

## RECRUITING ASSISTANCE FOR THE HR MANAGERS

---

### (DEVELOPER)

MENTOR NAME - **SRI.L.LAKSHMI NARYANA**

TEAM ID - **LTVIP2024TMID11734**

**TEAM LEADER** - **K.YESWANTH KUMAR**



EMAIL - [yk8727426@gmail.com](mailto:yk8727426@gmail.com)

### TEAM MEMBERS

**\*MEMBER 1** - **G.SAI SURYA**



EMAIL - [saisuryaguthikonda@gmail.com](mailto:saisuryaguthikonda@gmail.com)



**\*MEMBER 2 - Y.SATYA CHARAN**



EMAIL – [satyacharan1720@gmail.com](mailto:satyacharan1720@gmail.com)

**\*MEMBER 3 - G.RAMA SATYANARAYANA REDDY**



EMAIL – [goluguriramureddy@gmail.com](mailto:goluguriramureddy@gmail.com)

**\*MEMBER 4 - P.L.S.DURGA PRASAD**

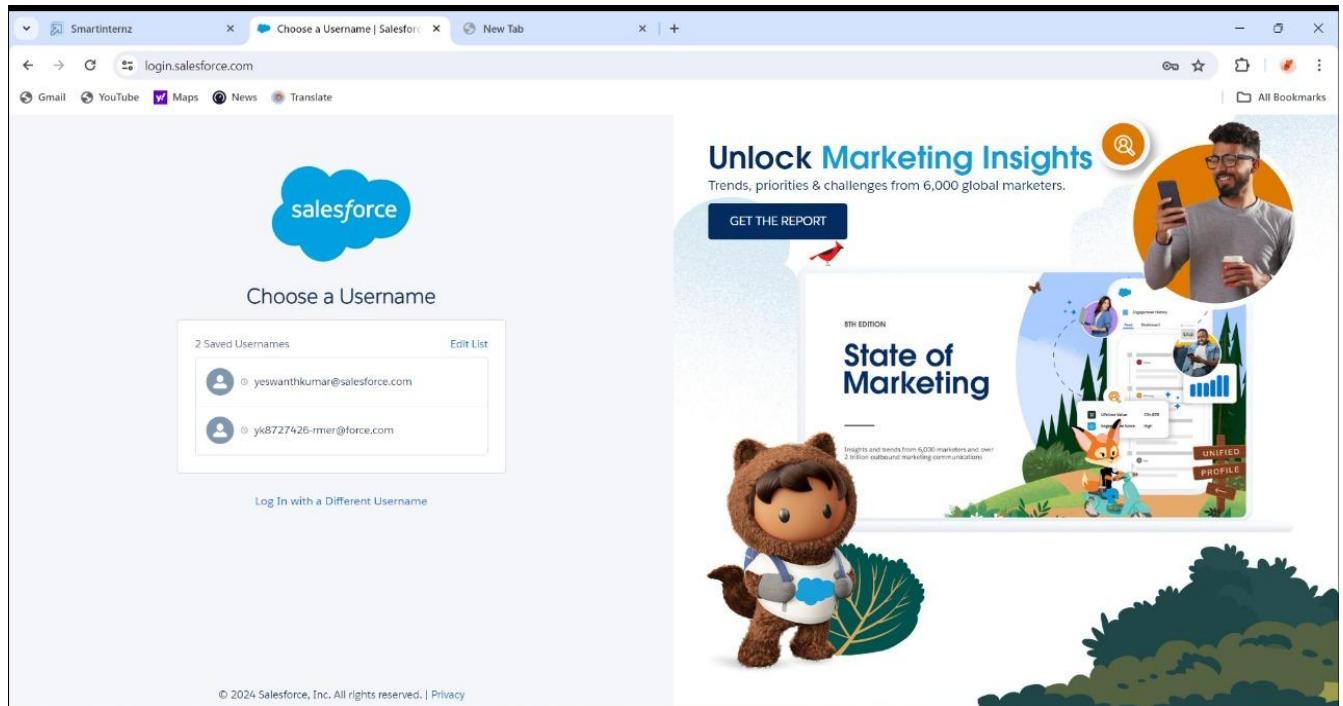


EMAIL - [plsdpyadav@gmail.com](mailto:plsdpyadav@gmail.com)

# creation of developer organization

*A Developer org has all the features and licenses you need to get started with Salesforce*

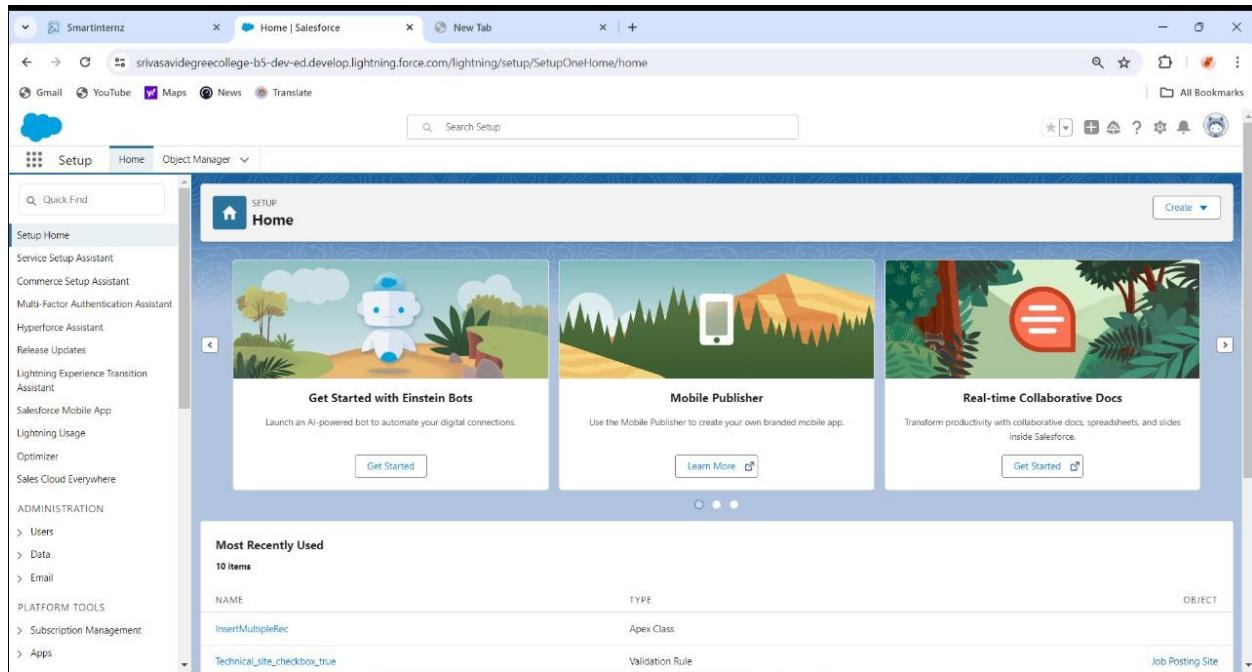
**Purpose and Goals:** Clearly define the purpose and goals of the developer organization. This includes identifying the target audience (internal developers, external developers, partners, etc.) and outlining what you aim to achieve through the organization. Whether it's fostering innovation, building a developer community, supporting API usage, or driving developer engagement, having a clear purpose helps align efforts and resources effectively.



**Resource Allocation and Support:** Allocate appropriate resources and provide necessary support to empower developers within the organization.

# Setup page

*The setup page will appear as below.*



In Salesforce, the setup page is a central hub where administrators can configure and customize various aspects of their Salesforce organization.

Here's some information about the setup page:

\*Navigation

\*Setup Menu

\*Administer

\*Objects and Fields

\*Develop

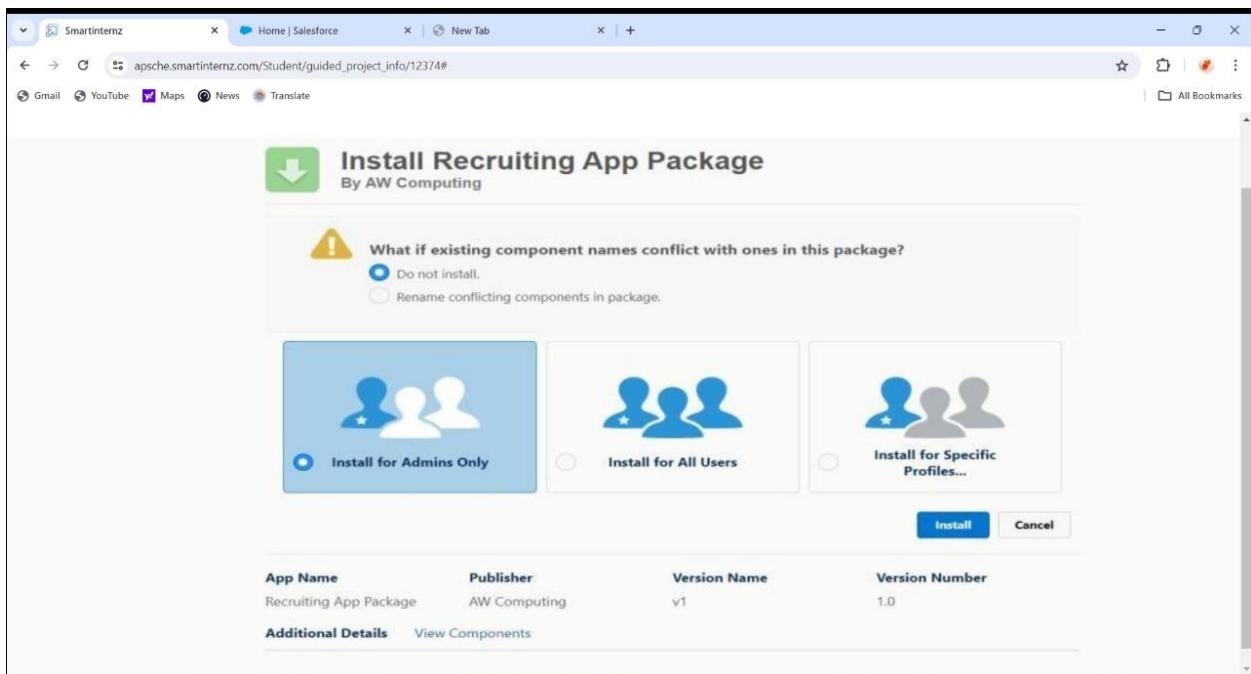
\*Customize

## Installing packages

Find the App: Locate the recruiting app you want to install from either Salesforce AppExchange or from a trusted third-party vendor.

Check Compatibility: Ensure that the app is compatible with your Salesforce edition and version. Review any system requirements or prerequisites specified by the app provider.

Install the Package:

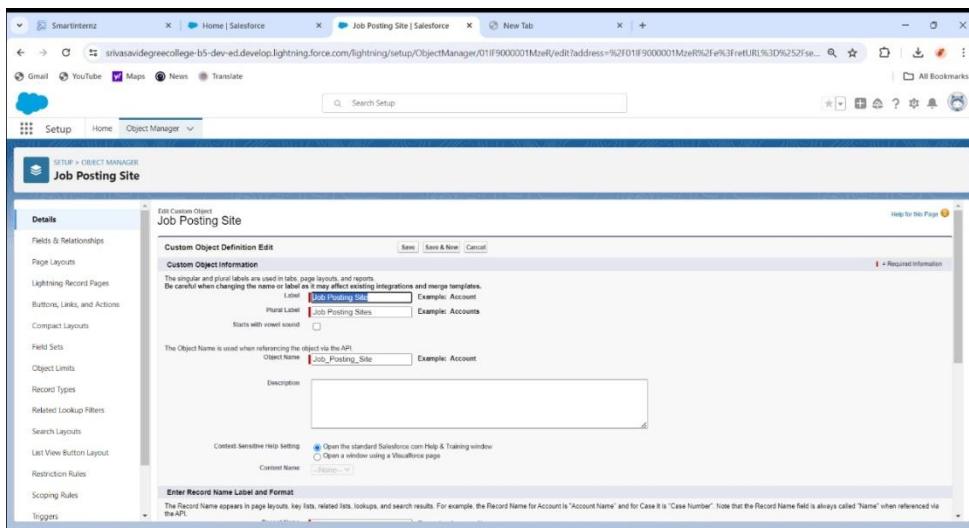


Navigate to the Salesforce Setup page by clicking on the gear icon (⚙) in the top-right corner and selecting "Setup."

# OBJECT

## Create A Custom Object For Job Posting Sites

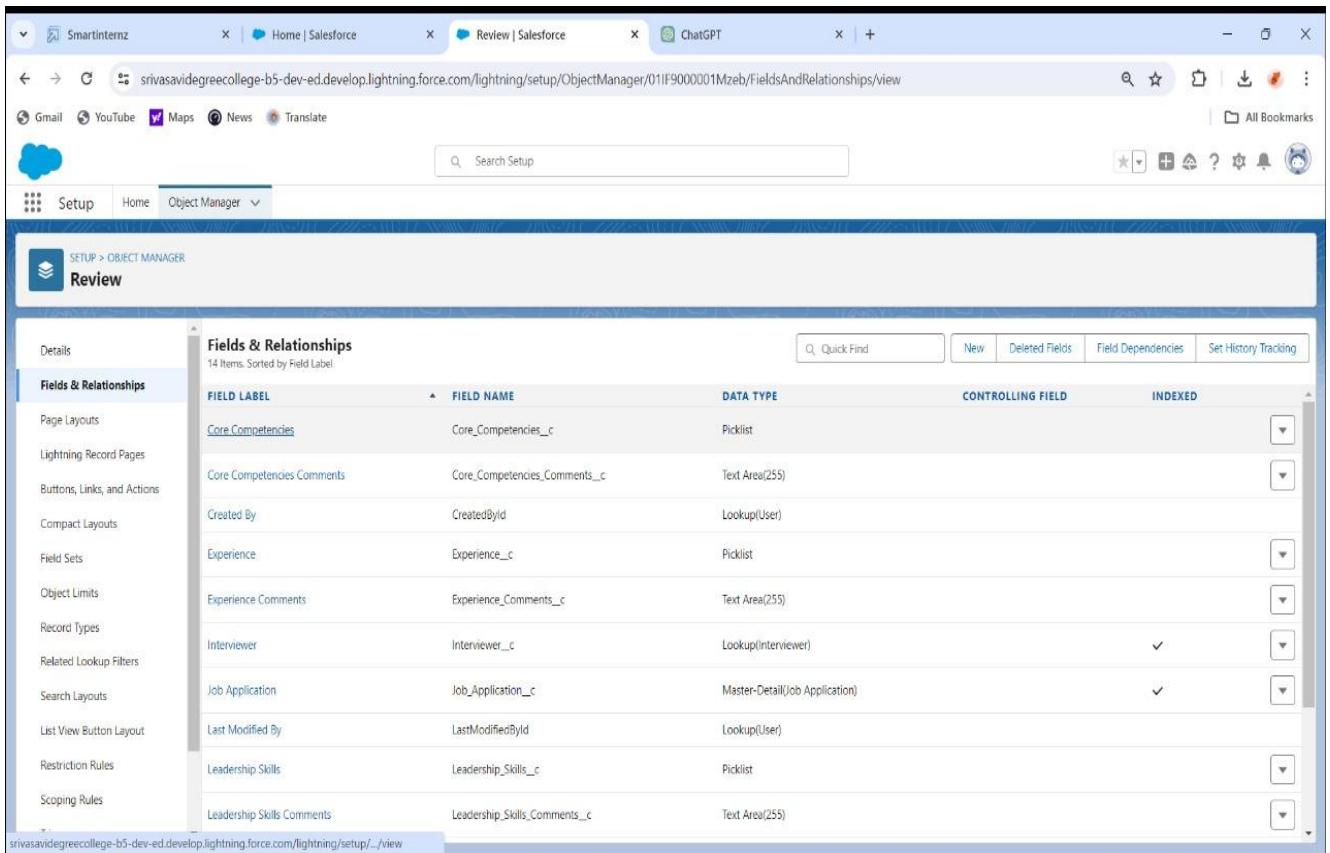
A custom object for job posting sites in Salesforce can streamline recruitment processes by centralizing job postings, tracking candidate applications, and integrating with other HR systems. It enables organizations to manage job listings, monitor application statuses, and analyze recruitment metrics efficiently.



