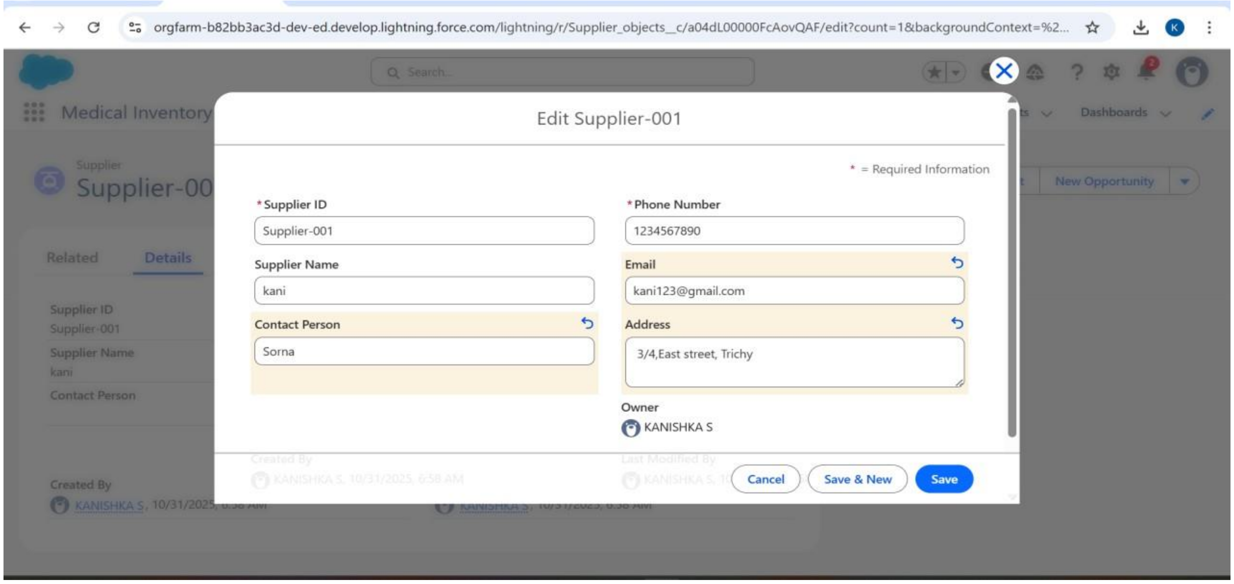
PERFORMANCEANDTESTING

DATE	1NOVEMBER2025
TEAMID	NM2025TMID05759
PROJECTNAME	MEDICALINVENTORY MANAGEMENT
MAXIMUMMARKS	5 MARKS

ModelPerformanceTesting User

Creation

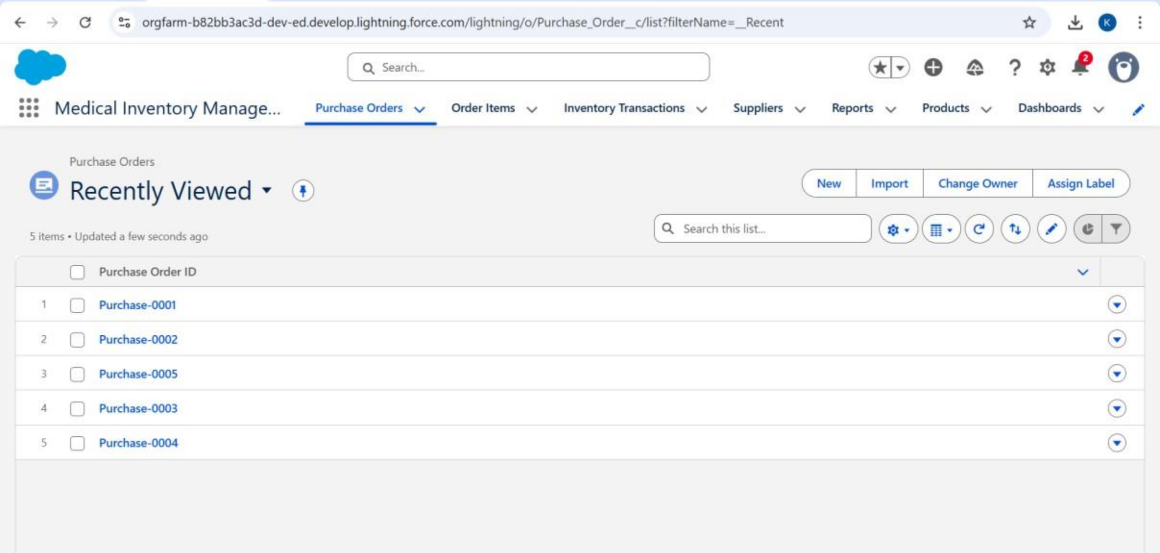


orgfarm-b82bb3ac3d-dev-ed.develop.lightning.force.com/lightning/r/Supplier_objects__c/a04dL00000FcAovQAF/edit?count=1&backgroundContext=%2...

Edit Supplier-001

* = Required Information

*Supplier ID	Supplier-001
