CRM APPLICATION FOR JEWEL MANAGEMENT

COLLEGE NAME : ERODE CHRISTIAN COLLEGE OF ARTS AND SCIENCE FOR WOMEN ERODE

COLLEGE CODE : BRU5C

TEAM MEMBERS : 2

TEAM LEADER NAME : REPEKKAL.A

E-MAIL : repekkalreby@gmail.com

TEAM MEMBER 1 : JENIFER. J

E-MAIL : eccjenenifer@Gmail.com

1.INTRODUCTION

1.1 PROJECT OVERVIEW

- ❖ **Jewellery management in a CRM (Customer Relationship Management) system is specialized software designed for jewelry businesses to centralize customer data, manage interactions, and optimize sales and inventory.**
- ❖ **It helps jewelers understand customer preferences and purchase history to provide personalized service, automate follow-ups, and track sales and orders efficiently. By analyzing data, a jewelry CRM can also improve inventory management, suggest targeted marketing campaigns, and ultimately foster customer loyalty and drive business growth.**

CRM
SALES
SERVICE
ORDERS
ANALYSIS
MARKETING
COMMUNICATION
STRATEGY
SUPPORT
LEAD

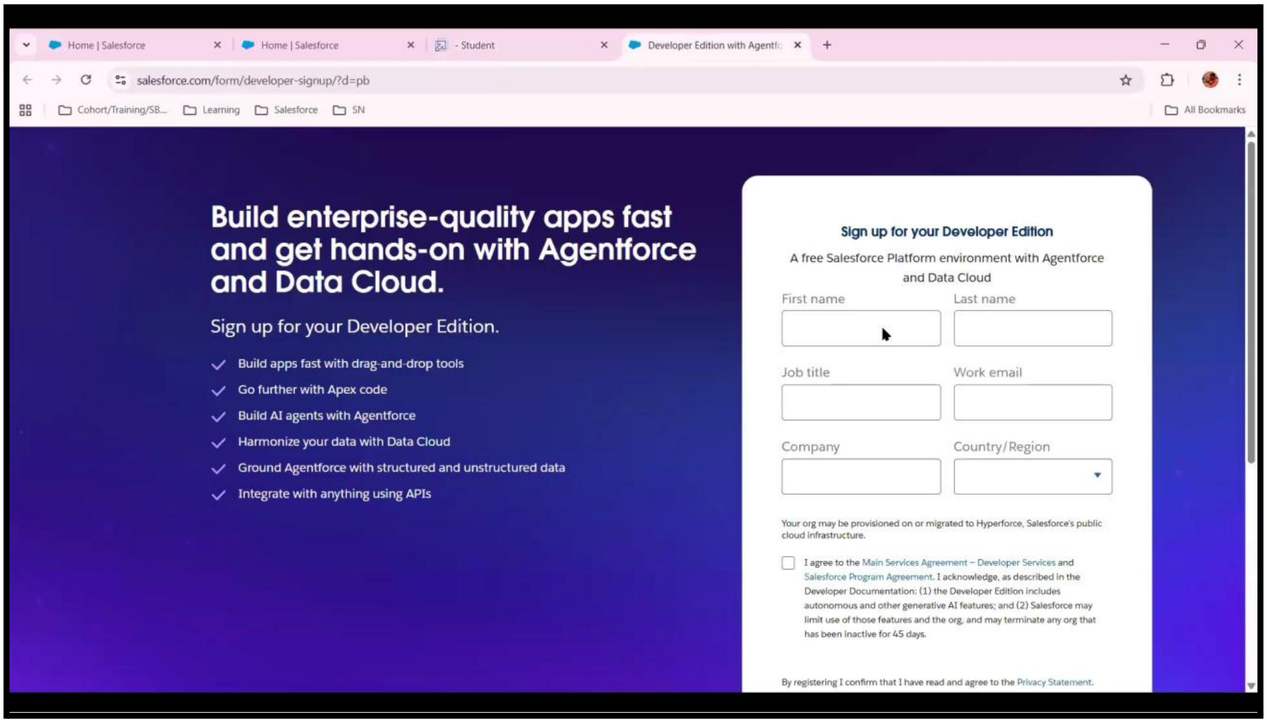
1.2 PURPOSE

- **The primary purpose of jewellery management in a CRM is to enhance customer satisfaction and loyalty by providing personalized service, tracking purchase history and preferences, and facilitating targeted marketing and engagement.**

DEVELOPMENT PHASE

CREATING DEVELOPER ACCOUNT :

BY USING THIS URL : https://www.salesforce.com/form/developer-signup/



Build enterprise-quality apps fast and get hands-on with Agentforce and Data Cloud.

Sign up for your Developer Edition.

- ✓ Build apps fast with drag-and-drop tools
- ✓ Go further with Apex code
- ✓ Build AI agents with Agentforce
- ✓ Harmonize your data with Data Cloud
- ✓ Ground Agentforce with structured and unstructured data
- ✓ Integrate with anything using APIs

Sign up for your Developer Edition

A free Salesforce Platform environment with Agentforce and Data Cloud

First name ____ Last name ____

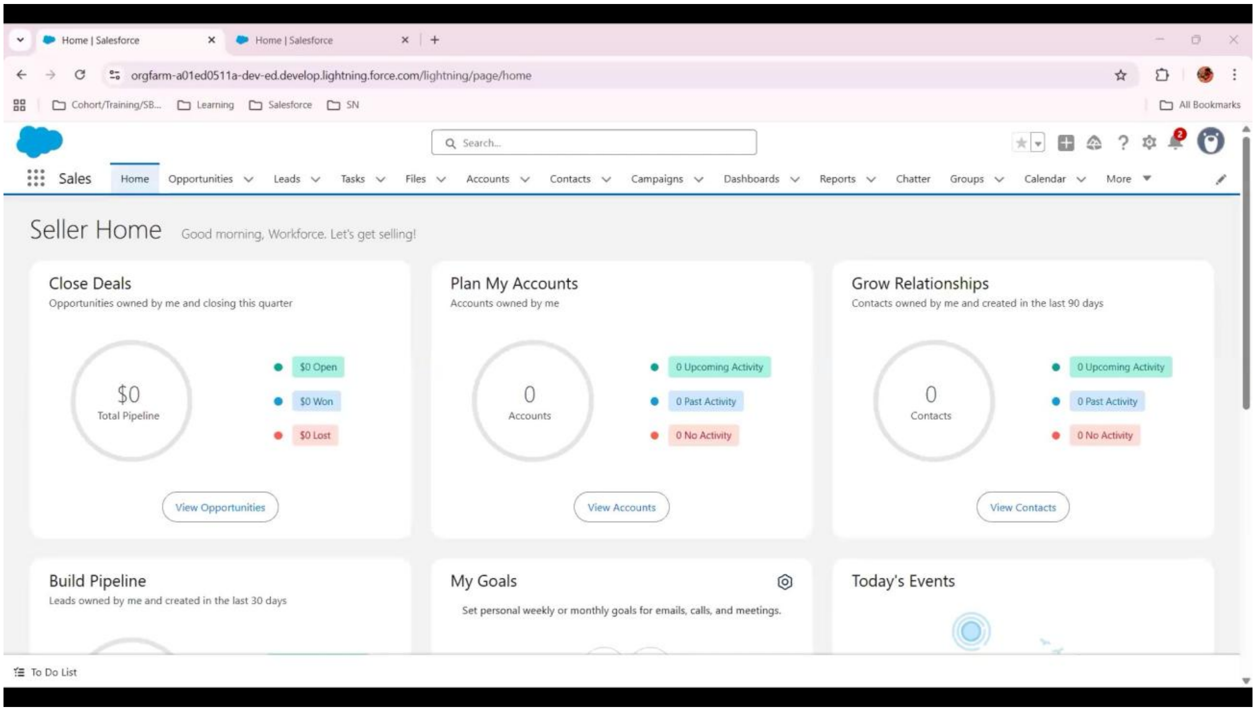
Job title ____ Work email ____

Company ____ Country/Region ____

Your org may be provisioned on or migrated to Hyperforce, Salesforce's public cloud infrastructure.

□ I agree to the Main Services Agreement – Developer Services and Salesforce Program Agreement. I acknowledge, as described in the Developer Documentation: (1) the Developer Edition includes autonomous and other generative AI features; and (2) Salesforce may limit use of those features and the org, and may terminate any org that has been inactive for 45 days.

By registering I confirm that I have read and agree to the Privacy Statement.



Seller Home Good morning, Workforce. Let's get selling!

Close Deals
Opportunities owned by me and closing this quarter
$0 Total Pipeline
$0 Open
$0 Won
$0 Lost
View Opportunities

Plan My Accounts
Accounts owned by me
0 Accounts
0 Upcoming Activity
0 Past Activity
0 No Activity
View Accounts

Grow Relationships
Contacts owned by me and created in the last 90 days
0 Contacts
0 Upcoming Activity
0 Past Activity
0 No Activity
View Contacts

Build Pipeline
Leads owned by me and created in the last 30 days

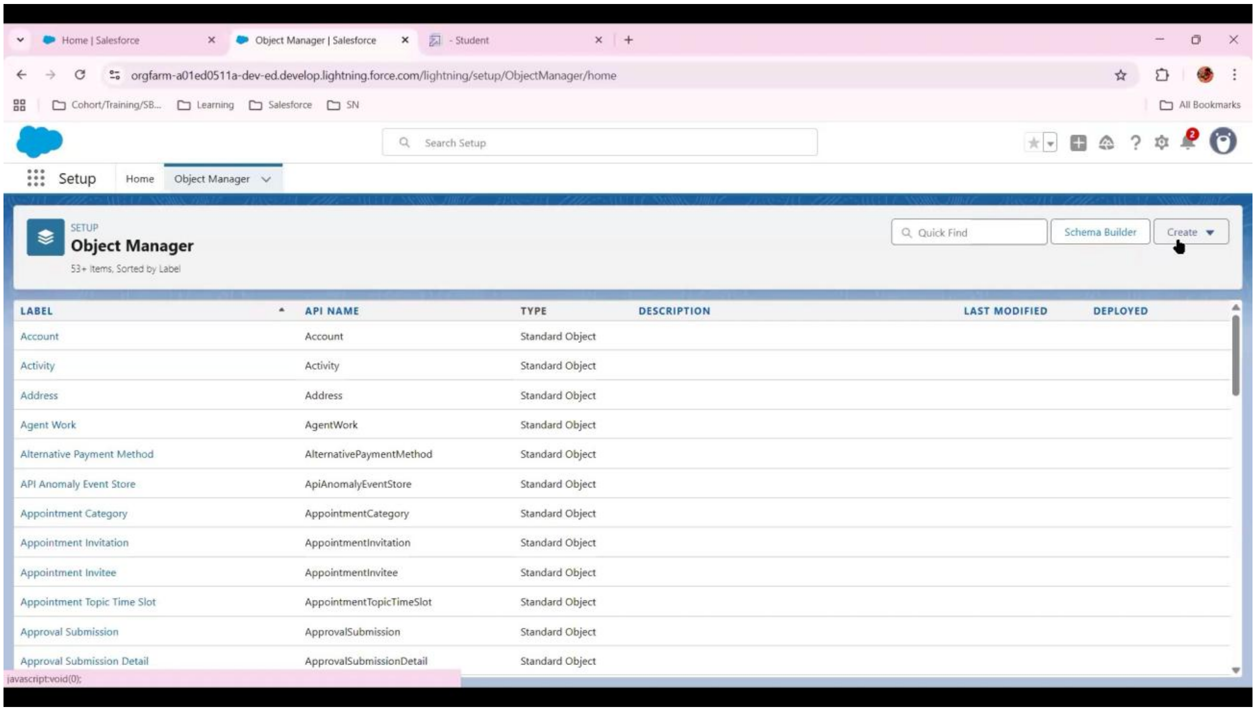
My Goals
Set personal weekly or monthly goals for emails, calls, and meetings.