Fields & Relationships					
6 items, Sorted by Field Label					
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED	
Created By	CreatedById	Lookup(User)			
Description	Description__c	Text Area(255)			
Job Posting Site URL	Job_Posting_Site_URL__c	URL(255)			
Last Modified By	LastModifiedById	Lookup(User)			
Owner	OwnerId	Lookup(User/Group)		✓	
Site Name	Name	Text(80)		✓	
Status	Status__c	Picklist			
Technical Site	Technical_Site__c	Checkbox			

## Create A Custom Object For Reviews

A custom object for reviews in Salesforce facilitates the collection, organization, and analysis of feedback from customers, employees, or products. It allows for tracking key metrics, sentiment analysis, and generating insights to improve products/services and enhance customer satisfaction. Integration with workflows and reporting tools enables efficient review management and actionable insights for decision-making.

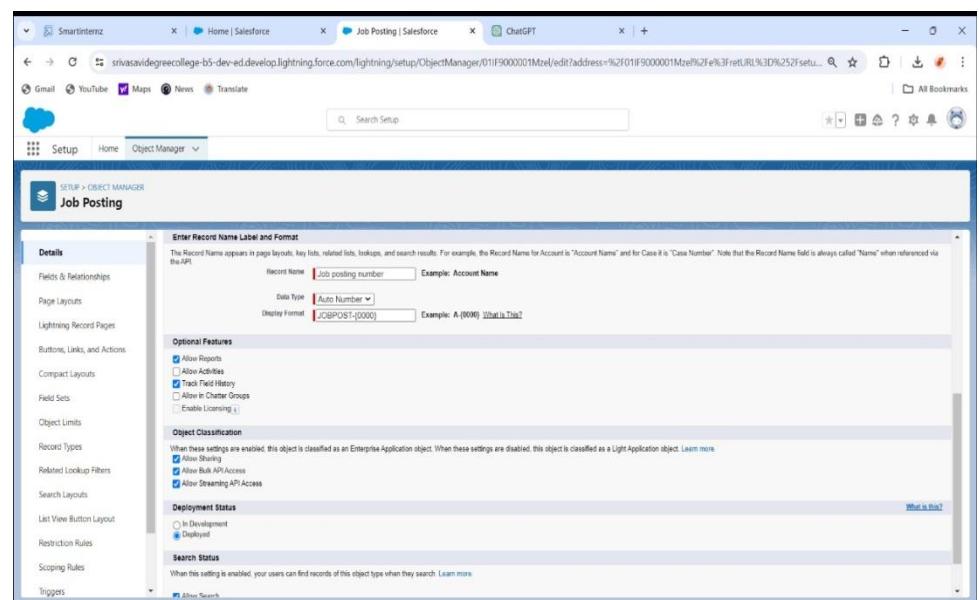
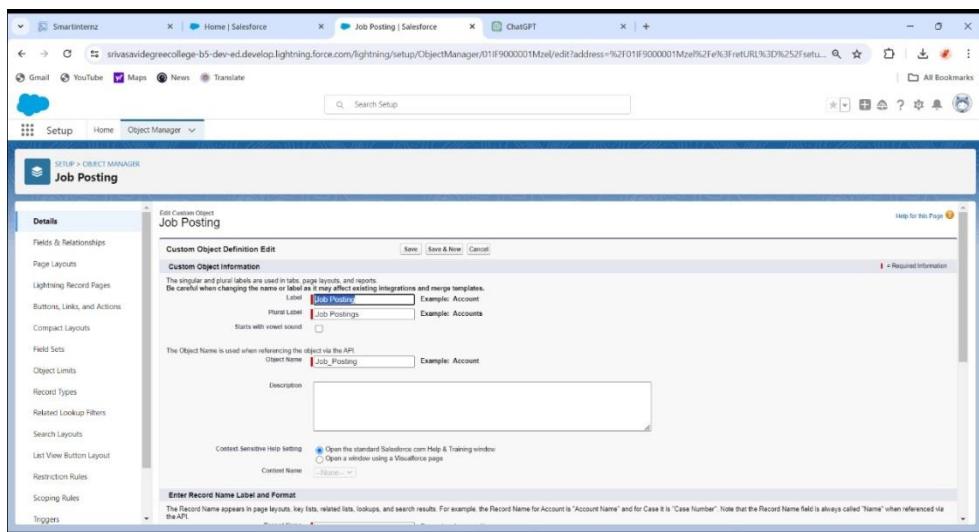


The screenshot shows the Salesforce Object Manager interface for the 'Review' custom object. The 'Fields & Relationships' tab is selected. The table lists 14 fields, each with its label, name, data type, and other properties. The fields are:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Core Competencies	Core_Competencies_c	Picklist		
Core Competencies Comments	Core_Competencies_Comments_c	Text Area(255)		
Created By	CreatedById	Lookup(User)		
Experience	Experience_c	Picklist		
Experience Comments	Experience_Comments_c	Text Area(255)		
Interviewer	Interviewer_c	Lookup(Interviewer)		✓
Job Application	Job_Application_c	Master-Detail(Job Application)		✓
Last Modified By	LastModifiedById	Lookup(User)		
Leadership Skills	Leadership_Skills_c	Picklist		
Leadership Skills Comments	Leadership_Skills_Comments_c	Text Area(255)		

# Junction Object

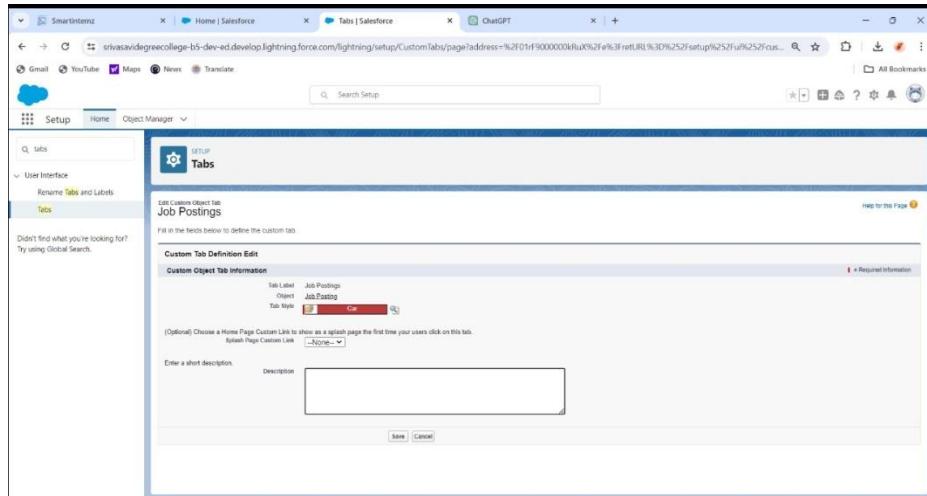
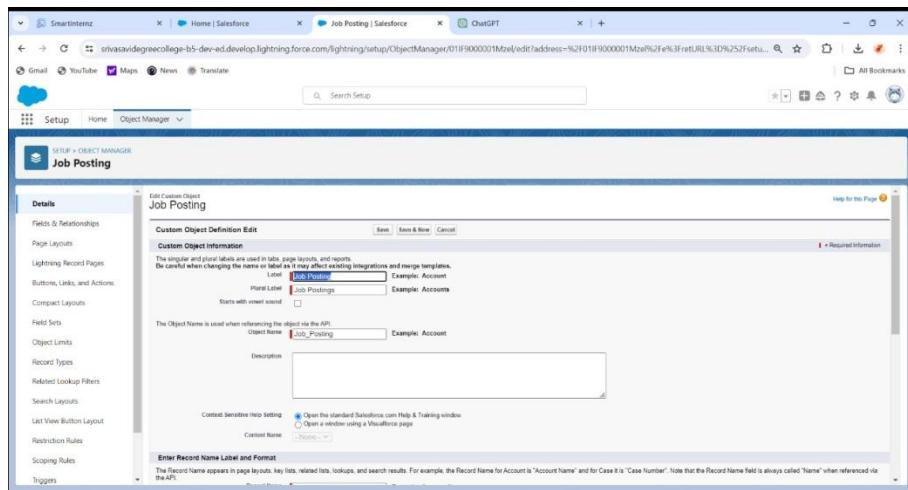
A junction object in Salesforce establishes a many-to-many relationship between two objects by serving as a bridge. It contains two master-detail relationships, linking records of the related objects. Junction objects are commonly used in scenarios where multiple records of one object need to be associated with multiple records of another object, such as in a many-to-many relationship between Accounts and Contacts.



# TABS

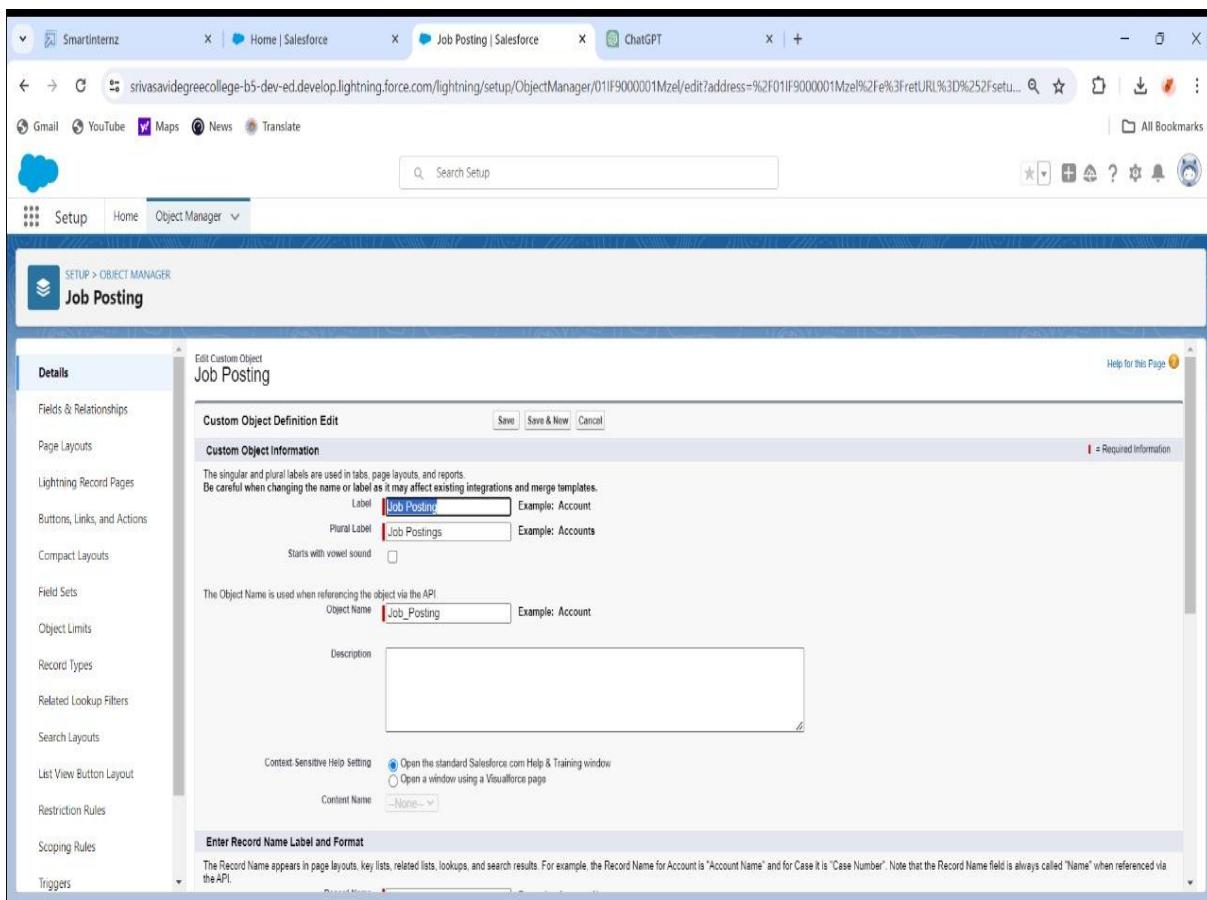
## Creation Of Job Posting Sites Tab

Creating a "Job Posting Sites" tab in Salesforce allows users to easily access and manage information related to various job posting platforms. This tab centralizes data on job posting sites, including site names, URLs, contact information, and associated job postings, streamlining recruitment processes. Integration with custom objects, workflows, and reports enhances efficiency in managing job posting partnerships and tracking recruitment efforts.



