toggle between the options.   
  
The default value for checked (on) is **1** and unchecked (off) is **0**. However, you can set your own values, with the two optional parameters.

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |
| **Default** | 1,0 |

**Parameters**

fieldName

The database field name

*(optional)* value\_on

The value set when the checkbox is checked (default is 1)

*(optional)* value\_off

The value set when the checkbox is unchecked (default is 0)

**Example**

//this will set 1 for checked and 0 for unchecked (perfect if your field is an INT or TINYINT)  
**$tblDemo->defineCheckbox("fldHasAHome");**  
  
//this will define it so that "on" is for checked and "off" is for unchecked   
//(obviously you want to use a varchar or a text now)  
**$tblDemo->defineCheckbox("fldILeftTheStove", "on", "off");**

**See also**

[defineAllowableValues](http://www.ajaxcrud.com/api/index.php?id=defineAllowableValues).

**defineRelationship**

defines a relationship between your table and another (via the foreign key)

void defineRelationship ( String dbMainTableFK, String dbOtherTable, String dbOtherTablePK, String dbOtherTableField, [ String dbOtherTableSortField = ""], [ INT required = 1 ], [String whereClause=""] )

This function sets the "allowable" values in a dropdown box for a 1-Many relationship. Your dropdown VALUE will be the foreign key and the option TEXT will be the field of your choosing in the associated table.

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |

**Parameters**

dbMainTableFK: Foreign key of the main table

dbOtherTable : Name of the associated table (table you're making a relationship to)

dbOtherTablePK : Primary key of the associated table (this is the VALUE of the dropdown)

dbOtherTableField : Field to return for the TEXT in the dropdown

*(optional)* dbOtherTableSortField

The name of the field in the associated table you wish to sort by (eg fldSort)

*(optional)* required

0 to make this field not require one of the choices (default) | 1 to require one be selected

*(optional)* whereClause : a where clause for the associated table to use

**Example**

$tblDemo->defineRelationship("fkOtherTableID", "tblOtherTable", "pkOtherTableID", "fldOtherTableField");   
  
OR (w/advanced features)  
  
$tblDemo->defineRelationship("fkOtherTableID", "tblOtherTable", "pkOtherTableID", "fldOtherTableField", "fldSort", 0, "WHERE fldOtherTableName LIKE 'Libertarian%'");   
  
DISPLAY MORE THAN ONE FIELD:  
  
$tblDemo->defineRelationship("fkOtherTableID", "tblOtherTable", "pkOtherTableID", "CONCAT(fldFirstName, " ", fldMiddleName, " ", fldLastName)")

**See also**

[defineAllowableValues](http://www.ajaxcrud.com/api/index.php?id=defineAllowableValues).

**disableTableHeaders**

Turns off html table headers for the table

void disableTableHeaders()

Use this method if you want to remove table headers from your data output

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |
| **Default** | table headers: true |

**Parameters**

none

**Example**

$tblDemo->disableTableHeaders();

**See also**

[turnOffAjaxADD](http://www.ajaxcrud.com/api/index.php?id=turnOffAjaxADD), [emptyTableMessage](http://www.ajaxcrud.com/api/index.php?id=emptyTableMessage).

**disallowAdd**

disallow new rows to be added to the table

void disallowAdd ()

This essentially does not display the "add item" button at the bottom.

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |

**Parameters**

none

**Example**

$tblDemo->disallowAdd();

**See also**

[disallowDelete](http://www.ajaxcrud.com/api/index.php?id=disallowDelete) [disallowEdit](http://www.ajaxcrud.com/api/index.php?id=disallowEdit).

**disallowDelete**

disallows row deletion

void disallowDelete ()

Use this function if you do not want to allow the user to delete rows from the table.

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |

**Parameters**

none

**Example**

$tblDemo->disallowDelete();

**See also**

[disallowAdd](http://www.ajaxcrud.com/api/index.php?id=disallowAdd) [disallowEdit](http://www.ajaxcrud.com/api/index.php?id=disallowEdit).

**disallowEdit**

disallows a field to be edited

void disallowEdit ( String fieldName )

Use this function to disable the ajax editing of a field in a table. This is often used for the fields where you want populated with data (on create) and never touched again (eg a Creation Datestamp). This function can be used in conjunction with the [addValueOnInsert](http://www.ajaxcrud.com/api/index.php?id=addValueOnInsert) method, so that fields such as "fldCreateDate" can be set when a row is created and then is not editable afterward.

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |

**Parameters**

fieldName

The database field name you don't want to be able to edit

**Example**

$tblDemo->disallowEdit('fldCreationDate');

**See also**

[turnOffAjaxEditing](http://www.ajaxcrud.com/api/index.php?id=turnOffAjaxEditing), [addValueOnInsert](http://www.ajaxcrud.com/api/index.php?id=addValueOnInsert), [disallowDelete](http://www.ajaxcrud.com/api/index.php?id=disallowDelete), [disallowAdd](http://www.ajaxcrud.com/api/index.php?id=disallowAdd).

**displayAddFormTop**

bring the Add Row form to the top of the page

void displayAddFormTop()

the "Add" form is below the table (by default). use displayAddFormTop() to bring it to the top

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |

**Example**

$tblDemo->displayAddFormTop();

**See also**

[disallowAdd](http://www.ajaxcrud.com/api/index.php?id=disallowAdd).

**displayAs**

displays the field name in the table under a pseudonym

void displayAs ( String fieldName, String fieldDisplayTitle )

Use this function to refer to a database field as another name (ie if your fields all have a prefix you can remove them when displaying them).

This display title is referred to in all places in the table (adding & editing)

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |
| **Default** | actual field name |

**Parameters**

fieldName

The database field you want to display under another name

fieldDisplayTitle

The alternate title to display

**Example**

$tblDemo->displayAs("fldMyField", "My Field");

**formatFieldWithFunction**

format data in a column with a custom function

void formatFieldWithFunction( String fieldName, String functionName )

Use this function when you wish to format data in your table with a custom function. For example, you'd use this if you wanted to format money to always display looking like a currency.   
  
The parameters of your custom function (which YOU define) must take in ONE argument (**$data**) and return your formated data.

