SALESFORCE

Workforce Administration Solution using Salesforce

Objective

The Workforce Administration Solution is a comprehensive software platform designed to enhance organizational efficiency by streamlining employee and asset management processes. It acts as a centralized system that integrates key functions such as employee project allocation, performance tracking, and asset assignment.

Salesforce

Salesforce is your customer success platform, designed to help you sell, service, market, analyze, and connect with your customers. Salesforce has everything you need to run your business from anywhere. Using standard products and features, you can manage relationships with prospects and customers, collaborate and engage with employees and partners, and store your data securely in the cloud.

Activity 1: Creating Developer Account :

1. Go to https://developer.salesforce.com/signup

2. On the sign up form, enter valid details.

3. Click on Sign Me Up.

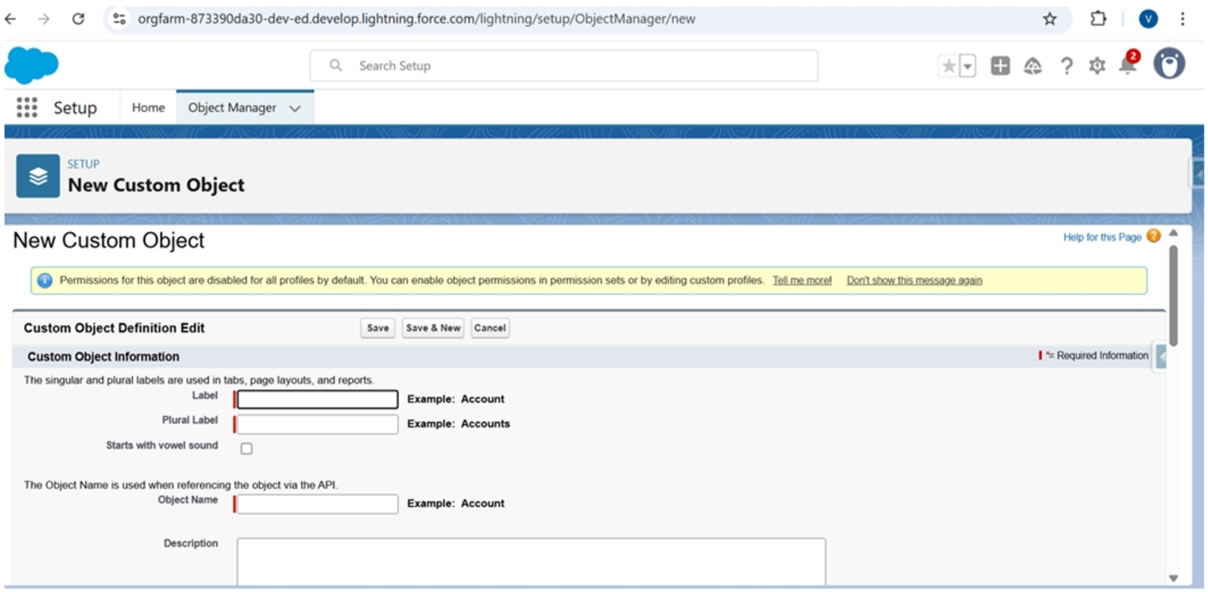
4. Go to your email inbox, verify account, and set password/security question.

5. You will be redirected to Salesforce setup page.

Objects

1. From the setup page --> Click on Object Manager --> Click on Create --> Click on Custom

Object.



1) Enter the label name: Employee
2) Plural label name: Employees
3) Enter Record Name Label and Format
    1. Record Name : Employee ID
    2. Data Type : Auto Number
    3. Display Format : EMS-{0000}
    4. Startng Number : 1
    5. Click on Allow reports
    6. Allow search --> Save.

4) Create 3 more objects with label names as ProjectTask, Asset, Asset Service.
Note: use “Text” as a data type and label Record Name as “Project Task Name”. a. These are the objects we have created in our project.

Tabs

1. Tabs are user interfaces to build/view object records.
   Types: Custom tabs, Web tabs, Visualforce tabs, Lightning component tabs, Lightning

page tabs.

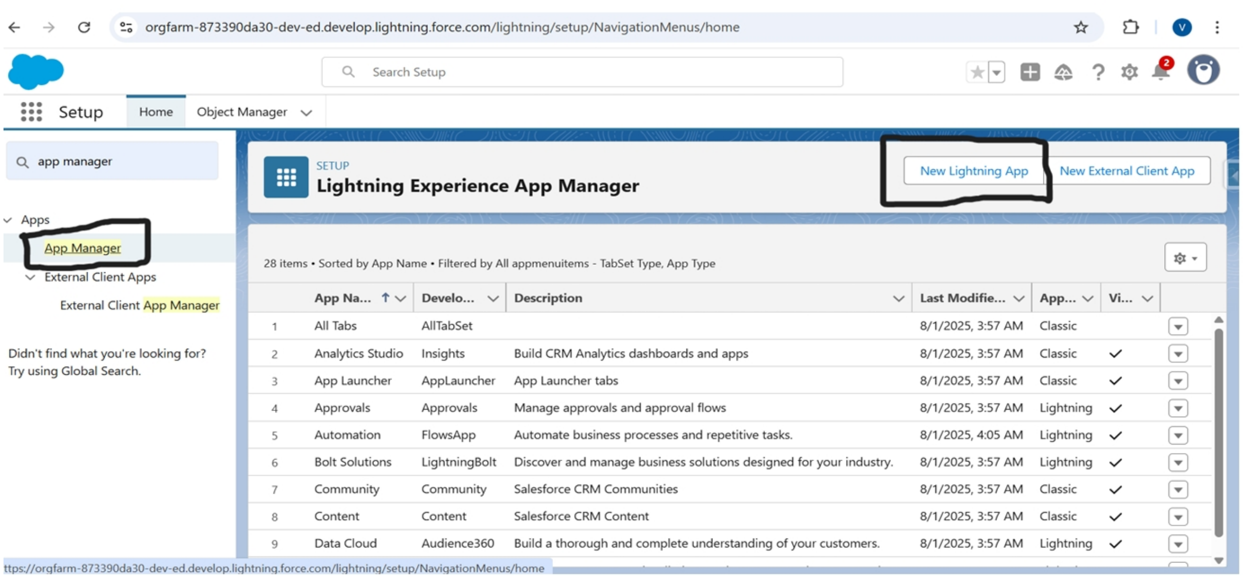
2. Go to setup page --> type Tabs in Quick Find bar --> click on tabs --> New (under custom object tab) 2. Select Object(Employee) --> Select any tab style --> Next (Add to profiles page) keep it as default --> Next (Add to Custom App) keep it as default --> Save.