Today's Events

To Do List

❖

Created objects: Property, Tenant, Lease, Payment

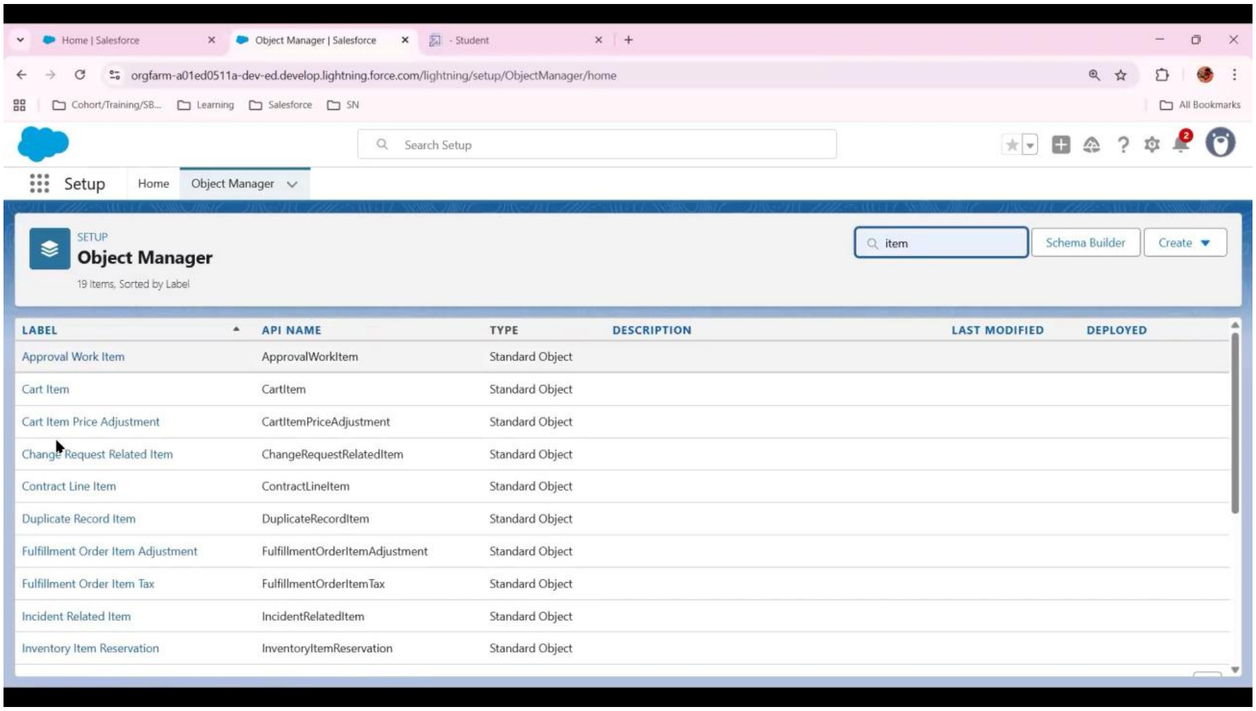


SETUP

Object Manager

53+ Items, Sorted by Label

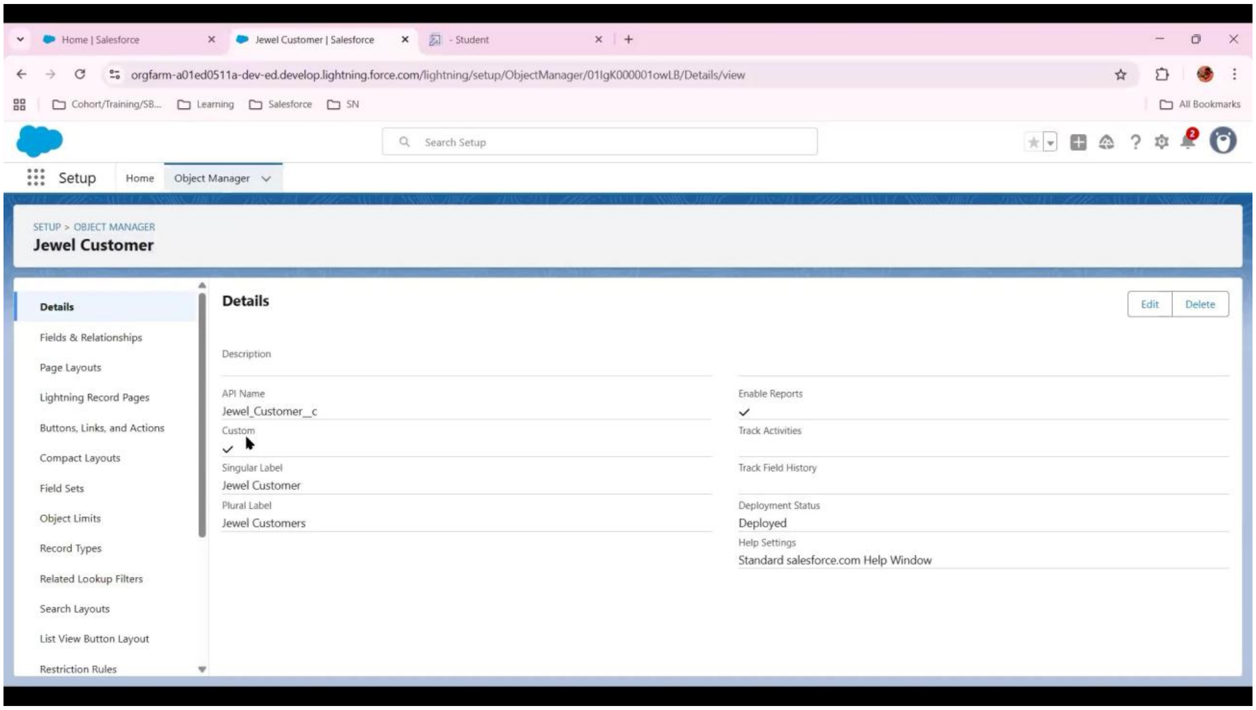
LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Account	Account	Standard Object			
Activity	Activity	Standard Object			
Address	Address	Standard Object			
Agent Work	AgentWork	Standard Object			
Alternative Payment Method	AlternativePaymentMethod	Standard Object			
API Anomaly Event Store	ApiAnomalyEventStore	Standard Object			
Appointment Category	AppointmentCategory	Standard Object			
Appointment Invitation	AppointmentInvitation	Standard Object			
Appointment Invitee	AppointmentInvitee	Standard Object			
Appointment Topic Time Slot	AppointmentTopicTimeSlot	Standard Object			
Approval Submission	ApprovalSubmission	Standard Object			
Approval Submission Detail	ApprovalSubmissionDetail	Standard Object			



LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Approval Work Item	ApprovalWorkItem	Standard Object			
Cart Item	CartItem	Standard Object			
Cart Item Price Adjustment	CartItemPriceAdjustment	Standard Object			
Change Request Related Item	ChangeRequestRelatedItem	Standard Object			
Contract Line Item	ContractLineItem	Standard Object			
Duplicate Record Item	DuplicateRecordItem	Standard Object			
Fulfillment Order Item Adjustment	FulfillmentOrderItemAdjustment	Standard Object			
Fulfillment Order Item Tax	FulfillmentOrderItemTax	Standard Object			
Incident Related Item	IncidentRelatedItem	Standard Object			
Inventory Item Reservation	InventoryItemReservation	Standard Object			

- ❖ **Configured fields and relationships**

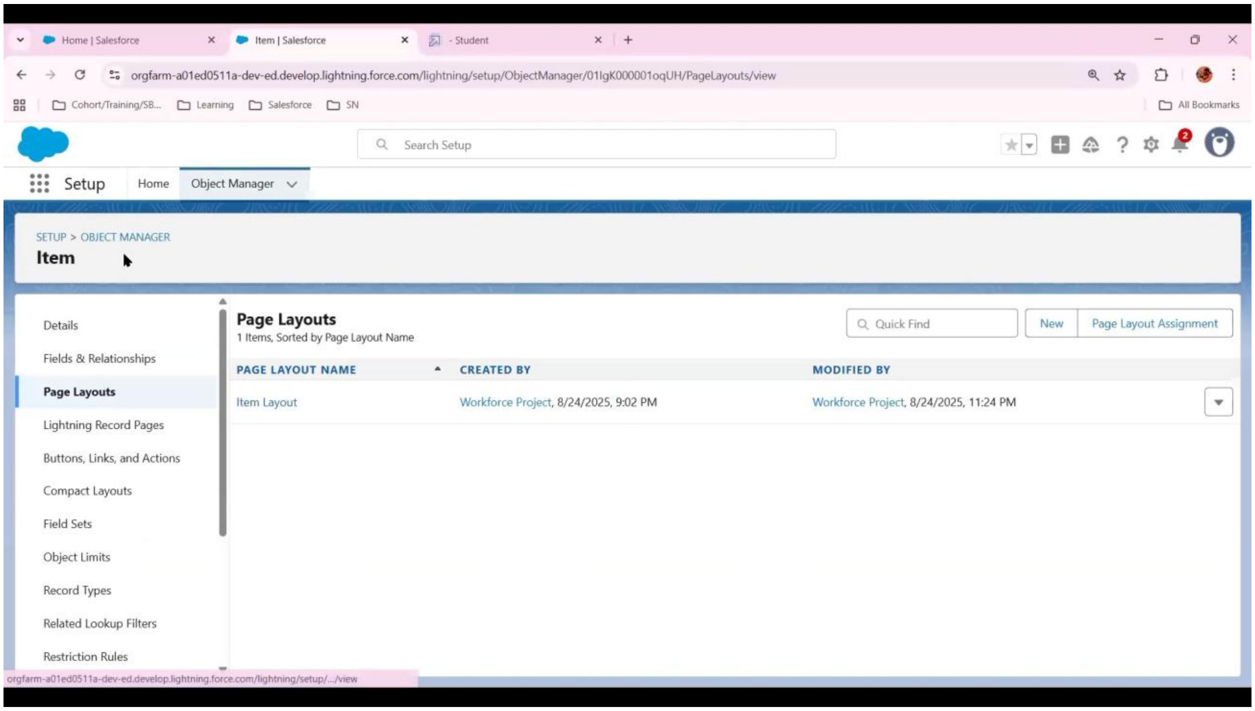
❖



Jewel Customer

Details

Description	
API Name: Jewel_Customer__c	Enable Reports: ✓
Custom: ✓	Track Activities
Singular Label: Jewel Customer	Track Field History
Plural Label: Jewel Customers	Deployment Status: Deployed
	Help Settings: Standard salesforce.com Help Window



