

# Custom Object Creation and Object Sharing Table

When we create a custom object say **Quote\_\_c**, the platform will default it's OWD to **Public Read/Write**.

UI:

View the custom objects: <https://na34.salesforce.com/p/setup/custent/CustomObjectsPage>

Create a custom object: <https://na34.salesforce.com/011/e?setupid=CustomObjects>

Custom Object

Quote

Standard Fields [4]

Custom Fields & Relationships [0]

Validation Rules [0]

Page Layouts [1]

Field Sets [0]

Compact Layouts [1]

Buttons, Links, and Actions [8]

Record Types [0]

Apex Sharing Reasons [0]

Apex Sharing Recalculation [0]

Object Limits [10]

Custom Object Definition Detail

EditDelete

Singular Label	Quote	Description
Plural Label	Quotes	Enable Reports <input type="checkbox"/>
Object Name	Quote	Track Activities <input type="checkbox"/>
API Name	Quote__c	Allow in Chatter Groups <input type="checkbox"/>
		Allow Sharing <input checked="" type="checkbox"/>
		Allow Bulk API Access <input checked="" type="checkbox"/>
		Allow Streaming API Access <input checked="" type="checkbox"/>
		Track Field History <input type="checkbox"/>
		Deployment StatusDeployed
		Allow Search <input type="checkbox"/>
		Help SettingsStandard salesforce.com Help Window
Created By	Mohan Chinnappan, 12/30/2015 9:16 AM	Modified ByMohan Chinnappan, 12/30/2015 9:16 AM

OWD of the newly created **Quote\_\_c** (Object Record Table)

Quote	Public Read/Write	Public Read/Write	<input checked="" type="checkbox"/>
-------	-------------------	-------------------	-------------------------------------

Since OWD for this object is least restrictive, Object Sharing Table (**Quote\_\_share**) is not yet created by the platform

describe Quote\_\_share

⚡

✖

⬅

➡

1

The requested resource does not exist

Let us make it less restrictive by changing OWD for this object to **Private**:

Quote	Private	Public Read/Write	<input checked="" type="checkbox"/>
-------	---------	-------------------	-------------------------------------

This will make the platform to setup Object Sharing Table (**Quote\_\_share**) along with other things. This is a background task. The Admin user who made this change will be notified when this background task is completed:

This page displays your organization's sharing settings. These settings specify the level of access your users have to each others' data.

An organization-wide default update has been initiated by Mohan Chinnappan on 12/30/2015 9:25 AM. You can't submit any changes prior to the completion of the operation. Mohan Chinnappan will receive an email when the operation finishes.

Let check the availability of the share table: **Quote\_\_share**

```
describe Quote__share
```

⚡

✖

⬅

➡

1

Response

Table

History

Sharing ERD

Arch

Sharing Table

Industries

Roles

Dict

Dict CSV

Drawings

Machine Learning

SLDS

Label	Name	Nullable	Picklist values	Reference to	Relationship name	Type	Unique	Updateable
Custom Object Share ID	Id	false	--	--	null	id	false	false
Parent ID	ParentId	false	--	[ "Quote__c" ]	Parent	reference	false	false
User/Group ID	UserOrGroupId	false	--	[ "Group", "User" ]	null	reference	false	false
Custom Object Access	AccessLevel	false	[ "Read", "Edit", "All" ]	--	null	picklist	false	true
Row Cause	RowCause	true	[ "Owner", "Manual", "Rule", "ImplicitChild", "ImplicitParent", "Team", "Territory", "TerritoryManual", "TerritoryRule", "Territory2Forecast" ]	--	null	picklist	false	false

Let us query Quote\_\_share: No records yet!:

```
SELECT ParentId, UserOrGroupId, AccessLevel, RowCause FROM Quote__share
```

⚡

✖

⬅

➡

1

Response

Table

History

Sharing ERD

Arch

Sharing Table

Industries

Roles

Dict

Dict CSV

Drawings

Machine Learning

SLDS

```
1 {
2   "totalSize": 0,
3   "done": true,
4   "records": []
5 }
```

Let us **create** a record in Object Record Table: Quote\_\_c:

```
$ cat newQuote.json
{
  "name": "New Quote-1"
}

$ curl https://na34.salesforce.com/services/data/v33.0/subjects/Quote__c/
-H "Content-Type: application/json"
-d @newQuote.json
-H 'Authorization: Bearer 00D61000000dcQB!AQ4AQPqcU.5ect3zyKjaSlf3ebSi.WTIWW4Np4W.B8SKgADRFsm9Aq.OmRf8MikJQukNrNtzVZ9iiM93m

$ {"id":"a0661000002hup1AAA","success":true,"errors":[]}
```