3. Creating tabs for the remaining objects Project, ProjectTask, Asset, AssetService, Leaves.

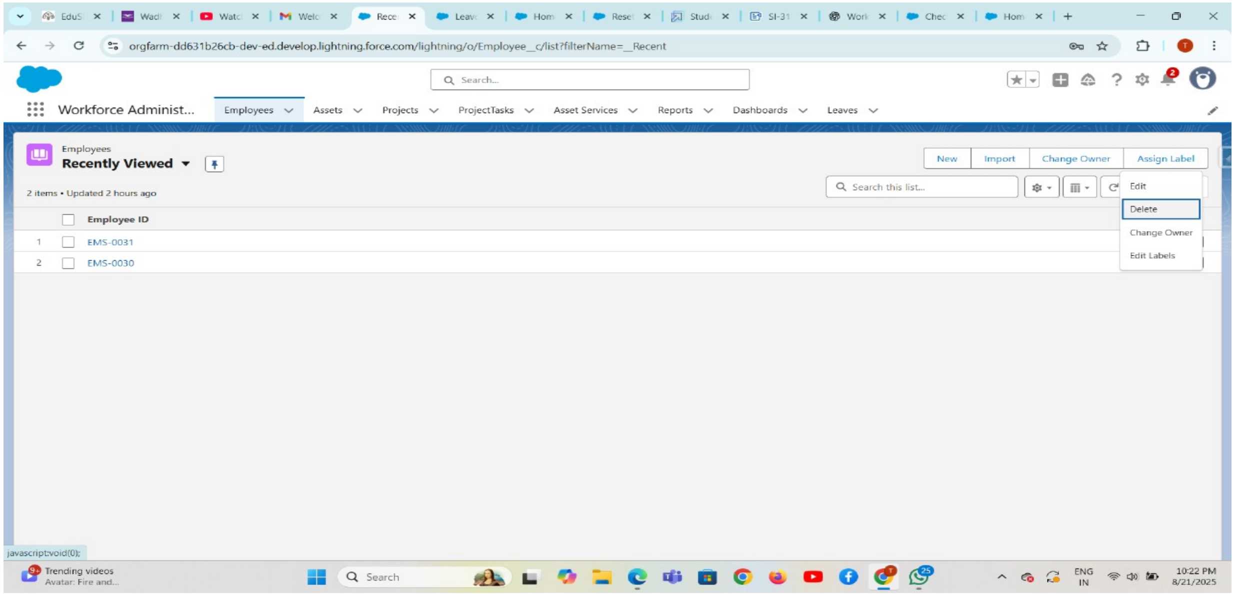
Lightning App

Lightning app bundles objects, tabs, and items in one navigation bar.

Steps:
1. Setup → App Manager → New Lightning App.
2. Fill details (Name: Workforce Administrator Solution).
3. Add Navigation Items (Employee, Projects, Assets, etc.).