**Details**

|  |  |
| --- | --- |
| **Requires** | a user-defined function to be set |
| **Type** | function |
| **Returns** | formatted data |
| **Default** | non-formatted data |

**Parameters**

fieldName

name of the field you want to format

functionName

name of the local function to be invoked

**Example**

**Example 1:**

$tblDemo->formatFieldWithFunction('fldCarPrice', 'formatAsCurrency');

function formatAsCurrency($data){

    return number\_format($data, 2, '.', ','); //displays 10 as 10.00

}

**Example 2:**

$tblDemo = formatFieldWithFunction("fldImage", "displayImage");

function displayImage($value){

//displays the image instead of just the filename

return "<img src='/images/{$value}' width='100' \>\n";

}

**See also**

[formatFieldWithFunctionAdvanced](http://www.ajaxcrud.com/api/index.php?id=formatFieldWithFunctionAdvanced), [modifyFieldWithClass](http://www.ajaxcrud.com/api/index.php?id=modifyFieldWithClass), [onAddExecuteCallBackFunction](http://www.ajaxcrud.com/api/index.php?id=onAddExecuteCallBackFunction), [onFileUploadExecuteCallBackFunction](http://www.ajaxcrud.com/api/index.php?id=onFileUploadExecuteCallBackFunction).

**formatFieldWithFunctionAdvanced**

format data in a column with a custom function

void formatFieldWithFunctionAdvanced( String fieldName, String functionName )

Use this function when you wish to format data in your table with a custom function which needs to use logic based on OTHER fields in the row. As such, you have access to the id (primary key) of the row to make logic-based styling decisions.   
  
The parameters of your custom function (which YOU define) must take in TWO arguments (**$data** and **$id**) and return your formatted data.

**Details**

|  |  |
| --- | --- |
| **Requires** | a user-defined function to be set |
| **Type** | function |
| **Returns** | formatted data |
| **Default** | non-formatted data |

**Parameters**

***fieldname***: name of the field you want to format

***functionName***: name of the local function to be invoked

**Example**

**Example:**

$tblCar->formatFieldWithFunctionAdvanced('fldPrice', 'displayCarPrice');

# this function returns the car's price if it's not sold; if so

# it says that the car is not for sale and displays the purchase date

function displayCarPrice($data, $id){

$carInfo = qr("SELECT fldStatus, fldPurchaseDate

FROM tblCar WHERE pkCarID = $id");

$status = $carInfo['fldStatus'];

$purchaseDate = $carInfo['fldPurchaseDate'];

$return = to\_money($data);

if ($status == 'purchased'){

$return = "CAR NO LONGER FOR SALE. Sold on $purchaseDate";

}

return $return;

}

**See also**

[formatFieldWithFunction](http://www.ajaxcrud.com/api/index.php?id=formatFieldWithFunction), [showOnly](http://www.ajaxcrud.com/api/index.php?id=showOnly).

**getNumRows**

Returns the number of rows returned

void getNumRows( )

Make sure you include your [addWhereClause](http://www.ajaxcrud.com/api/index.php?id=addWhereClause) statement(s) prior to calling this function

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | int |

**Parameters**

void

**Example**

$numRows = $tblProject->getNumRows();

if ($numRows > 0){

//do something

}

**See also**

[addWhereClause](http://www.ajaxcrud.com/api/index.php?id=addWhereClause), [emptyTableMessage](http://www.ajaxcrud.com/api/index.php?id=emptyTableMessage).

**insertRowsReturned**

Displays the number of rows returned (can be put anywhere in your application view).

void insertRowsReturned( )

Echos (a span html element) with the current row count. When/if you filter or add/delete rows, it will be dynamically updated via ajax

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | dynamic span w/row count |
| **Default** | 0 |

**Example**

Total Results: **insertRowsReturned();?>**

**See also**

[getNumRows](http://www.ajaxcrud.com/api/index.php?id=getNumRows).

**modifyFieldWithClass**

add a [css](http://www.ajaxcrud.com/api/index.php?id=css) class to a field (used for things like jquery validation or field masking)

void modifyFieldWithClass( String fieldName, String cssClassName )

Use this function when you wish to do some javascript/jquery validation or simply add a class to it to format it a certain way. For example, you'd use this function if you wanted make a field required (when adding) or to mask certain fields like phone number.

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** |  |
| **Default** | void |

**Parameters**

fieldName

name of the field you want to format

cssClassName

name [css](http://www.ajaxcrud.com/api/index.php?id=css) class(es) to be applied

**Example**

**Example 1:**

$tblDemo->modifyFieldWithClass("fldDateField", "datepicker");

**Example 2:**

$tblDemo->modifyFieldWithClass("fldZip", "zip required");

Any javascript validation based on the applied css class must be in the file *validation.js*

**See also**

[formatFieldWithFunction](http://www.ajaxcrud.com/api/index.php?id=formatFieldWithFunction).

## omitAddField

omits the input field from displaying when adding a row

void omitAddField ( String fieldName )

Use this function if you do not want to add a value when you create a row. It is useful for fields like fldLastModified, etc.

This function will NOT omit the field from being displayed in the table. To do this use [omitField](https://ajaxcrud.com/api/index.php?id=omitField). Also: you can use the [addValueOnInsert](https://ajaxcrud.com/api/index.php?id=addValueOnInsert) function in conjuntion to set a specific value to the field on row insert.

### Details

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |

### Parameters

fieldName

The name of the field you do not wish display an input box for (when creating a new row)

### Example

$tblDemo->omitAddField("fldLastModified");

### See also

[omitField](https://ajaxcrud.com/api/index.php?id=omitField) [addValueOnInsert](https://ajaxcrud.com/api/index.php?id=addValueOnInsert) [disallowEdit](https://ajaxcrud.com/api/index.php?id=disallowEdit) [disallowAdd](https://ajaxcrud.com/api/index.php?id=disallowAdd).

## omitField

omits a field from displaying in the table

void omitField ( String fieldName )

This function is used when you do not wish to DISPLAY a field (essentially omits the column for the table).

This function will NOT omit the field (and input box) from displaying when ADDING a row to the database. To do this use [omitAddField](https://ajaxcrud.com/api/index.php?id=omitAddField)

### Details

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |

### Parameters

fieldName

The field name you do not wish to display in the table

### Example

$tblDemo->omitField("fldLastModified");

### See also

[omitPrimaryKey](https://ajaxcrud.com/api/index.php?id=omitPrimaryKey) [omitAddField](https://ajaxcrud.com/api/index.php?id=omitAddField) [disallowEdit](https://ajaxcrud.com/api/index.php?id=disallowEdit).

