

Project Name: Sales Automobile Using Salesforce CRM

Team ID: LTVIP2024TMID11576

Team Leader: V Shilpa

Team member: R Surendra

Apex Trigger

Opportunity Automobile Quantity

UseCase : Whenever Opportunity Closed won Than Neglect / Minus the Quantity From Automobile Information on the Bases of Opportunity Automobile quantity.

1. Login to the respective trailhead account and navigate to the gear icon in the top right corner.
2. Click on the Developer console. Now you will see a new console window.
3. In the toolbar, you can see FILE. Click on it and navigate to new and create New apex class.
4. Name the class as “OpportunityHandlerClass”.

```
public class OpportunityHandlerClass {

    public static void opportunityAutomobileQuantity(List<Opportunity> LstOpportunity, Map<Id,Opportunity> OldMapOpportunity){
        set<Id> opportunityIds = new set<Id>();
        for(Opportunity opp : LstOpportunity){
            if(opp.StageName == 'Closed Won' ){
                opportunityIds.add(opp.Id);
            }
        }

        set<Id> opportunityIds = new set<Id>();
        for(Opportunity opp : LstOpportunity){
            if(opp.StageName == 'Closed Won' ){
                opportunityIds.add(opp.Id);
            }
        }

        Map<Id,Opportunity_Automobile__c> lstOpportunityAutomobile =new Map<Id,Opportunity_Automobile__c>{[SELECT Id, Opportunity__c, Automobile__c, Quantity__c, Unit_Price__c, Total_Price__c
FROM Opportunity_Automobile__c Where Opportunity__c IN: opportunityIds]};

        set<Id> AutoInformationIds = new set<Id>();
        for(Opportunity_Automobile__c OppAuto: lstOpportunityAutomobile.values()){
            if(OppAuto.Automobile__c != null){
                AutoInformationIds.add(OppAuto.Automobile__c);
            }
        }

        List<Automobile_Information__c> lstAutomobileInformation = new List<Automobile_Information__c>();
        Map<Id,Automobile_Information__c> MapAutomobileInformation = New Map<Id,Automobile_Information__c>{[SELECT Quantity__c, Price__c, Name, Id
FROM Automobile_Information__c
WHERE Id IN: AutoInformationIds]};

        For(Opportunity_Automobile__c AutoOpp : lstOpportunityAutomobile.Values()){
            decimal num = 0;
            if(AutoOpp.Automobile__c == MapAutomobileInformation.get(AutoOpp.Automobile__c).Id && OldMapOpportunity.get(AutoOpp.Opportunity__c).stagenam != 'Closed Won'){

                num = MapAutomobileInformation.get(AutoOpp.Automobile__c).Quantity__c- AutoOpp.Quantity__c;
                MapAutomobileInformation.get(AutoOpp.Automobile__c).quantity__c = num;
                lstAutomobileInformation.add(MapAutomobileInformation.get(AutoOpp.Automobile__c));
            }
        }
        If(!lstAutomobileInformation.isEmpty()){
            update lstAutomobileInformation;
        }
    }
}
```

