Trigger 1:

CREATE DEFINER=`cecs323sec01s35`@`10.%` TRIGGER `cecs323sec01s35`.`OrderedItems\_BEFORE\_INSERT` BEFORE INSERT ON `OrderedItems` FOR EACH ROW

BEGIN

if (select Orders.billtotal from Orders where Orders.orderID = NEW.orderID) is null

then update Orders set Orders.billtotal = 0 Where Orders.orderID = New.orderId;

end if;

if (select customer.SpicyChallenge #determines if the given customer has passed the spicy challenge

from customer

inner join knowncustomer on knowncustomer.email = customer.email

inner join orders on orders.orderid = new.orderid)

then

if new.orderid in (select orderid from PartyOrder) #if the given order is a partyorder

then if(select PartyOrder.size

from PartyOrder

where PartyOrder.orderID = New.OrderID) >= 6 # if the party size is >= 6

then update Orders set Orders.billtotal = new.price \* new.quantity \* (1/20) \* (11/10) + (select Orders.billtotal from Orders where Orders.orderID = NEW.orderID) Where Orders.orderID = New.orderId; #calculate bill with 5% discount and 10% tip

else update Orders set Orders.billtotal = new.price \* new.quantity \* (1/20) + (select Orders.billtotal from Orders where Orders.orderID = NEW.orderID) Where Orders.orderID = New.orderId; #calculate bill only with %5 discount

end if;

else update Orders set Orders.billtotal = new.price \* new.quantity \* (1/20) + (select Orders.billtotal from Orders where Orders.orderID = NEW.orderID) Where Orders.orderID = New.orderId; #calculate bill only with %5 discount

end if;

else

if new.orderid in (select orderid from PartyOrder) #if the given order is a partyorder

then if(select PartyOrder.size

from PartyOrder

where PartyOrder.orderID = NewOrderID) >= 6 -- if the party size is >= 6

then update Orders set Orders.billtotal = new.price \* new.quantity \* (11/10) + (select Orders.billtotal from Orders where Orders.orderID = NEW.orderID) Where Orders.orderID = New.orderId; #calculate bill with 10% tip

else update Orders set Orders.billtotal = new.price \* new.quantity + (select Orders.billtotal from Orders where Orders.orderID = NEW.orderID) Where Orders.orderID = New.orderId; #calculate bill with no tip and no discount

end if;

else update Orders set Orders.billtotal = new.price \* new.quantity + (select Orders.billtotal from Orders where Orders.orderID = NEW.orderID) Where Orders.orderID = New.orderId;#calculate bill with no tip and no discount

end if;

end if;

END

Trigger 2:

CREATE DEFINER=`cecs323sec01s35`@`10.%` TRIGGER `cecs323sec01s35`.`PartyOrder\_BEFORE\_INSERT` BEFORE INSERT ON `PartyOrder` FOR EACH ROW

BEGIN

if new.size > 30 #if the size of the part yis great than 30 then dont allow it

then signal sqlstate '45000' set message\_text = 'Party size cannot exceed 30';

end if;

END

Sample Output of trigger 2:

01:47:08 INSERT INTO PartyOrder VALUES (1,1,4,'O0008') 1 row(s) affected 0.047 sec

01:47:40 INSERT INTO PartyOrder VALUES (31,1,4,'O0009') Error Code: 1644. Party size cannot exceed 30 0.047 sec