## omitFieldCompletely

this function calls both [omitField](https://ajaxcrud.com/api/index.php?id=omitField) and [omitAddField](https://ajaxcrud.com/api/index.php?id=omitAddField) for a field

void omitFieldCompletely ( String fieldName )

Use this function when you want to both NOT display a field AND not allow a field to be added (created a null value (or default value) for that field)

### Details

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |
| **Default** | field not omitted |

### Parameters

fieldName

The field name you do not wish to display in the table

### Example

$tblDemo->omitFieldCompletely('fldActive');

### See also

[omitField](https://ajaxcrud.com/api/index.php?id=omitField) [omitAddField](https://ajaxcrud.com/api/index.php?id=omitAddField).

## omitPrimaryKey

omits the primary key from displaying in the table

void omitPrimaryKey()

You should use this function when you do not want to display the primary key (id) of your database table in your html table.

### Details

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |

### Parameters

none

### Example

$tblDemo->omitPrimaryKey();

### See also

[omitField](https://ajaxcrud.com/api/index.php?id=omitField) [omitAddField](https://ajaxcrud.com/api/index.php?id=omitAddField).

## onAddExecuteCallBackFunction

calls a local function when a new row is added

void onAddExecuteCallBackFunction ( String functionName )

Use this function when you want to apply certain logic (or display a certain message) when a new row is added. Use of this function is *similar* in a way to a java interface: **you must define a function with the name specified in the function call.**

The defined function must take ONE parameter (an array). The indexes of this array are the fields of the table. (with one extra index: "id")

**NOTE:** this function will TURN OFF ajax adding.

**NB:** if you wish to implement logic which will require headers not be sent prior (e.g. set cookies or header redirects) you will need to set the variable [***doActionOnShowTable***](https://ajaxcrud.com/api/index.php?id=doActionOnShowTable) to FALSE and then direct your executing script to call obj->doAction("add").

### Details

|  |  |
| --- | --- |
| Requires | (callback function) |
| Type | function |
| Returns | void |

### Parameters

**functionName**: name of the local, user-defined function to be invoked when a row is added

### Example

$demo->onAddExecuteCallBackFunction("doMyLogicForAdding");

//down below the showTable() function

function doMyLogicForAdding($array){

$pkMenuID = $array['pkMenuID'];//indexes are fields from db

$filename = $array['fldFilename'];

//update this table (or some other table)

if ($fldURL == ''){

$success = qr("UPDATE tblMenu SET fldURL = \"$filename\"

WHERE pkMenuID = $pkMenuID");

}

}

### See also

[onUpdateExecuteCallBackFunction](https://ajaxcrud.com/api/index.php?id=onUpdateExecuteCallBackFunction), [onFileUploadExecuteCallBackFunction](https://ajaxcrud.com/api/index.php?id=onFileUploadExecuteCallBackFunction), [onDeleteFileExecuteCallBackFunction](https://ajaxcrud.com/api/index.php?id=onDeleteFileExecuteCallBackFunction), [addValueOnInsert](https://ajaxcrud.com/api/index.php?id=addValueOnInsert), [turnOffAjaxADD](https://ajaxcrud.com/api/index.php?id=turnOffAjaxADD).

## onDeleteFileExecuteCallBackFunction

calls a local function when a file (that was previously uploaded) is deleted

void onDeleteFileExecuteCallBackFunction ( String functionName )

use this function when you want to apply certain logic (or display a certain message) when an (uploaded) file is deleted. Use of this function is *similiar* in a way to a java interface: **you must define a function with the name specifed in the function call.**

This function must take ONE parameter (an array). The indexes of this array are the following:

 id (the id of the row where the file was removed)

 field (the field for which the file was removed)

### Details

|  |  |
| --- | --- |
| Requires | [setFileUpload](https://ajaxcrud.com/api/index.php?id=setFileUpload), (callback function) |
| Type | function |
| Returns | void |

### Parameters

**functionName:** name of the local function to be invoked when a file is deleted

### Example

$table->onDeleteFileExecuteCallBackFunction("deleteDocument");

function deleteDocument($array){

$id = $array[id];

$field = $array[field];

if ($id != ''){

$success = qr("UPDATE tblMenu SET

fldURL = '' WHERE pkMenuID = $id");

}

}

### See also

[onFileUploadExecuteCallBackFunction](https://ajaxcrud.com/api/index.php?id=onFileUploadExecuteCallBackFunction), [onAddExecuteCallBackFunction](https://ajaxcrud.com/api/index.php?id=onAddExecuteCallBackFunction).

## onFileUploadExecuteCallBackFunction

calls a local function when a file is uploaded

void onFileUploadExecuteCallBackFunction ( String functionName )

use this function when you want to apply certain logic (or display a certain message) when a file is uploaded. Use of this function is *similiar* in a way to a java interface: **you must define a function with the name specifed in the function call.**

This function must take ONE parameter (an array). The indexes of this array are as follows:

 id (the id of the row for which the file was uploaded)

 *(your primary key)* (the id of the row for which the file was uploaded)

 field (the fieldname to which the file was uploaded [the filename was placed in this field: see [setFileUpload](https://ajaxcrud.com/api/index.php?id=setFileUpload)])

 fileName (the name of the file) [may have been renamed]

 fileSize (the filesize of the file)

 fldType (the filetype of the file)

### Details

|  |  |
| --- | --- |
| Requires | [setFileUpload](https://ajaxcrud.com/api/index.php?id=setFileUpload), *your callback function* |
| Type | function |
| Returns | void |

### Parameters

**functionName:** name of the local function to be invoked when a file is uploaded

### Example

$demo->setFileUpload("fldFilename");

$demo->onFileUploadExecuteCallBackFunction("uploadDocument");

function uploadDocument($array){

$id = $array[id]; //this will always be the value

$id = $array[pkID]; //same as above IF the pk is pkID

$fileName = $array[fileName];

$fileSize = $array[fileSize];

$fileType = $array[fldType];

if ($fileName != ''){

$success = qr("UPDATE tblDemo SET fldFilesize = '$fileSize'

WHERE pkDemoID = $id");

}

}

### See also

[setFileUpload](https://ajaxcrud.com/api/index.php?id=setFileUpload), [onAddExecuteCallBackFunction](https://ajaxcrud.com/api/index.php?id=onAddExecuteCallBackFunction), [onDeleteFileExecuteCallBackFunction](https://ajaxcrud.com/api/index.php?id=onDeleteFileExecuteCallBackFunction).

