42. Trigger Framework - 01 June 2022

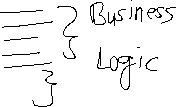
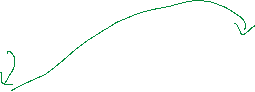
Trigger Framework 02 Types

**1] Handler and Helper Framework**

2] Dispatcher Framework

**1] Handler and Helper Framework**

**A] Code Re-usability B] Code Readability**



User Story 1] Create / Update Account Record. If Type is “Prospect”, then Description should be “Tu Tu Tu Tara”, else blank. Take backup of deleted Account Records.

|  |  |
| --- | --- |
| Event | Before |
| Operations | Insert, Update |
| Trigger Object | Account |
| Affecting Object | Account |

trigger AccountTrigger on Account (before insert, before update, before delete) {

if(trigger.isInsert && trigger.isBefore){

AccountTriggerHandler.beforeInsert(trigger.new);

}

if(trigger.isUpdate && trigger.isBefore){

AccountTriggerHandler.beforeUpdate(trigger.new, trigger.oldMap);

}

if(trigger.isDelete && trigger.isBefore){

AccountTriggerHandler.beforeDelete(trigger.old);

}

}

public class AccountTriggerHandler {

public static void beforeInsert(List<Account> accNewList){

AccountTriggerHelper.accountDescriptionSetting(accNewList, null);

}

public static void beforeUpdate(List<Account> accNewList, Map<Id,Account> accOldMap){

AccountTriggerHelper.accountDescriptionSetting(accNewList , accOldMap);

}

public static void beforeDelete(List<Account> accOldList){

AccountTriggerHelper.accountBackup(accOldList);

}

}

public class AccountTriggerHelper {

public static void accountBackup(List<Account> accOldList){

List<AccountBackup\_\_c> accBackList = new List<AccountBackup\_\_c>();

for(Account objAcc : accOldList){

AccountBackup\_\_c objAccBack = new AccountBackup\_\_c(oldAccName\_\_c=objAcc.Name, oldRating\_\_c=objAcc.Rating);

accBackList.add(objAccBack);

}

if(!accBackList.isEmpty())

Database.insert(accBackList,false);

}

public static void accountDescriptionSetting(List<Account> accNewList, Map<Id, Account> accOldMap){ //trigger.new

for(Account objAcc : accNewList){

if(String.isNotBlank(objAcc.Type)){

if(trigger.isInsert && trigger.isBefore){

if(objAcc.Type =='Prospect'){

objAcc.Description = 'Tu Tu Tu Tara';

}

else{

objAcc.Description = '';

}

}

if(trigger.isUpdate && trigger.isBefore){

if(objAcc.Type != accOldMap.get(objAcc.Id).Type){

if(objAcc.Type =='Prospect'){

objAcc.Description = 'Tu Tu Tu Tara';

}

else{

objAcc.Description = '';

}

}

}

}

}

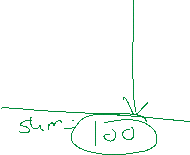
}

}

===================END================



if(!accMap.isEmpty()){ if((trigger.isInsert || trigger.isUpdate || trigger.isUndelete) && trigger.isAfter){ for(Opportunity objOpp: trigger.new){ sum= sum + objOpp.Amount; if(accMap.containsKey(objOpp.AccountId)){ if(accMap.get(objOpp.AccountId).Total\_Amount\_Opportunity\_\_c==null){ accMap.get(objOpp.AccountId).Total\_Amount\_Opportunity\_\_c = sum; } else{ accMap.get(objOpp.AccountId).Total\_Amount\_Opportunity\_\_c = accMap.get(objOpp.AccountId).Total\_Amount\_Opportunity\_\_c + sum;



}



trigger ContactTrigger31May on Contact (after insert, after update, after delete, after Undelete) {

Set<Id> accIdSet = new Set<Id>();

if( (trigger.isInsert || trigger.isUndelete || trigger.isUpdate) && trigger.isAfter){

for(Contact objCon : trigger.new){//C3

accIdSet.add(objCon.AccountId);//C3 = Cinemax

}

}

if( (trigger.isDelete || trigger.isUpdate) && trigger.isAfter){

for(Contact objCon : trigger.old){

accIdSet.add(objCon.AccountId);

}

}

Map<Id,Account> accMap = new Map<Id,Account>();

if(!accIdSet.isEmpty()){

for(Account objAcc : [select id, Count\_of\_Contacts\_\_c,TotalAmount\_\_c, (select id,Amount\_\_c from Contacts) from Account where Id IN : accIdSet]){//cinemax , C1,C2,C3

objAcc.Count\_of\_Contacts\_\_c = objAcc.Contacts.size();

accMap.put(objAcc.Id, objAcc);

}

}

for(Contact objCon : trigger.new){

if(accMap.containsKey(objCon.AccountId)){

List<Contact> ConList = accMap.get(objCon.AccountId).Contacts;

Decimal amount=0;

for(Contact objConExt : ConList){

amount = amount + objConExt.Amount\_\_c;

}

System.debug('Final Amount '+amount);

accMap.get(objCon.AccountId).TotalAmount\_\_c = amount;

}

}

if(!accMap.isEmpty()){

System.debug('Final Map = '+accMap.values());

Database.update(accMap.values(),false);

}

}