Code:

```
public class OpportunityHandlerClass {

    public static void opportunityAutomobileQuantity(List<Opportunity> LstOpportunity, Map<Id,Opportunity>
OldMapOpportunity){
        set<Id> opportunityIds = new set<Id>();
        for(Opportunity opp : LstOpportunity){
            if(opp.StageName == 'Closed Won' ){
                opportunityIds.add(opp.Id);
            }
        }

        Map<Id,Opportunity_Automobile__c> lstOpportunityAutomobile =new
Map<Id,Opportunity_Automobile__c>{[SELECT Id, Opportunity__c, Automobile__c, Quantity__c, Unit_Price__c,
Total_Price__c FROM Opportunity_Automobile__c Where Opportunity__c IN: opportunityIds]};

        set<Id> AutoInformationIds = new set<Id>();
```

```

for(Opportunity_Automobile__c OppAuto: lstOpportunityAutomobile.values()){
    if(OppAuto.Automobile__c != null){
        AutoInformationIds.add(OppAuto.Automobile__c);
    }
}
List<Automobile_Information__c> lstAutomobileInfomation = new List<Automobile_Information__c>();
Map<Id,Automobile_Information__c> MapAutomobileInformation = New
Map<Id,Automobile_Information__c>([SELECT Quantity__c, Price__c, Name, Id FROM
Automobile_Information__c WHERE Id IN: AutoInformationIds]);
For(Opportunity_Automobile__c AutoOpp : lstOpportunityAutomobile.Values()){
    decimal num = 0;
    if(AutoOpp.Automobile__c == MapAutomobileInformation.get(AutoOpp.Automobile__c).Id &&
OldMapOpportunity.get(AutoOpp.Opportunity__c).stagename != 'Closed Won'){

        num = MapAutomobileInformation.get(AutoOpp.Automobile__c).Quantity__c- AutoOpp.Quantity__c;
        MapAutomobileInformation.get(AutoOpp.Automobile__c).quantity__c = num;
        lstAutomobileInfomation.add(MapAutomobileInformation.get(AutoOpp.Automobile__c));
    }
}
If(!lstAutomobileInfomation.IsEmpty()){
    update lstAutomobileInfomation;
}

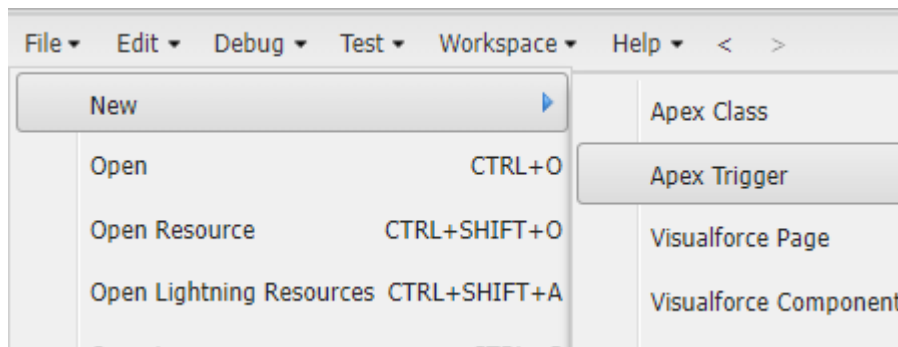
}
}

```

Trigger Handler :

How to create a new trigger :

1. While still in the account, navigate to the gear icon in the top right corner.
2. Click on developer console and you will be navigated to a new console window.
3. Click on the File menu in the toolbar, and click on new? Trigger.
4. Enter the trigger name and the object to be triggered.
5. Name : OpportunityTrigger
6. sObject : Opportunity



Syntax For creating trigger :

The syntax for creating trigger is :

```
Trigger [trigger name] on [object name]( Before/After event){
//block of code
}
```

In this project , trigger is called whenever the particular records sum exceed the threshold i.e minimum business requirement value. Then the code in the trigger will get executed.

1. Trigger for Opportunity Object.

```
1 trigger OpportunityTrigger on Opportunity (before update, After Update) {
2     if(trigger.isbefore && trigger.isUpdate){
3         OpportunityHandlerClass.opportunityAutomobileQuantity(trigger.new, trigger.oldMap);
4     }
5 }
6 }
```

Code:

```
trigger OpportunityTrigger on Opportunity (before update, After Update) {
    if(trigger.isbefore && trigger.isUpdate){
        OpportunityHandlerClass.opportunityAutomobileQuantity(trigger.new, trigger.oldMap);
    }
}
```

Opportunity-Automobile Error

UseCase : If Quantity of Automobile is Zero or Less than The Quantity from The Opportunity-Automobile Than Throw an error .

Login to the respective trailhead account and navigate to the gear icon in the top right corner.