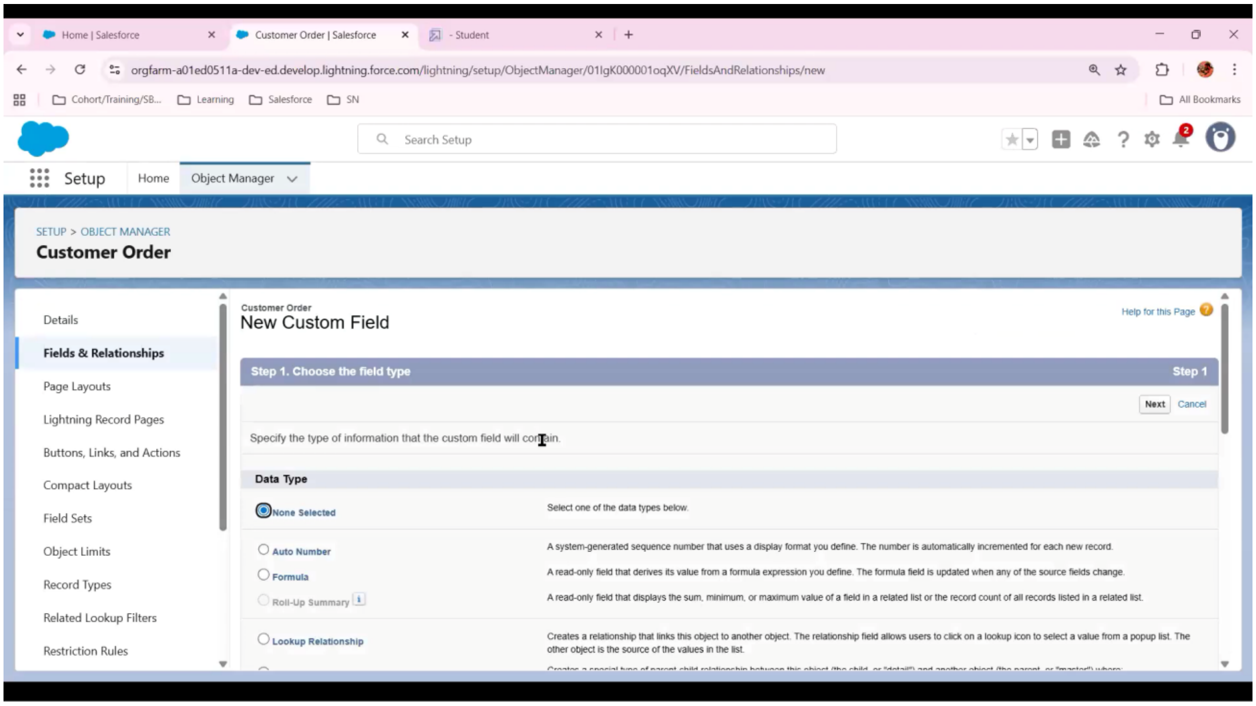
Item

Page Layouts

1 Items, Sorted by Page Layout Name

PAGE LAYOUT NAME	CREATED BY	MODIFIED BY
Item Layout	Workforce Project, 8/24/2025, 9:02 PM	Workforce Project, 8/24/2025, 11:24 PM

Create New Custom Object



Customer Order

Customer Order

New Custom Field

Step 1. Choose the field type — Step 1

Specify the type of information that the custom field will contain.

Data Type

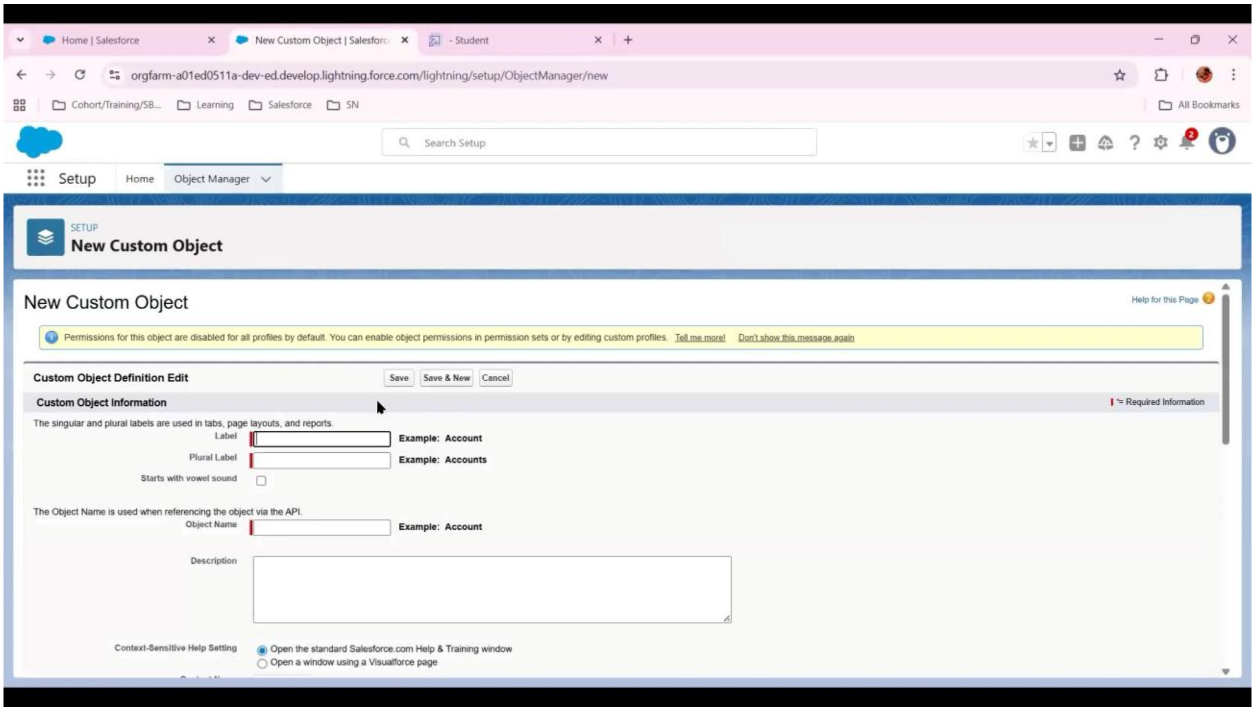
● None Selected — Select one of the data types below.

○ Auto Number — A system-generated sequence number that uses a display format you define. The number is automatically incremented for each new record.

○ Formula — A read-only field that derives its value from a formula expression you define. The formula field is updated when any of the source fields change.

○ Roll-Up Summary — A read-only field that displays the sum, minimum, or maximum value of a field in a related list or the record count of all records listed in a related list.

○ Lookup Relationship — Creates a relationship that links this object to another object. The relationship field allows users to click on a lookup icon to select a value from a popup list. The other object is the source of the values in the list.



New Custom Object

New Custom Object

Permissions for this object are disabled for all profiles by default. You can enable object permissions in permission sets or by editing custom profiles.

Custom Object Definition Edit

Custom Object Information — | = Required Information

The singular and plural labels are used in tabs, page layouts, and reports.

Label: | Example: Account

Plural Label: | Example: Accounts

Starts with vowel sound: □

The Object Name is used when referencing the object via the API.

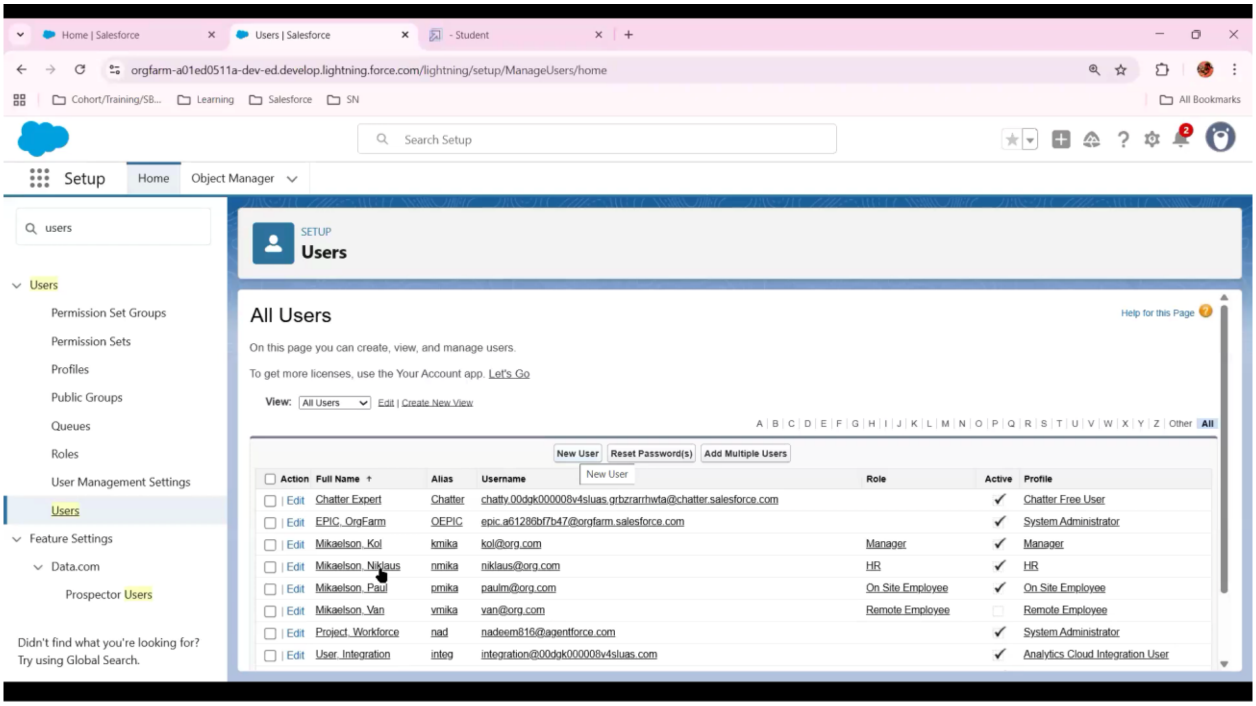
Object Name: | Example: Account

Description:

Context-Sensitive Help Setting: ● Open the standard Salesforce.com Help & Training window ○ Open a window using a Visualforce page