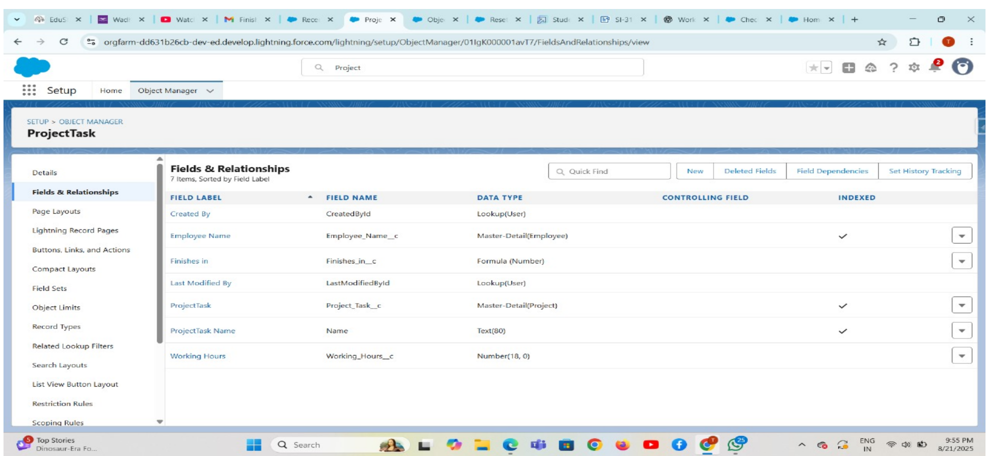


4. Assign User Profiles (System Administrator).
5. Save & Finish.

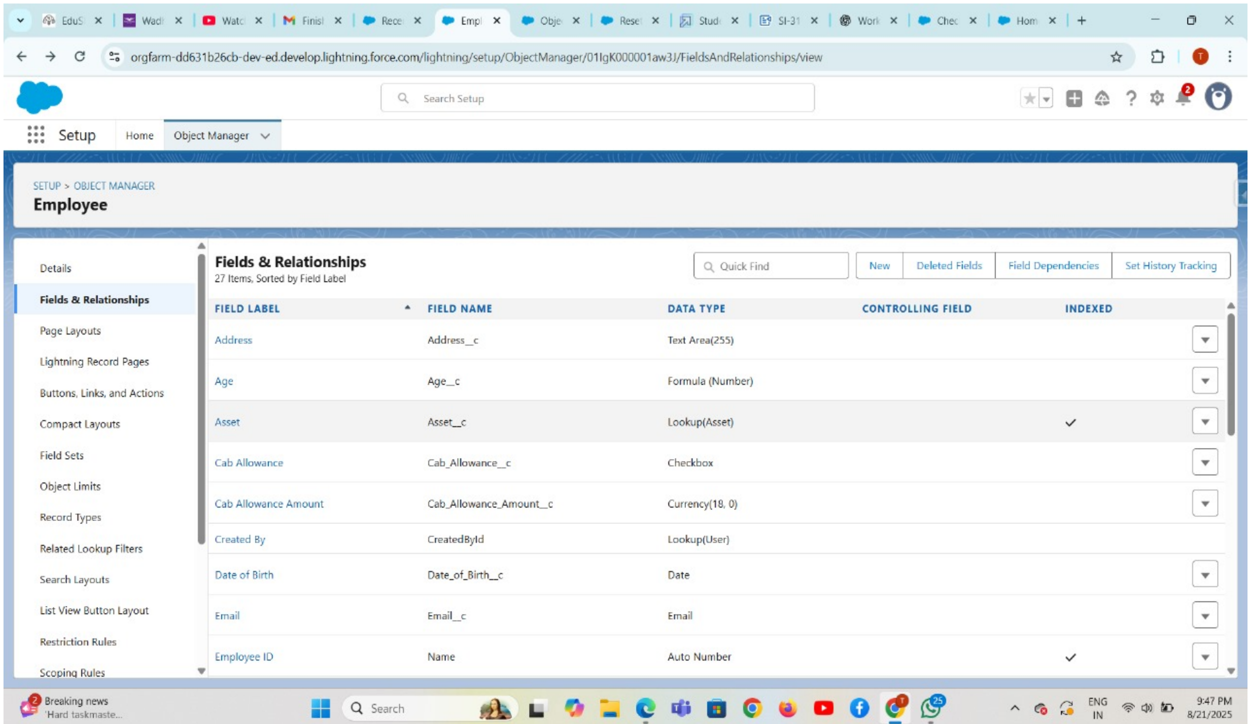


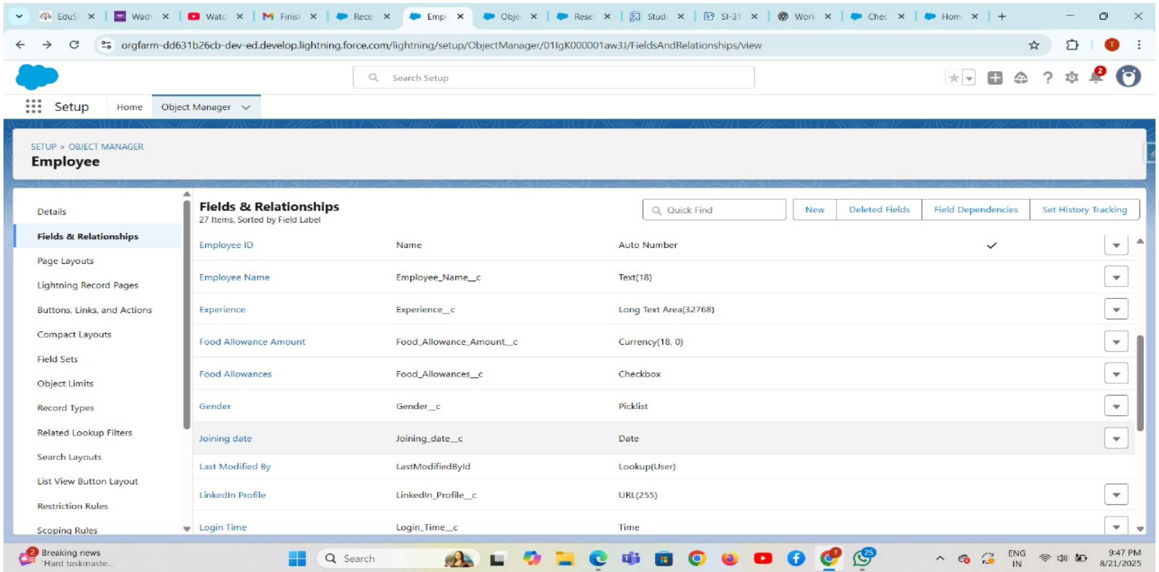
Fields & Relationships

1. Fields store data. Types: Standard, Custom.
2. Relationships link objects (Lookup, Master-Detail, Many-to-Many, Hierarchical).
3. Process: Setup → Object Manager → Object → Fields & Relationships → New → Choose Data type → Fill details → Save.