## onUpdateExecuteCallBackFunction

calls a local, user-defined php function after a particular field is edited/updated

void onUpdateExecuteCallBackFunction ( String fieldName, String functionName )

Use this function when you want to apply certain logic (or display a certain message) when a row is updated/edited. Use of this function is *similar* in a way to a java interface: **you must define a function with the name specified in the function call.**

The defined function must take ONE parameter (an array). The indexes of this array are the fields values of the entire edited row.

NOTE: this function will TURN OFF ajax editing (for that particular field)

### Details

|  |  |
| --- | --- |
| Requires | (callback function) |
| Type | function |
| Returns | void |

### Parameters

**fieldname:** name of the field which, after edited, will cause the callback function to be executed

**functionName:** name of the local, user-defined function to be invoked when the field is updated/edited

### Example

$demo->onUpdateExecuteCallBackFunction("fldBirthday",

"doMyLogicAfterUpdating");

//down below the showTable() function

function doMyLogicForUpdating($array){

$pkBirthdayID = $array['pkBirthdayID']; //example index - field in db

$fldName = $array['fldName']; //example index - field in db

$fldBirthday = $array['fldBirthday']; //example index - field in db

$age = calulateAge($fldBirthday, $fldName) //example function-not shown

//update this table (or some other table)

if ($fldURL == ''){

$success = qr("UPDATE tblPerson SET fldAge = \"$age\"

WHERE pkPersonID = 3");

}

}

### See also

[onAddExecuteCallBackFunction](https://ajaxcrud.com/api/index.php?id=onAddExecuteCallBackFunction), [onFileUploadExecuteCallBackFunction](https://ajaxcrud.com/api/index.php?id=onFileUploadExecuteCallBackFunction), [onDeleteFileExecuteCallBackFunction](https://ajaxcrud.com/api/index.php?id=onDeleteFileExecuteCallBackFunction).

## orderFields

sort the fields order in the database

void orderFields ( String fieldList )

Use this table if you want to rearrange the order of the database table without having to change the database table schema

### Details

|  |  |
| --- | --- |
| Type | function |
| Returns | re-sorted fields in table |
| Default | order in database schema |

### Parameters

**fieldList:** A list of fields (comma delimited) in the order you want to show them.

### Example

$tblFriend->orderFields("pkFriendID, fldAddress, fldName, fldState");

### See also

[addOrderBy](https://ajaxcrud.com/api/index.php?id=addOrderBy).

## setAddFieldNote

Puts a note (description of the field) next to it when adding a column

setAddFieldNote( String fieldName, String fieldCaption )

Puts a note (description of the field) next to it when adding a column

### Details

|  |  |
| --- | --- |
| Type | function |
| Returns | void |
| Default | null |

### Parameters

**fieldname:** name of the field

**fieldCaption:** the [caption](https://ajaxcrud.com/highslide/doc.php#adding-caption) you want to display

### Example

$tblTest->setAddFieldNote("fldField", "Enter your first name");

### See also

[setAddPlaceholderText](https://ajaxcrud.com/api/index.php?id=setAddPlaceholderText).

## setAddPlaceholderText

Puts placeholder text on the input fields

function setAddPlaceholderText ( String fieldName, String placeholderText )

Only applies when ADDing a new row.

### Details

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | null |
| **Default** | null |

### Parameters

fieldName

name of the field

placeholderText

text in the placeholder

### Example

$tblAttendee->setAddPlaceholderText("fldTimeArriving", "5:30pm");

### See also

[setAddFieldNote](https://ajaxcrud.com/api/index.php?id=setAddFieldNote).

## setAjaxFilterBoxSize

**\*\*THIS FUNCTION IS DEPRECATED AS OF V6.0; use second (optional) param of** [**addAjaxFilterBox**](https://ajaxcrud.com/api/index.php?id=addAjaxFilterBox) **instead\*\***  
sets the size (lengh of the textbox) of an ajax filter box

void setAjaxFilterBoxSize( String fieldName, int size)

Use this function when you want to increase or decrease the size of an ajax filter box. You must use the [addAjaxFilterBox](https://ajaxcrud.com/api/index.php?id=addAjaxFilterBox)(fieldname) or [addAjaxFilterBoxAllFields](https://ajaxcrud.com/api/index.php?id=addAjaxFilterBoxAllFields)() function in order to make this work

### Details

|  |  |
| --- | --- |
| **Requires** | [addAjaxFilterBox](https://ajaxcrud.com/api/index.php?id=addAjaxFilterBox) or [addAjaxFilterBoxAllFields](https://ajaxcrud.com/api/index.php?id=addAjaxFilterBoxAllFields) |
| **Type** | function |
| **Returns** |  |
| **Default** | 10 |

### Parameters

fieldName

name of the field you want to search by

size

size of the textbox (eg 5, 10...)

### Example

$tblDemo->addAjaxFilterBox('fldField1'); $tblDemo->setAjaxFilterBoxSize('fldField1', 4);

### See also

[addAjaxFilterBox](https://ajaxcrud.com/api/index.php?id=addAjaxFilterBox), [addAjaxFilterBoxAllFields](https://ajaxcrud.com/api/index.php?id=addAjaxFilterBoxAllFields).

## setCSSFile

Sets a [css](https://ajaxcrud.com/api/index.php?id=css) file as the stylesheet to be used to render the table.

void setCSSFile ( String cssFile )

Use this function when you want to apply a stylesheet to the table. The class is compatible with the stylesheets provided here: <http://icant.co.uk/csstablegallery/>

### Details

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |
| **Default** | default.css |

### Parameters

cssFile

the [css](https://ajaxcrud.com/api/index.php?id=css) file (must be in the same folder as the [ajaxCRUD](https://ajaxcrud.com/api/index.php?id=ajaxCRUD) library in the **/[css](https://ajaxcrud.com/api/index.php?id=css)** folder)

### Example

$tblDemo->setCSSFile('mycss.css');

### See also

[cellspacing](https://ajaxcrud.com/api/index.php?id=cellspacing), [addTableBorder](https://ajaxcrud.com/api/index.php?id=addTableBorder).

