Apex Managed Sharing In Salesforce :

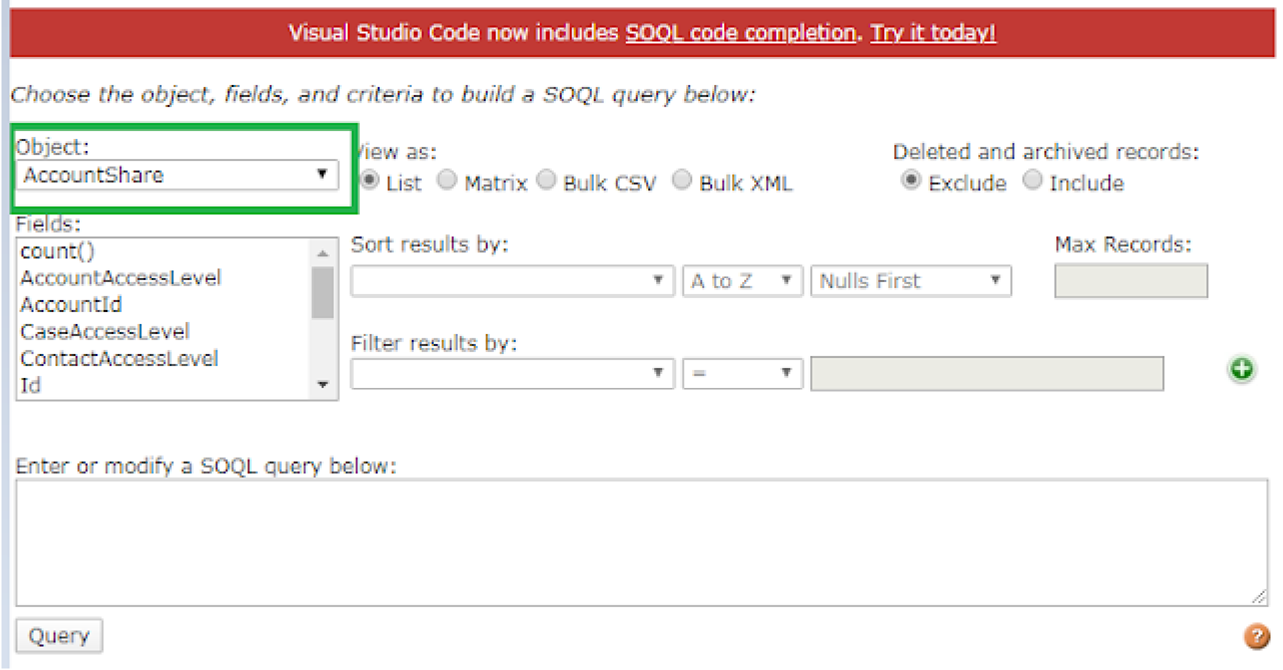
Reason For Usage Of Apex Sharing :

The client might required to share the record access to different set of users based on COMPLEX CRITERIA/CONDITIONS which cannot be done using sales force native features like Sharing Rules/OWD.

Apex Sharing :

Providing Sobject Access to targeted user by using Apex is called Apex Sharing.

By Default if OWD of the object is not set to most permissive access level(Public Read/Write) ,Then there will be automatically share object created for the existing object .
For Example if OWD for 'Account' is not set to Public Read/Write then 'AccountShare' object will be existing in the back-ground .
ScreenShot 1:



Visual Studio Code now includes SOQL code completion. Try it today!

Choose the object, fields, and criteria to build a SOQL query below:

Object: AccountShare

View as: ● List ○ Matrix ○ Bulk CSV ○ Bulk XML

Deleted and archived records: ● Exclude ○ Include

Fields:
count()
AccountAccessLevel
AccountId
CaseAccessLevel
ContactAccessLevel
Id

Sort results by: A to Z Nulls First

Max Records:

Filter results by: =

Enter or modify a SOQL query below:

Query

Share Object Technical Details :

Like Account object have different fields existed ,In the Same way Share object also contains the set of fields .Please find the below table.

Field Name	Field Description
ParentId	Salesforce ID of the record for which you wanted to share the access to targeted users.
AccessLevel	Level of access to be shared with target user or Group
UserOrGroupId	The ID of the User whom you wanted to share the access. The ID can also be Public Group ID or Role ID
RowCause(Apex Sharing Reason)	Reason for providing the access .This will differentiate weather access provided by OWD or Sharing Rules or Apex Sharing.

Scenario Based Example:

Prerequisite:

Objects Used in the discussion:

1.Project--Object contains all the business task details which require to be completed .

2.Project Member--Are the members who work on the project which has assigned to them.
Project Member is junction object which is master-Detail with Project and Contact

3.Contact--Sales force Contact Object
Assume all the contacts Owner is assigned with their own User ID .
Eg .We have user Kris who is also Owner for associated contact named Kris .

Use Case :

1.Whenever status of the project changed from 'New' to 'Assigned' The respective project record access should be shared with all the Project Members .

2.Once the Project Members completed their assigned task and Status of the Project changes to 'Completed' then Access should be Revoked/Removed from the Project Team Members .