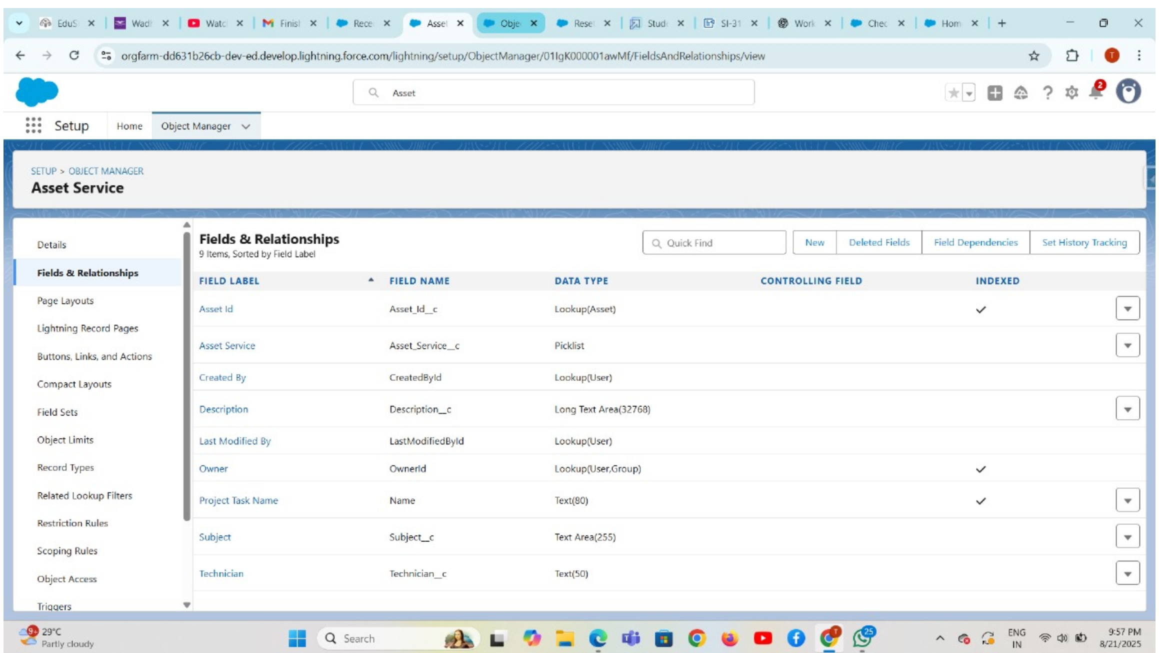


1. Fields and relationships of ProjectTask

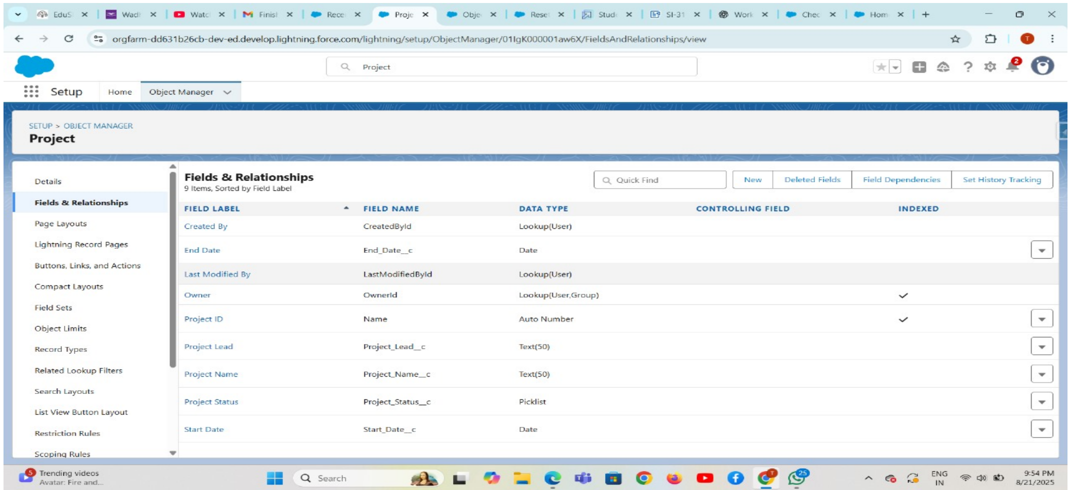




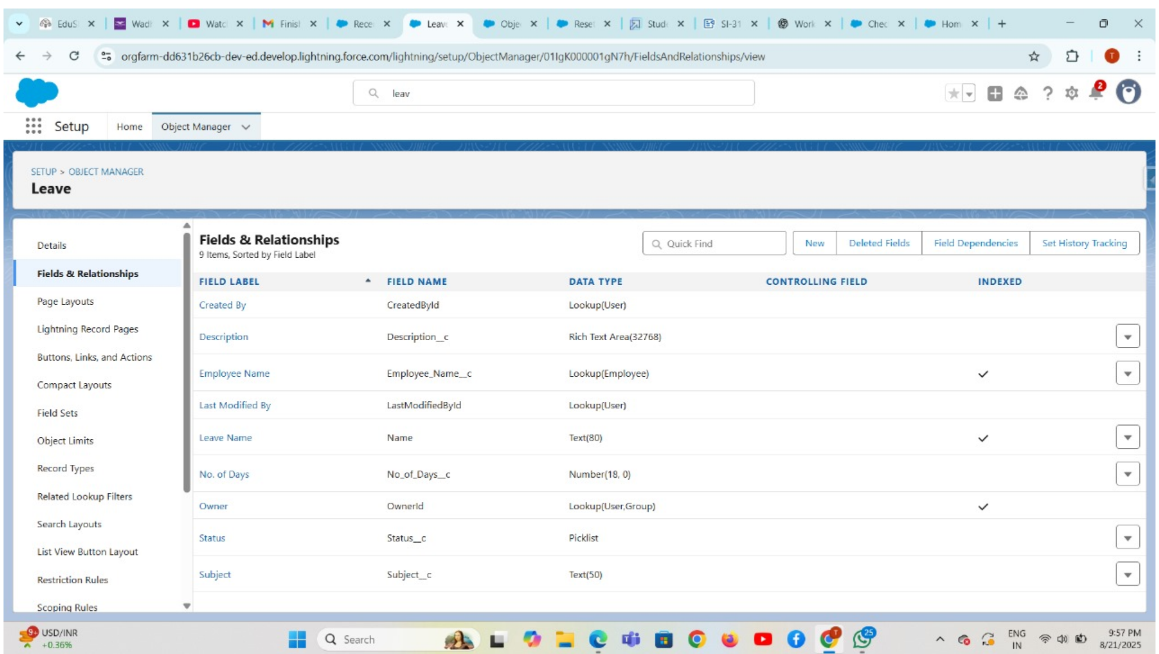
2. Fields and relationships of Employee



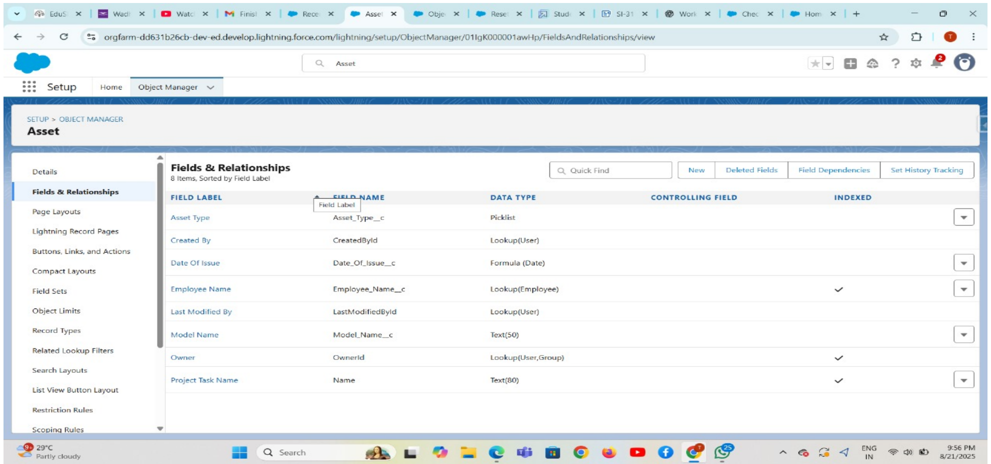
3. Fields and relationships of Asset Service



4. Fields and relationships of Project



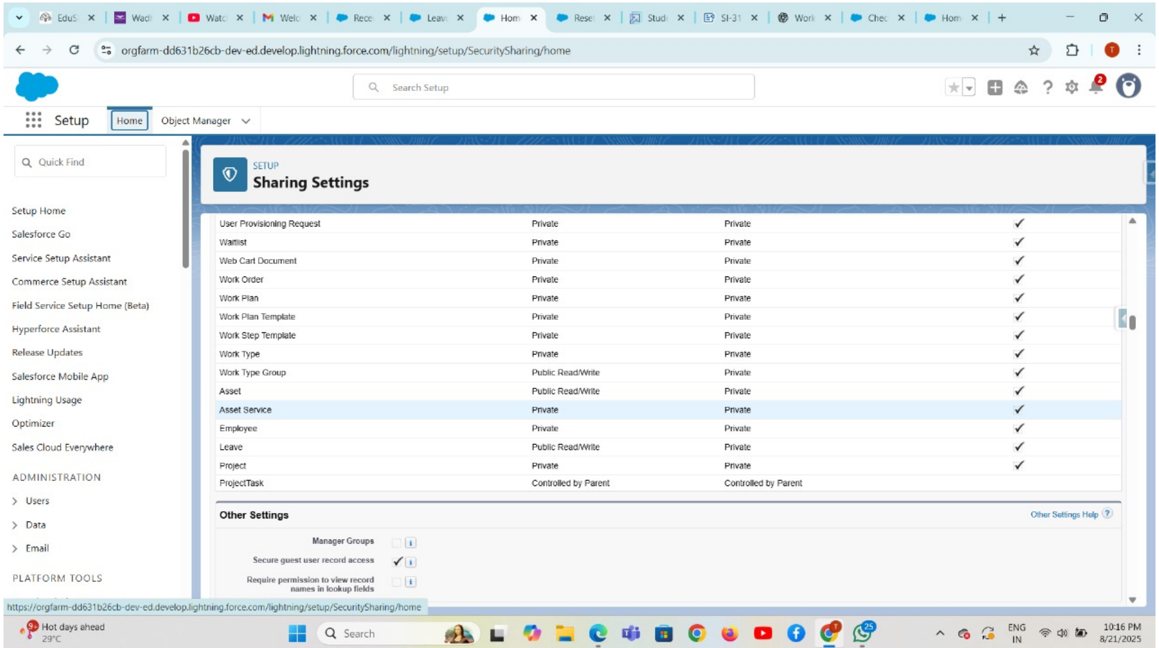
5. Fields and relationships of Leave



6. Fields and relationships of Asset

OWD Settings

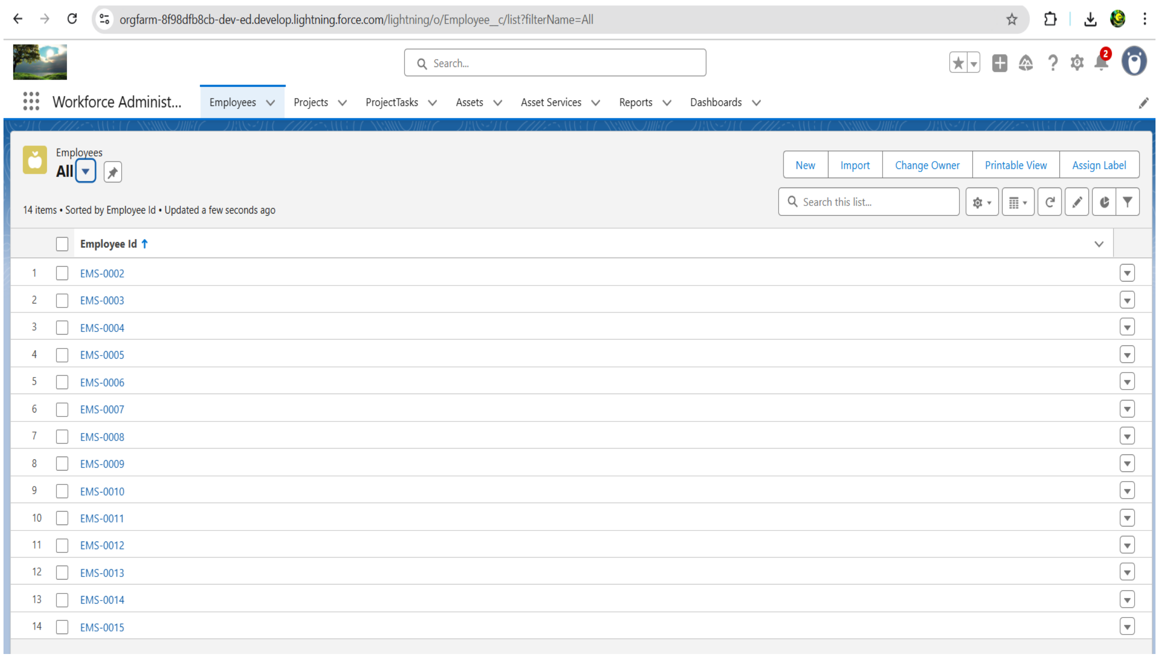
1. Organization-Wide Defaults define access rules.
2. Steps: Setup → Sharing Settings → Edit → Change objects (Employee, Project, AssetService) → Set access to Private → Grant access using hierarchies → Save.