1. Click on the Developer console. Now you will see a new console window.
2. In the toolbar, you can see FILE. Click on it and navigate to new and create New apex class.
3. Name the class as “OpportunityAutomobileHandler”

```

1 public class OpportunityAutomobileHandler {
2     public static void quantityErrorOnAutomobileInformation(List<Opportunity_Automobile__c> lstOpportunityAutomobile){
3         set<Id> AutomobileIds = new Set<Id>();
4         For(Opportunity_Automobile__c OppAutomobile : lstOpportunityAutomobile){
5             if(OppAutomobile.Automobile__c != null){
6                 AutomobileIds.add(OppAutomobile.Automobile__c);
7             }
8         }
9         Map<Id, Automobile_Information__c> lstAutomobileInformation = new Map<Id, Automobile_Information__c>([SELECT Id, CreatedById, Quantity__c, Price__c
10                                                         FROM Automobile_Information__c WHERE Id IN: AutomobileIds]);
11         For(Opportunity_Automobile__c OppAutomobile : lstOpportunityAutomobile){
12             if(OppAutomobile.Automobile__c == lstAutomobileInformation.get(OppAutomobile.Automobile__c).Id && lstAutomobileInformation.get(OppAutomobile.Automobile__c).Quantity__c < OppAutomobile.Quantity__c){
13                 OppAutomobile.addError('the Number of Automobile u want are not Available !! the Automobile are Available Count is ' + lstAutomobileInformation.get(OppAutomobile.Automobile__c).Quantity__c);
14             }
15         }
16     }
17 }

```

Code:

```

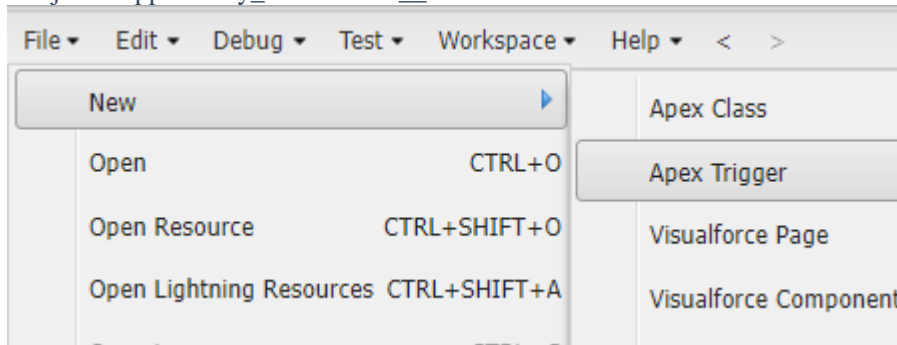
public class OpportunityAutomobileHandler {
    public static void quantityErrorOnAutomobileInformation(List<Opportunity_Automobile__c>
lstOpportunityAutomobile){
        set<Id> AutomobileIds = new Set<Id>();
        For(Opportunity_Automobile__c OppAutomobile : lstOpportunityAutomobile){
            if(OppAutomobile.Automobile__c != null){
                AutomobileIds.add(OppAutomobile.Automobile__c);
            }
        }
        Map<Id, Automobile_Information__c> lstAutomobileInformation = new
map<Id, Automobile_Information__c>([SELECT Id, CreatedById, Quantity__c, Price__c FROM
Automobile_Information__c WHERE Id IN: AutomobileIds]);
        For(Opportunity_Automobile__c OppAutomobile : lstOpportunityAutomobile){
            If(OppAutomobile.Automobile__c == lstAutomobileInformation.get(OppAutomobile.Automobile__c).Id &&
lstAutomobileInformation.get(OppAutomobile.Automobile__c).Quantity__c < OppAutomobile.Quantity__c){
                OppAutomobile.addError('the Number of Automobile u want are not Available !! the Automobile are
Available Count is ' + lstAutomobileInformation.get(OppAutomobile.Automobile__c).Quantity__c);
            }
        }
    }
}

```

Trigger Handler :

How to create a new trigger :

