CODE:

***Order Confirmation Email Template:-***

*Dear {!HandsMen\_Order\_\_c.HandsMen\_Customer\_\_c},*

*Your order #{!HandsMen\_Order\_\_c.Name} has been confirmed!*

*Thank you for shopping with us.*

*Best Regards,*

*Sales Team*

**Low Stock Alert Email Template:-**

Dear Inventory Manager,

This is to inform you that the stock for the following product is running low:

Product Name: {!Inventory\_\_c.HandsMen\_Product\_\_c}

Current Stock Quantity: {!Inventory\_\_c.Stock\_Quantity\_\_c}

Please take the necessary steps to restock this item immediately.

Best Regards,

Inventory Monitoring System

**Loyality Points Email Template:-**

Congratulations! You are now a {!HandsMen\_Customer\_\_c.Loyalty\_Status\_\_c} member and you are eligible for our Loyalty Rewards Program.

Enjoy exclusive discounts, early access to offers, and special member benefits.

Thank you for your continued Support.

**Update Order Total :-**

trigger OrderTotalTrigger on HandsMen\_Order\_\_c (before insert, before update) {

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

    for (HandsMen\_Order\_\_c order : Trigger.new) {

        if (order.HandsMen\_Product\_\_c != null) {

            productIds.add(order.HandsMen\_Product\_\_c);

        }

    }

    Map<Id, HandsMen\_Product\_\_c> productMap = new Map<Id, HandsMen\_Product\_\_c>(

        [SELECT Id, Price\_\_c FROM HandsMen\_Product\_\_c WHERE Id IN :productIds]

    );

    for (HandsMen\_Order\_\_c order : Trigger.new) {

        if (order.HandsMen\_Product\_\_c != null && productMap.containsKey(order.HandsMen\_Product\_\_c)) {

            HandsMen\_Product\_\_c product = productMap.get(order.HandsMen\_Product\_\_c);

            if (order.Quantity\_\_c != null) {

                order.Total\_Amount\_\_c = order.Quantity\_\_c \* product.Price\_\_c;

            }

        }

    }

}

**Stock Deduction Trigger:-**

trigger StockDeductionTrigger on HandsMen\_Order\_\_c (after insert, after update) {

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

    for (HandsMen\_Order\_\_c order : Trigger.new) {

        if (order.Status\_\_c == 'Confirmed' && order.HandsMen\_Product\_\_c != null) {

            productIds.add(order.HandsMen\_Product\_\_c);

        }

    }

    if (productIds.isEmpty()) return;

    // Query related inventories based on product

    Map<Id, Inventory\_\_c> inventoryMap = new Map<Id, Inventory\_\_c>(

        [SELECT Id, Stock\_Quantity\_\_c, HandsMen\_Product\_\_c

         FROM Inventory\_\_c

         WHERE HandsMen\_Product\_\_c IN :productIds]

    );

    List<Inventory\_\_c> inventoriesToUpdate = new List<Inventory\_\_c>();

    for (HandsMen\_Order\_\_c order : Trigger.new) {

        if (order.Status\_\_c == 'Confirmed' && order.HandsMen\_Product\_\_c != null) {

            for (Inventory\_\_c inv : inventoryMap.values()) {

                if (inv.HandsMen\_Product\_\_c == order.HandsMen\_Product\_\_c) {

                    inv.Stock\_Quantity\_\_c -= order.Quantity\_\_c;

                    inventoriesToUpdate.add(inv);

                    break;

                }

            }

        }

    }

    if (!inventoriesToUpdate.isEmpty()) {

        update inventoriesToUpdate;

    }

}