## Creation of Reviews Tab

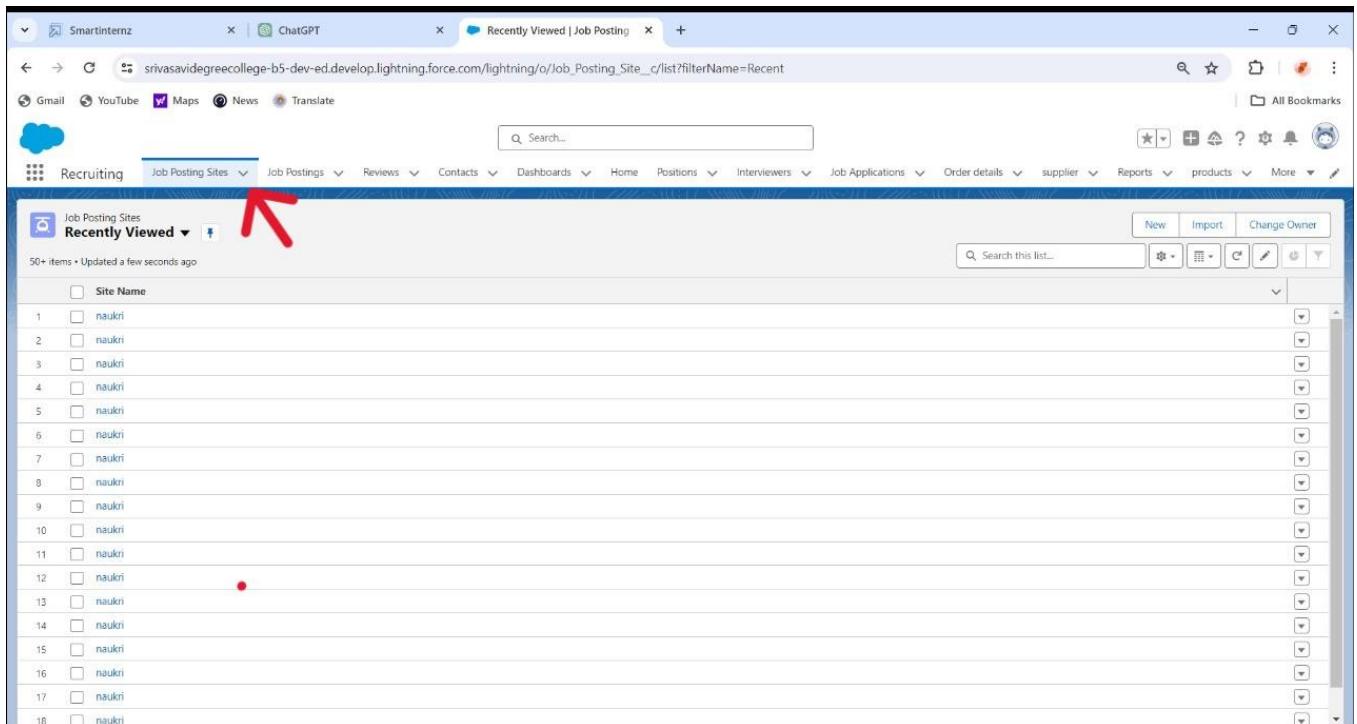
Creating a "Reviews" tab in Salesforce provides a dedicated space for managing feedback and evaluations from customers, employees, or products. This tab enables easy access to review records, including ratings, comments, and sentiment analysis, facilitating data-driven decision-making and actionable insights. Integration with workflows and reports enhances review management and analysis, improving overall performance and customer satisfaction.



# Lightning App

## Adding Job Posting Sites Tab

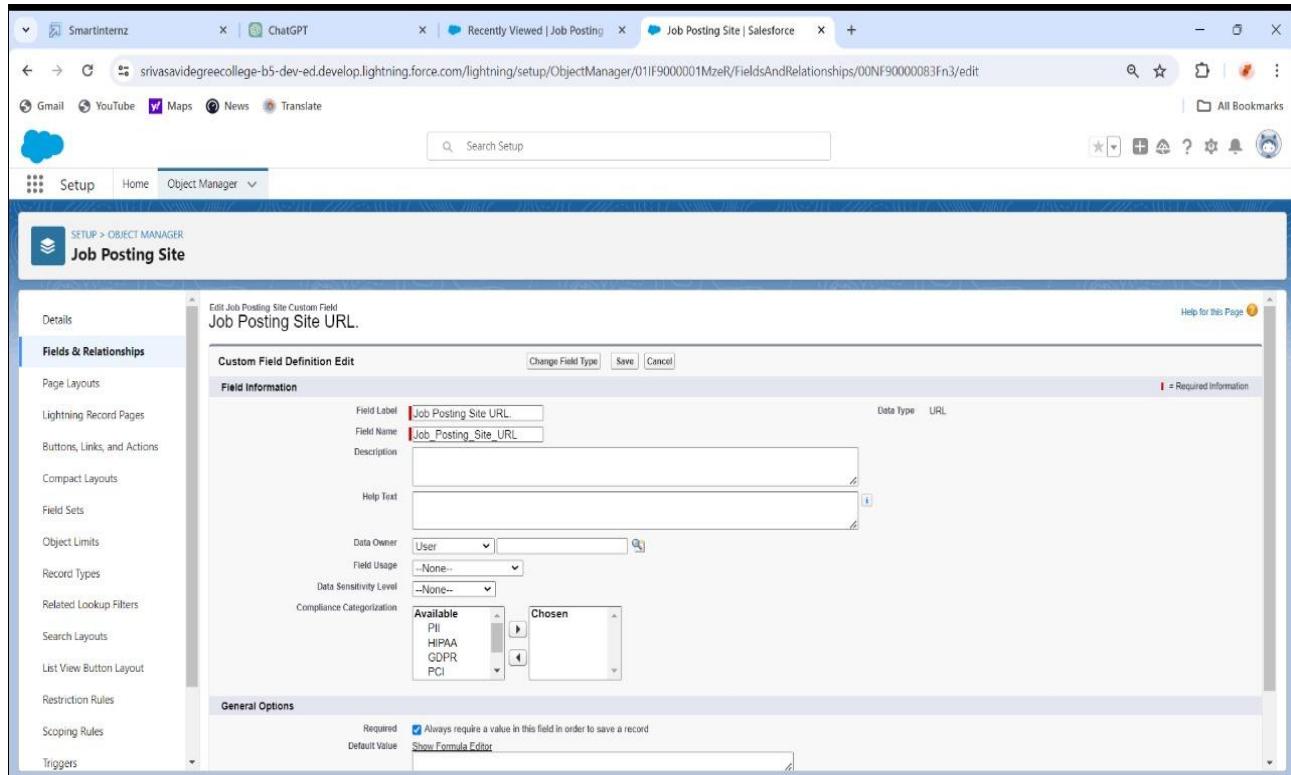
Adding a "Job Posting Sites" tab in Salesforce streamlines recruitment efforts by centralizing information about various job posting platforms. This tab enables easy access to site details such as names, URLs, contacts, and associated job postings, facilitating efficient partnership management. Integration with custom objects and reports enhances visibility and tracking of job posting site data, optimizing recruitment strategies.



# Fields and relationships

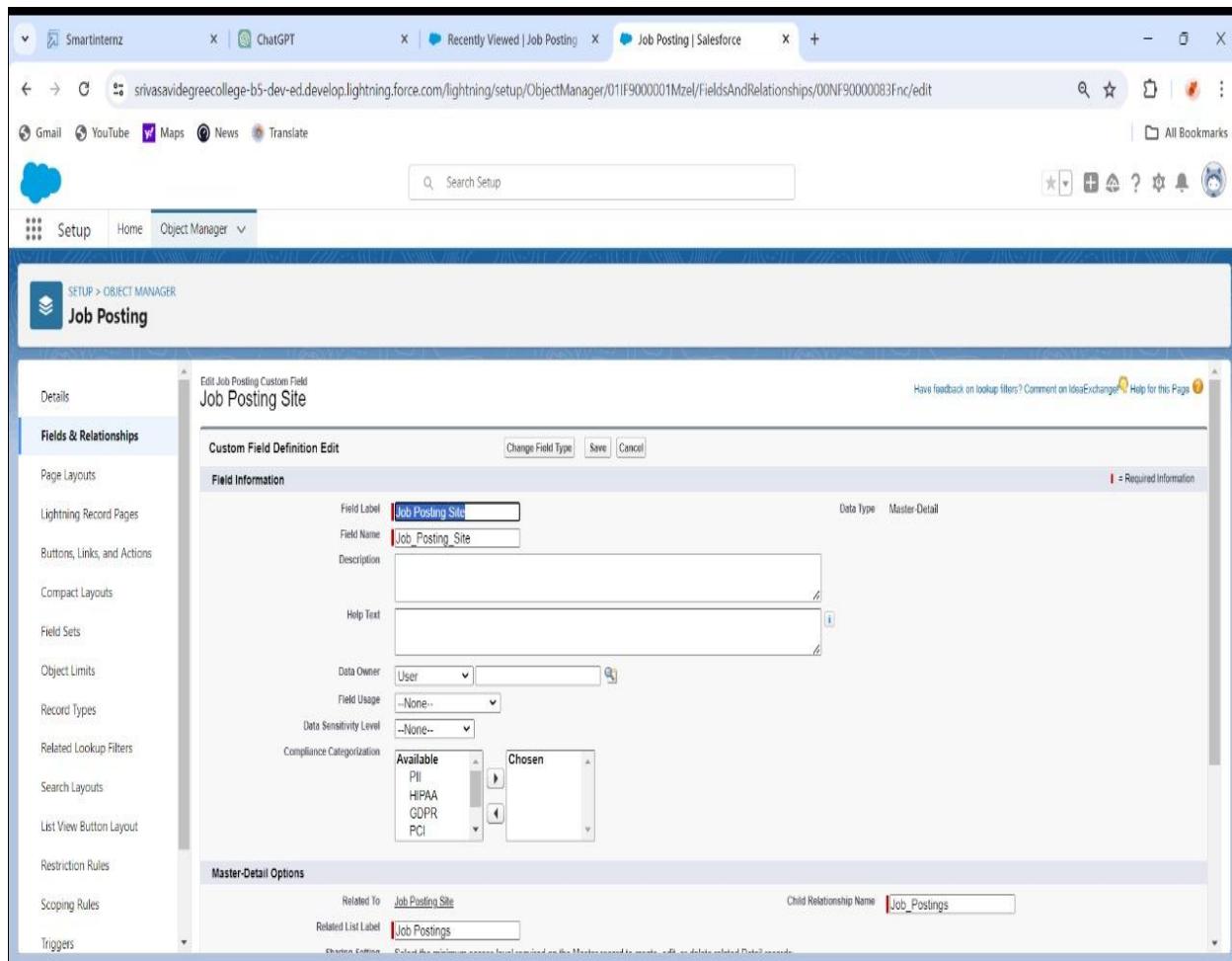
## Create New Field For Job Posting Site

Creating a "Job Posting Site" field in Salesforce allows users to associate job postings with specific posting platforms, enhancing organization and categorization. This custom field captures crucial information such as site name, URL, or identifier, facilitating streamlined management and analysis of recruitment efforts. Integration with workflows and reports enables efficient tracking of job postings across different platforms, optimizing recruitment strategies.



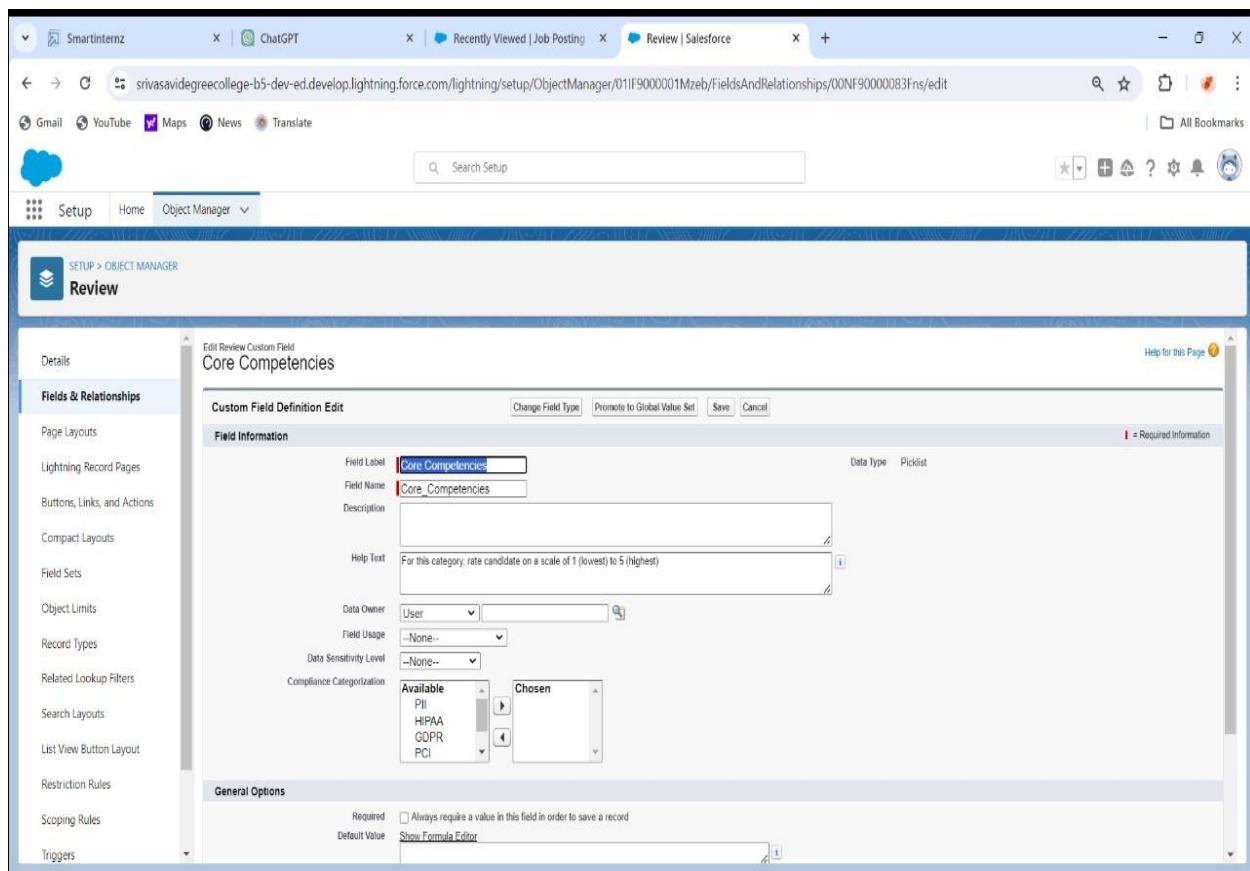
## Create Relationships For Job Posting

Establishing relationships for job postings in Salesforce enables effective linking between job postings and related entities, such as job applicants or job posting sites. These relationships streamline recruitment processes, allowing for easy tracking of candidate applications, site partnerships, and associated data. Integration with automation tools and reporting functionalities enhances visibility and analysis, optimizing recruitment strategies and decision-making.