1. While still in the trailhead account, navigate to the gear icon in the top right corner.
2. Click on developer console and you will be navigated to a new console window.
3. Click on the File menu in the toolbar, and click on new? Trigger.
4. Enter the trigger name and the object to be triggered.
5. Name : OpportunityAutoMobileTrigger
6. sObject : Opportunity_Automobile__c



Trigger :

Handler for the Opportunity_Automobile__c Object

```

1 trigger OpportunityAutoMobileTrigger on Opportunity_Automobile__c (before insert, before Update) {
2     if(trigger.isbefore && trigger.isinsert || trigger.isupdate){
3         OpportunityAutomobileHandler.quantityErrorOnAutomobileInformation(trigger.new);
4     }
5 }

```

Code:

```

trigger OpportunityAutoMobileTrigger on Opportunity_Automobile__c (before insert, before Update) {
    if(trigger.isbefore && trigger.isinsert || trigger.isupdate){
        OpportunityAutomobileHandler.quantityErrorOnAutomobileInformation(trigger.new);
    }
}

```

Invoice Creation Trigger

UseCase : Whenever an opportunity is Closed won then create the Invoice on the Bases of Opportunity Automobile Data.

Login to the respective trailhead account and navigate to the gear icon in the top right corner.

1. Click on the Developer console. Now you will see a new console window.
2. In the toolbar, you can see FILE. Click on it and navigate to new and create New apex class.
3. Name the class as "InvoiceCreation".

```

1 public class InvoiceCreation {
2     public static void OpportunityClosedwonInvoiceGeneration(List<Opportunity> lstOpportunity, Map<Id,Opportunity>OldMapOpportunity){
3         set<Id> oppIds = new Set<Id>();
4         For(Opportunity opp : lstOpportunity){
5             if(opp.StageName == 'Closed Won' && OldMapOpportunity.get(opp.Id).StageName != opp.StageName){
6                 oppIds.add(opp.Id);
7             }
8         }
9         List<Opportunity_Automobile__c> lstOpportunityAutomobile = [SELECT Unit_Price__c, Total_Price__c, Automobile__c, Quantity__c, Opportunity__c, Id FROM Opportunity_Automobile__c WHERE Opportunity__c IN: oppIds];
10        List<Invoice__c> lstInvoice = new List<Invoice__c>();
11        For(Opportunity_Automobile__c oppAuto : lstOpportunityAutomobile){
12            Invoice__c i = new Invoice__c();
13            i.Quantity__c = oppAuto.Quantity__c;
14            i.Unit_Price__c = oppAuto.Unit_Price__c;
15            i.Total_Price__c = oppAuto.Total_Price__c;
16            i.Purchase_Date__c = date.today();
17            i.Opportunity__c = oppAuto.Opportunity__c;
18            lstInvoice.add(i);
19        }
20        If(lstInvoice.isEmpty()){
21            Insert lstInvoice;
22        }
23    }
24 }