❖ Create New User and Roles

❖



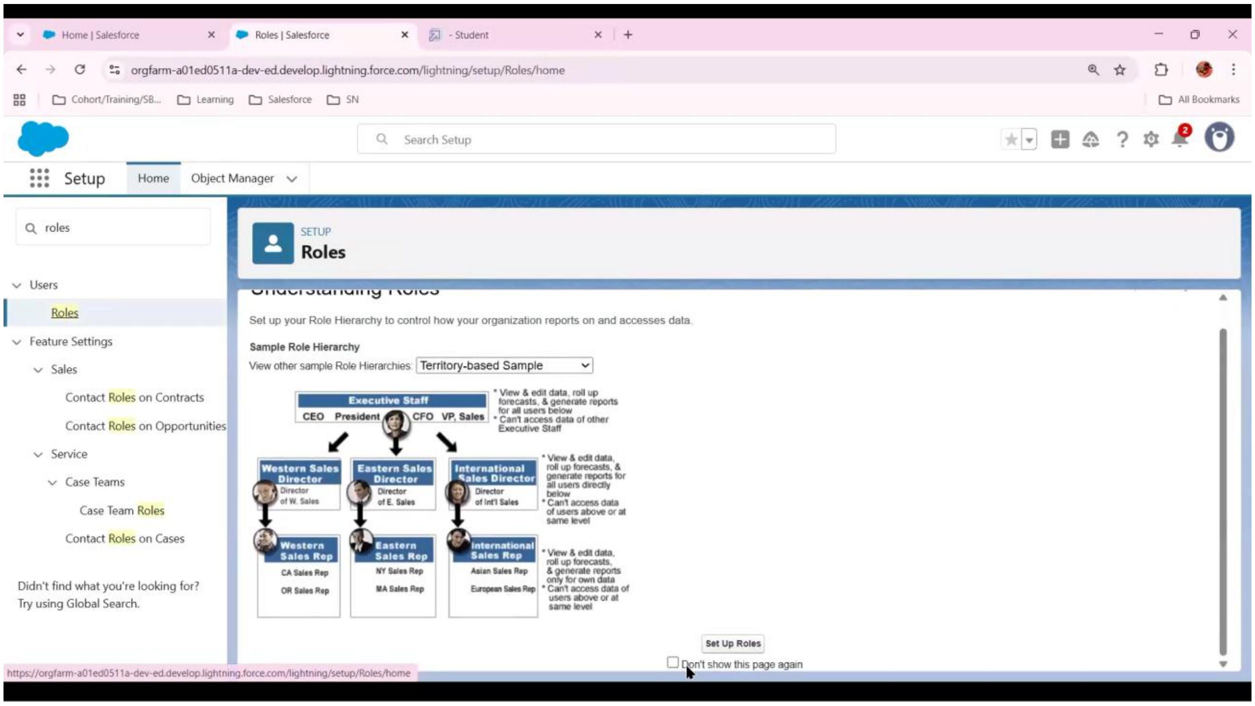
All Users

On this page you can create, view, and manage users.

To get more licenses, use the Your Account app. Let's Go

View: All Users Edit | Create New View

□ Action	Full Name ↑	Alias	Username	Role	Active	Profile
□ | Edit	Chatter Expert	Chatter	chatty.00dgk000008v4sluas.grbzrarrhwta@chatter.salesforce.com		✓	Chatter Free User
□ | Edit	EPIC, OrgFarm	OEPIC	epic.a61286bf7b47@orgfarm.salesforce.com		✓	System Administrator
□ | Edit	Mikaelson, Kol	kmika	kol@org.com	Manager	✓	Manager
□ | Edit	Mikaelson, Niklaus	nmika	niklaus@org.com	HR	✓	HR
□ | Edit	Mikaelson, Paul	pmika	paulm@org.com	On Site Employee	✓	On Site Employee
□ | Edit	Mikaelson, Van	vmika	van@org.com	Remote Employee	□	Remote Employee
□ | Edit	Project, Workforce	nad	nadeem816@agentforce.com		✓	System Administrator
□ | Edit	User, Integration	integ	integration@00dgk000008v4sluas.com		✓	Analytics Cloud Integration User


Home | Salesforce
Roles | Salesforce
- Student
orgfarm-a01ed0511a-dev-ed.develop.lightning.force.com/lightning/setup/Roles/home
Cohort/Training/SB...
Learning
Salesforce
SN
All Bookmarks
Search Setup
Setup
Home
Object Manager
roles
Users
Roles
Feature Settings
Sales
Contact Roles on Contracts
Contact Roles on Opportunities
Service
Case Teams
Case Team Roles
Contact Roles on Cases
Didn't find what you're looking for? Try using Global Search.
SETUP
Roles
Executive Staff
CEO
President
CFO
VP, Sales
* View & edit data, roll up forecasts, & generate reports for all users below
* Can't access data of other Executive Staff
Western Sales Director
Director of W. Sales
Eastern Sales Director
Director of E. Sales
International Sales Director
Director of Int'l Sales
* View & edit data, roll up forecasts, & generate reports for all users directly below
* Can't access data of users above or at same level
Western Sales Rep
CA Sales Rep
OR Sales Rep
Eastern Sales Rep
NY Sales Rep
MA Sales Rep
International Sales Rep
Asian Sales Rep
European Sales Rep
* View & edit data, roll up forecasts, & generate reports only for own data
* Can't access data of users above or at same level
Set Up Roles
https://orgfarm-a01ed0511a-dev-ed.develop.lightning.force.com/lightning/setup/Roles/home

Understanding Roles

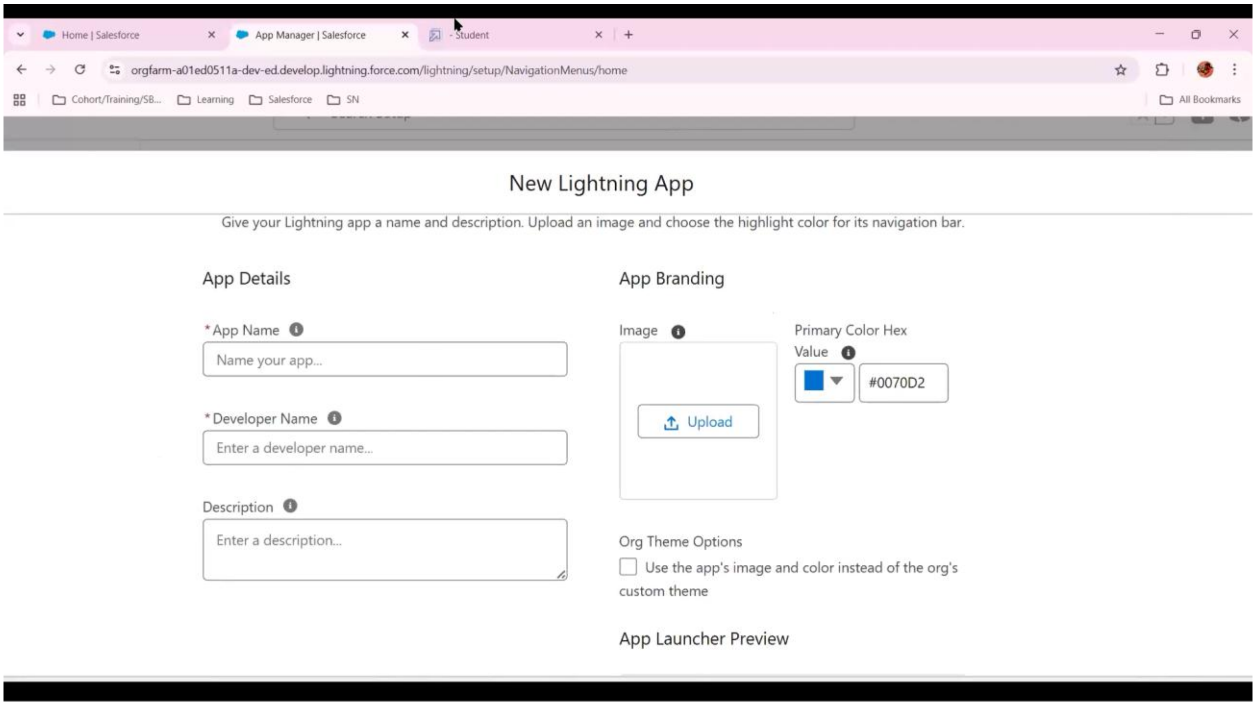
Set up your Role Hierarchy to control how your organization reports on and accesses data.

Sample Role Hierarchy

View other sample Role Hierarchies: Territory-based Sample

□ Don't show this page again

Create New Lighting App



New Lightning App

Give your Lightning app a name and description. Upload an image and choose the highlight color for its navigation bar.

App Details

*App Name: Name your app...

*Developer Name: Enter a developer name...

Description: Enter a description...