## Create New Field For Reviews

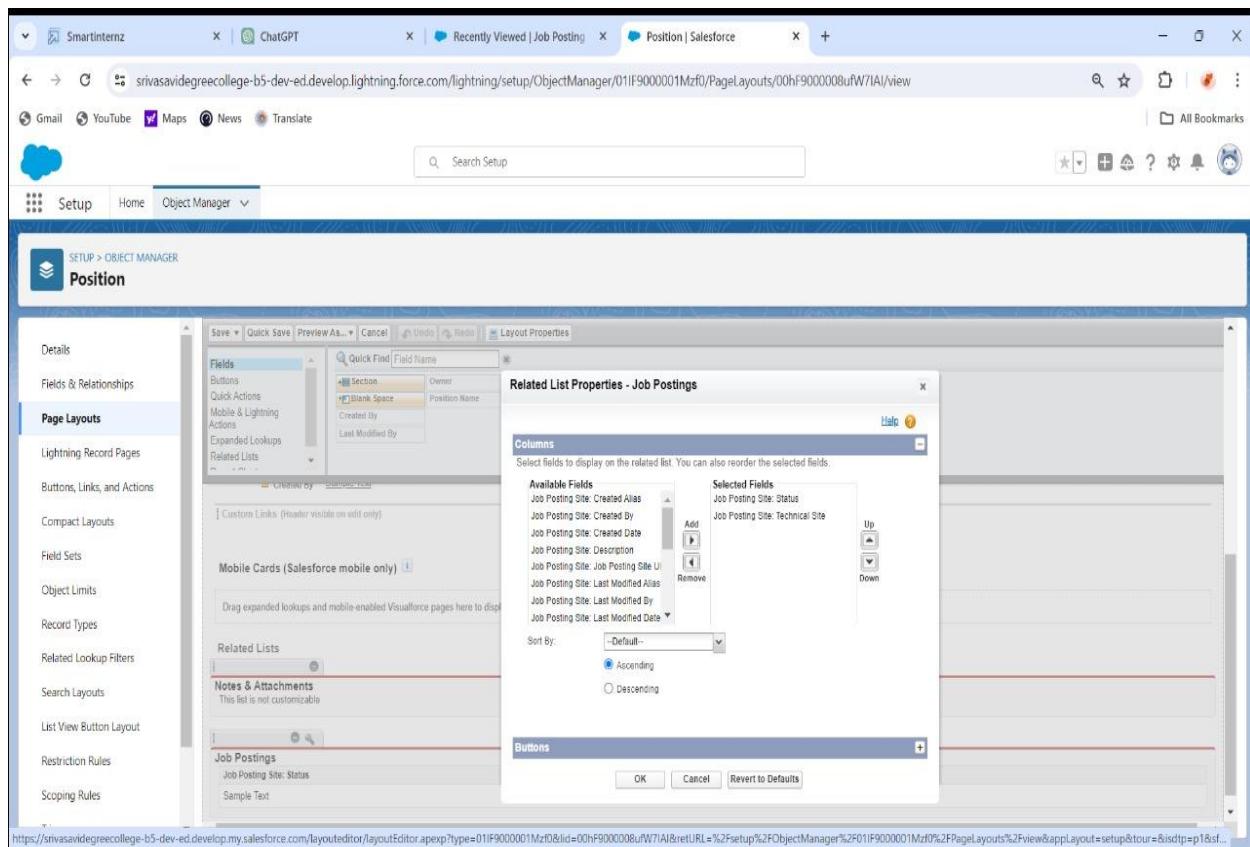
Introducing a "Review" field in Salesforce allows for systematic collection and analysis of feedback, enhancing customer insights and satisfaction. This custom field captures important details such as review ratings, comments, and timestamps, facilitating informed decision-making and targeted improvements. Integration with workflows and reports empowers efficient review management and performance tracking, driving continuous enhancement of products or services.



# Page Layout

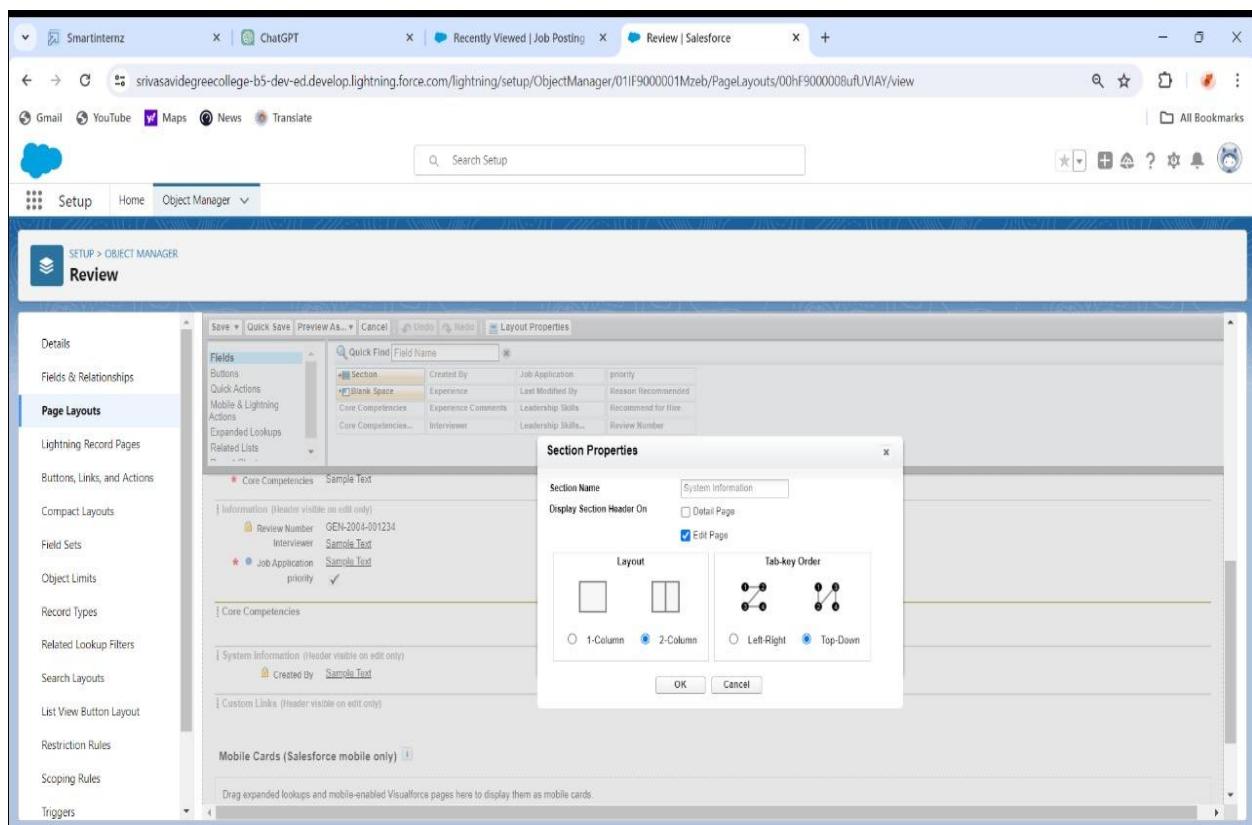
## Modifying The Page Layouts

Modifying page layouts in Salesforce enables customization of user interfaces to align with specific business needs and workflows. This includes rearranging, adding, or removing fields, related lists, and components to optimize user experience and productivity. Integration with profiles and record types ensures tailored experiences for different user roles and scenarios, enhancing efficiency and usability.



## Create A Page Layout For Review Object

Creating a page layout for the Review object in Salesforce facilitates efficient data entry and review management. This layout organizes fields and related lists to streamline the review process, ensuring essential information such as ratings, comments, and reviewer details are easily accessible. Integration with record types enables customized layouts for different review types or stages, optimizing user experience and data accuracy.



# Validation Rules

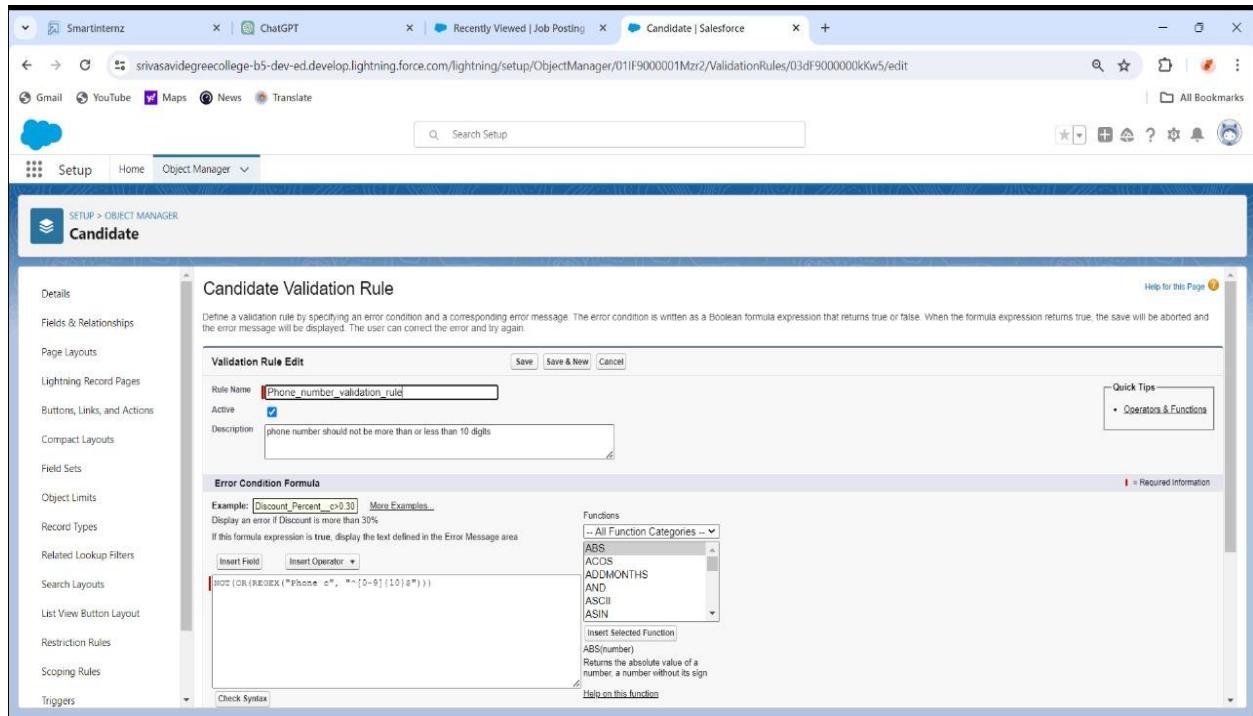
## Creating A Validation Rule

Specify the field or fields to be validated and establish the conditions for acceptance or rejection.

Test the rule thoroughly with representative data samples to ensure it effectively identifies errors or inconsistencies.

Document the validation rule comprehensively, including its purpose, conditions, and potential impacts, for future reference and maintenance.

Document the validation rule comprehensively, including its purpose, conditions, and potential impacts, for future reference and maintenance.



## Create A Validation Rule For Technical Site (Checkbox Is Equal To True)

Rule name: Technical site checkbox true.

Active: checked

Description: Technical site checkbox should be check.

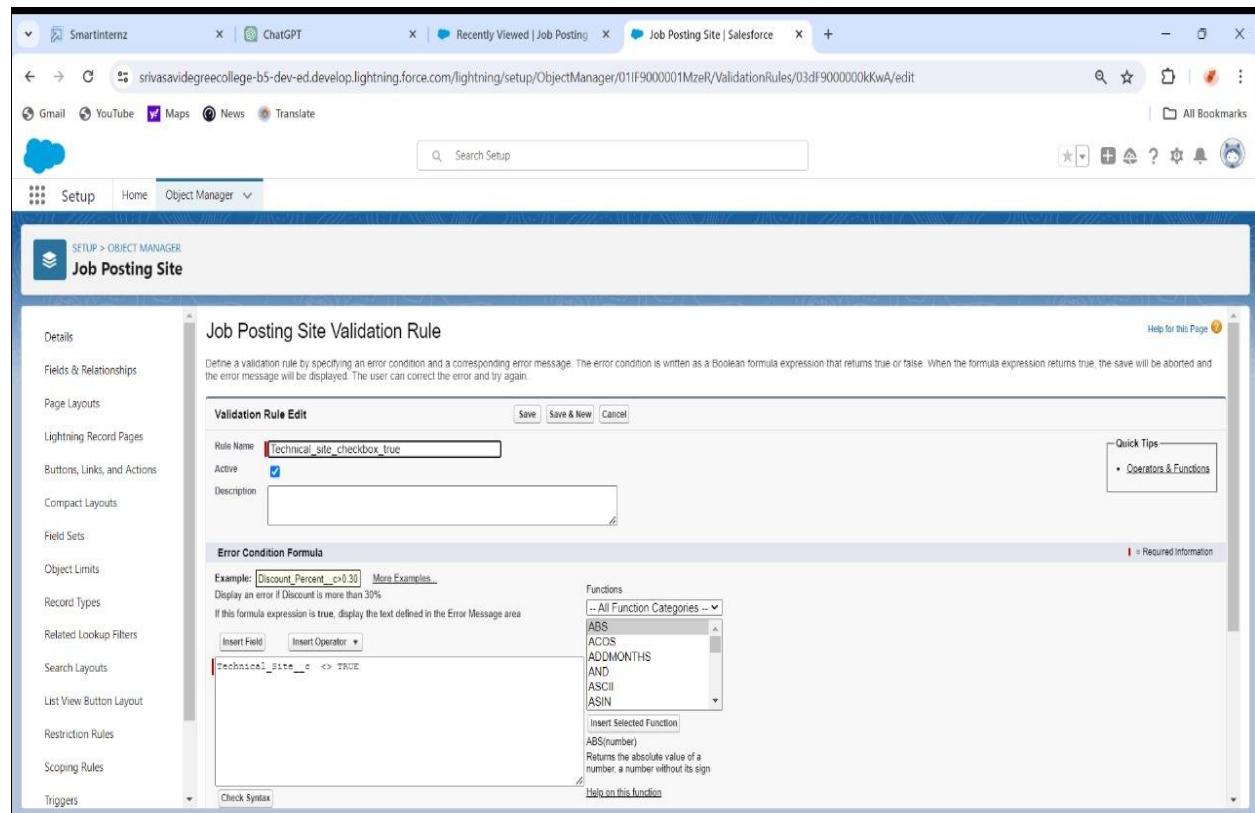
Under Error Condition Formula: write the condition using insert field, insert operator, insert function Technical\_Site c != TRUE

Using check syntax: check if the formula you entered is valid or not.

Error Message: Please select check box of technical site.

Error location: select field (Technical site).

Save.