User Adoption

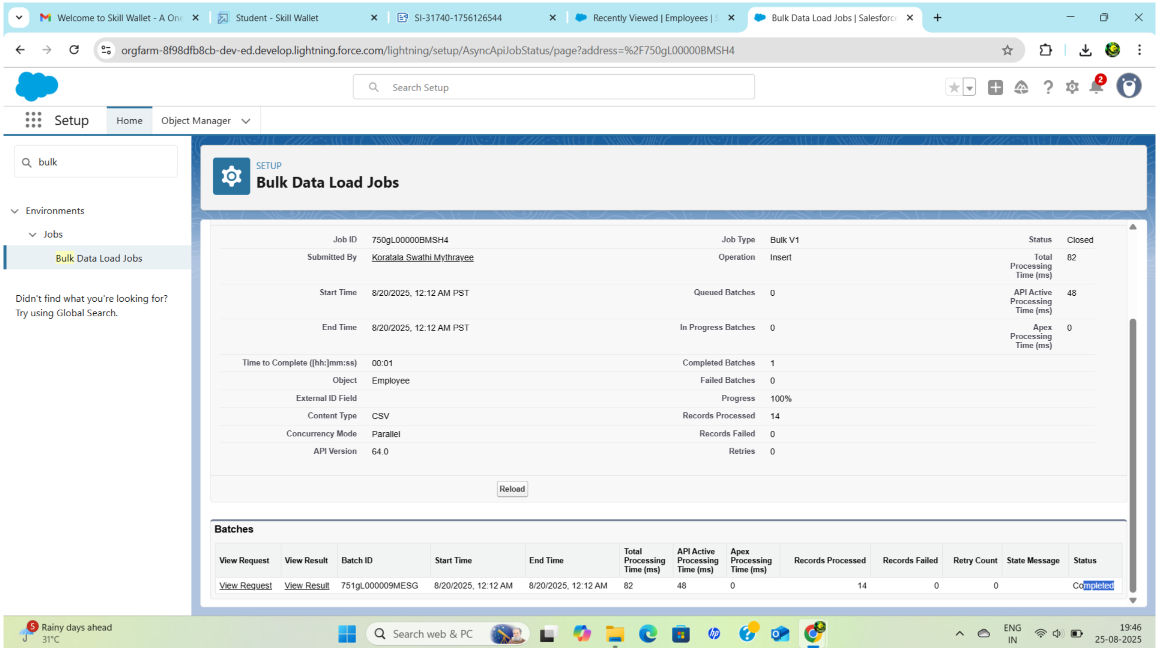
1. User tasks: Create/Edit Users, Reset Passwords, Grant Permissions, Configure Data Access.

2. Example: Creating an Employee record via App Launcher → Employee Tab → New → Save.



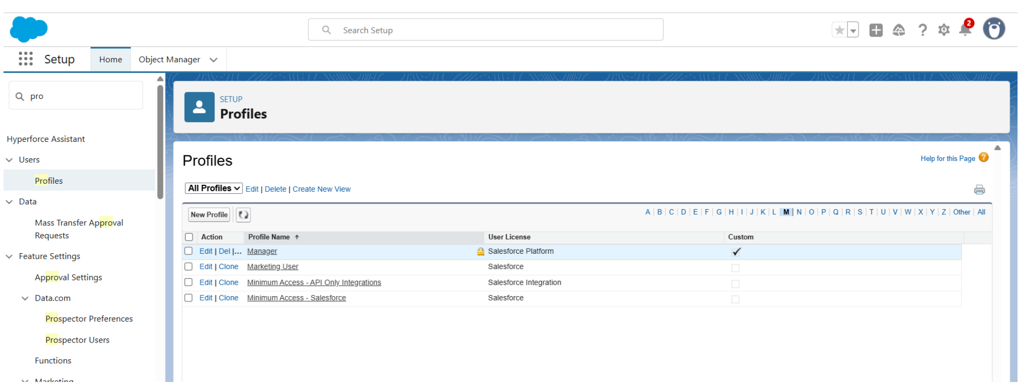
Import Data

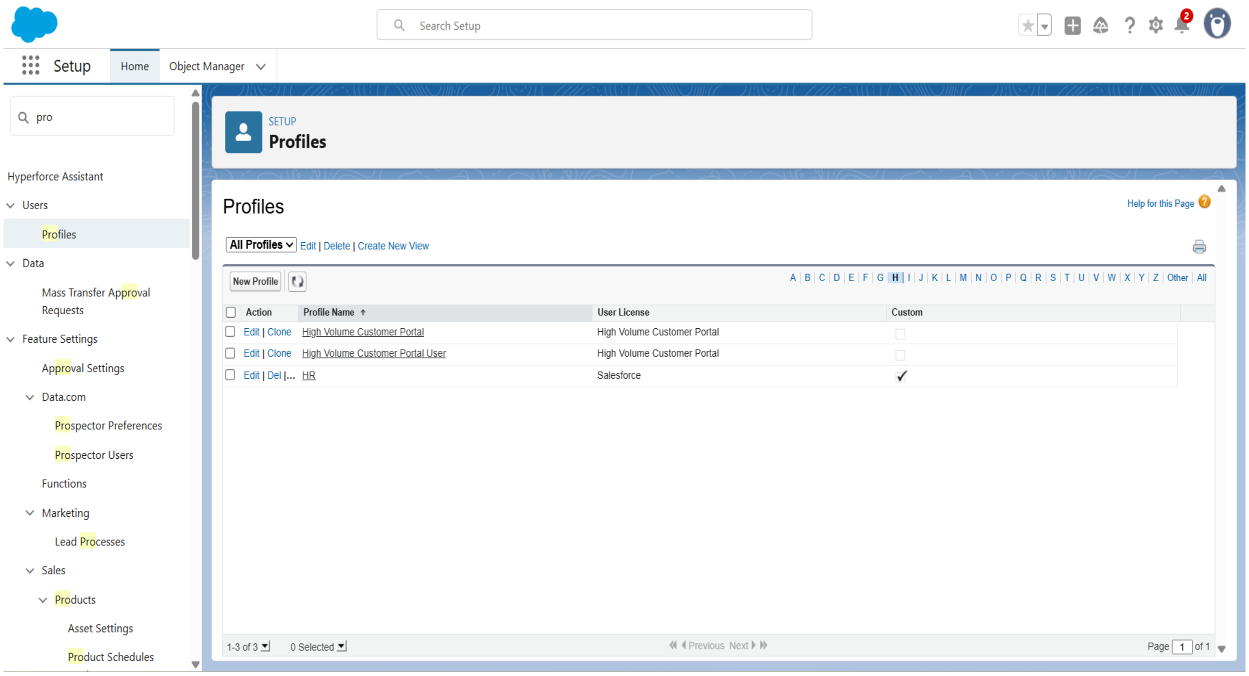
1. Data Import Wizard allows external data upload.
2. Steps: Setup → Data Import Wizard → Launch → Choose Object → Upload CSV → Map Fields → Start Import → Verify.
3. After importing we must get record failed as 0.