Now, let us query Quote\_\_share:

```
SELECT ParentId, UserOrGroupId, AccessLevel, RowCause
FROM Quote__share
```



Response	Table	History	Sharing ERD	Arch	Sharing Table	Industries	Roles	Dict	Dict CSV	Drawings	Machine Learning
----------	-------	---------	-------------	------	---------------	------------	-------	------	----------	----------	------------------

SLDS

ParentId		UserOrGroupId	AccessLevel	RowCause
1	a0661000002hup1AAA	0056100000195Y2AAI	All	Owner

You can see a tuple (record) with RowCase: **Owner** and AccessLevel: **All** This is the work of the platform's managed sharing: providing full access to the record owner.

Now we can create a Apex class to provide Read access to non-owner of this record.

Let us find the users in the org:

```
SELECT id,name FROM User
```



Response	Table	History	Sharing ERD	Arch	Sharing Table	Industries	Roles	Dict	Dict CSV	Drawings	Machine Learning
----------	-------	---------	-------------	------	---------------	------------	-------	------	----------	----------	------------------

SLDS

	<b>Id</b>	<b>Name</b>
1	0056100000195Y2AAI	Mohan Chinnappan
2	0056100000195Y9AAI	Integration User
3	0056100000195YAAAY	Security User
4	00561000001VmT8AAK	Joe Simple
5	0056100000195Y8AAI	Chatter Expert

Here we will provide Read access to the user Joe Simple (id:00561000001VmT8AAK aka UserOrGroupId in Record Sharing table) for this record (id:a0661000002hup1AAA aka ParentId in Record Sharing table )

Create Apex class: QuoteSharing with method:

```
manualShareRead(ParentId, userOrGroupId)
```

UI: <https://na34.salesforce.com/01p>

```

public class QuoteSharing {

    public static boolean manualShareRead(Id recordId, Id userOrGroupId){
        // Create new sharing object for the custom object Quote.
        Quote__Share quoteShr  = new Quote__Share();

        // Set the ID of record being shared.
        quoteShr.ParentId = recordId;

        // Set the ID of user or group being granted access.
        quoteShr.UserOrGroupId = userOrGroupId;

        // Set the access level.
        quoteShr.AccessLevel = 'Read';

        // Set rowCause to 'manual' for manual sharing.
        // This line can be omitted as 'manual' is the default value for sharing objects.
        quoteShr.RowCause = Schema.Job__Share.RowCause.Manual;

        // Insert the sharing record and capture the save result.
        // The false parameter allows for partial processing if multiple records passed
        // into the operation.
        Database.SaveResult sr = Database.insert(quoteShr,false);

        // Process the save results.
        if(sr.isSuccess()){
            // Indicates success
            return true;
        }
        else {
            // Get first save result error.
            Database.Error err = sr.getErrors()[0];

            // Check if the error is related to trival access level.
            // Access levels equal or more permissive than the object's default
            // access level are not allowed.
            // These sharing records are not required and thus an insert exception is acceptable.
            if(err.getStatusCode() == StatusCode.FIELD_FILTER_VALIDATION_EXCEPTION  &&
                err.getMessage().contains('AccessLevel')){
                // Indicates success.
                return true;
            }
            else{
                // Indicates failure.
                return false;
            }
        }
    }
}

```

Running using tooling API:

```
QuoteSharing.manualShareRead('a0661000002hup1AAA', '00561000001VmT8AAK');
```

The screenshot shows the Salesforce Developer Console interface. At the top, there's a text area with the following code: `/tooling/executeAnonymous/?anonymousBody=QuoteSharing.manualShareRead('a0661000002hup1AAA', '00561000001VmT8AAK');`. Below the text area are several icons: a red lightning bolt, a blue 'X', a blue left arrow, a blue right arrow, and a grey circle with the number '1'. Below these icons is a horizontal menu with tabs: 'Response' (selected), 'Table', 'History', 'Sharing ERD', 'Arch', 'Sharing Table', 'Industries', 'Roles', 'Dict', 'Dict CSV', 'Drawings', and 'Machine Learning'. Below the menu, the word 'SLDS' is displayed. At the bottom, there's a code editor showing the JSON response:

```

1 {
2   "line": -1,
3   "column": -1,
4   "compiled": true,
5   "success": true,
6   "compileProblem": null,
7   "exceptionStackTrace": null,
8   "exceptionMessage": null
9 }