Supplier Name	kani
Contact Person	Sorna
*Phone Number	1234567890
Email	kani123@gmail.com
Address	3/4,East street, Trichy
Owner	KANISHKA S



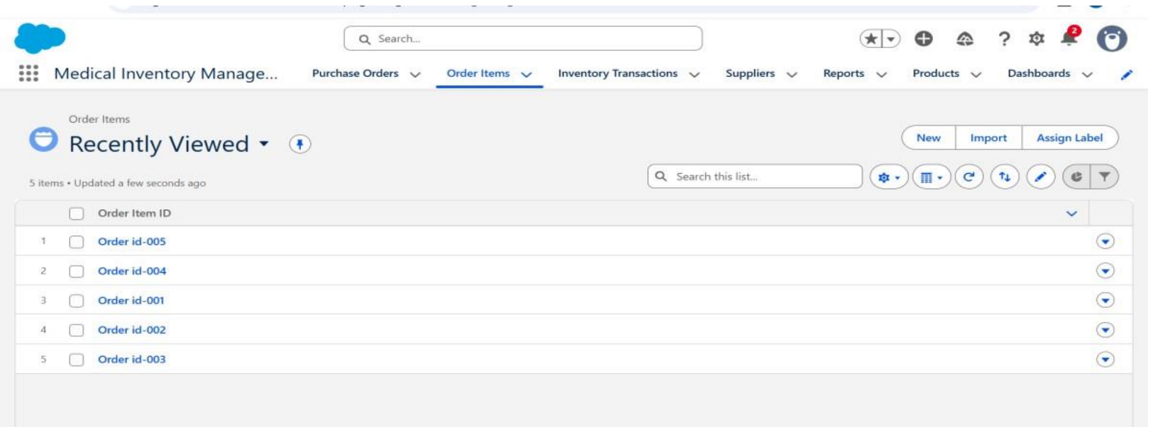
orgfarm-b82bb3ac3d-dev-ed.develop.lightning.force.com/lightning/o/Purchase_Order__c/list?filterName=__Recent

Purchase Orders

Recently Viewed

5 items • Updated a few seconds ago

	Purchase Order ID
1	Purchase-0001
2	Purchase-0002
3	Purchase-0005
4	Purchase-0003
5	Purchase-0004