```

Code:

```

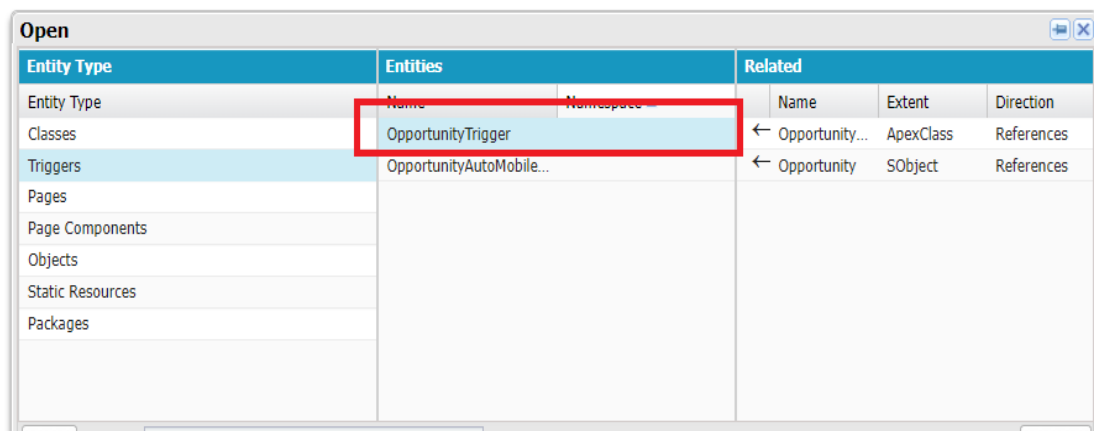
public class InvoiceCreation {
    public static void OpportunityClosedwonInvoiceGeneration(List<Opportunity> lstOpportunity,
Map<Id,Opportunity>OldMapOpportunity){
        set<Id> oppIds = new Set<Id>();
        For(Opportunity opp : lstOpportunity){
            if(Opp.StageName == 'Closed Won' && OldMapOpportunity.get(opp.Id).StageName != opp.StageName){
                oppIds.add(opp.Id);
            }
        }
        List<Opportunity_Automobile__c> lstOpportunityAutomobile = [SELECT Unit_Price__c, Total_Price__c,
Automobile__c, Quantity__c,Opportunity__c, Id FROM Opportunity_Automobile__c WHERE Opportunity__c IN:
oppIds];
        List<Invoice__c> lstInvoice = new List<Invoice__c>();
        For(Opportunity_Automobile__c oppAuto : lstOpportunityAutomobile){
            Invoice__c i = new Invoice__c();
            i.Quantity__c = oppAuto.Quantity__c;
            i.Unit_Price__c = oppAuto.Unit_Price__c;
            i.Total_Price__c = oppAuto.Total_Price__c;
            i.Purchase_Date__c = date.today();
            i.Opportunity__c = oppAuto.Opportunity__c;
            lstInvoice.add(i);
        }
        if(!lstInvoice.isEmpty()){
            insert lstInvoice;
        }
    }
}

```

Trigger Handler :

For this class we don't need to create any trigger, we will call this Code in "Opportunity Trigger".

1. Go on files and click on open.
2. Click on triggers.
3. Double click on OpportunityTrigger.



```

Code Coverage: None | API Version: 59
1 trigger OpportunityTrigger on Opportunity (before update, After Update) {
2     if(trigger.isbefore && trigger.isUpdate){
3         OpportunityHandlerClass.opportunityAutomobileQuantity(trigger.new, trigger.oldMap);
4     }
5 }
6 IF(trigger.isafter && trigger.isupdate){
7     InvoiceCreation.OpportunityClosedwonInvoiceGeneration(trigger.new, trigger.oldMap);
8 }
9 }

```

Trigger:

```

trigger OpportunityTrigger on Opportunity (before update, After Update) {
    if(trigger.isbefore && trigger.isUpdate){
        OpportunityHandlerClass.opportunityAutomobileQuantity(trigger.new, trigger.oldMap);
    }
    IF(trigger.isafter && trigger.isupdate){
        InvoiceCreation.OpportunityClosedwonInvoiceGeneration(trigger.new, trigger.oldMap);
    }
}

```

Check Contact Role

UseCase : Whenever an opportunity is Going to Closed won then check it has the contact role or Not.

Login to the respective trailhead account and navigate to the gear icon in the top right corner.

1. Click on the Developer console. Now you will see a new console window.
2. In the toolbar, you can see FILE. Click on it and navigate to new and create New apex class.
3. Name the class as “ContactRoleCheck”.

```

public class ContactRoleCheck {
    public static void CheckcontactRoleonOpportunity(List<Opportunity> lstOpportunity, Map<Id,Opportunity>OldMapOpportunity){
        List<OpportunityContactRole> lstContactRole = [SELECT Id From OpportunityContactRole WHERE OpportunityId IN: OldMapOpportunity.keySet()];
        For(Opportunity opp : lstOpportunity){
            if(opp.StageName == 'Closed Won' && OldMapOpportunity.get(opp.Id).StageName != opp.StageName){
                If(lstContactRole.isEmpty()){
                    opp.adderror('Please add contact Role on opportunity whenever Opportunity is Going to Closed Won.');
```