```

Now, query **Quote\_\_share**:

```
SELECT ParentId, UserOrGroupId, AccessLevel, RowCause FROM Quote__share
```



[Response](#) [Table](#) [History](#) [Sharing ERD](#) [Arch](#) [Sharing Table](#) [Industries](#) [Roles](#) [Dict](#) [Dict CSV](#) [Drawings](#) [Machine Learning](#)

SLDS

	ParentId	UserOrGroupId	AccessLevel	RowCause
1	a0661000002hup1AAA	0056100000195Y2AAI	All	Owner
2	a0661000002hup1AAA	00561000001VmT8AAK	Read	Manual

You can see new tuple (record) with RowCase: **Manual** and AccessLevel: **Read**

This is the result of our work: **User Managed Sharing** Using **Apex**.

## Apex Managed Sharing:

Only users with “**Modify All Data**” permission can add or change Apex managed sharing on a record. Apex managed sharing is maintained across record owner changes.

Apex managed sharing uses **Apex sharing reason** (provide developers ability to track why they shared a record with a user or group of users)

Let us create an **Apex sharing reason**:

Deployment Status

Monitor

System Overview

Imports

Outbound Messages

Time-Based Workflow

Automated Process Actions

Case Escalations

API Usage Notifications

Mass Emails

Email Snapshots

Jobs

Logs

Compact Layouts

New Compact Layout Assignment Compact Layouts Help

Action	Label	API Name	Primary	Modified By
	System Default	SYSTEM	✓	

Buttons, Links, and Actions

New Action New Button or Link Buttons, Links, and Actions Help

Action	Label	Name	Description	Type	Content Source	Icon	Overridden
Edit	Accept	Accept			Standard Salesforce.com Page		<input type="checkbox"/>
Edit	Clone	Clone			Standard Salesforce.com Page		<input type="checkbox"/>
Edit	Delete	Delete			Standard Salesforce.com Page		<input type="checkbox"/>
Edit	Edit	Edit			Standard Salesforce.com Page		<input type="checkbox"/>
Edit	List	List			Standard Salesforce.com Page		<input type="checkbox"/>
Edit	New	New			Standard Salesforce.com Page		<input type="checkbox"/>
Edit	Quotes Tab	Tab			Standard Salesforce.com Page		<input type="checkbox"/>
Edit	View	View			Standard Salesforce.com Page		<input type="checkbox"/>

Record Types

New Page Layout Assignment Record Types Help

No record types defined

Apex Sharing Reasons

New Apex Sharing Reasons Apex Sharing Reasons Help

No Apex sharing reasons defined.

Apex Sharing Recalculation

New Recalculate Apex Sharing Apex Sharing Recalculation Help

No Apex sharing recalculations defined

Object Limits

Object Limits Help

Item	Usage	Limit	% used	Message
Custom Fields	0	500	0%	



## New Apex Sharing Reason

Apex sharing reasons are used by developers when adding sharing to a record programmatically. Using an apex sharing reason prevents standard users from deleting the sharing, and allows the developer to track why they added the sharing.

**Apex Sharing Reason Edit**

Save

The Label is shown on the sharing detail page for a record.

Reason Label

Quote Write Sharing

i

Reason Name

Quote\_Write\_Sharing

i

Save

Save & New

Cancel

### Apex Sharing Reasons

New

Apex Sharing Reasons Help ?

Action	Label	Modified By
<a href="#">Edit</a>   <a href="#">Del</a>	<a href="#">Quote Write Sharing</a>	Mohan Chinnappan, 12/30/2015 1:22 PM

Check the Apex Sharing Reason:

https://na34.salesforce.com/\_ui/common/apex/debug/ApexCSI

File Edit Debug Test Workspace Help < >

Log executeAnonymous @12/30/2015, 4:25:23 PM

**Execution Log**

Timestamp	Event
16:25:23:022	EXECUTION_STARTED
16:25:23:022	CODE_UNIT_STARTED
16:25:23:022	HEAP_ALLOCATE
16:25:23:023	HEAP_ALLOCATE
16:25:23:023	HEAP_ALLOCATE
16:25:23:023	HEAP_ALLOCATE
16:25:23:023	HEAP_ALLOCATE
16:25:23:023	HEAP_ALLOCATE
16:25:23:023	HEAP_ALLOCATE
16:25:23:023	STATEMENT_EXECUTE
16:25:23:023	STATEMENT_EXECUTE
16:25:23:023	HEAP_ALLOCATE
16:25:23:023	HEAP_ALLOCATE
16:25:23:023	HEAP_ALLOCATE
16:25:23:023	HEAP_ALLOCATE
16:25:23:023	USER_DEBUG
16:25:23:023	CUMULATIVE_LIMIT_USAGE
16:25:23:023	LIMIT_USAGE_FOR_NS
16:25:23:000	LIMIT_USAGE_FOR_NS
16:25:23:000	LIMIT_USAGE_FOR_NS
16:25:23:000	LIMIT_USAGE_FOR_NS

