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\_Pricec, Total\_Pricec, Automobilec, Quantityc,Opportunityc, Id FROM Opportunity\_Automobilec 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 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);

}

}