Solution:

We require to create Apex Trigger on Project object and create the share object records and delete the access after Project completed

The below Trigger collects all the project records for which status changed to 'Assigned' Or changed to 'Completed' and send them to Apex class to create and Revoke the access respectively by using apex sharing .

Apex Trigger :

```
trigger ProjectShare on Project__c (before insert,after update) {

    List<Project__c > projectShareList=new List<Project__c >();

    List<Project__c > revokeShareList=new List<Project__c >();

    //Share the access

    if(Trigger.isafter&&Trigger.isUpdate){

        for(Project__c Pro:Trigger.new){

            if(Pro.Status__c=='Assigned'&&trigger.oldmap.get(Pro.id).Status__c!='Assigned'){

                projectShareList.add(Pro);

                system.debug('Share the access'+Pro.id);

            }

        }

        if(projectShareList!=null){

            ProjectShareclass.provideAccess(studyshareList);

        }


        //Revoke the access

        for(Project__c Pro:Trigger.new){

            if(Pro.Status__c!='Assigned'&&trigger.oldmap.get(Pro.id).Status__c=='Assigned'){

                system.debug('Revoke the access'+Pro.id);

                revokeShareList.add(std);

            }

        }
        if(revokeShareList!=null){

            ProjectShareclass.revokeAccess(revokeshareList);

        }
    }
}
```

Apex Class :

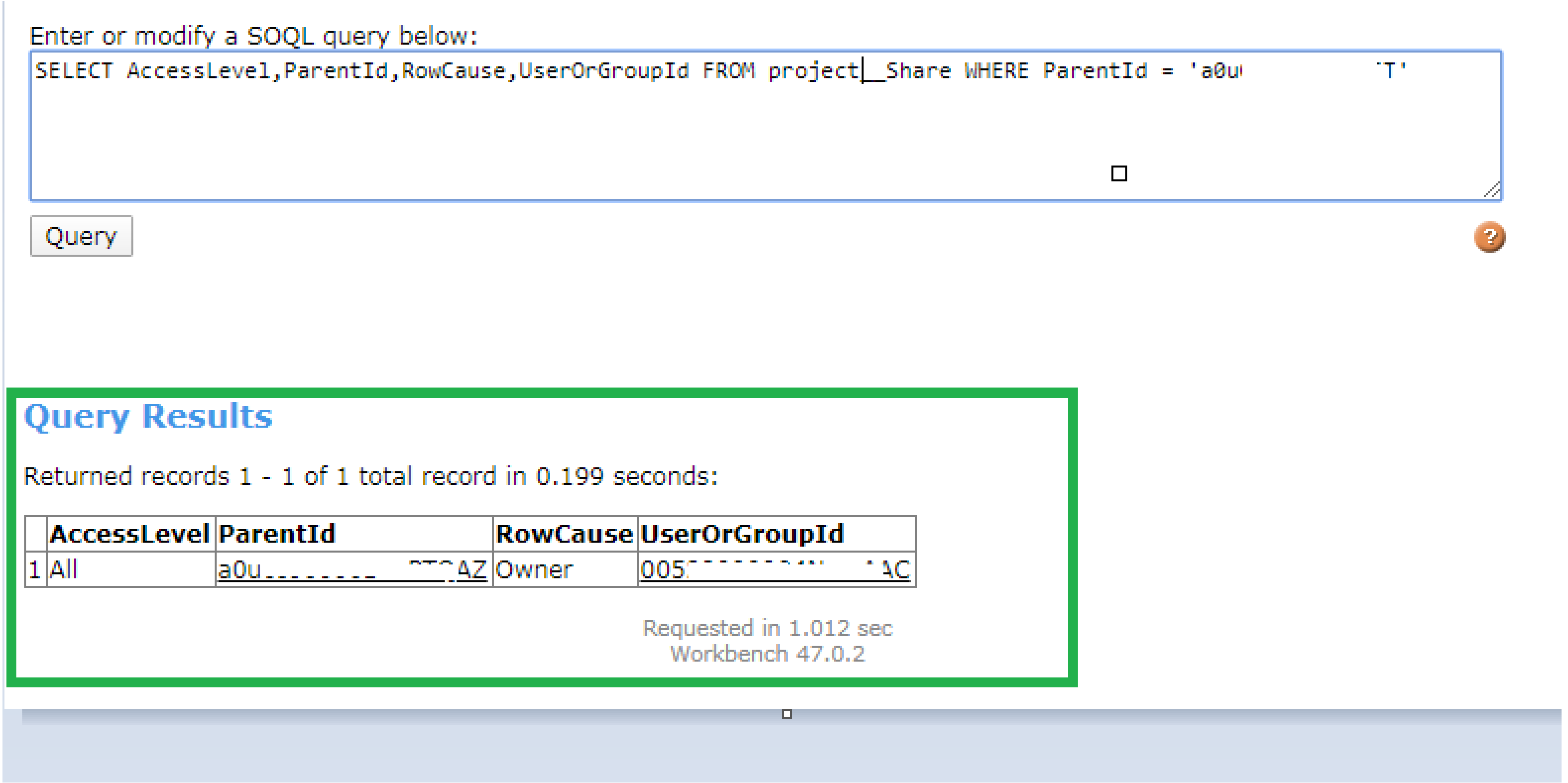
```
    public static void  provideAccess(List<Project__c > pros){

        List<Project_member__c> tmList=new List<Project_member__c>();

        map<id,Project_member__c> tmmap=new map<id,Project_member__c>();
```

```
     List<Project__Share> shareList=new List<Project__Share>();

     tmList=[select id,project__c,MasterDetailWithcontact__r.owner.id from Project_member__c where project__c IN:pros];

     userList=[select id from user];

     for(Project_member__c s:tmList){

        tmmap.put(s.Project__c,s);

     }

     for(project_c rec:Pros){

        Project__Share share=new Project__Share();

        share.ParentId = rec.id;

        share.AccessLevel = 'Edit';

        share.RowCause = Schema.Project__Share.RowCause.Manual;

        share.UserOrGroupId = tmmap.get(rec.id).MasterDetailWithcontact__r.owner.id;

        shareList.add(share);

     }

     insert shareList;

  }

  public static void  revokeAccess(List<Project__c> pros){

     List<Project__Share> shareList=new List<Project__Share>();

     shareList=[select id from Project__Share where parentId IN:pros AND RowCause = 'Manual'];

     delete shareList;
  }
}
```