☐ This Frame ☐ Executable ☐ Debug Only ☐ Filter [Click here to filter the log](#)

**Logs** Tests Checkpoints Query Editor View State Progress Pro

User	Application	Operation
Mohan Chinnappan	000000000001007	/services/data/v33.0/tooling/executeAnonymous/
Mohan Chinnappan	000000000001007	/services/data/v35.0/tooling/executeAnonymous/
Mohan Chinnappan	000000000001007	/services/data/v35.0/tooling/executeAnonymous/
Mohan Chinnappan	000000000001007	/services/data/v35.0/tooling/executeAnonymous/
Mohan Chinnappan	000000000001007	/services/data/v35.0/tooling/executeAnonymous/
Mohan Chinnappan	000000000001007	/services/data/v35.0/tooling/executeAnonymous/

ToolSet

SALESFORCE PLUS SHARING DEBUGGER SOAP CONSOLE CPQ WORKERS

/tooling/executeAnonymous/?anonymousBody=System.debug(Schema.Quote\_\_Share.rowCause.Quote\_Write\_Sharing\_\_c);

⚡

✕

←

→

2

Response

Table

History

Sharing ERD

Arch

Sharing Table

Industries

Roles

Dict

Dict

SLDS

```
1 {
2   "line": -1,
3   "column": -1,
4   "compiled": true,
5   "success": true,
6   "compileProblem": null,
7   "exceptionStackTrace": null,
8   "exceptionMessage": null
9 }
```

Let us create an Apex class to do Apex Managed Sharing:

```

public class QuoteApexManagedSharing {

    public static boolean addShareEdit(Id recordId, Id userOrGroupId){
        // Create new sharing object for the custom object Quote.
        Quote__Share quoteShr = new Quote__Share();

        // Set the ID of record being shared.
        quoteShr.ParentId = recordId;

        // Set the ID of user or group being granted access.
        quoteShr.UserOrGroupId = userOrGroupId;

        // Set the access level.
        quoteShr.AccessLevel = 'edit';

        // Set rowCause to Schema.Quote__Share.rowCause.Quote_Write_Sharing__c
        quoteShr.RowCause = Schema.Quote__Share.rowCause.Quote_Write_Sharing__c;

        // Insert the sharing record and capture the save result.
        // The false parameter allows for partial processing if multiple records passed
        // into the operation.
        Database.SaveResult sr = Database.insert(quoteShr,false);

        // Process the save results.
        if(sr.isSuccess()){
            // Indicates success
            return true;
        }
        else {
            // Get first save result error.
            Database.Error err = sr.getErrors()[0];

            // Check if the error is related to trivial access level.
            // Access levels equal or more permissive than the object's default
            // access level are not allowed.
            // These sharing records are not required and thus an insert exception is acceptable.
            if(err.getStatusCode() == StatusCode.FIELD_FILTER_VALIDATION_EXCEPTION &&
                err.getMessage().contains('AccessLevel')){
                // Indicates success.
                return true;
            }
            else{
                // Indicates failure.
                return false;
            }
        }
    }
}

```

Run using tooling API:

```
QuoteApexManagedSharing.addShareEdit('a0661000002hup1AAA', '00561000001VmT8AAK');
```

```
/tooling/executeAnonymous/?anonymousBody=QuoteApexManagedSharing.addShareEdit('a0661000002hup1AAA', '00561000001VmT8AAK');
```

Response

Table

History

Sharing ERD

Arch

Sharing Table

Industries

Roles

Dict

Dict CSV

Drawings

Machine Learning

SLDS

```

1 {
2   "line": -1,
3   "column": -1,
4   "compiled": true,
5   "success": true,
6   "compileProblem": null,
7   "exceptionStackTrace": null,
8   "exceptionMessage": null
9 }