SETUP > OBJECT MANAGER

Job Posting Site

Job Posting Site Validation Rule

Validation Rule Edit

Rule Name: Technical\_site\_checkbox\_true

Active:

Description: (empty)

Error Condition Formula

Example: Discount\_Percent\_c>30 [More Examples...](#)

If this formula expression is true, display the text defined in the Error Message area

Functions

- ABS
- ACOS
- ADDMONTHS
- AND
- ASCII
- ASIN

Insert Field

Technical\_Site\_c <> TRUE

Check Syntax

# PROFILE

## Creation On Profile

\*To create a validation rule in Salesforce for a profile, follow these steps:

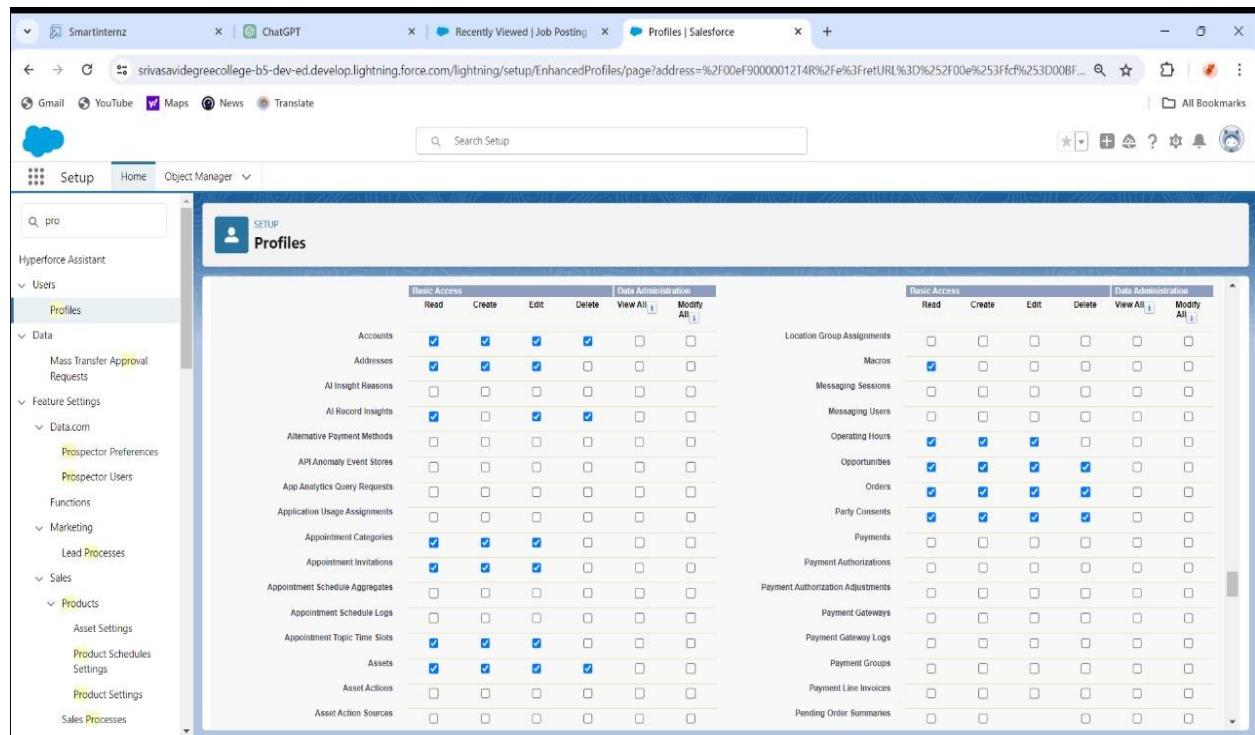
\*Access Setup: Log in to your Salesforce account and navigate to the Setup page.

\*Open Object Manager: In the Quick Find box, type "Object Manager" and select it.

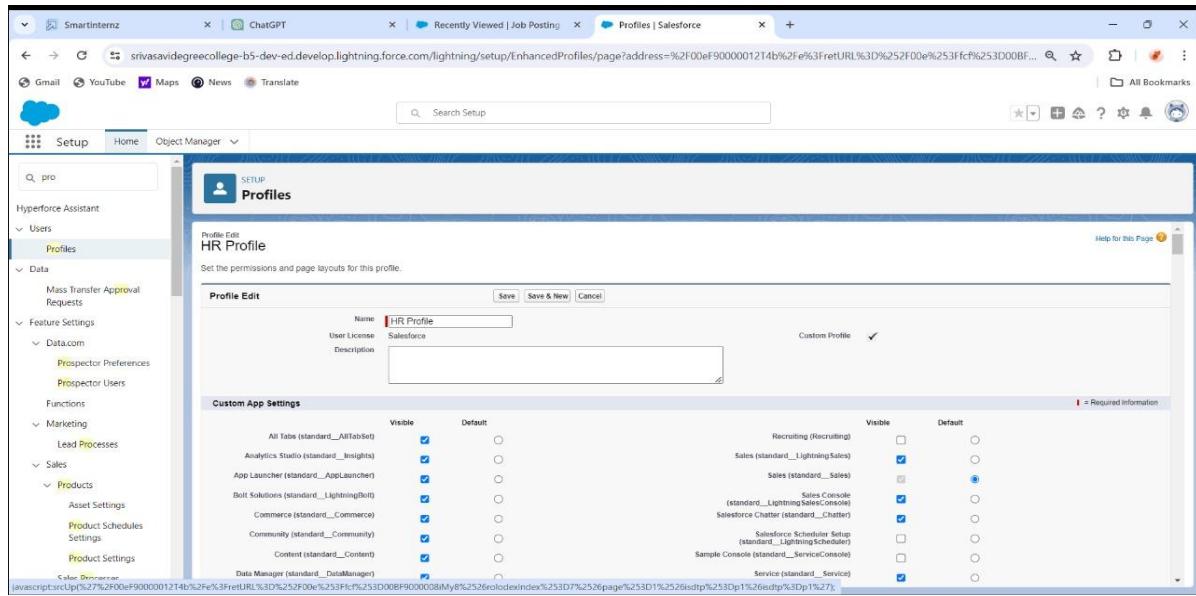
\*Choose Object: Select the object (e.g., the object where the checkbox field resides) for which you want to create the validation rule.

\*Validation Rules: In the left pane, under "Validation Rules," click on "New Validation Rule."

\*Define Criteria: Enter a name and description for the validation rule. In the formula editor, write the validation rule formula.



## Create A Profile With The Profile Name As "HR Profile"



To create a profile in Salesforce named "HR Profile," follow these steps:

\*Access Setup: Log in to your Salesforce account and navigate to the Setup page.

\*Profiles: In the Quick Find box, type "Profiles" and select it.

\*New Profile: Click on the "New Profile" button.

\*file Settings: Enter "HR Profile" as the profile name.

\*Permissions: Set the appropriate permissions for the profile based on the requirements of your HR users. This may include object permissions, field permissions, and other settings related to data access and functionality.

\*Save: Once you have configured the profile settings, click "Save" to create the "HR Profile."

\*Assign Users: After creating the profile, assign users to it by editing their user records and selecting the "HR Profile" from the Profile dropdown menu.

\*Test: Test the profile by logging in as a user assigned to the "HR Profile" to ensure that the permissions and settings are configured correctly.

# USER

## Creating A User

To create a user in Salesforce, follow these steps:

Access Setup: Log in to your Salesforce account and navigate to the Setup page.

Users: In the Quick Find box, type "Users" and select it.

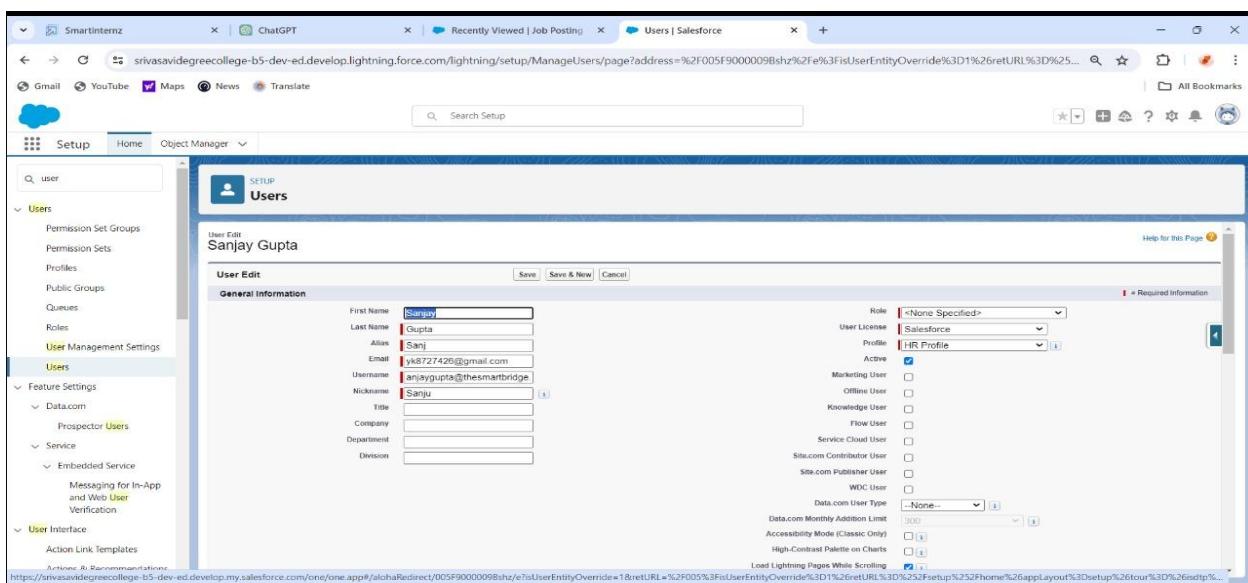
New User: Click on the "New User" button.

User Information: Fill in the required user information, including First Name, Last Name, Email, Username, Role, Profile, etc.

Role Assignment: Assign a role to the user if applicable. Roles define the level of access a user has to records in Salesforce.

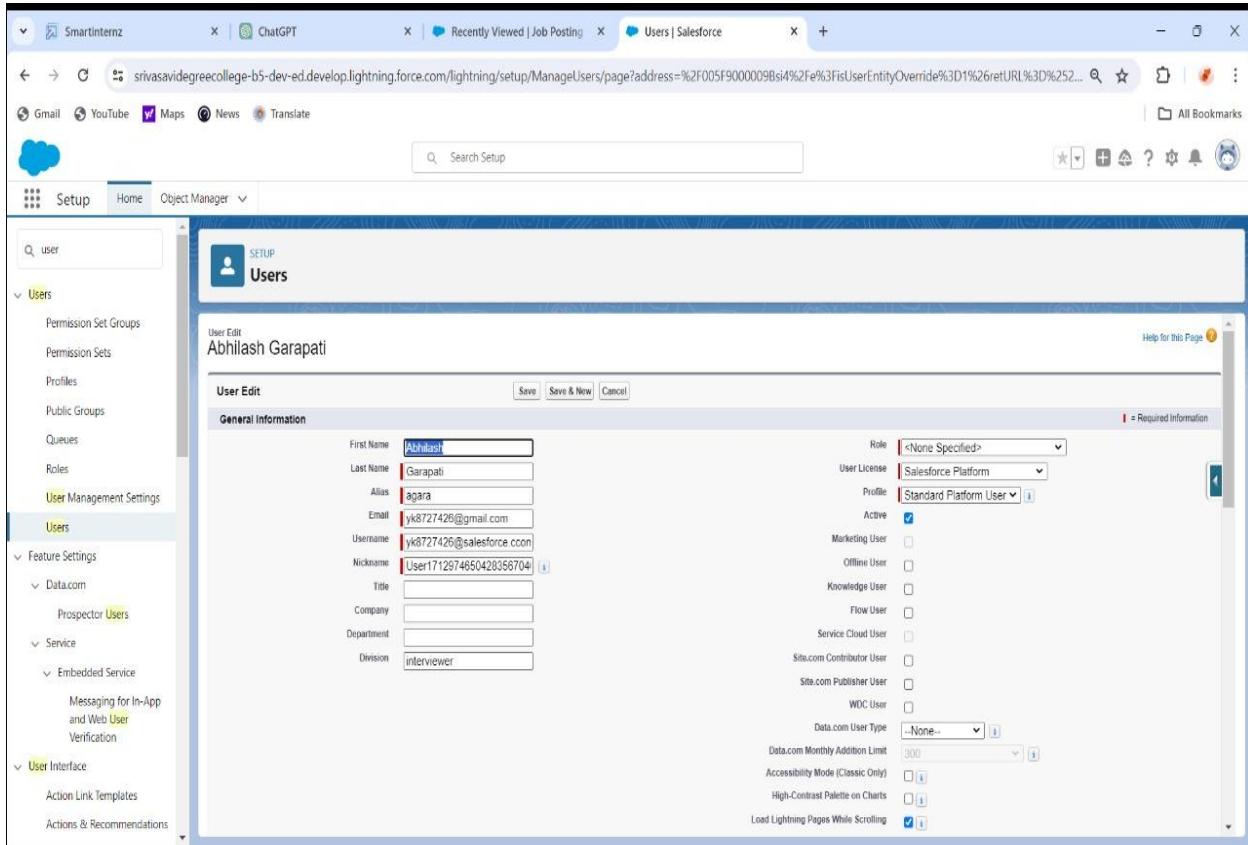
Profile Assignment: Assign a profile to the user. Profiles determine the permissions and settings available to the user.

Save: Once you have entered all the necessary information, click "Save" to create the user.



## Create Another User

To Create a user with a username as “Abhilash Garapati”, and assign him the interviewer profile.



\*Access Setup: Log in to your Salesforce account and navigate to the Setup page.

\*Users: In the Quick Find box, type "Users" and select it.

\*New User: Click on the "New User" button.

\*User Information:

First Name: Abhilash

Last Name: Garapati

\*Then Save It.

# Permission set

## Creating A Permission Set

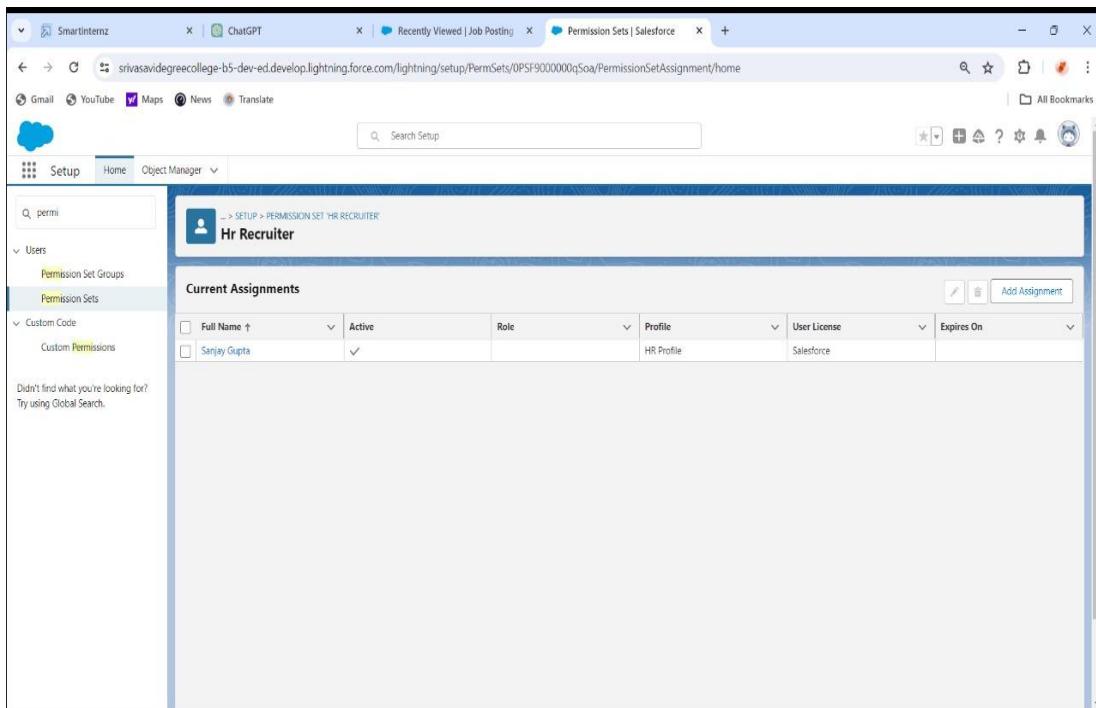
New Permission Set: Click on the "New" button to create a new permission set.