## setExactSearchField

Table filter assist function; sets a field to use an EXACT search.

void setExactSearchField ( String fieldName )

Since the default search (for table filtering) is an approximate search, this function sets a field to use an EXACT search. In other words a = instead of a LIKE %search%

### Details

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |

### Parameters

fieldName

The field name

### Example

$tblDemo->setExactSearchField('fldFieldName');

### See also

[addAjaxFilterBox](https://ajaxcrud.com/api/index.php?id=addAjaxFilterBox), [addAjaxFilterBoxAllFields](https://ajaxcrud.com/api/index.php?id=addAjaxFilterBoxAllFields).

## setFileUpload

allows a field to accept file uploads

void setFileUpload ( String fieldName, String serverUploadFolder, String webUploadFolder, [Array permittedExts] )

This function allows a field (or multiple fields) in your table to allow file uploads. What this essentially means is that the field you set to handle file uploads inserts the file's **relative path** to the file.

Note: you must chmod the folder to which you're uploading files to **777** (world writable)

### Details

|  |  |
| --- | --- |
| Type | function |
| Returns | web-safe filename of uploaded file (on upload |

### Parameters

**fieldname**: The field name you want to accept file uploads

**serverUploadFolder**: The **full** OR **relative** path to the folder which will contain the uploaded file(s) **on the server**

**webUploadFolder**: The relative and **web accessible path** to the folder which contains the uploaded file(s)

**permittedExts** *(optional):* An array with allowed file extensions supported by this upload

### Example

**Example 1:**  
$tblDemo->setFileUpload("fldFilename", "uploads/", "uploads/");   
  
**Example 2:**  
$allowedExts = array("png", "jpg", "jpeg", "gif", "bmp", "tif", "tiff"); $profile->setFileUpload("fldAvatar", "profiles/", "profiles/", $allowedExts);

### See also

[appendUploadFilename](https://ajaxcrud.com/api/index.php?id=appendUploadFilename), [onFileUploadExecuteCallBackFunction](https://ajaxcrud.com/api/index.php?id=onFileUploadExecuteCallBackFunction).

## setInitialAddFieldValue

Set the initial value for a field (when adding a row)

function setInitialAddFieldValue ( String fieldName, String value )

### Details

|  |  |
| --- | --- |
| Type | function |
| Returns | null |
| Default | null |

### Parameters

fieldName

name of the field

value

value you want

### Example

$tblDemo->setInitialAddFieldValue ("fldTime", "5:00");

### See also

[setAddPlaceholderText](https://ajaxcrud.com/api/index.php?id=setAddPlaceholderText), [setAddFieldNote](https://ajaxcrud.com/api/index.php?id=setAddFieldNote).

## setLimit

set the limit of number of rows to display

void setLimit ( int limit )

Use this function to specify the maximum amount of rows you wish to display **on one page**. This function does not truncate the total amount of rows returned. It merely "pages" the results. So if there are 90 rows in the table and you've set a limit of 30, then there will be 3 pages.

### Details

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |
| **Default** | 50 |

### Parameters

limit

The maximum limit of rows to be returned per page. Value must be an int

### Example

$tblDemo->setLimit(30);

### See also

[addWhereClause](https://ajaxcrud.com/api/index.php?id=addWhereClause) [addOrderBy](https://ajaxcrud.com/api/index.php?id=addOrderBy).

## setOrientation

set the orientation of the table (i.e. set to veritcal)

void setOrientation ( String orientation )

Use this function is you want to set the orientation of the table to vertical instead of horizontal (which is default). One would use this function if they had a long list of fields in their database OR if their field names were long, causing the page display to require scrolling

### Details

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |
| **Default** | horizontal |

### Parameters

orientation

the orientation (only valid values are 'horizontal' or 'vertical')

### Example

$tblDemo->setOrientation("vertical");

### See also

[setLimit](https://ajaxcrud.com/api/index.php?id=setLimit), [showOnly](https://ajaxcrud.com/api/index.php?id=showOnly).

## setTextareaHeight

Sets the textarea height for a field

void setTextareaHeight ( String fieldName, int size )

This function should be used where the known input text will most likely be longer (than for a regular text field) eg a "Description" field or a field to input HTML.

*This function automatically sets a field to use a* ***textarea*** *for editing / adding*

### Details

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |
| **Default** | input box |

### Parameters

fieldName

The field name which will use a textarea input to handle the user data

size

The height (in pixels) of the textarea

### Example

$tblDemo->setTextareaHeight('fldLongField', 200);

### See also

[setTextboxWidth](https://ajaxcrud.com/api/index.php?id=setTextboxWidth).

## setTextboxWidth

Sets the width size for a textbox field

void setTextboxWidth ( String fieldName, int size )

Sets the width of textboxes to something less or more than 20. This is useful if/when making [ajaxCRUD](https://ajaxcrud.com/api/index.php?id=ajaxCRUD) forms mobile-friendly

### Details

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |
| **Default** | 20 |

### Parameters

fieldName

The field name which will use a textarea input to handle the user data

size

The height (in pixels) of the textarea

### Example

$tblDemo->setTextboxWidth('fldMyField', 10);

### See also

[setTextareaHeight](https://ajaxcrud.com/api/index.php?id=setTextareaHeight).

## showCheckboxAll

Set ALL checkboxes for a field (in a column) to be on of off.

void showCheckboxAll ( String fieldName, String display\_data )

To be used as a "header" checkbox. When selected it sets ALL checkboxes for that field to be of a certain value.   
  
Use this function only if you have an **true/false** type of field (two options). As such you want a simple checkbox to toggle between the options. This method will allow a single checkbox to select ALL columns

### Details

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |

### Parameters

fieldName

The database field name

display\_data

Show or do not show the value when clicking it (true or false)

### Example

$tblDemo->showCheckboxAll('fldField3', true);   
  
Click for active example [here](https://ajaxcrud.com/examples/demo-showcheckboxes.php)

### See also

[defineCheckbox](https://ajaxcrud.com/api/index.php?id=defineCheckbox), [defineAllowableValues](https://ajaxcrud.com/api/index.php?id=defineAllowableValues).