```

Execution Log		
Timestamp	Event	Details
17:00:04:089	VARIABLE_SCO...	[23] sr Database.SaveResult true raise
17:00:04:089	VARIABLE_ASSI...	[23] sr {"success":true,"id":"02c6100000BC9RMAA1"} 0x1b01aac1
17:00:04:089	STATEMENT_EX...	[26]
17:00:04:090	STATEMENT_EX...	[28]
17:00:04:090	HEAP_ALLOCATE	[28] Bytes:20
17:00:04:090	USER_DEBUG	[28] DEBUG sharing reason added

View Log:

Query Object Sharing Table: **Quote\_\_share**:

SELECT ParentId, UserOrGroupId, AccessLevel, RowCause FROM Quote\_\_share

⚡

✕

←

→

3

Response

Table

History

Sharing ERD

Arch

Sharing Table

Industries

Roles

Dict

Dict CSV

Drawings

Machine Learning

SLDS

	ParentId	UserOrGroupId	AccessLevel	RowCause
1	a0661000002hup1AAA	0056100000195Y2AAI	All	Owner
2	a0661000002hup1AAA	00561000001VmT8AAK	Edit	Quote_Write_Sharing__c
3	a0661000002hup1AAA	00561000001VmT8AAK	Read	Manual

You can see new tuple (record) with RowCase: **Quote\_Write\_Sharing\_\_c** and AccessLevel: **Edit**

This is the result of our work: **Apex Managed Sharing** .

As you here, Object-id: a0661000002hup1AAA for the UserOrGroupId: 00561000001VmT8AAK has both **Read** and **Edit**, Platform will provide more permissive access (**Edit**, in this case) to this user.

## Notes

If a manual share access level is set to **Read** and you insert a new one that's set to **Write**, the original share rows are updated to **Write**, indicating the higher level of access

Testing this:

```
public class QuoteSharingGeneric {
```



```

public static boolean manualShare(Id recordId, Id userOrGroupId, String accessLevel){
    // Create new sharing object for the custom object Quote.
    Quote__Share quoteShr  = new Quote__Share();

    // Set the ID of record being shared.
    quoteShr.ParentId = recordId;

    // Set the ID of user or group being granted access.
    quoteShr.UserOrGroupId = userOrGroupId;

    // Set the access level.
    quoteShr.AccessLevel = accessLevel;

    // Set rowCause to 'manual' for manual sharing.
    // This line can be omitted as 'manual' is the default value for sharing objects.
    quoteShr.RowCause = Schema.Job__Share.RowCause.Manual;

    // Insert the sharing record and capture the save result.
    // The false parameter allows for partial processing if multiple records passed
    // into the operation.
    Database.SaveResult sr = Database.insert(quoteShr,false);

    // Process the save results.
    if(sr.isSuccess()){
        // Indicates success
        return true;
    }
    else {
        // Get first save result error.
        Database.Error err = sr.getErrors()[0];

        // Check if the error is related to trivial access level.
        // Access levels equal or more permissive than the object's default
        // access level are not allowed.
        // These sharing records are not required and thus an insert exception is acceptable.
        if(err.getStatusCode() == StatusCode.FIELD_FILTER_VALIDATION_EXCEPTION  &&
            err.getMessage().contains('AccessLevel')){
            // Indicates success.
            return true;
        }
        else{
            // Indicates failure.
            return false;
        }
    }
}
}

```

Run: QuoteSharingGeneric.manualShare('a0661000002hup1AAA', '00561000001VmT8AAK', 'Edit');

```
/tooling/executeAnonymous/?anonymousBody=QuoteSharingGeneric.manualShare('a0661000002hup1AAA', '00561000001VmT8AAK', 'Edit');
```

Response

Table

History

Sharing ERD

Arch

Sharing Table

Industries

Roles

Dict

Dict CSV

Drawings

Machine Learning

SLDS

```

1 {
2   "line": -1,
3   "column": -1,
4   "compiled": true,
5   "success": true,
6   "compileProblem": null,
7   "exceptionStackTrace": null,
8   "exceptionMessage": null
9 }

```

SELECT ParentId, UserOrGroupId, AccessLevel, RowCause FROM Quote\_\_Share



Response   **Table**   History   Sharing ERD   Arch   Sharing Table   Industries   Roles   Dict   Dict CSV   Drawings   Machine Learning

SLDS

	ParentId	UserOrGroupId	AccessLevel	RowCause
1	a0661000002hup1AAA	0056100000195Y2AAI	All	Owner
2	a0661000002hup1AAA	00561000001VmT8AAK	Edit	Quote_Write_Sharing__c
3	a0661000002hup1AAA	00561000001VmT8AAK	Edit	Manual