Profiles

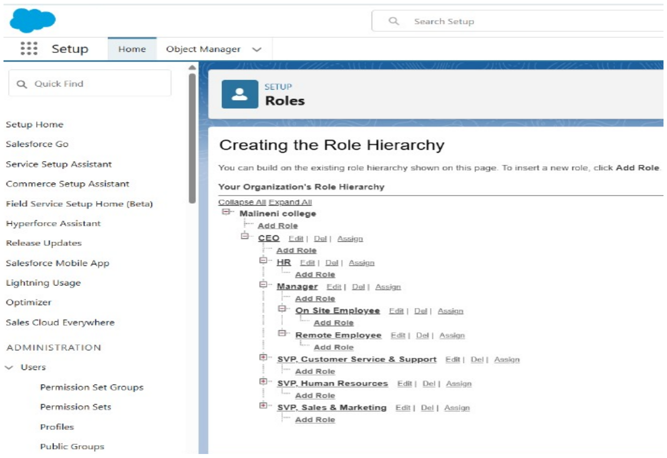
1. Profiles define permissions (Object, Field, Tab, App, etc.).
2. Steps: Setup → Profiles → Clone Standard Profile → Name → Edit → Assign Object Permissions → Save.
3. The profiles for this project are HR ,Manager ,On-Site Employee





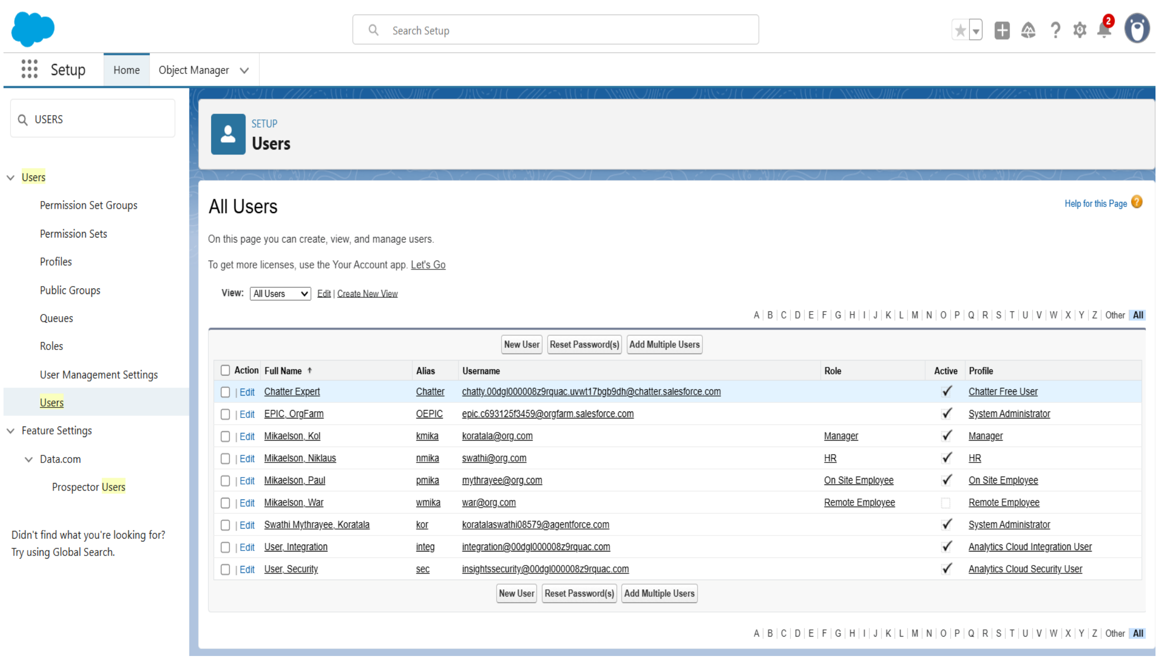
Roles

1. Roles define record-level visibility.
2. Steps: Setup → Roles → Expand All → Add Role → Label (HR, CEO, Manager, etc.) → Save.
3. Role hierarchy: CEO → HR→ Manager → Onsite/Remote Employees.



Users

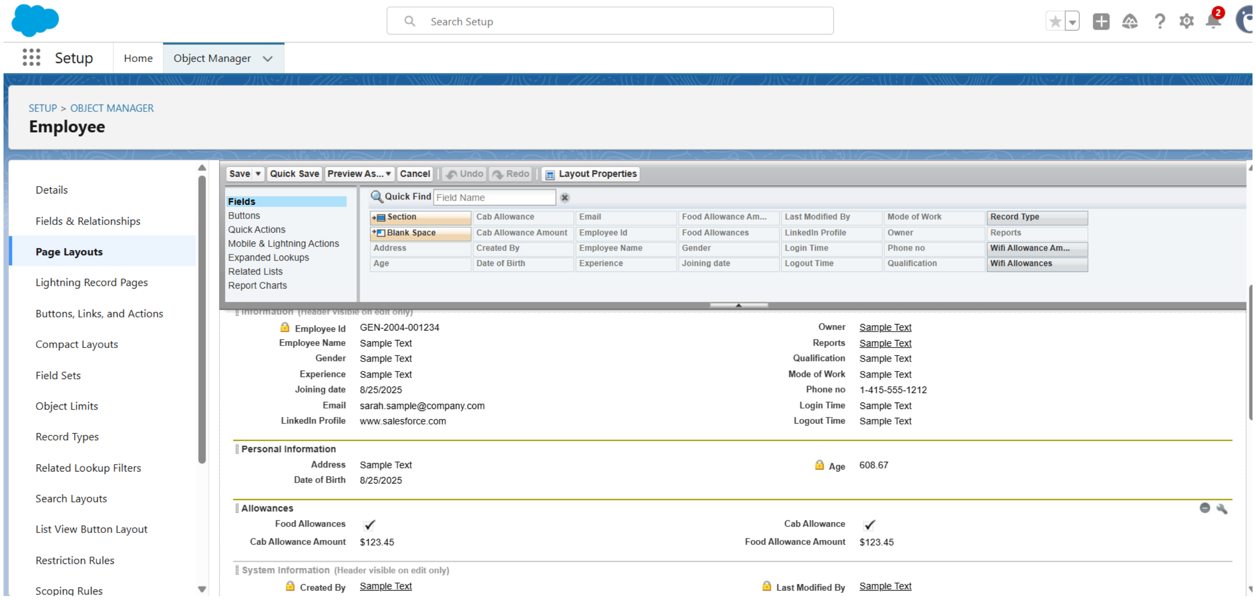
Users log in to Salesforce. Each has an account.
Steps: Setup → Users → New User → Fill Details → Assign Role & License → Save.