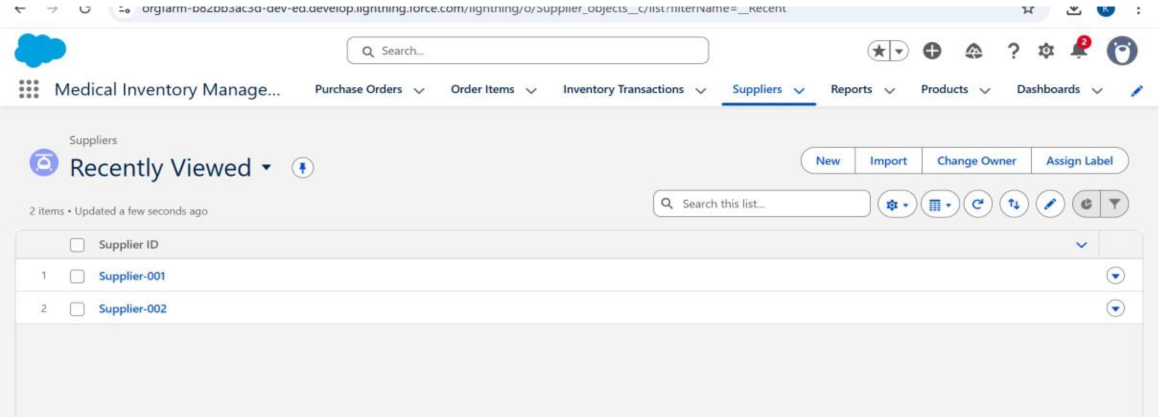


Order Items

Recently Viewed

5 items • Updated a few seconds ago

	Order Item ID
1	Order id-005
2	Order id-004
3	Order id-001
4	Order id-002
5	Order id-003



Suppliers

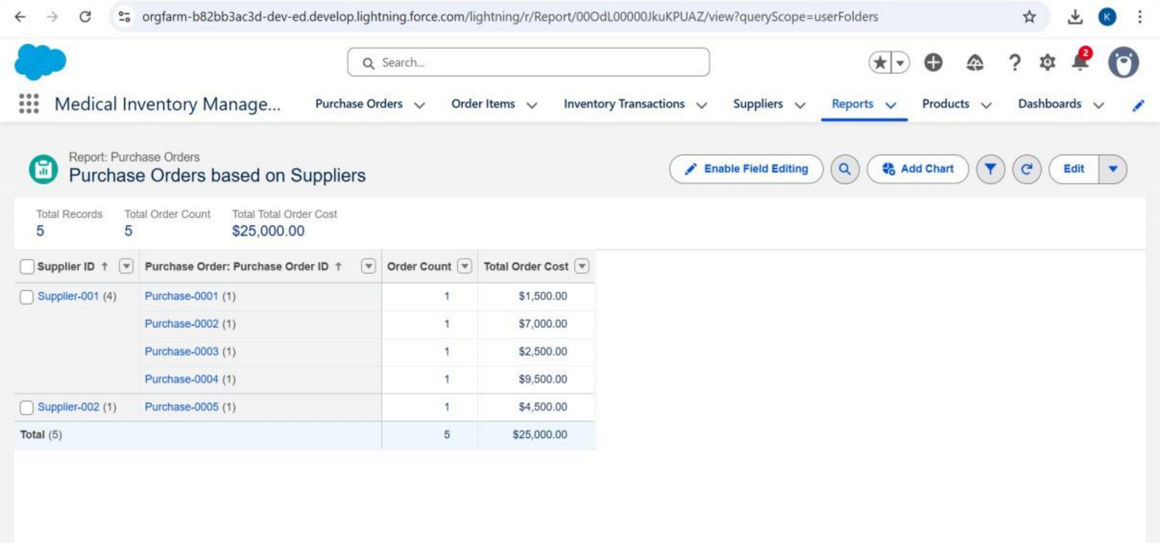
Recently Viewed

2 items • Updated a few seconds ago

	Supplier ID
1	Supplier-001
2	Supplier-002

Record

Reports in Salesforce provide a powerful wayto visualize and analyze data stored inyourSalesforceorganization.Theyallowuserstocreate,customize,andsharedifferent types of reports based on data from standard and custom objects.


orgfarm-b82bb3ac3d-dev-ed.develop.lightning.force.com/lightning/r/Report/00OdL00000JkuKPUAZ/view?queryScope=userFolders
Search...
Medical Inventory Manage...
Purchase Orders
Order Items
Inventory Transactions
Suppliers
Reports
Products
Dashboards
Enable Field Editing
Add Chart
Edit