Permissions: Click on the "Permissions" tab to configure the permissions for the permission set

License: Select the appropriate license if required. This determines the features and functionalities available to users assigned this permission set.

Save: Once you have configured the permissions, click "Save" to create the permission set.

Assign Users: After creating the permission set, assign users to it by clicking on the "Manage Assignments" button and then clicking "Add Assignments". You can assign users individually or by using a permission set group.



# User adoption

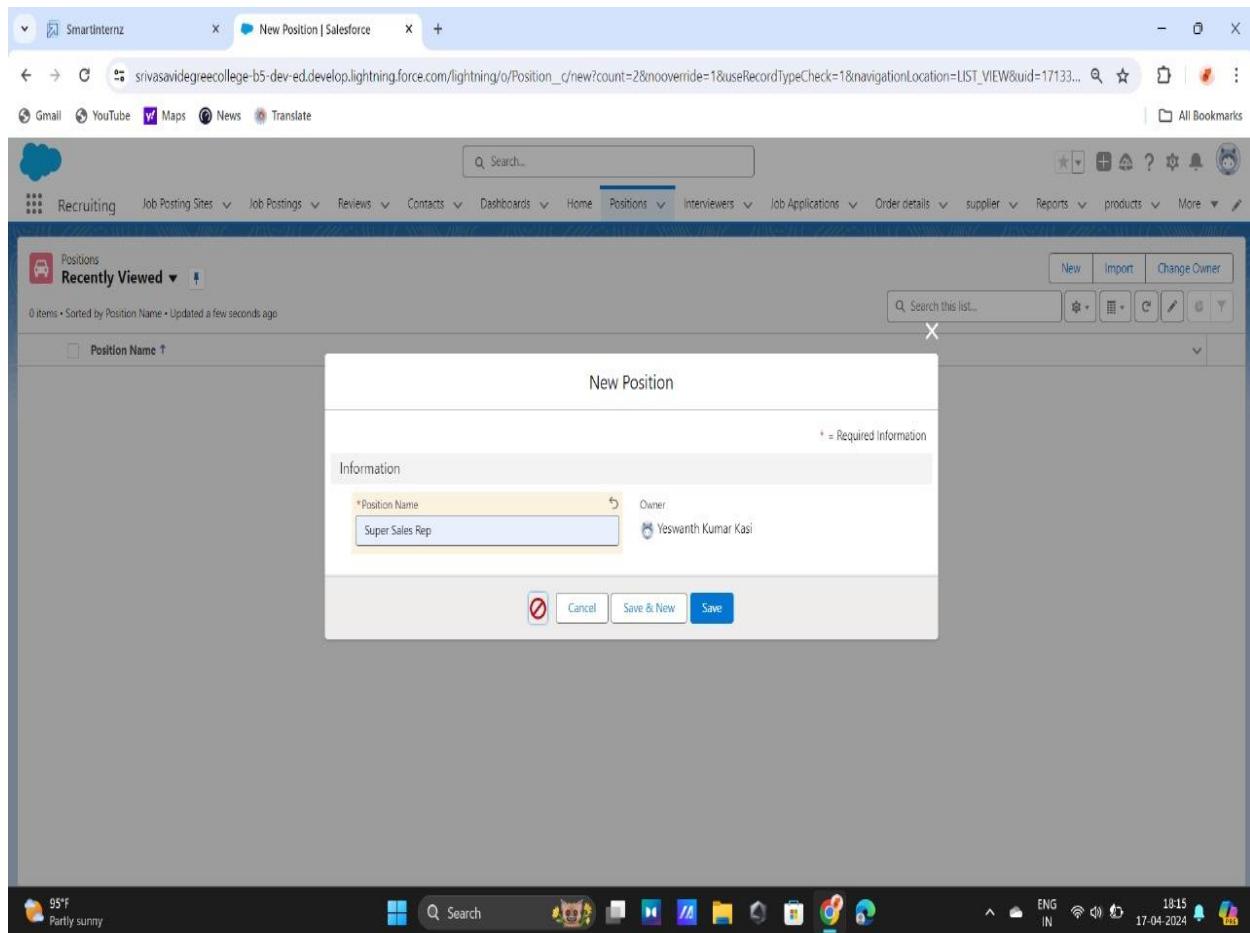
## Create a record (positions)

\*Click on App Launcher on left side of screen.

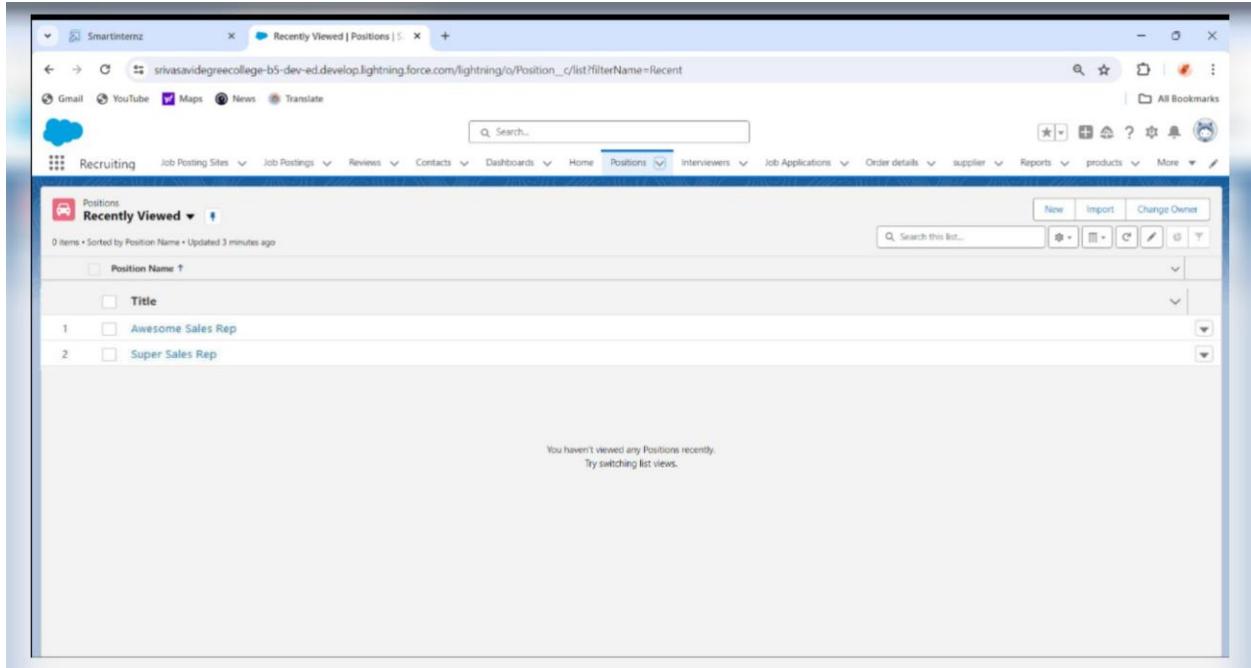
\*Search Recruiting & click on it.

\*Click on Positions Tab.

\*Click new and fill details & Save



## View a record (positions)



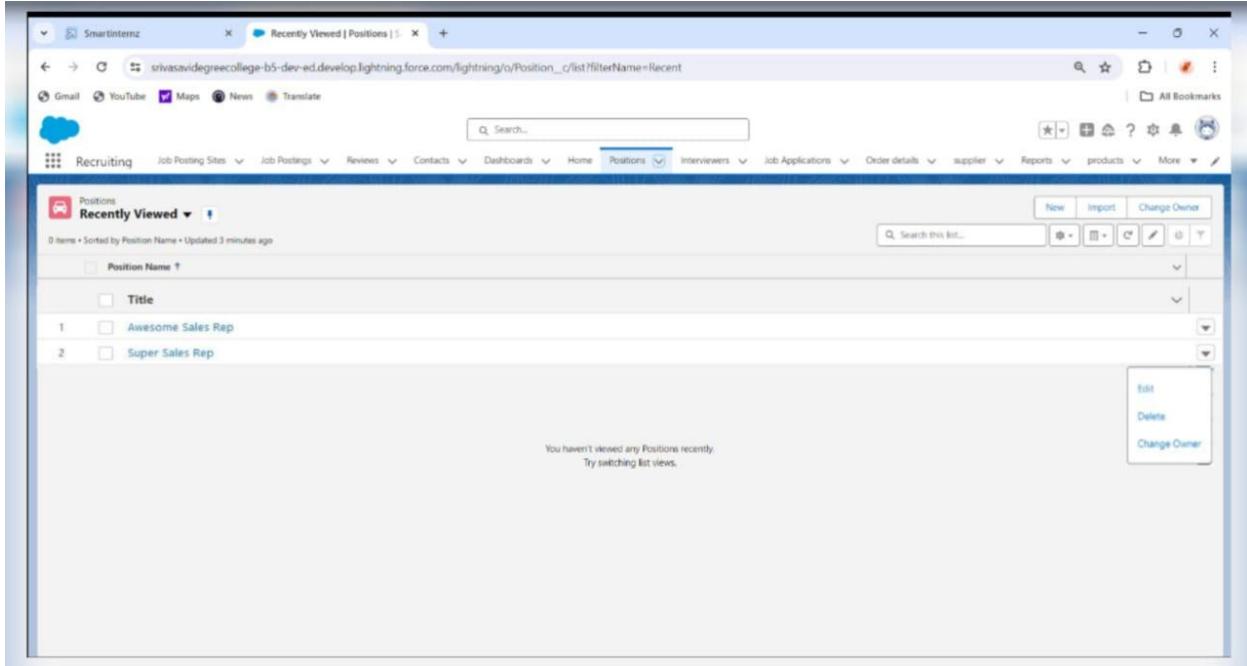
\*Click on App Launcher on left side of screen.

\*Search Recruiting & click on it.

\*Click on Positions Tab.

\*Click on any record name. you can see the details of the Positions.

# Delete a Record (positions)



- \*Click on App Launcher on left side of screen.
- \*Search Recruiting & click on it.
- \*Click on Positions Tab.
- \*Click on Arrow at right hand side on that Particular record.
- \*Click delete and delete again.

# REPORTS

## Create a Report

\*Navigate to the Report Builder

\*Log in to your Salesforce account.

\*Go to the "Reports" tab.

\*Choose Report Type:

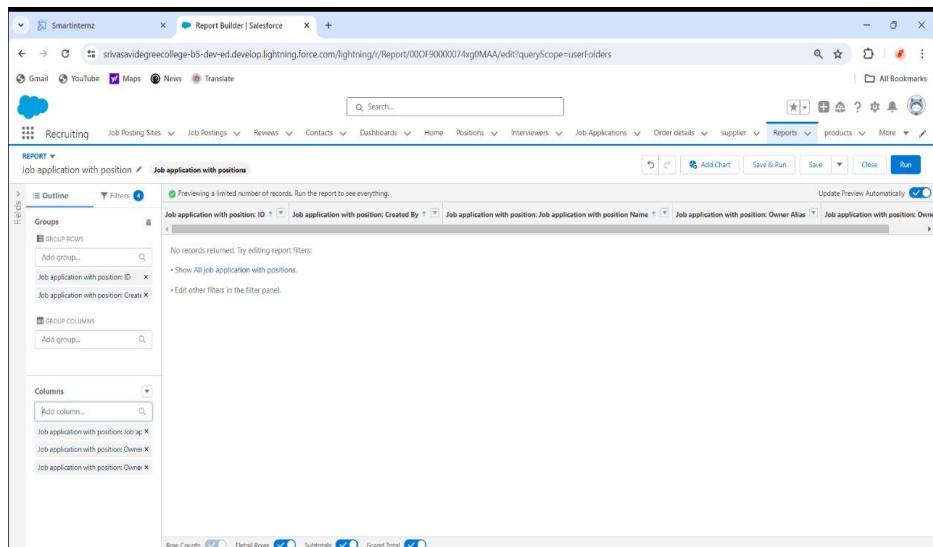
\*Click on the "New Report" button.

\*Select Fields:

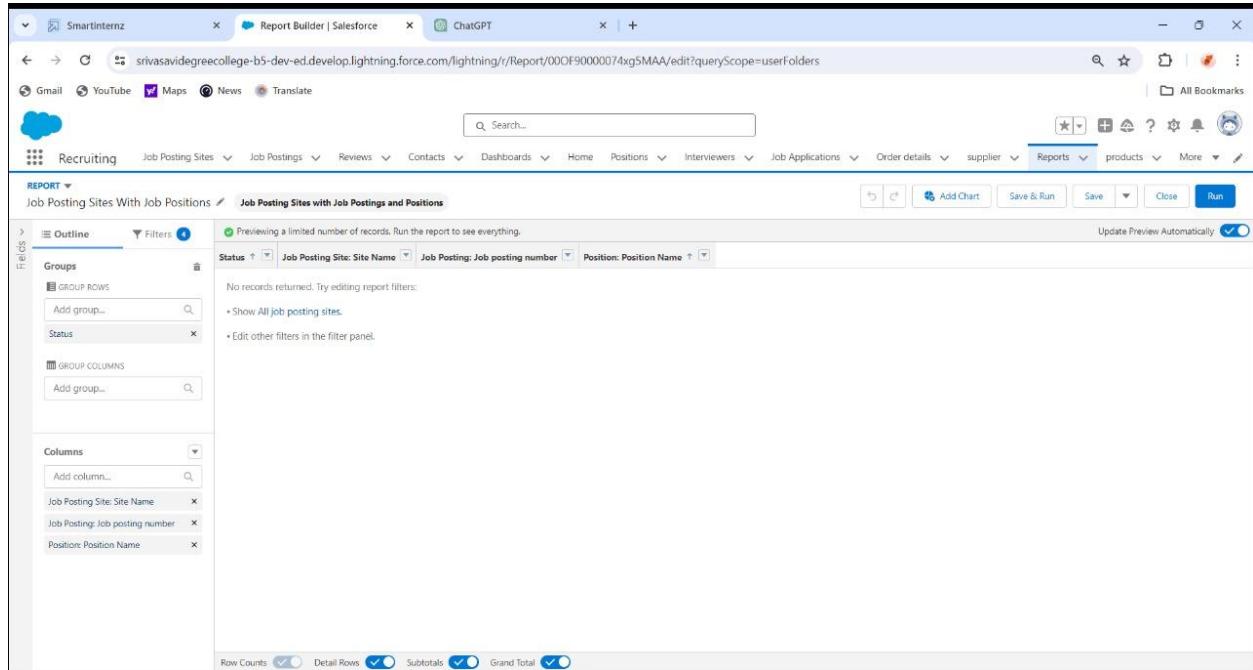
\*Choose the fields you want to include in the report. You can drag and drop fields from the left panel to the report builder.