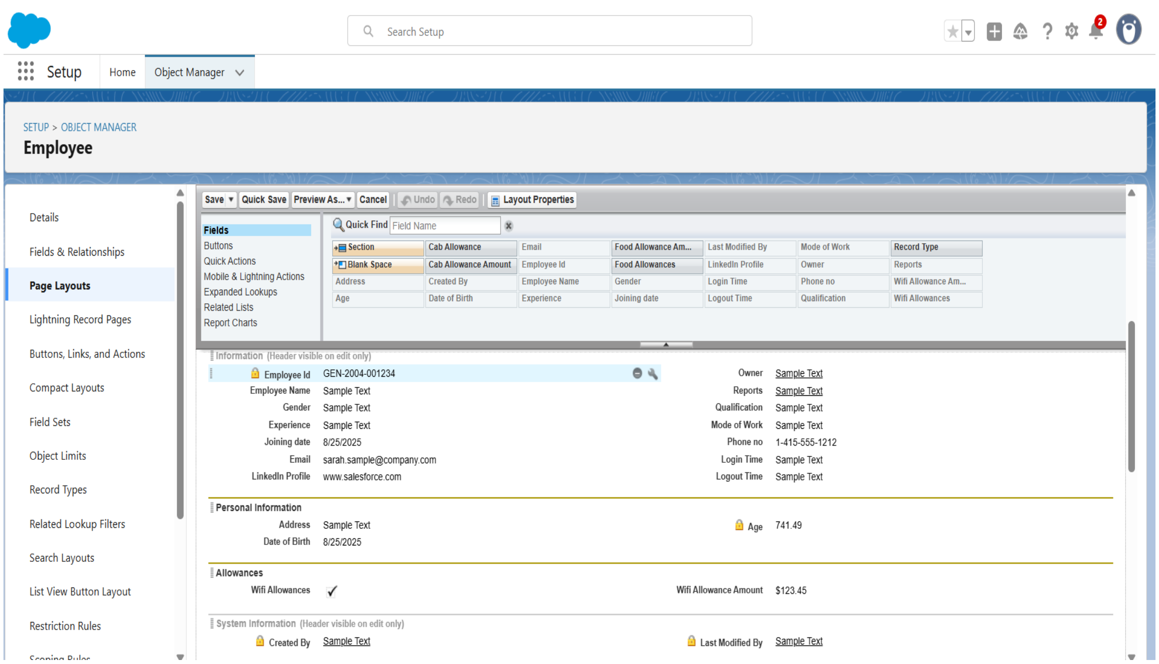


Page Layouts

Customize design of record detail/edit pages.

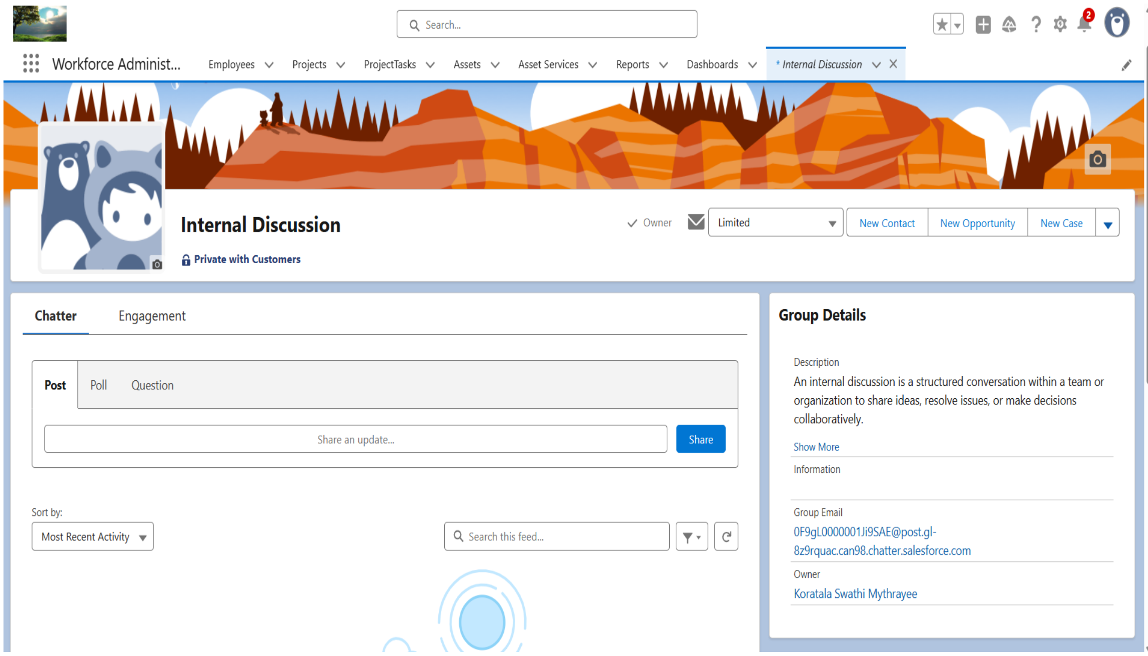
Steps: Setup → Object Manager → Object → Page Layouts → New → Drag & Drop fields/sections → Save.





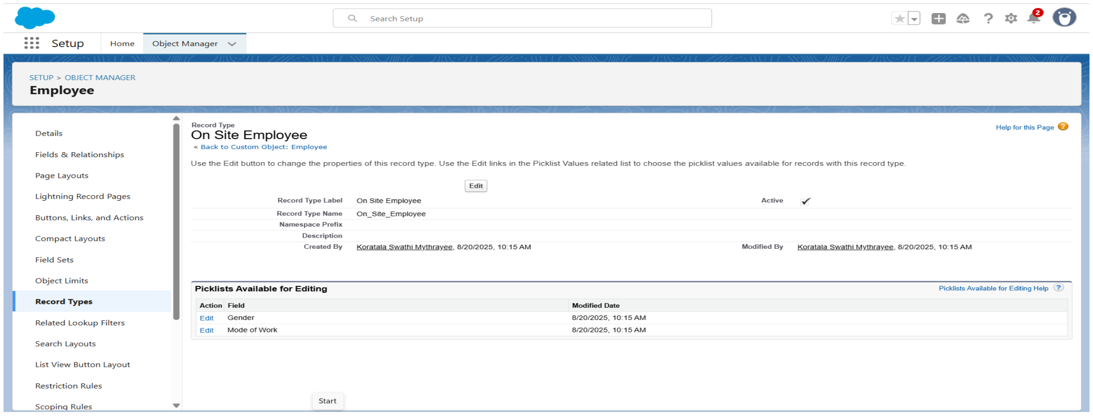
Chatter Groups

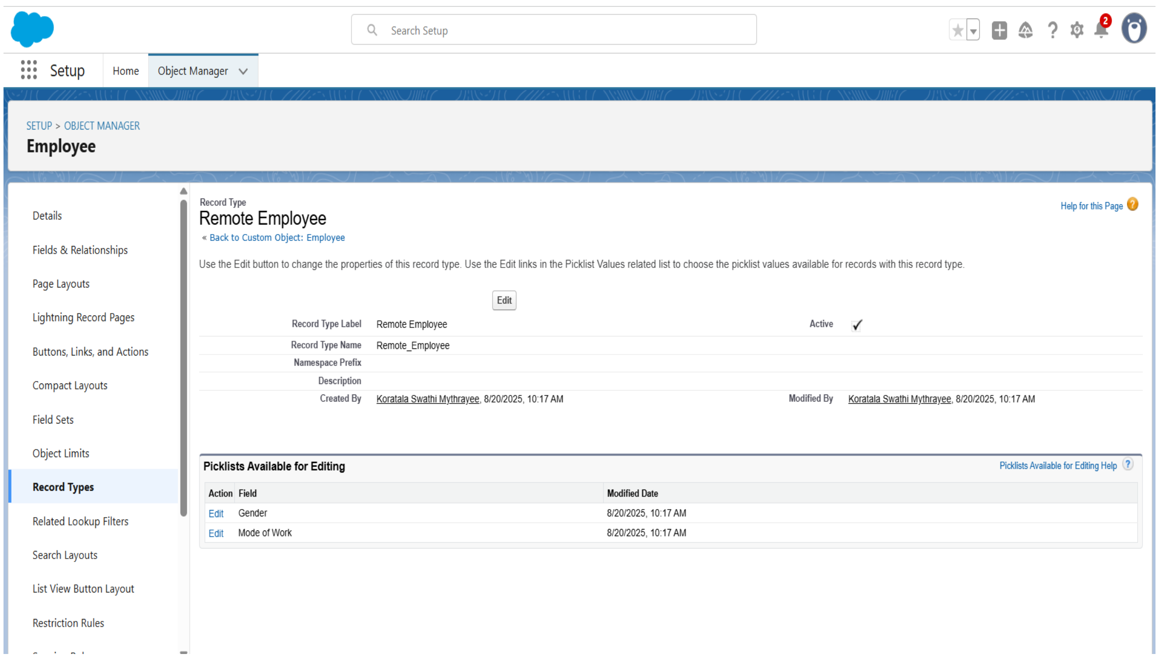
Collaboration spaces for team discussions, file sharing, updates.
 Steps: App Launcher → Groups → New → Fill details (Name, Description, Private) → Add Members → Save.