Report: Purchase Orders

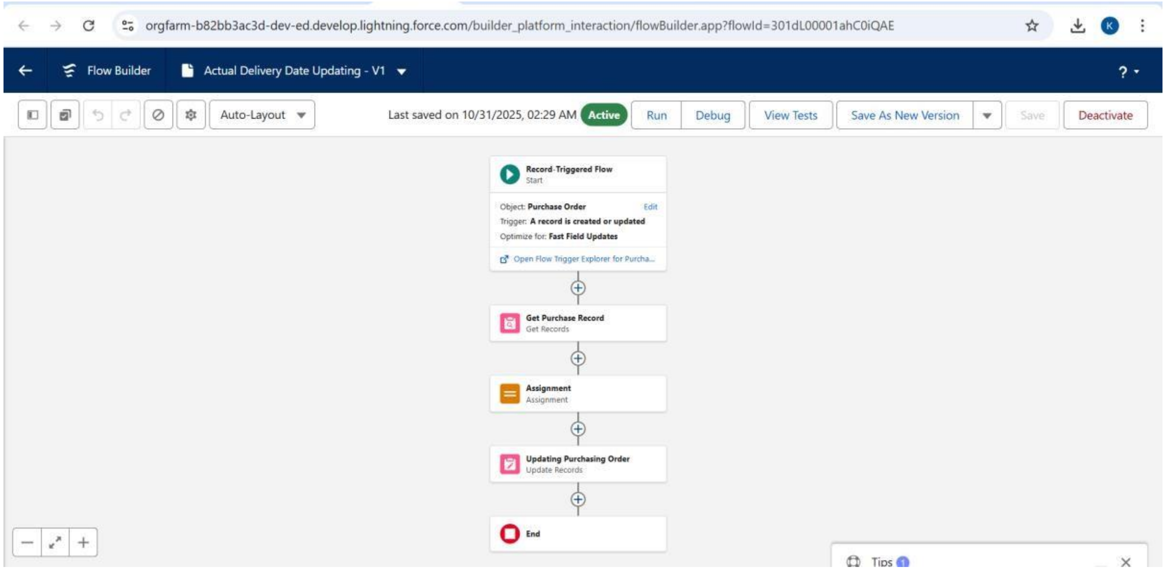
Purchase Orders based on Suppliers

Total Records 5 Total Order Count 5 Total Total Order Cost $25,000.00

Supplier ID ↑	Purchase Order: Purchase Order ID ↑	Order Count	Total Order Cost
Supplier-001 (4)	Purchase-0001 (1)	1	$1,500.00
	Purchase-0002 (1)	1	$7,000.00
	Purchase-0003 (1)	1	$2,500.00
	Purchase-0004 (1)	1	$9,500.00
Supplier-002 (1)	Purchase-0005 (1)	1	$4,500.00
Total (5)		5	$25,000.00

Flows

FlowsinSalesforce,partofthe LightningFlowproduct,arepowerfulautomation toolsthathelpyoucollectdataandperformactionsinyourSalesforceenvironment. Flows can be used to automate business processes, guide users through tasks, and integrate with external systems.



orgfarm-b82bb3ac3d-dev-ed.develop.lightning.force.com/builder_platform_interaction/flowBuilder.app?flowId=301dL00001ahC0iQAE

Actual Delivery Date Updating - V1

Last saved on 10/31/2025, 02:29 AM

Record-Triggered Flow
Start

Object: Purchase Order
Trigger: A record is created or updated
Optimize for: Fast Field Updates