App Branding

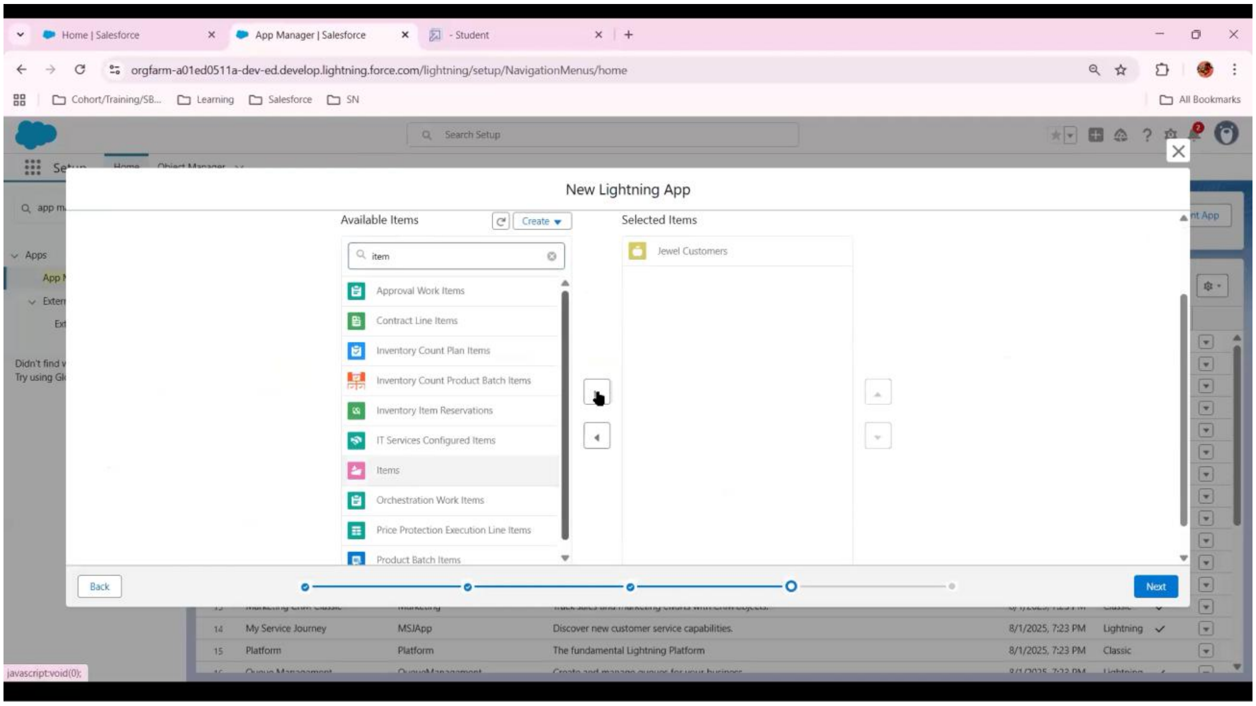
Image: Upload

Primary Color Hex Value: #0070D2

Org Theme Options

☐ Use the app's image and color instead of the org's custom theme

App Launcher Preview



New Lightning App

Available Items

Search: item

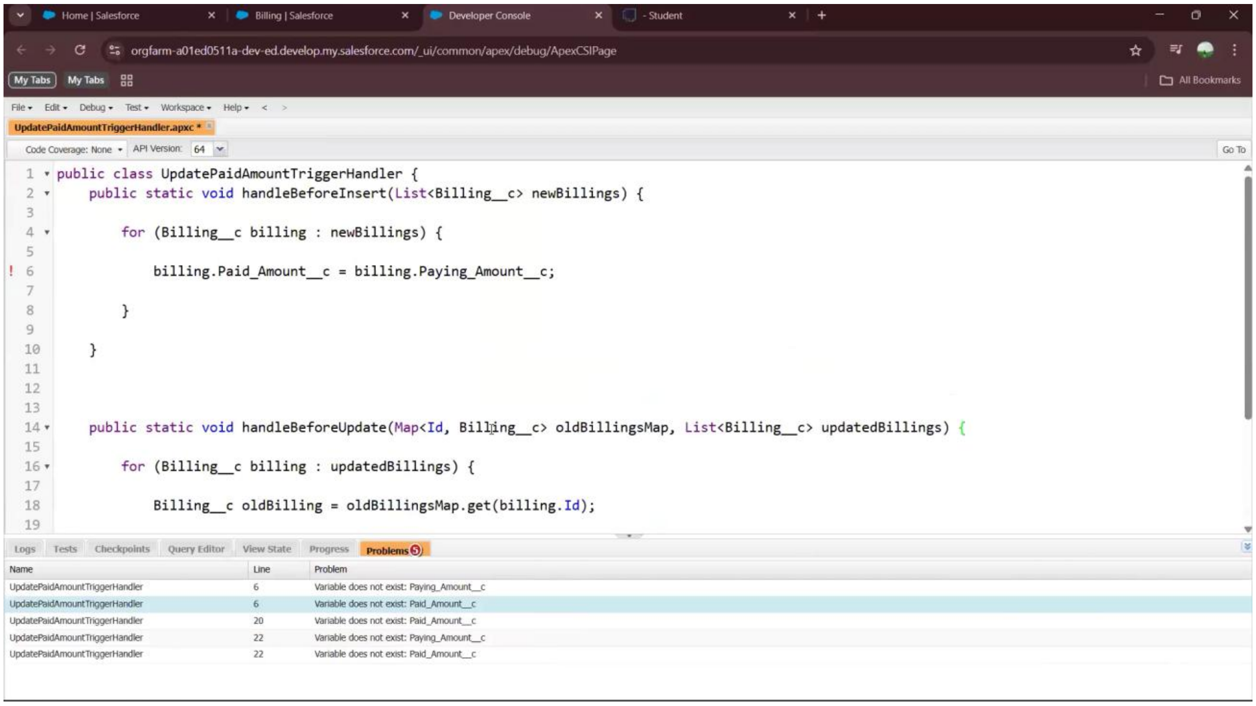
- Approval Work Items
- Contract Line Items
- Inventory Count Plan Items
- Inventory Count Product Batch Items
- Inventory Item Reservations
- IT Services Configured Items
- Items
- Orchestration Work Items
- Price Protection Execution Line Items
- Product Batch Items

Selected Items

- Jewel Customers

14	My Service Journey	MSJApp	Discover new customer service capabilities.	8/1/2025, 7:23 PM	Lightning	✓
15	Platform	Platform	The fundamental Lightning Platform	8/1/2025, 7:23 PM	Classic	

❖ Create An Apex Class

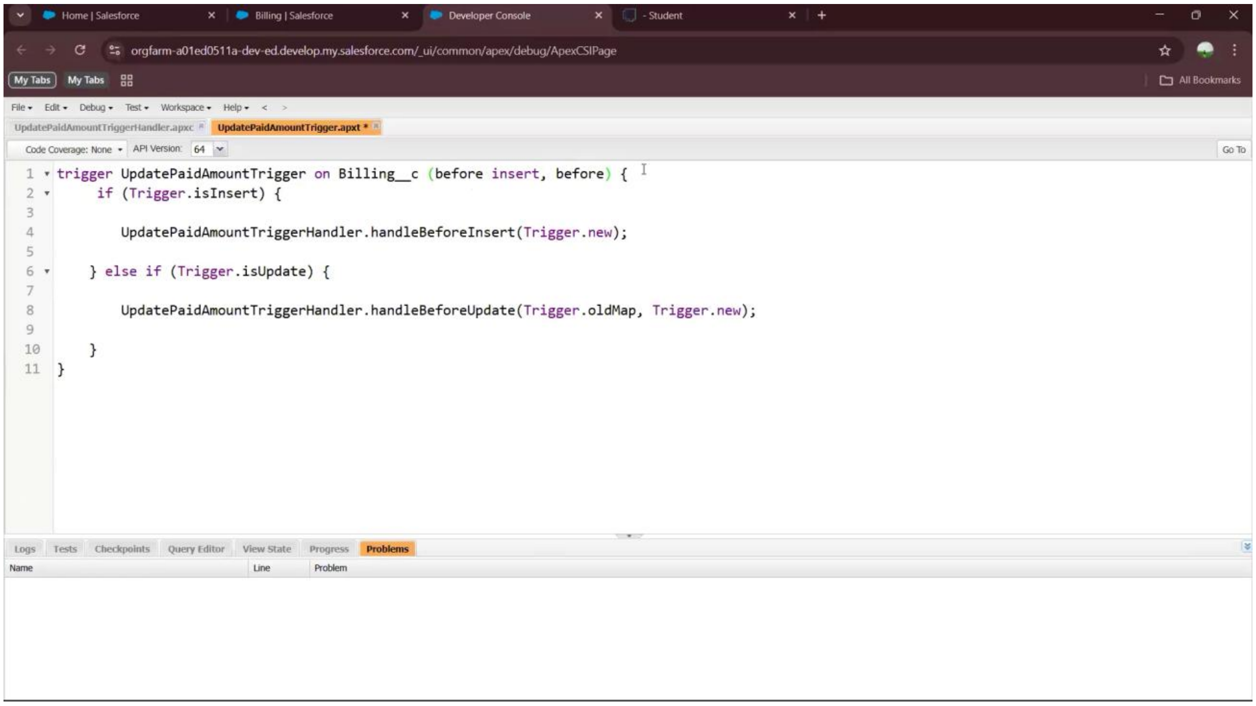


orgfarm-a01ed0511a-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

Code Coverage: None ▾ API Version: 64

```
1   public class UpdatePaidAmountTriggerHandler {
2       public static void handleBeforeInsert(List<Billing__c> newBillings) {
3
4           for (Billing__c billing : newBillings) {
5
6               billing.Paid_Amount__c = billing.Paying_Amount__c;
7
8           }
9
10      }
11
12
13
14      public static void handleBeforeUpdate(Map<Id, Billing__c> oldBillingsMap, List<Billing__c> updatedBillings) {
15
16          for (Billing__c billing : updatedBillings) {
17
18              Billing__c oldBilling = oldBillingsMap.get(billing.Id);
19
```

Name	Line	Problem
UpdatePaidAmountTriggerHandler	6	Variable does not exist: Paying_Amount__c
UpdatePaidAmountTriggerHandler	6	Variable does not exist: Paid_Amount__c
UpdatePaidAmountTriggerHandler	20	Variable does not exist: Paid_Amount__c
UpdatePaidAmountTriggerHandler	22	Variable does not exist: Paying_Amount__c
UpdatePaidAmountTriggerHandler	22	Variable does not exist: Paid_Amount__c



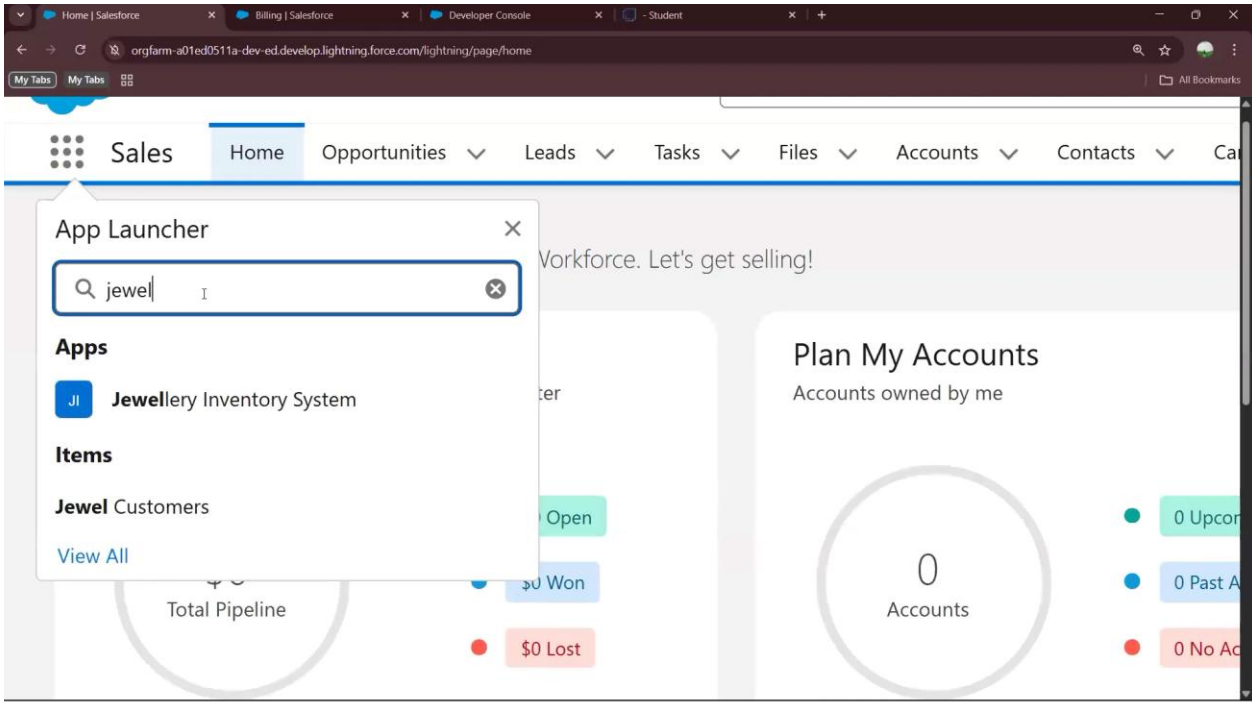
orgfarm-a01ed0511a-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