Trigger:

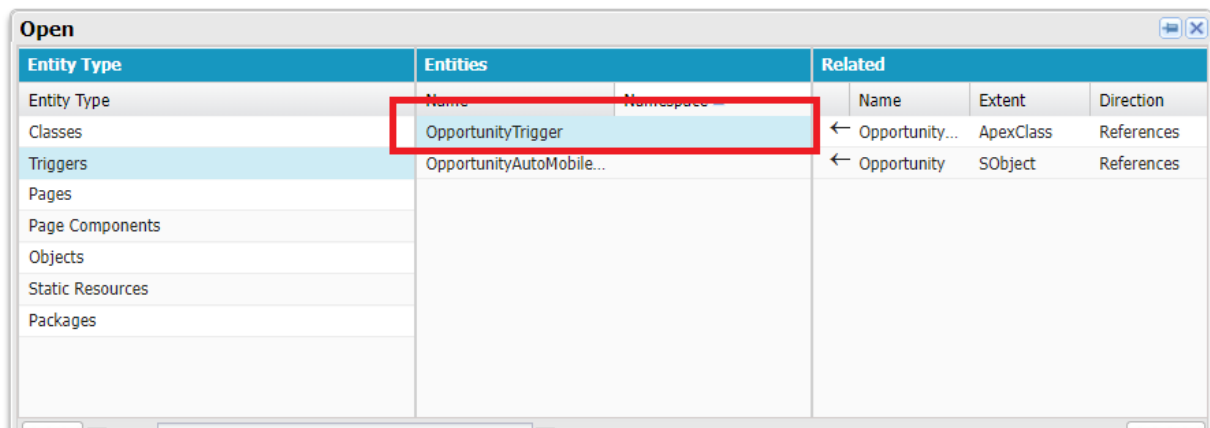
```

public class ContactRoleCheck {
    public static void CheckcontactRoleonOpportunity(List<Opportunity> lstOpportunity,
Map<Id,Opportunity>OldMapOpportunity){
        List<OpportunityContactRole> lstContactRole = [SELECT Id From OpportunityContactRole WHERE
OpportunityId IN: OldMapOpportunity.keySet()];
        For(Opportunity opp : lstOpportunity){
            if(opp.StageName == 'Closed Won' && OldMapOpportunity.get(opp.Id).StageName != opp.StageName){
                If(lstContactRole.isEmpty()){
                    opp.adderror('Please add contact Role on opportunity whenever Opportunity is Going to Closed Won.');
```


Trigger Handler :

For this class we don't need to create any trigger, we will call this Code in "Opportunity Trigger".

1. Go on files and click on open.
2. Click on triggers.
3. Double click on OpportunityTrigger.



Trigger Code :

```
Code Coverage: None | API Version: 59
1 trigger OpportunityTrigger on Opportunity (before update, After Update) {
2     if(trigger.isbefore && trigger.isupdate){
3         OpportunityHandlerClass.opportunityAutomobileQuantity(trigger.new, trigger.oldMap);
4         ContactRoleCheck.CheckcontactRoleonOpportunity(trigger.new, trigger.oldMap);
5     }
6     IF(trigger.isafter && trigger.isupdate){
7         InvoiceCreation.OpportunityClosedwonInvoiceGeneration(trigger.new, trigger.oldMap);
8     }
9 }
```

Trigger:

```
trigger OpportunityTrigger on Opportunity (before update, After Update) {
    if(trigger.isbefore && trigger.isupdate){
        OpportunityHandlerClass.opportunityAutomobileQuantity(trigger.new, trigger.oldMap);
        ContactRoleCheck.CheckcontactRoleonOpportunity(trigger.new, trigger.oldMap);
    }
    IF(trigger.isafter && trigger.isupdate){
        InvoiceCreation.OpportunityClosedwonInvoiceGeneration(trigger.new, trigger.oldMap);
    }
}
```