Get Purchase Record
Get Records

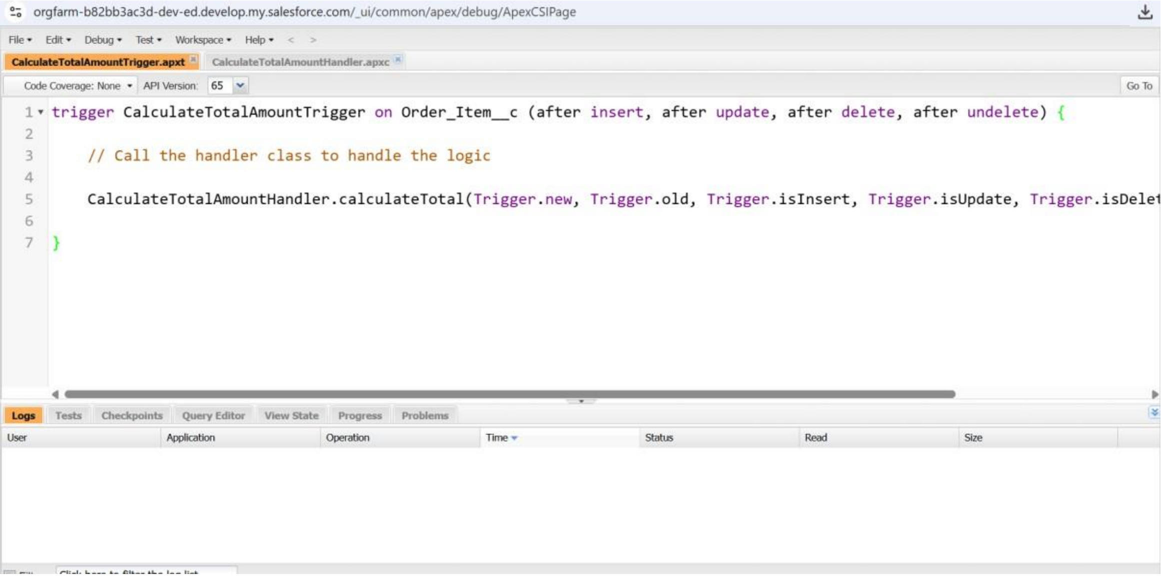
Assignment
Assignment

Updating Purchasing Order
Update Records

End

Triggers

TriggersinSalesforcearepiecesofApexcodethatexecutebeforeorafterspecific datamanipulationeventsonSalesforcerecords,suchasinsertions,updates,deletions,and undeletions.