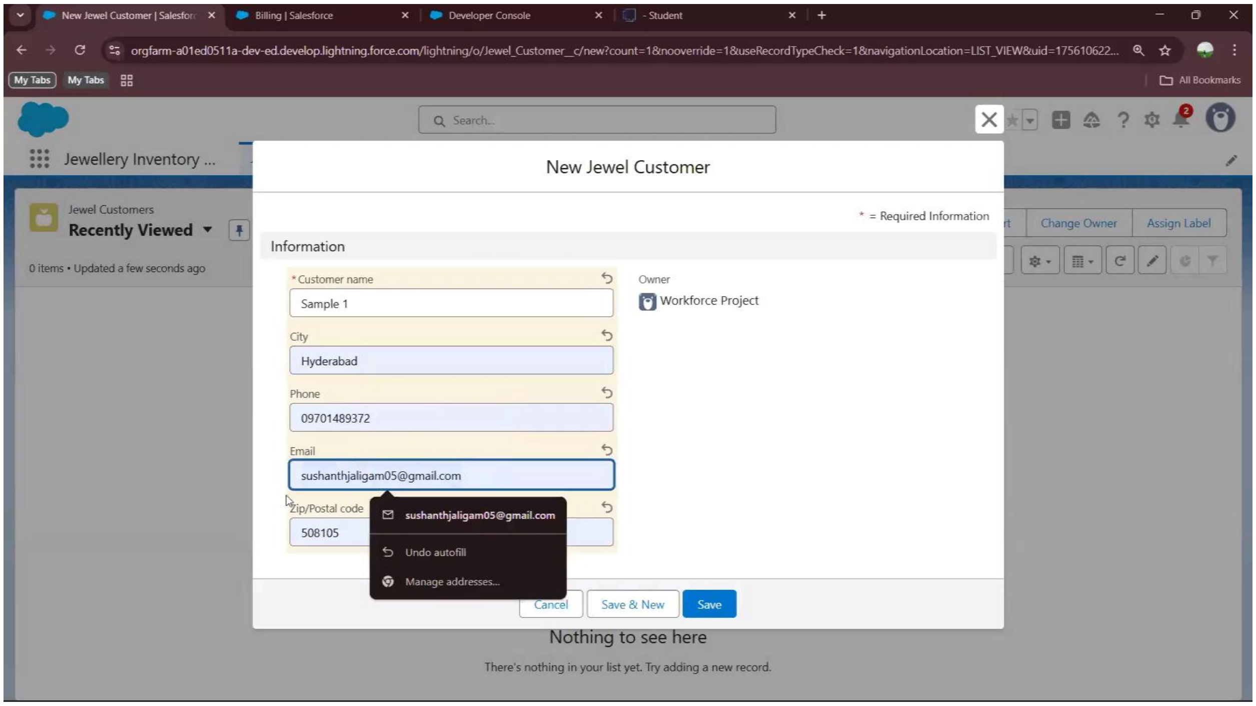
Code Coverage: None ▾ API Version: 64

```
trigger UpdatePaidAmountTrigger on Billing__c (before insert, before) {
     if (Trigger.isInsert) {

        UpdatePaidAmountTriggerHandler.handleBeforeInsert(Trigger.new);

    } else if (Trigger.isUpdate) {

        UpdatePaidAmountTriggerHandler.handleBeforeUpdate(Trigger.oldMap, Trigger.new);

    }
}
```

Name	Line	Problem

- **Go to Home and Search For Jewellery Inventory System**




New Jewel Customer | Salesfor
Billing | Salesforce
Developer Console
- Student
orgfarm-a01ed0511a-dev-ed.develop.lightning.force.com/lightning/o/Jewel_Customer__c/new?count=1&nooverride=1&useRecordTypeCheck=1&navigationLocation=LIST_VIEW&uid=175610622...
My Tabs
My Tabs
All Bookmarks
Search...
Jewellery Inventory ...
Jewel Customers
Recently Viewed
0 items • Updated a few seconds ago
Change Owner
Assign Label
sushanthjaligam05@gmail.com
Undo autofill
Manage addresses...
Cancel
Save & New
Save
Nothing to see here
There's nothing in your list yet. Try adding a new record.

New Jewel Customer

* = Required Information

Information

* Customer name: Sample 1

City: Hyderabad

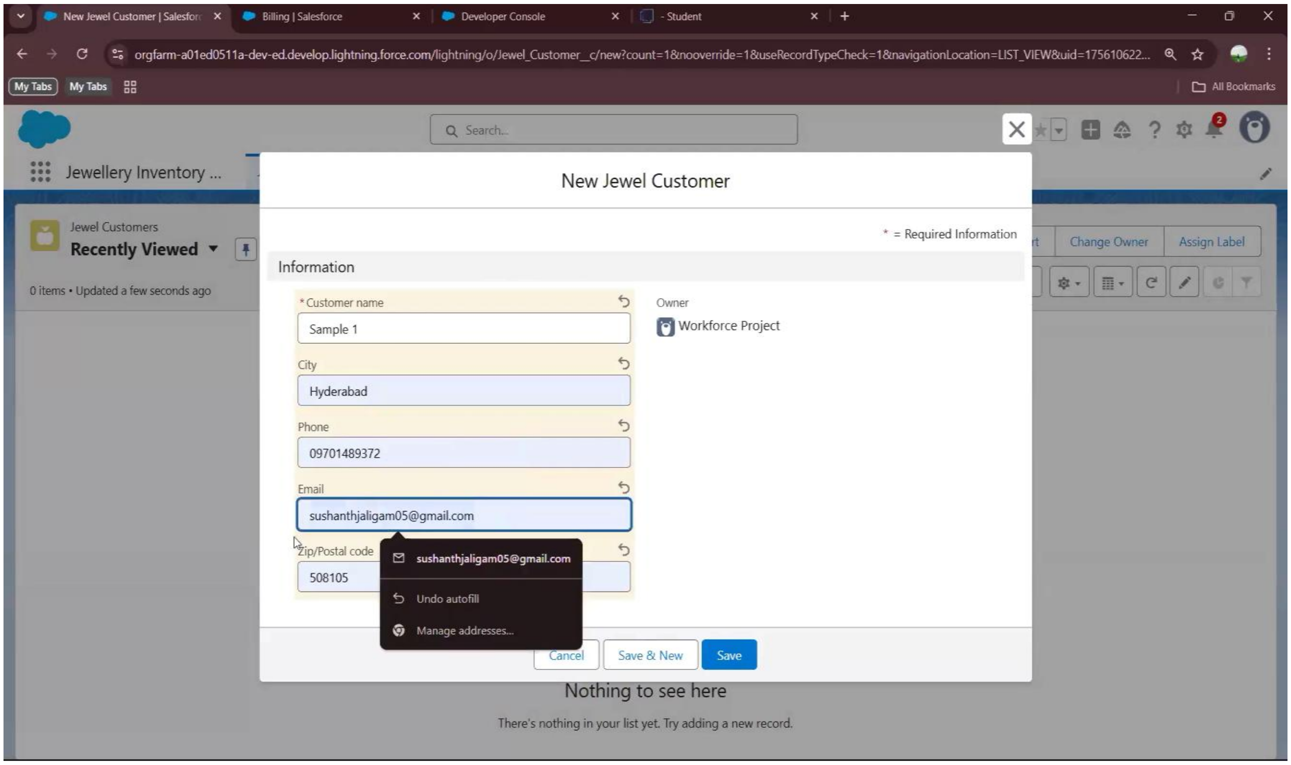
Phone: 09701489372

Email: sushanthjaligam05@gmail.com

Zip/Postal code: 508105

Owner: Workforce Project

- **Create New Customer and Give Price**


New Jewel Customer | Salesforce
Billing | Salesforce
Developer Console
- Student
orgfarm-a01ed0511a-dev-ed.develop.lightning.force.com/lightning/o/Jewel_Customer__c/new?count=1&nooverride=1&useRecordTypeCheck=1&navigationLocation=LIST_VIEW&uid=175610622...
My Tabs
My Tabs
All Bookmarks
Search...
Jewellery Inventory ...
Jewel Customers
Recently Viewed
0 items • Updated a few seconds ago
Change Owner
Assign Label
Nothing to see here
There's nothing in your list yet. Try adding a new record.
Undo autofill
Manage addresses...
Cancel
Save & New
Save

New Jewel Customer

* = Required Information

Information

*Customer name: Sample 1

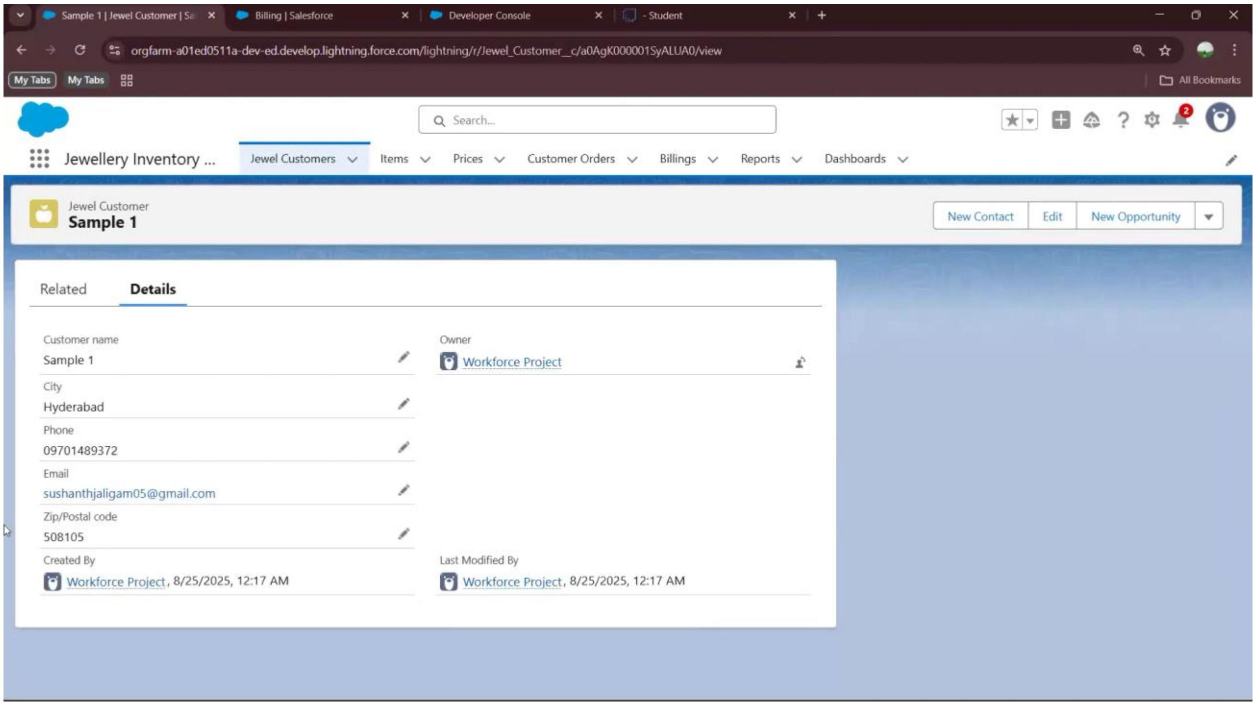
City: Hyderabad

Phone: 09701489372

Email: sushanthjaligam05@gmail.com

Zip/Postal code: 508105

Owner: Workforce Project



orgfarm-a01ed0511a-dev-ed.develop.lightning.force.com/lightning/r/Jewel_Customer__c/a0AgK000001SyALUA0/view