Result :

Once the project record is created into the database ,By default only owner of the record will have access to record as shown in Screenshot 2 :

Screenshot 2:



Enter or modify a SOQL query below:

```
SELECT AccessLevel,ParentId,RowCause,UserOrGroupId FROM project__Share WHERE ParentId = 'a0u[redacted]T'
```

Query Results

Returned records 1 - 1 of 1 total record in 0.199 seconds:

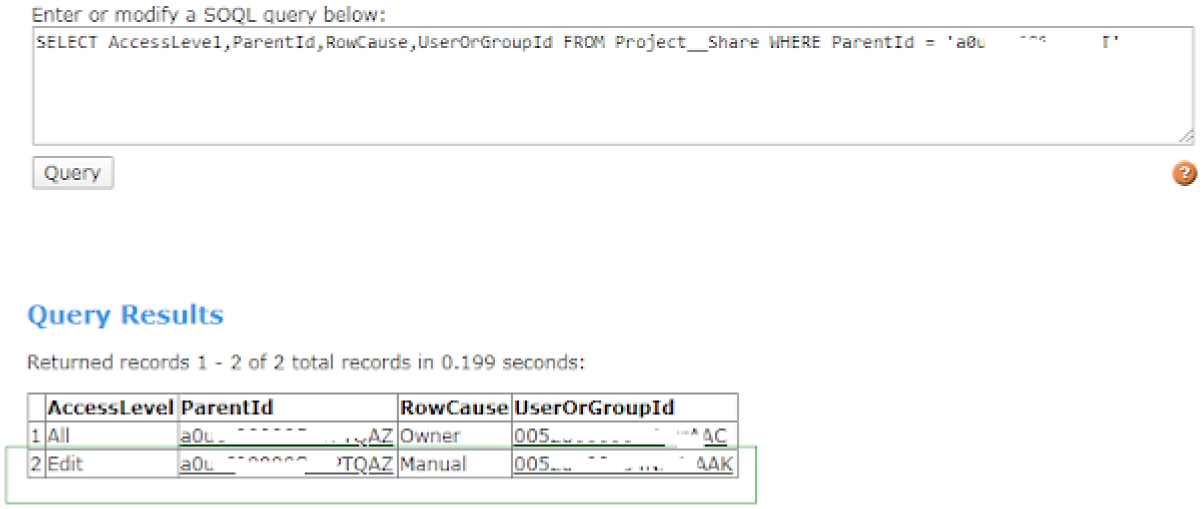
	AccessLevel	ParentId	RowCause	UserOrGroupId
1	All	a0u[redacted]AZ	Owner	005[redacted]AC

Requested in 1.012 sec
Workbench 47.0.2

Once the above project record updated to the 'Assigned' status ,Access of the project record will be shared to project team member . Please find the ScreenShot 3.

First Share record with all access Level belongs to the default record owner access . And Second Share record with Edit access and RowCause 'Manual' belongs to Share record created by above Apex trigger and Apex class.

ScreenShot 3 :



Enter or modify a SOQL query below:

```
SELECT AccessLevel,ParentId,RowCause,UserOrGroupId FROM Project__Share WHERE ParentId = 'a0u[redacted]T'
```

Query Results

Returned records 1 - 2 of 2 total records in 0.199 seconds:

	AccessLevel	ParentId	RowCause	UserOrGroupId
1	All	a0u[redacted]AZ	Owner	005[redacted]AC
2	Edit	a0u[redacted]TQAZ	Manual	005[redacted]AAK

If you update the Status to 'Completed' again the 'Edit' access would be revoked/Deleted to all the Project Members associated to Project object.