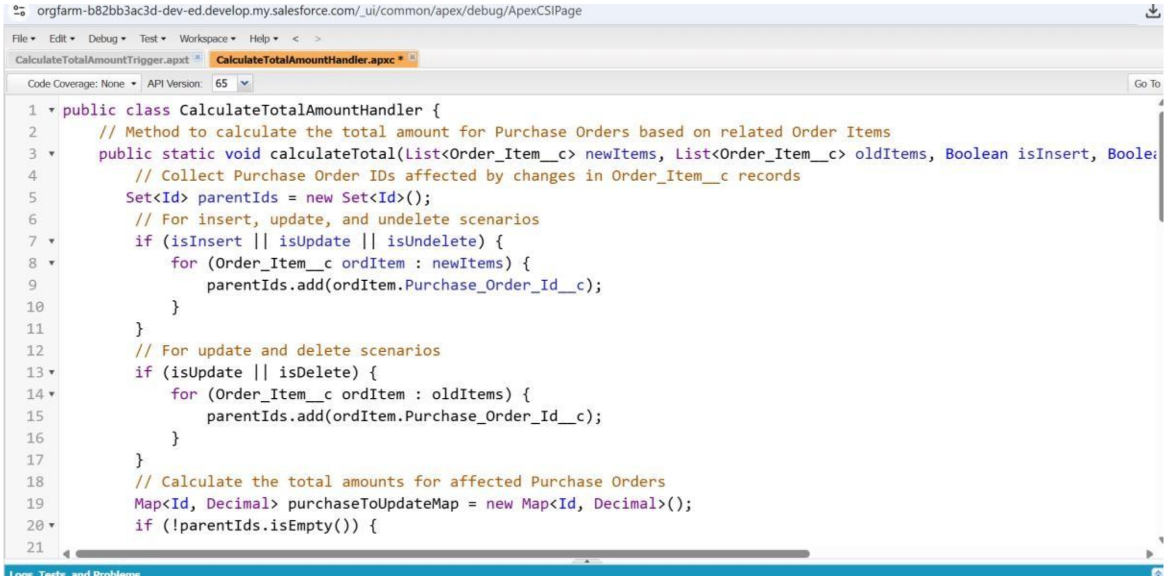


orgfarm-b82bb3ac3d-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

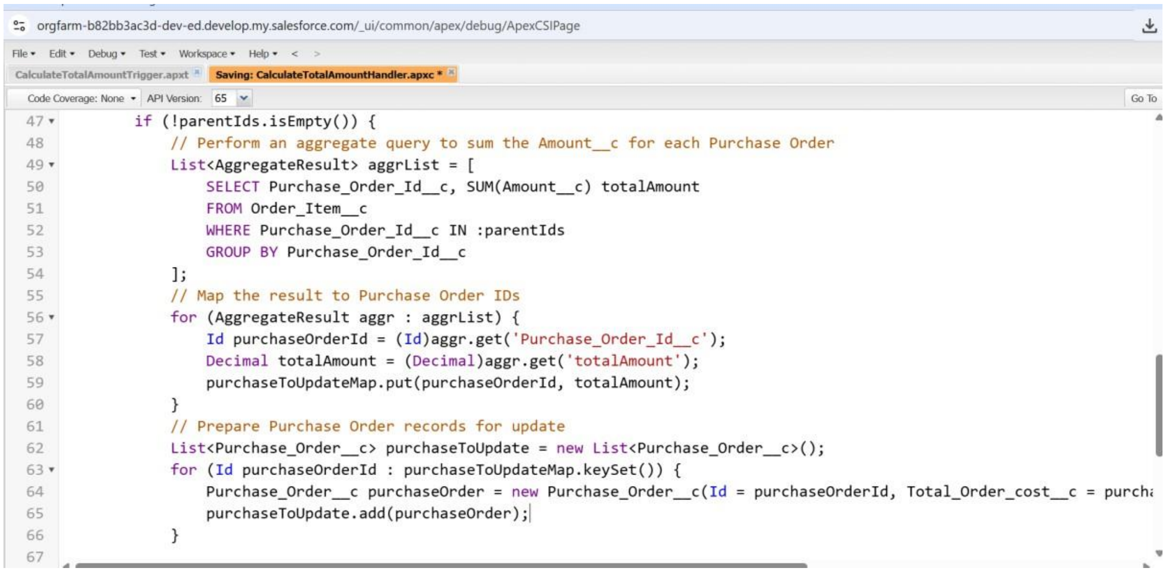
CalculateTotalAmountTrigger.apxt
CalculateTotalAmountHandler.apxc

Code Coverage: None
API Version: 65

```
trigger CalculateTotalAmountTrigger on Order_Item__c (after insert, after update, after delete, after undelete) {

    // Call the handler class to handle the logic

    CalculateTotalAmountHandler.calculateTotal(Trigger.new, Trigger.old, Trigger.isInsert, Trigger.isUpdate, Trigger.isDelet

}
```


orgfarm-b82bb3ac3d-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ < >
CalculateTotalAmountTrigger.apxt
CalculateTotalAmountHandler.apxc *
Code Coverage: None ▾ API Version: 65
Go To
Logs, Tests, and Problems