Jewel Customer
Sample 1

Customer name: Sample 1

City: Hyderabad

Phone: 09701489372

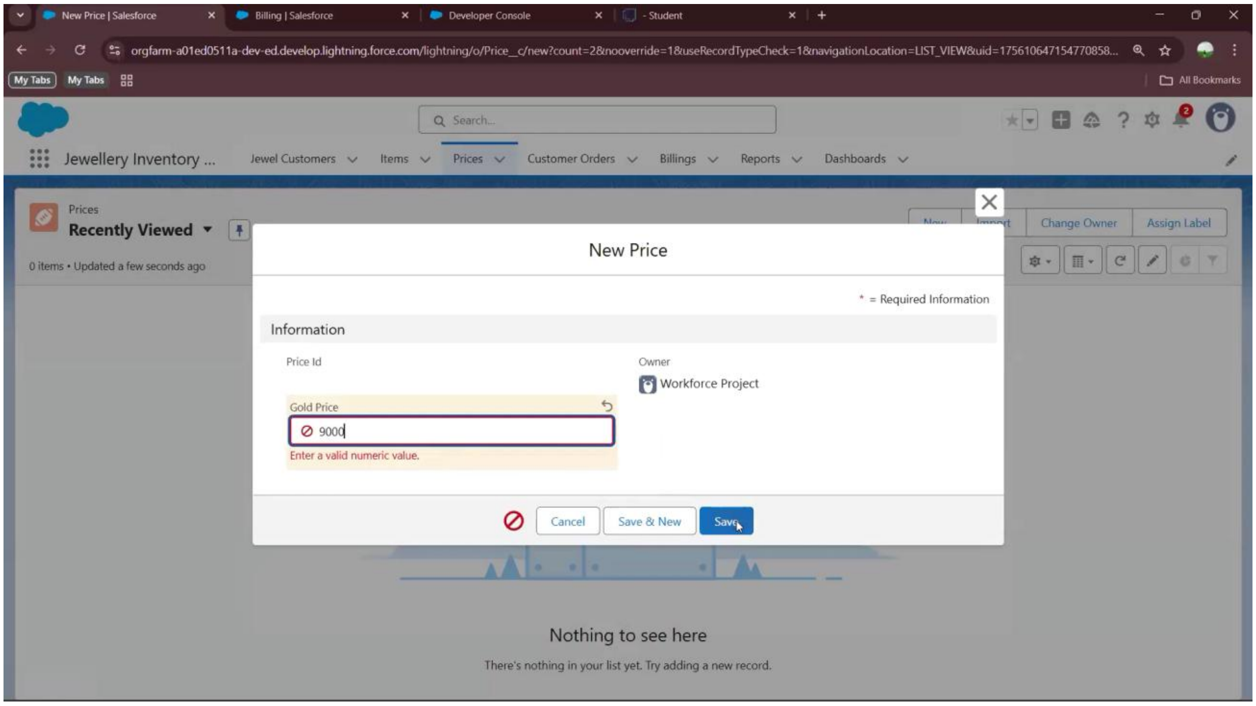
Email: sushanthjaligam05@gmail.com

Zip/Postal code: 508105

Created By: Workforce Project, 8/25/2025, 12:17 AM

Owner: Workforce Project

Last Modified By: Workforce Project, 8/25/2025, 12:17 AM



orgfarm-a01ed0511a-dev-ed.develop.lightning.force.com/lightning/o/Price__c/new?count=2&nooverride=1&useRecordTypeCheck=1&navigationLocation=LIST_VIEW&uid=175610647154770858...

Prices
Recently Viewed
0 items • Updated a few seconds ago

New Price

* = Required Information

Information

Price Id

Owner: Workforce Project

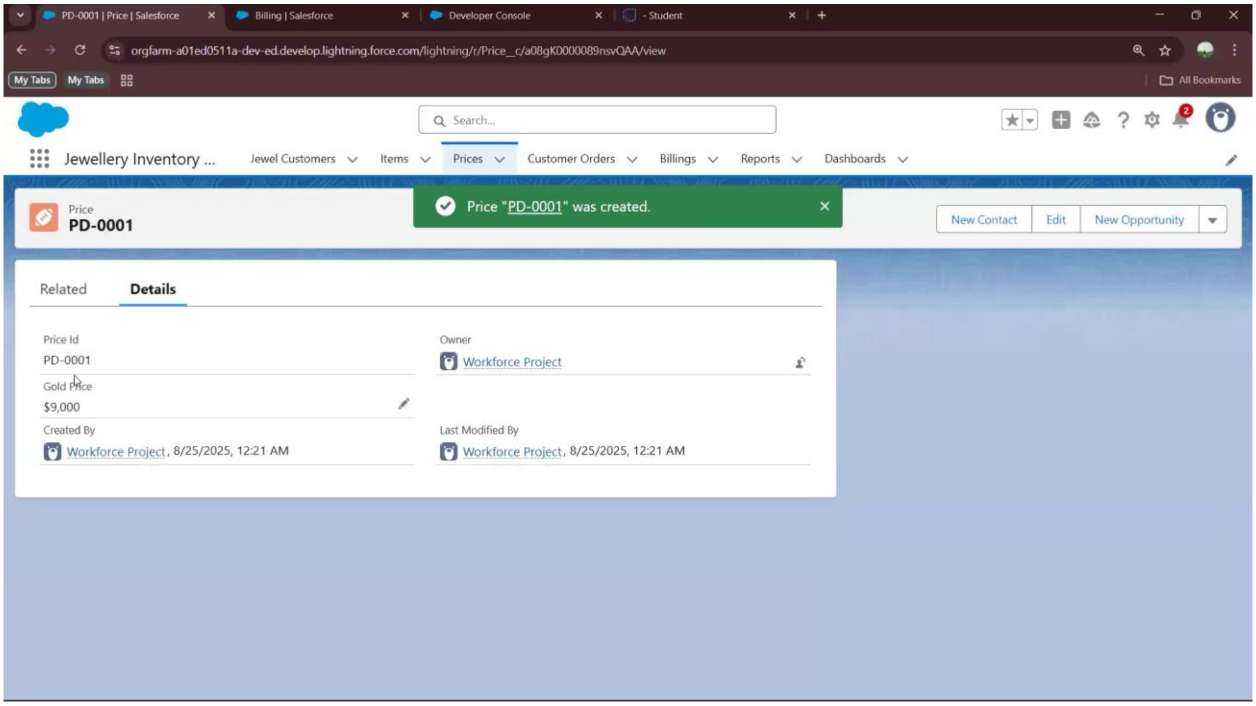
Gold Price: 9000

Enter a valid numeric value.

Nothing to see here

There's nothing in your list yet. Try adding a new record.

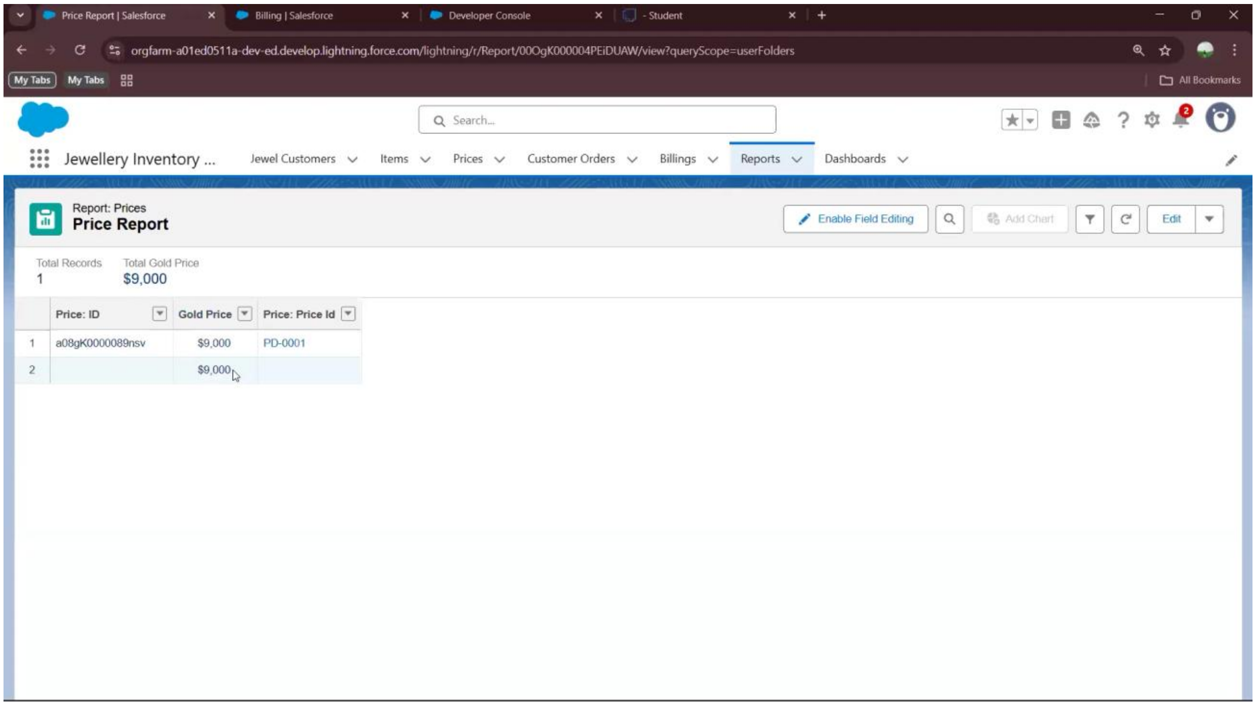
- ❖ **Check For Price Report**



Price "PD-0001" was created.

Price
PD-0001

Price Id PD-0001	Owner Workforce Project
Gold Price $9,000	
Created By Workforce Project, 8/25/2025, 12:21 AM	Last Modified By Workforce Project, 8/25/2025, 12:21 AM



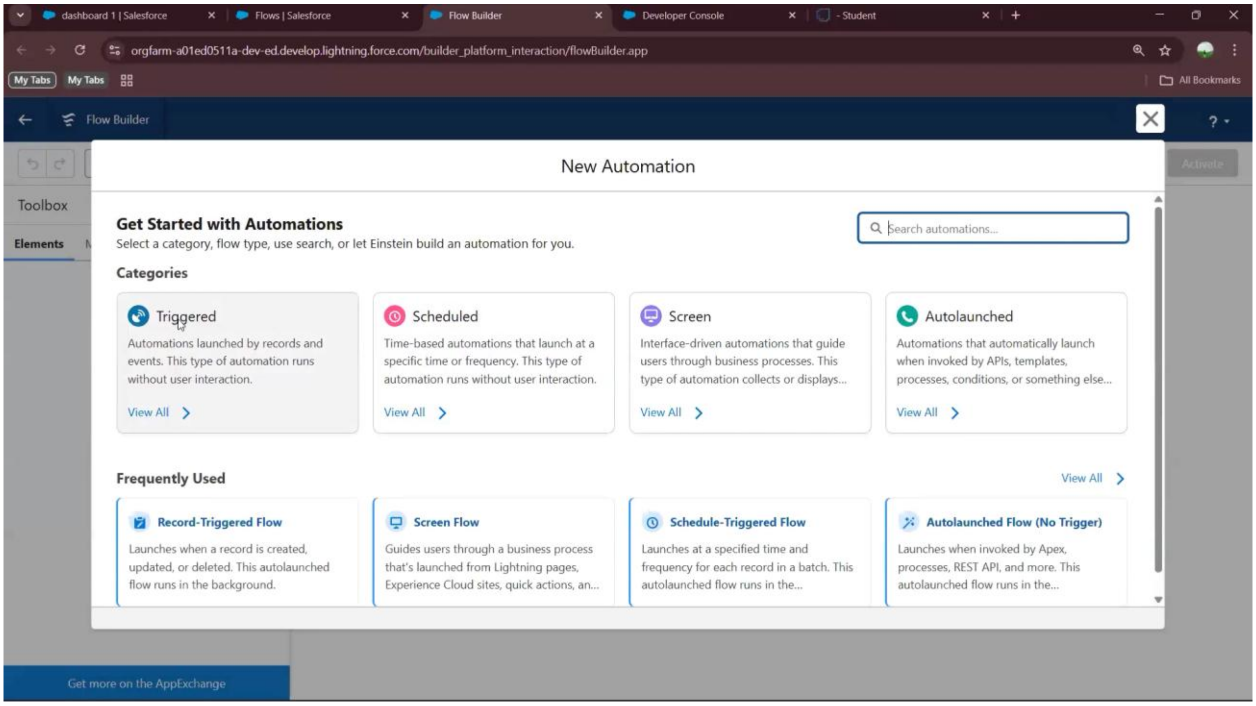
Report: Prices
Price Report

Total Records
1

Total Gold Price
$9,000

	Price: ID	Gold Price	Price: Price Id
1	a08gK0000089nsv	$9,000	PD-0001
2		$9,000	

❖ Create New Automation and Start



New Automation

Get Started with Automations

Select a category, flow type, use search, or let Einstein build an automation for you.