\*Save and Run the Report:

\*Once you're satisfied with the report setup, click on "Save & Run" to generate the report.



# Create a Report For Job Posting Sites With Job Positions And Positions



\*Log in to your Salesforce account.

\*Go to the "Reports" tab.

\*Choose the fields you want to include in the report.

\*Customize the report layout by adjusting column widths, rearranging fields, and adding headers or footers as needed.

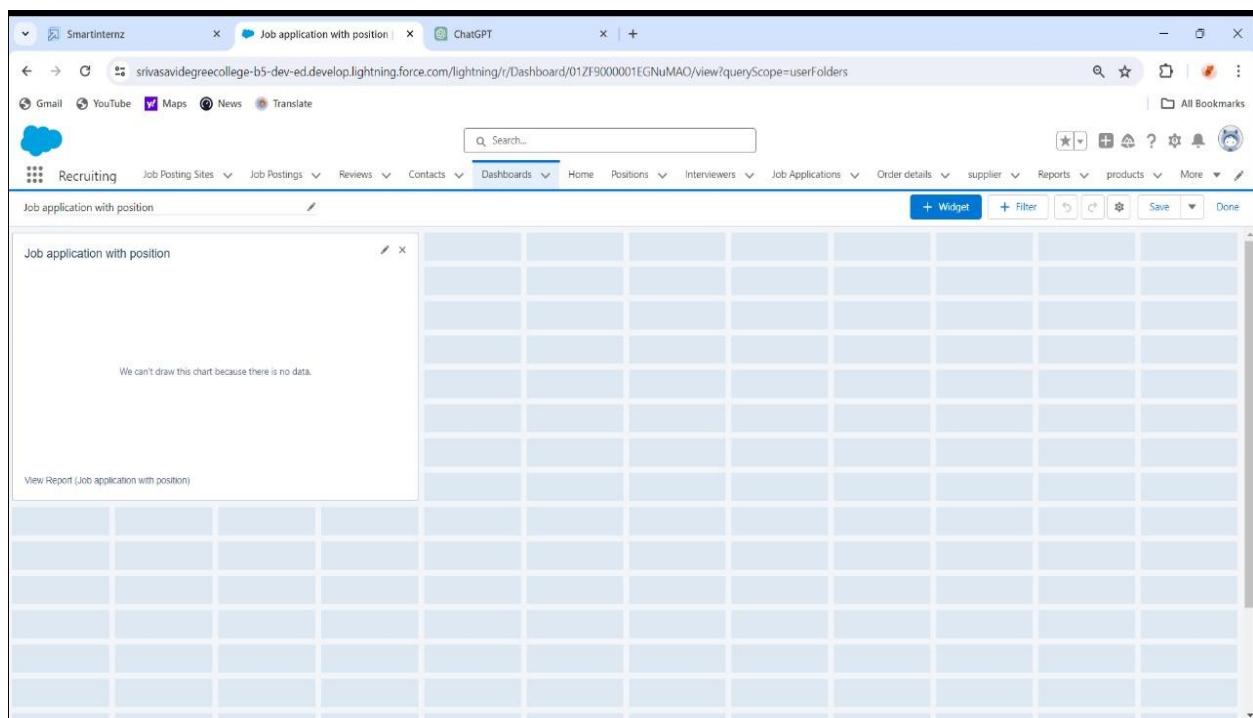
\*Save and Run the Report:

\*Once you're satisfied with the report setup, click on "Save & Run" to generate the report.

# DASHBOARD

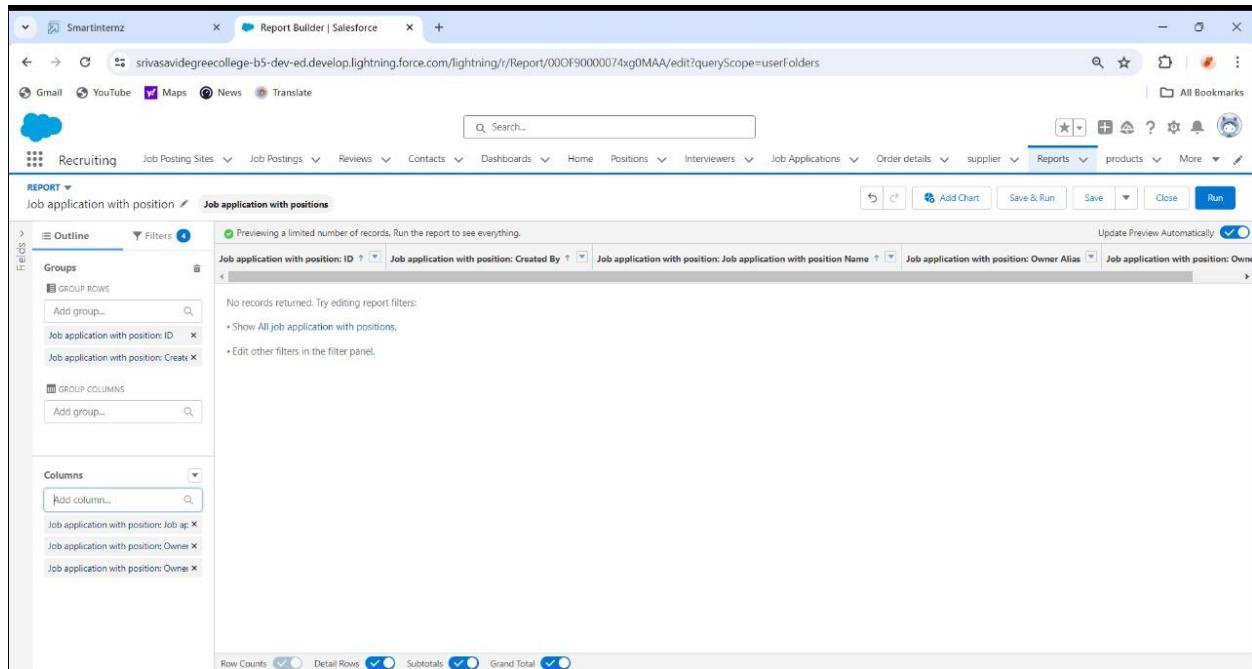
## Create a dashboard

- Click the Dashboards tab.
- Click New Dashboard.
- Name the dashboard Job application with position and click Create.
- Click +Component.
- Select Job application with position and click Select.
- Select the Vertical Bar Chart component and click Add.
- Click Save and then Done.



# View Reports and Dashboards

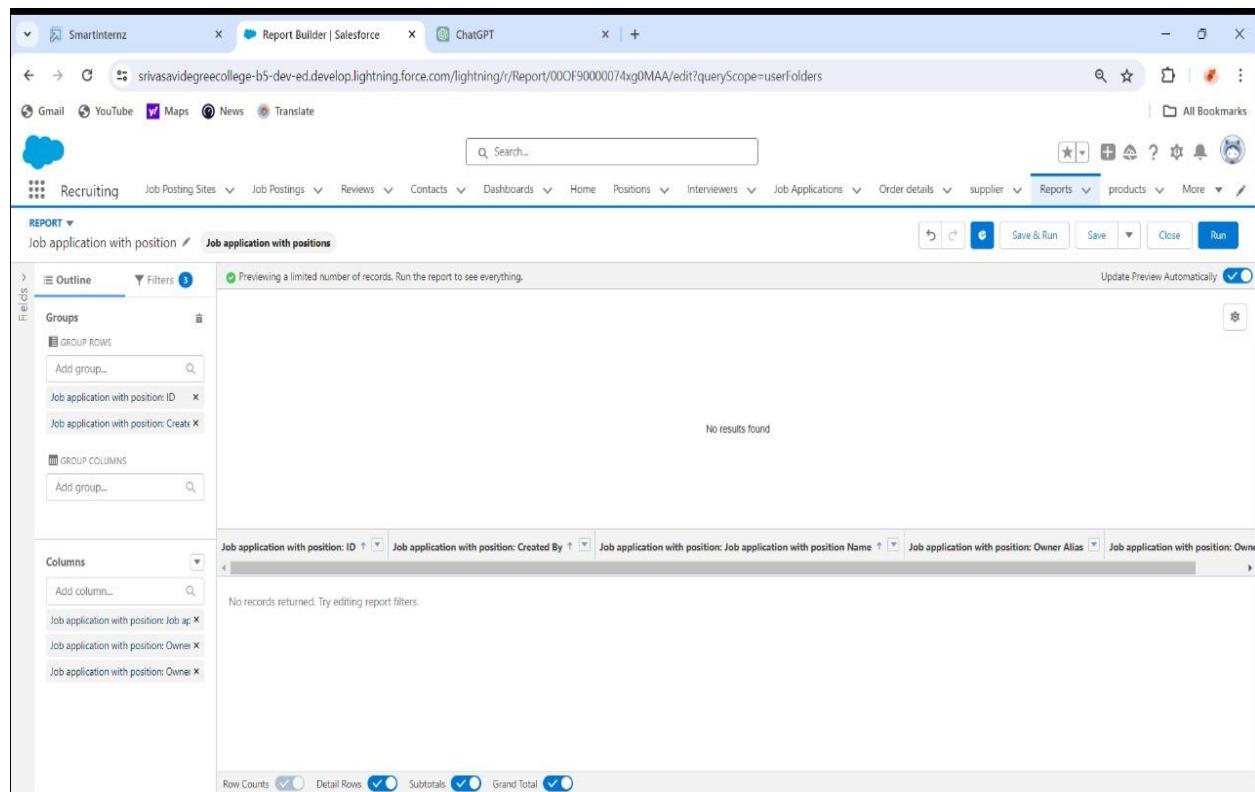
## Report



- Click on App Launcher on left side of screen.
- Search Recruiting & click on it.
- Click on Reports Tab.
- Click on Job application with position & see records

# Dashboards

- Click on App Launcher on left side of screen.
- Search Recruiting & click on it.
- Click on Dashboard Tab.
- Click on Job application with position & see records

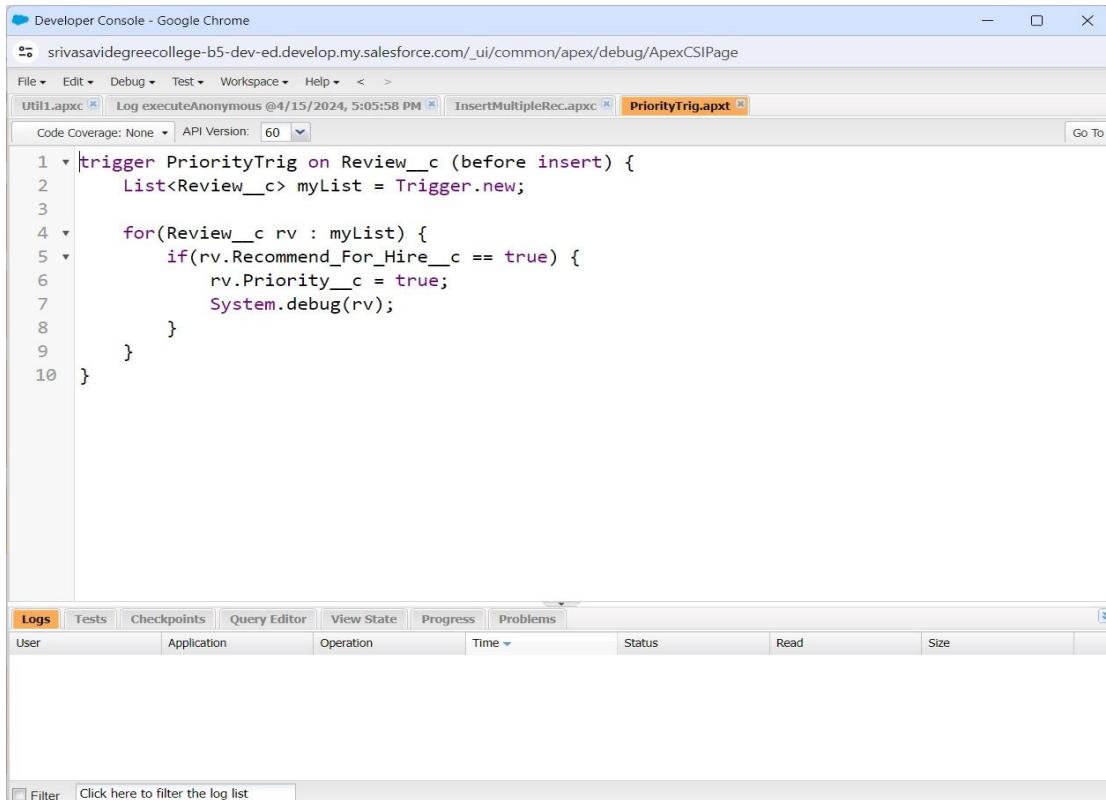


# Apex Triggers

## Creation of the Trigger

### Use Case:

HR is struggling!! Not knowing whom to hire on priority so she contacted the developer. Now the developer made the HR task bit easy with this changes . There is the Review\_\_c Object and there is 2 field priority and Recommended for Hire(CheckBox) so condition is like if suppose if we checked the checkbox than priority should be high



The screenshot shows the Salesforce Developer Console in Google Chrome. The URL is [srivasavidegreecollege-b5-dev-ed.develop.my.salesforce.com/\\_ui/common/apex/debug/ApexCSIPage](https://srivasavidegreecollege-b5-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage). The tabs at the top are Util1.apxc, Log executeAnonymous @4/15/2024, 5:05:58 PM, InsertMultipleRec.apxc, and PriorityTrig.apxt (which is the active tab). The code editor contains the following Apex trigger code:

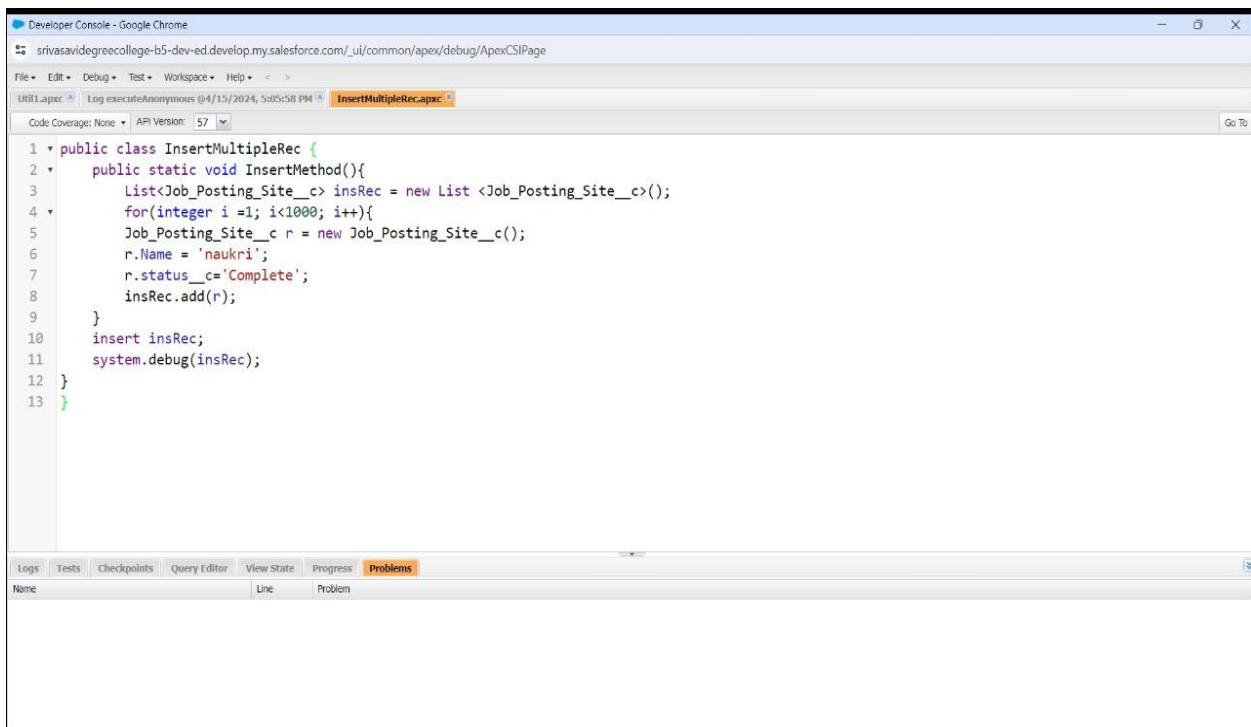
```
1 trigger PriorityTrig on Review__c (before insert) {
2     List<Review__c> myList = Trigger.new;
3
4     for(Review__c rv : myList) {
5         if(rv.Recommend_For_Hire__c == true) {
6             rv.Priority__c = true;
7             System.debug(rv);
8         }
9     }
10 }
```

The developer console interface includes a toolbar with File, Edit, Debug, Test, Workspace, Help, and a Go To button. Below the code editor is a logs table with columns: User, Application, Operation, Time, Status, Read, and Size. The logs section is currently empty. At the bottom, there is a filter input field with the placeholder "Click here to filter the log list".

# DML OPERATIONS AND BATCHABLE

## Create the DML Insert for the Job Posting Site

The HR Manager was looking in the Application and just checking how many candidates have applied for the job and he came to know that there are too many Candidates who applied for the job. So what the HR manager is Doing whenever there are too many Records he stores the information in the Excel sheet as per the month's Records. So he went to the Developer and asked to fetch the Record of the Job Posting Site. So This Task will be Executed with Apex Batch

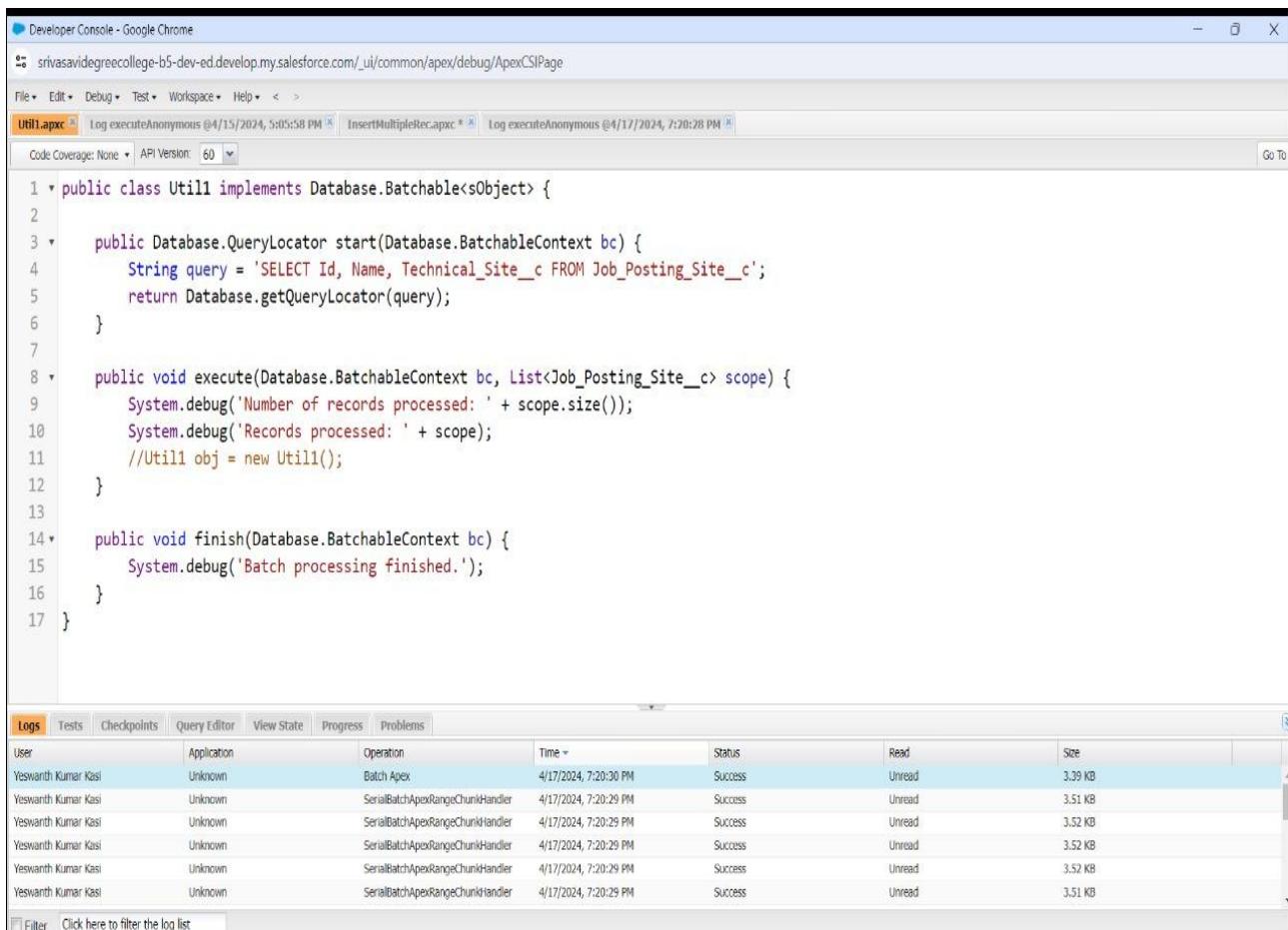


The screenshot shows the Salesforce Developer Console in Google Chrome. The URL is [srivasavidegreecollege-b5-dev-ed.develop.my.salesforce.com/\\_ui/common/apex/debug/ApexCSIPage](https://srivasavidegreecollege-b5-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage). The tab bar shows 'Util1.apxc' and 'InsertMultipleRec.apxc'. The code editor contains the following Apex code:

```
1 public class InsertMultipleRec {
2     public static void InsertMethod(){
3         List<Job_Posting_Site__c> insRec = new List <Job_Posting_Site__c>();
4         for(integer i =1; i<1000; i++){
5             Job_Posting_Site__c r = new Job_Posting_Site__c();
6             r.Name = 'naukri';
7             r.status__c='Complete';
8             insRec.add(r);
9         }
10        insert insRec;
11        system.debug(insRec);
12    }
13 }
```

The code is a batch class named 'InsertMultipleRec' with a static method 'InsertMethod'. It creates a list of 1000 'Job\_Posting\_Site\_\_c' records, each with the name 'naukri' and status 'Complete', and inserts them into the database. The developer console interface includes a 'Logs' tab, a 'Tests' tab, a 'Checkpoints' tab, a 'Query Editor' tab, a 'View State' tab, a 'Progress' tab, and a 'Problems' tab. The 'Problems' tab is currently selected.

# Create the Batch



The screenshot shows the Salesforce Developer Console in Google Chrome. The URL is `srivasavidegrecollge-b5-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage`. The tabs at the top are `Util1.apex` (selected), `Log executeAnonymous @4/15/2024, 5:05:58 PM`, `InsertMultipleRec.apxc`, and `Log executeAnonymous @4/17/2024, 7:20:28 PM`. The code editor contains the following Apex code:

```
1 * public class Util1 implements Database.Batchable<sObject> {
2
3     public Database.QueryLocator start(Database.BatchableContext bc) {
4         String query = 'SELECT Id, Name, Technical_Site__c FROM Job_Posting_Site__c';
5         return Database.getQueryLocator(query);
6     }
7
8     public void execute(Database.BatchableContext bc, List<Job_Posting_Site__c> scope) {
9         System.debug('Number of records processed: ' + scope.size());
10        System.debug('Records processed: ' + scope);
11        //Util1 obj = new Util1();
12    }
13
14     public void finish(Database.BatchableContext bc) {
15         System.debug('Batch processing finished.');
16     }
17 }
```

The logs section shows the following entries:

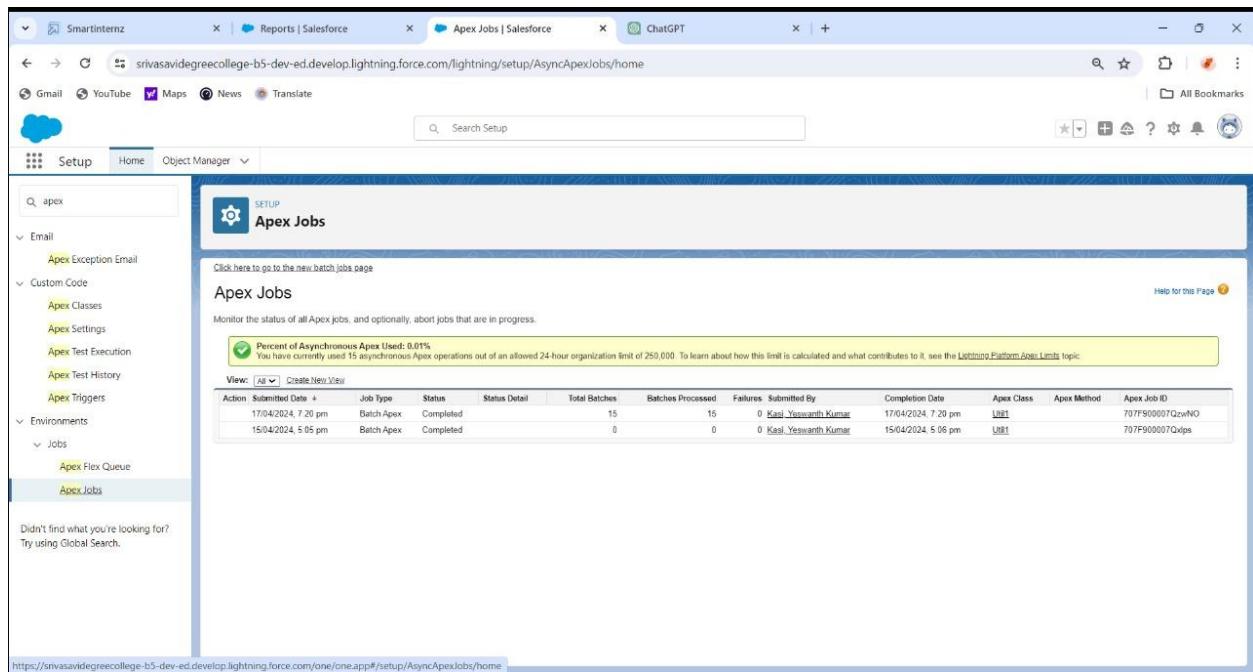
User	Application	Operation	Time	Status	Read	Size
Yeswanth Kumar Kasi	Unknown	Batch Apex	4/17/2024, 7:20:30 PM	Success	Unread	3.39 KB
Yeswanth Kumar Kasi	Unknown	SerialBatchApexRangeChunkHandler	4/17/2024, 7:20:29 PM	Success	Unread	3.51 KB
Yeswanth Kumar Kasi	Unknown	SerialBatchApexRangeChunkHandler	4/17/2024, 7:20:29 PM	Success	Unread	3.52 KB
Yeswanth Kumar Kasi	Unknown	SerialBatchApexRangeChunkHandler	4/17/2024, 7:20:29 PM	Success	Unread	3.52 KB
Yeswanth Kumar Kasi	Unknown	SerialBatchApexRangeChunkHandler	4/17/2024, 7:20:29 PM	Success	Unread	3.52 KB
Yeswanth Kumar Kasi	Unknown	SerialBatchApexRangeChunkHandler	4/17/2024, 7:20:29 PM	Success	Unread	3.51 KB

At the bottom, there is a filter input field with the placeholder `Click here to filter the log list`.

- Write a class that implements the `Database.Batchable` interface. This class will contain the logic to process your data in batches.
- Define the `start`, `execute`, and `finish` methods required by the interface.
- In the `start` method, query for the records you want to process.
- In the `execute` method, process the records in the batch.
- In the `finish` method, perform any final actions after all batches have been processed.

# Check the Batch

- Click on the gear icon
- click on setup
- Click on the home button and search for
- Apex Jobs than you can see the success of the chunks it run 200 batch for the total record



The screenshot shows a Salesforce setup page for 'Apex Jobs'. The left sidebar has a 'Search Setup' bar and a list of Apex-related items: Email (Apex Exception Email), Custom Code (Apex Classes, Apex Settings), Apex Test Execution, Apex Test History, Apex Triggers, Environments (Apex Flex Queue, Apex Jobs), and Jobs. The 'Apex Jobs' item is selected. The main content area has a 'SETUP Apex Jobs' header with a 'Click here to go to the new batch jobs page' link. Below it is a 'Percent of Asynchronous Apex Used: 0.01%' message. A table lists completed batches: 15 total batches, 15 processed, 0 failures, submitted by Kasl\_Yeswanth Kumar on 17/04/2024 at 7:20 pm, and completed on 15/04/2024 at 5:05 pm.

Action	Submitted Date	Job Type	Status	Status Detail	Total Batches	Batches Processed	Failures	Submitted By	Completion Date	Apex Class	Apex Method	Apex Job ID
	17/04/2024, 7:20 pm	Batch Apex	Completed		15	15	0	Kasl_Yeswanth Kumar	17/04/2024, 7:20 pm	Unit		707F900007QzwNO
	15/04/2024, 5:05 pm	Batch Apex	Completed		0	0	0	Kasl_Yeswanth Kumar	15/04/2024, 5:05 pm	Unit		707F900007Qxjps