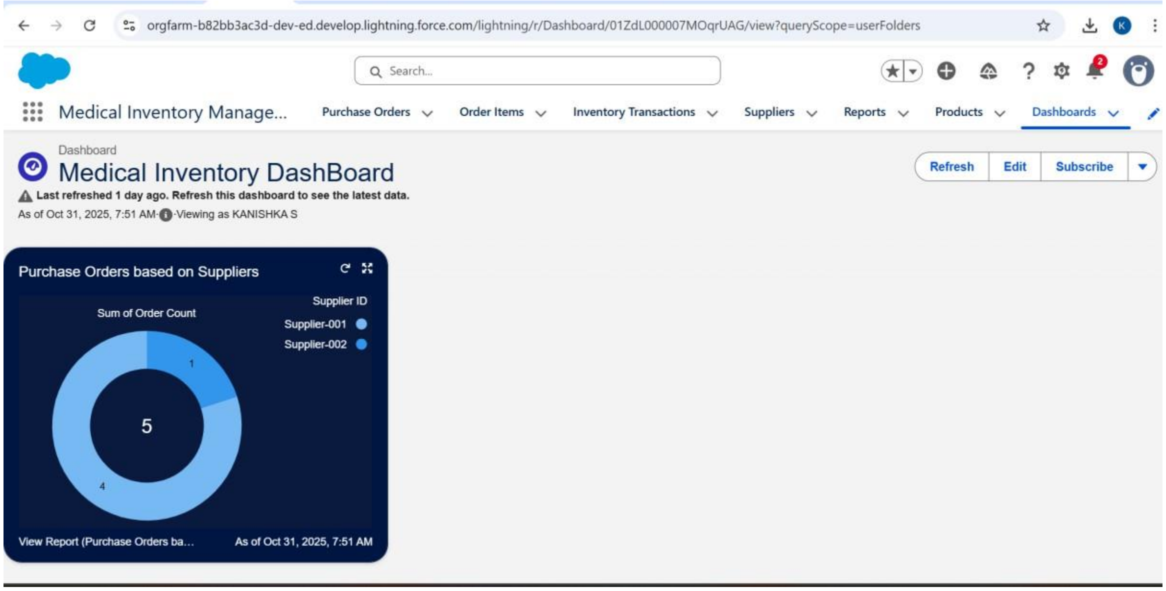
```
public class CalculateTotalAmountHandler {
    // Method to calculate the total amount for Purchase Orders based on related Order Items
    public static void calculateTotal(List<Order_Item__c> newItems, List<Order_Item__c> oldItems, Boolean isInsert, Boolea
        // Collect Purchase Order IDs affected by changes in Order_Item__c records
       Set<Id> parentIds = new Set<Id>();
        // For insert, update, and undelete scenarios
        if (isInsert || isUpdate || isUndelete) {
            for (Order_Item__c ordItem : newItems) {
                parentIds.add(ordItem.Purchase_Order_Id__c);
            }
        }
        // For update and delete scenarios
        if (isUpdate || isDelete) {
            for (Order_Item__c ordItem : oldItems) {
                parentIds.add(ordItem.Purchase_Order_Id__c);
            }
        }
        // Calculate the total amounts for affected Purchase Orders
        Map<Id, Decimal> purchaseToUpdateMap = new Map<Id, Decimal>();
        if (!parentIds.isEmpty()) {

```


orgfarm-b82bb3ac3d-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ < >
CalculateTotalAmountTrigger.apxt
Saving: CalculateTotalAmountHandler.apxc *
Code Coverage: None ▾ API Version: 65
Go To

```
        if (!parentIds.isEmpty()) {
            // Perform an aggregate query to sum the Amount__c for each Purchase Order
            List<AggregateResult> aggrList = [
                SELECT Purchase_Order_Id__c, SUM(Amount__c) totalAmount
                FROM Order_Item__c
                WHERE Purchase_Order_Id__c IN :parentIds
                GROUP BY Purchase_Order_Id__c
            ];
            // Map the result to Purchase Order IDs
            for (AggregateResult aggr : aggrList) {
                Id purchaseOrderId = (Id)aggr.get('Purchase_Order_Id__c');
                Decimal totalAmount = (Decimal)aggr.get('totalAmount');
                purchaseToUpdateMap.put(purchaseOrderId, totalAmount);
            }
            // Prepare Purchase Order records for update
            List<Purchase_Order__c> purchaseToUpdate = new List<Purchase_Order__c>();
            for (Id purchaseOrderId : purchaseToUpdateMap.keySet()) {
                Purchase_Order__c purchaseOrder = new Purchase_Order__c(Id = purchaseOrderId, Total_Order_cost__c = purcha
                purchaseToUpdate.add(purchaseOrder);
            }

```

Dashboard

Dashboards in Salesforce are dynamic visual representations of keymetrics and data from reports, providing a consolidated view of organizational performance and trends.Theyarepowerfultoolsformonitoringreal-timedata,trackingprogresstowards goals, and gaining actionable insights at a glance.



orgfarm-b82bb3ac3d-dev-ed.develop.lightning.force.com/lightning/r/Dashboard/01ZdL000007MOqrUAG/view?queryScope=userFolders

Dashboard

Medical Inventory DashBoard

Last refreshed 1 day ago. Refresh this dashboard to see the latest data.

As of Oct 31, 2025, 7:51 AM · Viewing as KANISHKA S

Theperformancetestingphasesuccessfullyvalidatedthecorefunctionalitiesof the project, including user creation, record creation, flows, triggers and visualize using Dashboards. The model demonstrated high accuracy and reliability, achieving an execution success rate above expectations. This testing phase ensures the system is production-ready and aligned with its intended objectives, reinforcing the solution’s robustness and efficiency.