**showCSVExportOption**

Displays a button which outputs table to CSV format

showCSVExportOption()

Use this function if you want to be able to allow users to output your table to a CSV file ("comma-separated-values" aka "comma delimited file" aka "Excel format").

**Details**

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |

**Example**

$tblMyTable->showCSVExportOption();

## showOnly

show only a few fields from your table instead of all of them

void showOnly ( String fieldList )

Use this function when you have a large table and only want to show a FEW fields by default instead of all of them. This prevents you from having to call [omitField](https://ajaxcrud.com/api/index.php?id=omitField), [omitAddField](https://ajaxcrud.com/api/index.php?id=omitAddField) or [omitFieldCompletely](https://ajaxcrud.com/api/index.php?id=omitFieldCompletely) for each field you don't want to show.   
  
You can also use this function to REORDER your database fields (different from the order defined by your schema), if desired.   
  
You can also use this function to ADD custom-defined fields to your HTML table, which do **not** necessarily need to exist in your db table. Then you can apply custom display logic on them with [formatFieldWithFunction](https://ajaxcrud.com/api/index.php?id=formatFieldWithFunction) or [formatFieldWithFunctionAdvanced](https://ajaxcrud.com/api/index.php?id=formatFieldWithFunctionAdvanced).

### Details

|  |  |
| --- | --- |
| Type | function |
| Returns | void |
| Default | all fields |

### Parameters

**fieldList**: A list of fields (comma delimited) you want to show.

### Example: $tblDemo->showOnly("fldTitle, fldName, fldPhone");

**Example 2:**

$tblDemo->showOnly("fldTitle, fldName, fld\_not\_in\_db, fldPhone");

$tblDemo->formatFieldWithFunctionAdvanced("fld\_not\_in\_db",

"getCarPurchaseDate");

function getCarPurchaseDate($data, $id){

//tblCar can be ANOTHER table completely

$carInfo = qr("SELECT fldPurchaseDate

FROM tblCar WHERE fkCarBuyerID = $id");

$purchaseDate = $carInfo['fldPurchaseDate'];

return $purchaseDate;

}

### See also

[omitField](https://ajaxcrud.com/api/index.php?id=omitField), [omitAddField](https://ajaxcrud.com/api/index.php?id=omitAddField), [omitFieldCompletely](https://ajaxcrud.com/api/index.php?id=omitFieldCompletely), [formatFieldWithFunctionAdvanced](https://ajaxcrud.com/api/index.php?id=formatFieldWithFunctionAdvanced).

## showTable

function to "run" the script / display the table

void showTable ()

This function must be called AFTER all other functions are called. It effectively shows the table and allows the CRUD operations.

This function can only be called once

### Details

|  |  |
| --- | --- |
| Requires | [ajaxCRUD](https://ajaxcrud.com/api/index.php?id=ajaxCRUD) (the constructor) |
| Type | function |
| Returns | void (displays the table) |

### Parameters

none

### Example

$tblDemo->showTable();

### See also

[ajaxCRUD](https://ajaxcrud.com/api/index.php?id=ajaxCRUD) [addTableBorder](https://ajaxcrud.com/api/index.php?id=addTableBorder).

## turnOffAjaxADD

Turns off ajax for the CREATE crud operation (adding a row).

void turnOffAjaxADD()

Use this method if you want ADDING a row to be done via POST (traditional way) instead of via ajax.   
  
***Note:*** this happens AUTOMATICALLY if you use file handling ([setFileUpload](https://ajaxcrud.com/api/index.php?id=setFileUpload)) OR want to use a callback method when adding a row ([onAddExecuteCallBackFunction](https://ajaxcrud.com/api/index.php?id=onAddExecuteCallBackFunction)).

### Details

|  |  |
| --- | --- |
| Type | function |
| Returns | void |
| Default | (ajax used for adding a row) |

### Parameters

none

### Example

$tblDemo->turnOffAjaxADD();

### See also

[onAddExecuteCallBackFunction](https://ajaxcrud.com/api/index.php?id=onAddExecuteCallBackFunction), [setFileUpload](https://ajaxcrud.com/api/index.php?id=setFileUpload), [disallowAdd](https://ajaxcrud.com/api/index.php?id=disallowAdd), [turnOffPaging](https://ajaxcrud.com/api/index.php?id=turnOffPaging), [turnOffAjaxEditing](https://ajaxcrud.com/api/index.php?id=turnOffAjaxEditing), [turnOffSorting](https://ajaxcrud.com/api/index.php?id=turnOffSorting).

## turnOffAjaxEditing

turns off the ability to edit ALL fields.

void turnOffAjaxEditing ()

Use this function if you do not want to allow editing of any fields (eg 'view' only access).

### Details

|  |  |
| --- | --- |
| Type | function |
| Returns | void |

### Example

$tblDemo->turnOffAjaxEditing()

### See also

[turnOffPaging](https://ajaxcrud.com/api/index.php?id=turnOffPaging), [turnOffAjaxADD](https://ajaxcrud.com/api/index.php?id=turnOffAjaxADD), [turnOffSorting](https://ajaxcrud.com/api/index.php?id=turnOffSorting), [disallowEdit](https://ajaxcrud.com/api/index.php?id=disallowEdit), [disallowAdd](https://ajaxcrud.com/api/index.php?id=disallowAdd), [disallowDelete](https://ajaxcrud.com/api/index.php?id=disallowDelete).

## turnOffPaging

turn off paging so total rows returns maxes at a value

void turnOffPaging( int maxRowsToReturn )

Use this function if you have a large database (thousands or millions of records) and want to limit number of rows returned.

### Details

|  |  |
| --- | --- |
| **Type** | function |
| **Returns** | void |
| **Default** | paging at 50/page |

### Parameters

maxRowsToReturn

integer for number you want to return (max)

### Example

$tblDogs->turnOffPaging(100);

### See also

[setLimit](https://ajaxcrud.com/api/index.php?id=setLimit), [turnOffAjaxADD](https://ajaxcrud.com/api/index.php?id=turnOffAjaxADD), [turnOffAjaxEditing](https://ajaxcrud.com/api/index.php?id=turnOffAjaxEditing), [turnOffSorting](https://ajaxcrud.com/api/index.php?id=turnOffSorting).

