17. Write a trigger to update Total Opportunity Amount associated with account when an opportunity is 'Updated' or 'Deleted' or 'Undeleted'

Object: Opportunity

```
Trigger OpportunityTrigger on Opportunity (After Update)
{
  if (trigger.isAfter && trigger.isUpdate)
   {
         OpportunityTriggerHandler. handleAfterUpdate(Trigger.New,
     Trigger.Oldmap);
   }
If (trigger.isAfter && trigger.isDelete)
   {
      OpportunityTriggerHandler. handleAfterDelete(trigger.old);
   }
 If (trigger.isAfter && trigger.isUndelete)
   {
      OpportunityTriggerHandler. handleAfterUndelete(trigger.new);
   }
}
```

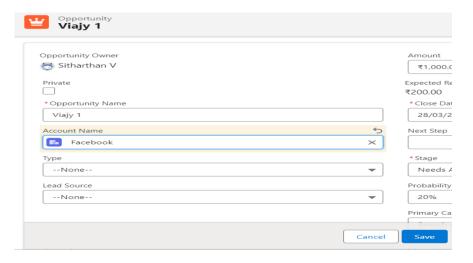
SWIPE THE PAGE TO SEE HANDLER CLASS...

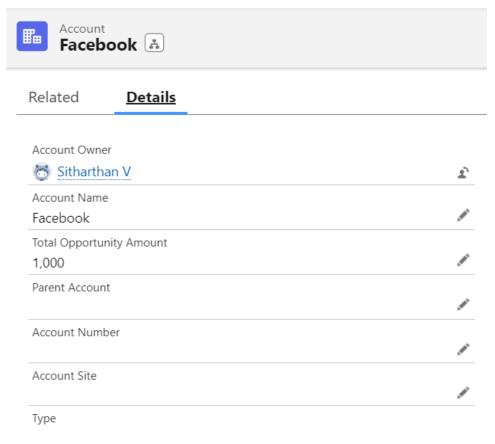
```
Public class OpportunityTriggerHandler
{
 Public static void
handleAfterUpdate(List<Opportunity>Opplist,Map<id,Opportunity>Oldmap)
 {
    Set<id>accids = new set<id>();
    for(Opportunity O : Opplist)
     {
       if(O.Accountid!=null)
      {
        accids.add(O.AccountId);
      }
      else if (Oldmap.ContainsKey(O.Id) && Oldmap.get(O.Id).AccountId !=
O.AccountId)
      {
        accids.add(Oldmap.get(O.Id).AccountId);
      }
     }
  List<AggregateResult> aggr = [SELECT AccountId, SUM(Amount) totalAmount FROM
Opportunity WHERE AccountId IN :accIds GROUP BY AccountId];
    List<Account> accToUpdate = new List<Account> ();
   for (AggregateResult a: aggr)
    {
      Account acc = new Account ();
      acc.Id = (Id) a.get('AccountId');
      acc.Total_Opportunity_Amount__c = (Decimal) a.get('totalAmount');
      accToUpdate.add(acc);
    }
```

```
if (!accToUpdate.isEmpty())
{
    update accToUpdate;
}
```

No table of contents entries found.

RESULT:





Public static void handleAfterDelete(List<Opportunity>Opplist)
{
 set<id>accids = new set<id>();

```
set<id>accids = new set<id>();
for(Opportunity O:Opplist)
{
   if(O.AccountId!=null)
   {
     accids.add(O.AccountId);
   }
}
```

List<AggregateResult> aggr = [SELECT AccountId, SUM(Amount) totalAmount FROM Opportunity WHERE AccountId IN :accIds GROUP BY AccountId];

```
List<Account> accToUpdate = new List<Account>();
```

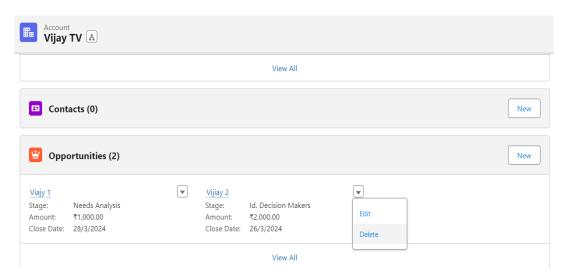
```
for (AggregateResult a: aggr) {
```

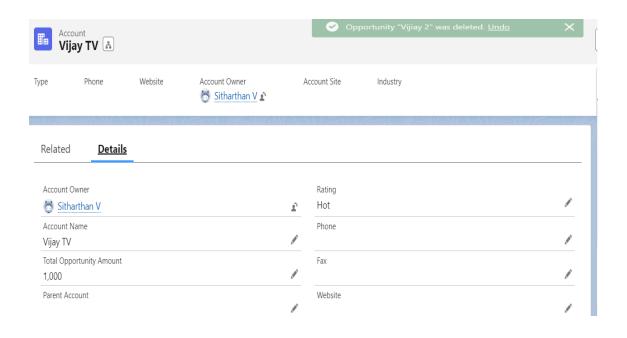
```
Account acc = new Account ();
    acc.Id = (Id) a.get('AccountId');
    acc.Total_Opportunity_Amount__c = (Decimal) a.get('totalAmount');
    accToUpdate.add(acc);
}

if(!accToUpdate.isEmpty())
{
    update accToUpdate;
}
```

RESULT:

}





```
Public static void handleAfterUndelete(list<Opportunity>Opplist)
{
    set<id>accids = new set<id>();

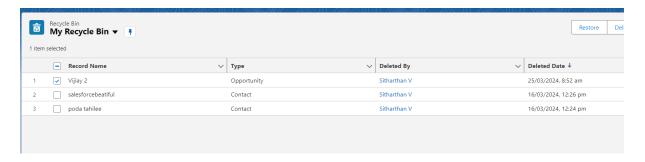
    for(Opportunity O : Opplist)
    {
        if(O.AccountId!=null)
        {
            accids.add(O.AccountId);
        }
}
```

```
}
```

List<AggregateResult> aggr = [SELECT AccountId, SUM(Amount) totalAmount FROM Opportunity WHERE AccountId IN :accIds GROUP BY AccountId];

```
List<Account> accToUpdate = new List<Account> ();
  for (AggregateResult a: aggr)
  {
    Account acc = new Account ();
    acc.Id = (Id) a.get('AccountId');
    acc.Total_Opportunity_Amount__c = (Decimal) a.get('totalAmount');
    accToUpdate.add(acc);
  }
  if(!accToUpdate.isEmpty())
    update accToUpdate;
  }
}
```

RESULT:





Account Owner Sitharthan V Account Name Vijay TV Total Opportunity Amount 3,000 Parent Account Account Number