Search automations...

Categories

Triggered — Automations launched by records and events. This type of automation runs without user interaction. View All

Scheduled — Time-based automations that launch at a specific time or frequency. This type of automation runs without user interaction. View All

Screen — Interface-driven automations that guide users through business processes. This type of automation collects or displays... View All

Autolaunched — Automations that automatically launch when invoked by APIs, templates, processes, conditions, or something else... View All

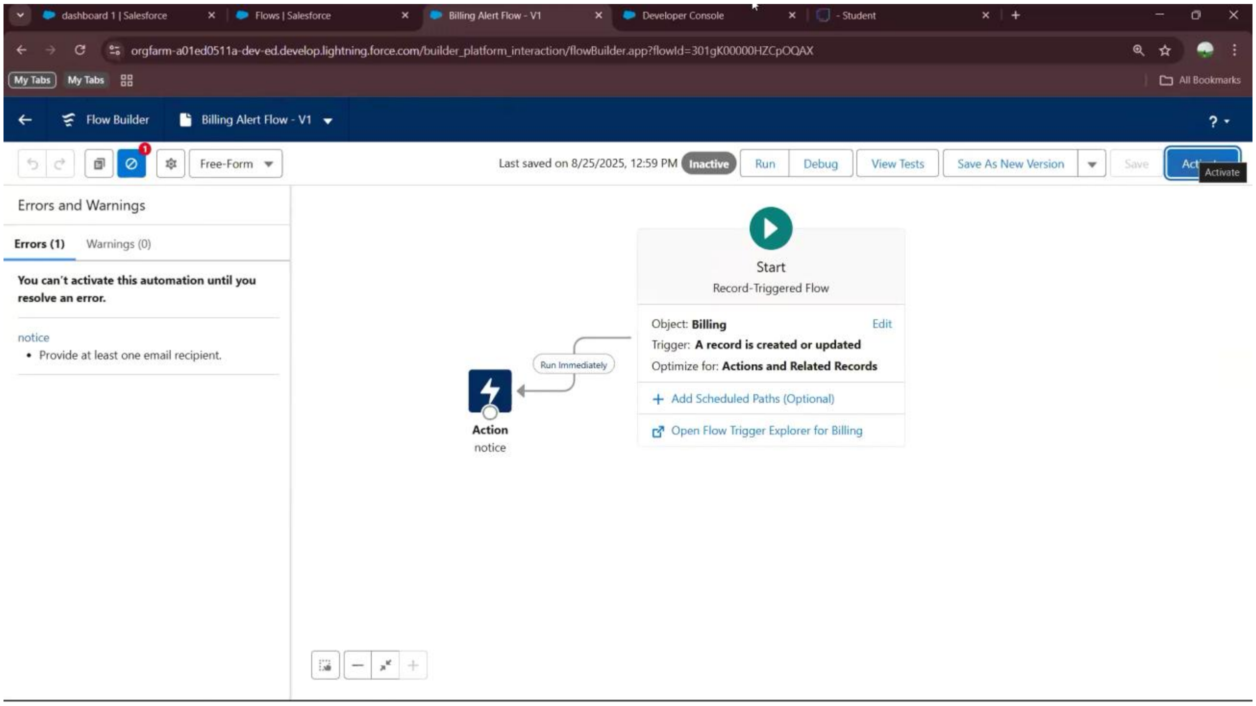
Frequently Used View All

Record-Triggered Flow — Launches when a record is created, updated, or deleted. This autolaunched flow runs in the background.

Screen Flow — Guides users through a business process that's launched from Lightning pages, Experience Cloud sites, quick actions, an...

Schedule-Triggered Flow — Launches at a specified time and frequency for each record in a batch. This autolaunched flow runs in the...

Autolaunched Flow (No Trigger) — Launches when invoked by Apex, processes, REST API, and more. This autolaunched flow runs in the...



Last saved on 8/25/2025, 12:59 PM

Errors and Warnings

Errors (1) Warnings (0)

You can't activate this automation until you resolve an error.

notice
- Provide at least one email recipient.

Start

Record-Triggered Flow

Object: **Billing** Edit

Trigger: **A record is created or updated**

Optimize for: **Actions and Related Records**

+ Add Scheduled Paths (Optional)

Open Flow Trigger Explorer for Billing

ADVANTAGES :

1. DOMAIN-SPECIFIC SOLUTION :

- **Focused on jewelry business needs like inventory tracking (gold, diamonds, stones), customized billing, and customer preferences.**

- **Easier to target a niche market compared to generic CRMs.**

2. SCALABILITY & FLEXIBILITY :

- **Developers can design the system to handle both small jewelry shops and larger chains.**

- **Can integrate features like loyalty programs, gemstone certifications, and repair tracking.**

3. INTEGRATION OPPORTUNITIES :

- **Can connect with POS systems, e-commerce websites, and accounting tools.**

- **Increases product value and gives developers scope to expand modules.**

4. DATA-DRIVEN INSIGHTS :

- **Offers jewelers customer insights (purchase history, trends).**

- **Developers can implement analytics dashboards, improving usability and business decision-making.**

5. USER-FRIENDLY UI FOCUS :

- **Jewelry staff may not be very tech-savvy, so creating an intuitive interface is a strong developer achievement.**

- **Opportunity to showcase skill in UI/UX design.**

6. SECURITY-ORIENTED DEVELOPMENT :

- **Sensitive customer and pricing data require strong encryption.**
- **Developers gain experience in secure coding practices (GDPR, data privacy compliance).**

DISADVANTAGES :

1. COMPLEX CUSTOMIZATION :

- **Jewelry items vary in weight, purity, stone count, certification, etc.**
- **Building flexible yet simple modules can be technically challenging.**

2. HIGH DEVELOPMENT COST & TIME :

- **Requires multiple features (CRM + inventory + invoicing + reporting).**
- **Can be overwhelming for a student project or small developer team.**

3. SECURITY RISKS :

- **Handling customer identity data, purchase values, and payment details.**
- **Vulnerable to cyber-attacks if not built with strong security layers.**

4. MAINTENANCE CHALLENGES :

- **Jewelry businesses may request frequent changes (new schemes, pricing structures, GST/tax updates).**
- **Developers need to ensure easy maintainability and version upgrades.**

5. INTEGRATION DIFFICULTIES :

- **Not all jewelers use modern systems; integrating with old accounting software or manual processes may be hard.**

6. MARKET COMPETITION :

- **Existing CRM tools (Zoho, Salesforce, etc.) already dominate.**
- **Niche customization may limit scalability outside the jewelry domain.**

CONCLUSION :

- **The development of the CRM Application for Jewel Management has successfully integrated customer relationship management, sales tracking, and inventory control into a single streamlined system. By centralizing customer data, purchase history, and stock information, this application not only sakthivel enhances business efficiency but also improves customer satisfaction through personalized services and timely responses.**
- **From a developer’s perspective, the project demonstrated the importance of database design, system integration, and user-friendly interface development in building a reliable solution. This application sakthi can be further expanded with advanced features such as data analytics, AI-driven recommendations, and mobile compatibility, ensuring long-term adaptability in the jewelry business.**
- **In conclusion, the project proves that a well-designed CRM system can bridge the gap between technology and traditional jewelry business operations, resulting in better decision-making, improved sales performance, and stronger customer relationships.**

APPENDIX :

```
public with sharing class JewelCRMManager {

   // Add Customer
   public static Id addCustomer(String name, String phone, String email) {
      Customer__c cust = new Customer__c(
         Name = name,
         Phone__c = phone,
         Email__c = email
      );
      insert cust;
return cust.Id;
   }
   // Get All Customers     public static List<Customer__c>
getCustomers() {        return [SELECT Id, Name, Phone__c, Email__c
FROM Customer__c];
   }

   // Add Jewel Item
   public static Id addJewel(String name, String type, Decimal price, Integer stock) {
      Jewel__c jewel = new Jewel__c(
         Name = name,
         Type__c = type,
         Price__c = price,
         Stock__c = stock
      );
      insert jewel;
return jewel.Id;
   }

   // Get All Jewelry Inventory     public static
List<Jewel__c> getInventory() {
      return [SELECT Id, Name, Type__c, Price__c, Stock__c FROM Jewel__c];
   }

   // Record Sale
   public static Id addSale(Id customerId, Id jewelId, Integer qty, Date sDate) {
      Jewel__c jewel = [SELECT Id, Stock__c FROM Jewel__c WHERE Id = :jewelId LIMIT 1];
if (jewel.Stock__c < qty) {          throw new AuraHandledException('Not enough stock
available!');
      }

      // Reduce stock
jewel.Stock__c -= qty;       update
jewel;

      Sale__c sale = new Sale__c(
         Customer__c = customerId,
         Jewel__c = jewelId,
         Quantity__c = qty,
         Date__c = sDate
      );
      insert sale;
return sale.Id;
   }

   // Get All Sales     public static
List<Sale__c> getSales() {        return [
         SELECT Id, Customer__r.Name, Jewel__r.Name, Quantity__c, Date__c
FROM Sale__c
      ];
   }
```

```
}
```

APEX TEST CLASS :

```
@isTest public class
JewelCRMManagerTest {
    @isTest
    static void testCRMFlow() {
        // Add customer
        Id custId = JewelCRMManager.addCustomer('Sakthivel', '7845770375', 'alice@test.com');

        // Add jewel
        Id jewelId = JewelCRMManager.addJewel('Gold Ring', 'Ring', 15000, 10);

        // Record sale
        Id saleId = JewelCRMManager.addSale(custId, jewelId, 2, Date.today());

        // Query results
        List<Customer__c> customers = JewelCRMManager.getCustomers();
        List<Jewel__c> jewels = JewelCRMManager.getInventory();
        List<Sale__c> sales = JewelCRMManager.getSales();

        System.assertEquals(1, customers.size());
        System.assertEquals(1, jewels.size());
        System.assertEquals(1, sales.size());
    }
}
```