## turnOffSorting

Turns off the ability to sort table by clicking the header titles

void turnOffSorting ()

Use this function if you do not want to show the html table headers (i.e. if they are obvious by looking at the data)

### Details

|  |  |
| --- | --- |
| Type | function |
| Returns | void |
| Default | sorting: on |

### Example

$tblDemo->turnOffSorting()

### See also

[turnOffAjaxEditing](https://ajaxcrud.com/api/index.php?id=turnOffAjaxEditing), [turnOffPaging](https://ajaxcrud.com/api/index.php?id=turnOffPaging), [turnOffAjaxADD](https://ajaxcrud.com/api/index.php?id=turnOffAjaxADD), [disallowEdit](https://ajaxcrud.com/api/index.php?id=disallowEdit), [disallowAdd](https://ajaxcrud.com/api/index.php?id=disallowAdd), [disallowDelete](https://ajaxcrud.com/api/index.php?id=disallowDelete).

## validateDeleteWithFunction

Allows you to set logic allowing if deletion of certain rows is permissible or not

void validateDeleteWithFunction ( String functionName )

Use this function when you wish to use logic to determine if a row can be deleted or not. In other words - you can base the decision on data in OTHER fields in the row. As such, the user-defined function will have access to the id (primary key) of the row to make logic-based decision and must return TRUE or FALSE. The parameters of your custom function (which YOU define) must take in ONE (1) argument ($id) and return TRUE or FALSE.

### Details

|  |  |
| --- | --- |
| Requires | a user-defined function to be set |
| Type | function |
| Returns | user-defined method returns true or false |

### Parameters

**functionName:** name of the user-defined function to be invoked

### Example

$tblUser->validateDeleteWithFunction("canRowBeModifiedOrDeleted");

# this function returns true or false, depending

# on if his IP address matches the stored ip address

# for the user (i.e. very lame authentication)

function canRowBeModifiedOrDeleted($id){

$ip = q1("SELECT fldIPAddress FROM tblUser

WHERE pkUserID = $id");

if ($ip == $\_SERVER["REMOTE\_ADDR"])

return true;

return false;

}

### See also

[formatFieldWithFunctionAdvanced](https://ajaxcrud.com/api/index.php?id=formatFieldWithFunctionAdvanced), [validateUpdateWithFunction](https://ajaxcrud.com/api/index.php?id=validateUpdateWithFunction).

## validateUpdateWithFunction

Allows you to set logic allowing if updating a row is permissible or not

void validateUpdateWithFunction ( String functionName )

Use this function when you wish to use logic to determine if a row can be updated or not. In other words - you can base the decision on data in OTHER fields in the row. As such, the user-defined function will have access to the id (primary key) of the row to make logic-based decision and must return TRUE or FALSE. The parameters of your custom function (which YOU define) must take in ONE (1) argument ($id) and return TRUE or FALSE.

### Details

|  |  |
| --- | --- |
| Requires | a user-defined function to be set |
| Type | function |
| Returns | user-defined method returns true or false |

### Parameters

functionName

name of the user-defined function to be invoked

### Example

$tblUser->validateUpdateWithFunction("canRowBeModifiedOrDeleted");

# this function returns true or false, depending

# on if his IP address matches the stored ip address

# for the user (i.e. very lame authentication)

function canRowBeModifiedOrDeleted($id){

$ip = q1("SELECT fldIPAddress FROM tblUser

WHERE pkUserID = $id");

if ($ip == $\_SERVER["REMOTE\_ADDR"])

return true;

return false;

}

### See also

[formatFieldWithFunctionAdvanced](https://ajaxcrud.com/api/index.php?id=formatFieldWithFunctionAdvanced), [validateDeleteWithFunction](https://ajaxcrud.com/api/index.php?id=validateDeleteWithFunction).

**Variables**

## actionText

Change the text 'Action' that appears in the table header

### Details

|  |  |
| --- | --- |
| **Type** | variable |
| **Default value** | Action |

### Example

$tblDemo->actionText = "Do Something";

### See also

[addText](https://ajaxcrud.com/api/index.php?id=addText), [deleteText](https://ajaxcrud.com/api/index.php?id=deleteText), [fileDeleteText](https://ajaxcrud.com/api/index.php?id=fileDeleteText), [fileEditText](https://ajaxcrud.com/api/index.php?id=fileEditText).

## addButtonText

Overwrites the text of the Add button

Use this variable if you want to change the text that the add button displays

### Details

|  |  |
| --- | --- |
| **Type** | variable |
| **Default value** | Add {Item} |

### Example

$tblDemo->addButtonText = "Register Me for this Event";

### See also

[addText](https://ajaxcrud.com/api/index.php?id=addText).

## addButtonToRowWindowOpen

When a button is added to row ([addButtonToRow](https://ajaxcrud.com/api/index.php?id=addButtonToRow)) the window in which to open page (ALL buttons).

Only valid options are "same" and "new". This variable will affect ALL buttons. IF you want to influence only specific buttons, use the 5th parameter on [addButtonToRow](https://ajaxcrud.com/api/index.php?id=addButtonToRow) function

### Details

|  |  |
| --- | --- |
| **Type** | variable |
| **Default value** | same |

### Example

$tblDemo->addButtonToRowWindowOpen = "same";

### See also

[addButtonToRow](https://ajaxcrud.com/api/index.php?id=addButtonToRow).

## addMessage

The message that displays after you add a new item (and you're not using ajax adding)

Use this variable to change the confirmation message that appears after adding an item to the database

### Details

|  |  |
| --- | --- |
| **Type** | variable |
| **Default value** | {Item} added |

### Example

$tblDemo->addMessage = "Thank you for adding your entry";

### See also

[emptyTableMessage](https://ajaxcrud.com/api/index.php?id=emptyTableMessage), [addText](https://ajaxcrud.com/api/index.php?id=addText).

## addText

The word that displays in the button for adding a new row

### Details

|  |  |
| --- | --- |
| **Type** | variable |
| **Default value** | Add |

### Example

$tblDemo->addText= "Añadir"

### See also

[deleteText](https://ajaxcrud.com/api/index.php?id=deleteText).

## ajaxcrud\_where\_clause

Most recent/active sql WHERE clause used by the [ajaxcrud](https://ajaxcrud.com/api/index.php?id=ajaxCRUD) table.

This is a SESSION variable which can be leveraged in any (other) php page and contains the active WHERE clause used by [ajaxCRUD](https://ajaxcrud.com/api/index.php?id=ajaxCRUD). Introduced in version v8.64. It is set automatically in the [addWhereClause](https://ajaxcrud.com/api/index.php?id=addWhereClause) method and used for both filtering and manually-defined WHERE conditions. It is stored in a bi-dimentional array, so you need to specify "[ajaxcrud](https://ajaxcrud.com/api/index.php?id=ajaxCRUD)\_where\_clause" as index 0 and then the table name as index 1.   
  
See example file */examples/where\_clause\_test.php*

### Details

|  |  |
| --- | --- |
| **Requires** | session\_start(); |
| **Type** | variable |
| **Default value** |  |

### Example

if (isset($\_SESSION['ajaxcrud\_where\_clause']['tblDemo'])){

$whereClause = $\_SESSION['ajaxcrud\_where\_clause']['tblDemo'];

echo "ajaxCRUD where clause: " . $whereClause;

}

else{

echo "No WHERE clause set (yet)";

}

## cancelText

The word that displays in the button for Cancelling out of an add form (slides the form up)

### Details

|  |  |
| --- | --- |
| **Type** | variable |
| **Default value** | Cancel |

### Example

$tblDemo->cancelText= "Añadir"

### See also

[addText](https://ajaxcrud.com/api/index.php?id=addText), [deleteText](https://ajaxcrud.com/api/index.php?id=deleteText).

**cellspacing**

cellspacing for the html table that displays the database table

**Details**

|  |  |
| --- | --- |
| **Type** | variable |
| **Default value** | 0 |

**Example**

$demo->cellspacing = 5;

## css

set to false if you don't want to use a css stylesheet

Use this if you format your html tables using a central stylesheet (and don't want to use a 'plug in' one)

### Details

|  |  |
| --- | --- |
| **Type** | variable |
| **Default value** | true |

### See also

[setCSSFile](https://ajaxcrud.com/api/index.php?id=setCSSFile).

## deleteText

The word that displays in the button for deleting an existing row

### Details

|  |  |
| --- | --- |
| **Type** | variable |
| **Default value** | Delete |

### Example

$tblDemo->deleteText= "Borrar"

### See also

[addText](https://ajaxcrud.com/api/index.php?id=addText).

## doActionOnShowTable

when set to true CRUD actions will happen when [showTable](https://ajaxcrud.com/api/index.php?id=showTable)() is called

CRUD operations will happen (by default) when the [showTable](https://ajaxcrud.com/api/index.php?id=showTable)() function is called; however, this may not always be desired -- for example when executing callback functions like [onAddExecuteCallBackFunction](https://ajaxcrud.com/api/index.php?id=onAddExecuteCallBackFunction) which might require headers not to bet set. Set this to FALSE if you want to have more control over the execution path

### Details

|  |  |
| --- | --- |
| Type | variable |
| Default value | true |

### Example

$obj = new ajaxCRUD("Me", "tblEventAttendee",

"pkAttendeeID", "../ajaxcrud/");

//ensures headers won't be set first

$obj->doActionOnShowTable = false;

$obj->onAddExecuteCallBackFunction("onAddCallBackFunction");

// this needs to be very low in the excution path to

// ensure the obj is loaded correctly with every other piece of logic

if (isset($\_REQUEST['action']) && $\_REQUEST['action'] != ''){

$obj->doAction($\_REQUEST['action']);

}

function onAddCallBackFunction($array){

global $idsAdded;

//print\_r($array);

$newID = $array['pkAttendeeID'];

$idsAdded[] = $newID; //array push

//set cookie 10 years in the future

setcookie(

"ajaxCRUDEventID\_$newID",

"cookieset",

time() + (10 \* 365 \* 24 \* 60 \* 60)

);

//print\_r($\_COOKIE);

}

### See also

[onAddExecuteCallBackFunction](https://ajaxcrud.com/api/index.php?id=onAddExecuteCallBackFunction).

**emptyTableMessage**

This message that appears when a table is empty.

**Details**

|  |  |
| --- | --- |
| **Type** | variable |
| **Default value** |  |

**Example**

$tblDemo->emptyTableMessage = "No rows in this table. Add one using the Add button below."

## fileDeleteText

Change the word that displays for the link that handles file deletions.

### Details

|  |  |
| --- | --- |
| **Type** | variable |
| **Default value** | del |

### Example

$tblDemo->fileDeleteText = "Delete This File";

### See also

[fileEditText](https://ajaxcrud.com/api/index.php?id=fileEditText).

## fileEditText

Change the word that displays for the link that handles file updates.

### Details

|  |  |
| --- | --- |
| **Type** | variable |
| **Default value** | edit |

### Example

$tblDemo->fileEditText = "Replace File";

### See also

[fileDeleteText](https://ajaxcrud.com/api/index.php?id=fileDeleteText).

## hover\_color

color when you hover over a cell to change its value

(this function is deprecicated. now use stylesheets via the [setCSSFile](https://ajaxcrud.com/api/index.php?id=setCSSFile) function)

### Details

|  |  |
| --- | --- |
| **Type** | variable |
| **Default value** | #B1CFF5 (a light blue) |

### Example

$demo->hover\_color = "#E9E8DC";

## showCheckbox

puts a checkbox in front of each row

Use this function when you want to put a checkbox in front of each row; when clicked it will pass the primary key of the row (with the value of the primary key) back to the table.   
  
You can do the same type of thing with the [addButtonToRow](https://ajaxcrud.com/api/index.php?id=addButtonToRow) function.

### Details

|  |  |
| --- | --- |
| **Type** | variable |
| **Default value** | false |

### Example

$tblUserAlerts->showCheckbox = true;

### See also

[defineCheckbox](https://ajaxcrud.com/api/index.php?id=defineCheckbox), [addButtonToRow](https://ajaxcrud.com/api/index.php?id=addButtonToRow).