Record Types

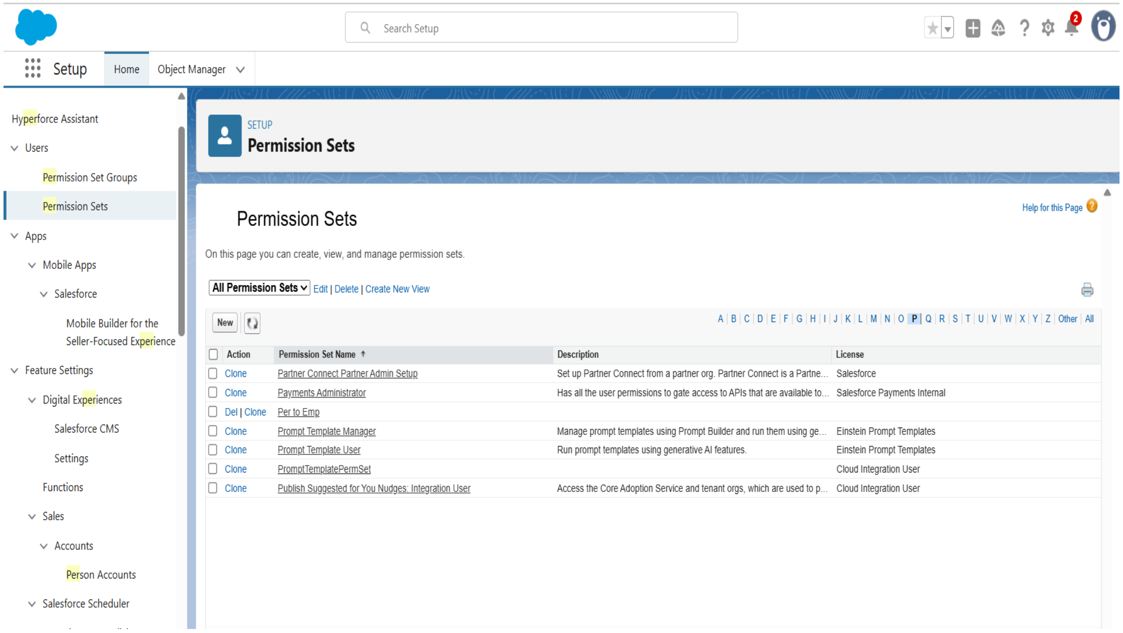
Group records of the same object. Allow different layouts, picklists, processes.
 Steps: Setup → Object Manager → Object → Record Types → New → Fill details → Assign Profiles → Select Page Layout → Save.

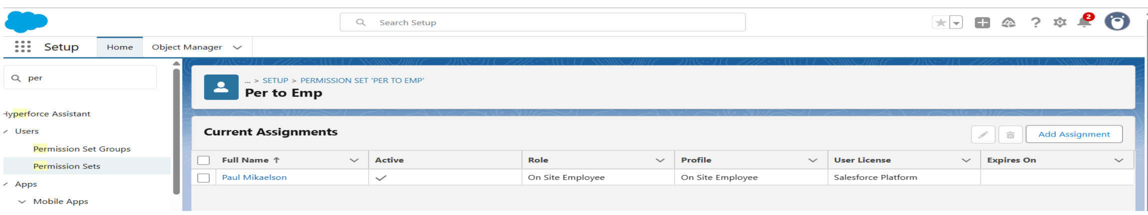




Permission Sets

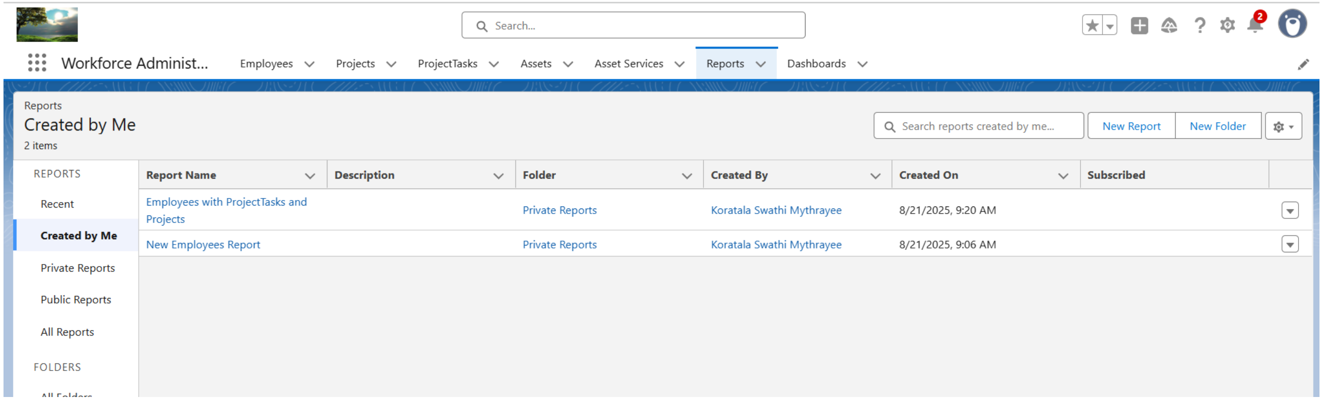
Additional permissions beyond profiles.
Steps: Setup → Permission Sets → New → Label (Per to Emp) → Object Settings → Employee → Assign Permissions → Save → Assign Users.





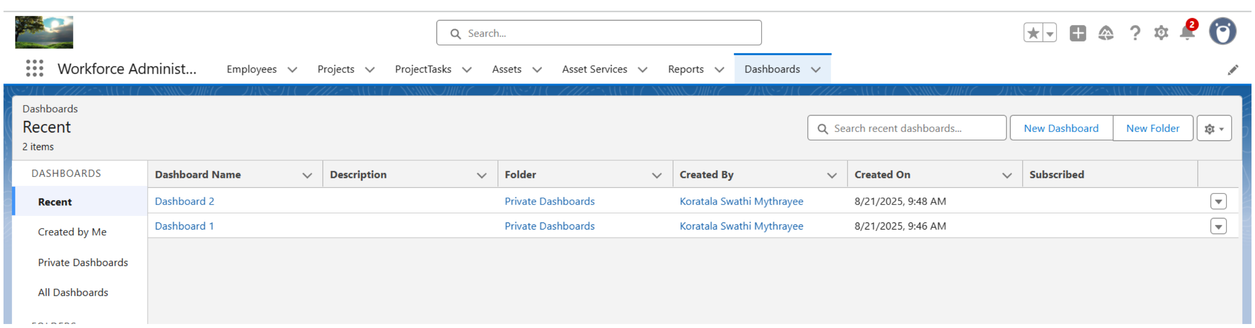
Reports

1. Reports display Salesforce data.
2. Steps: Reports Tab → New Report → Select Report Type → Customize → Save/Run.
3. Reports created: Employees with ProjectTasks, Employees with Assets, Employee.



Dashboards

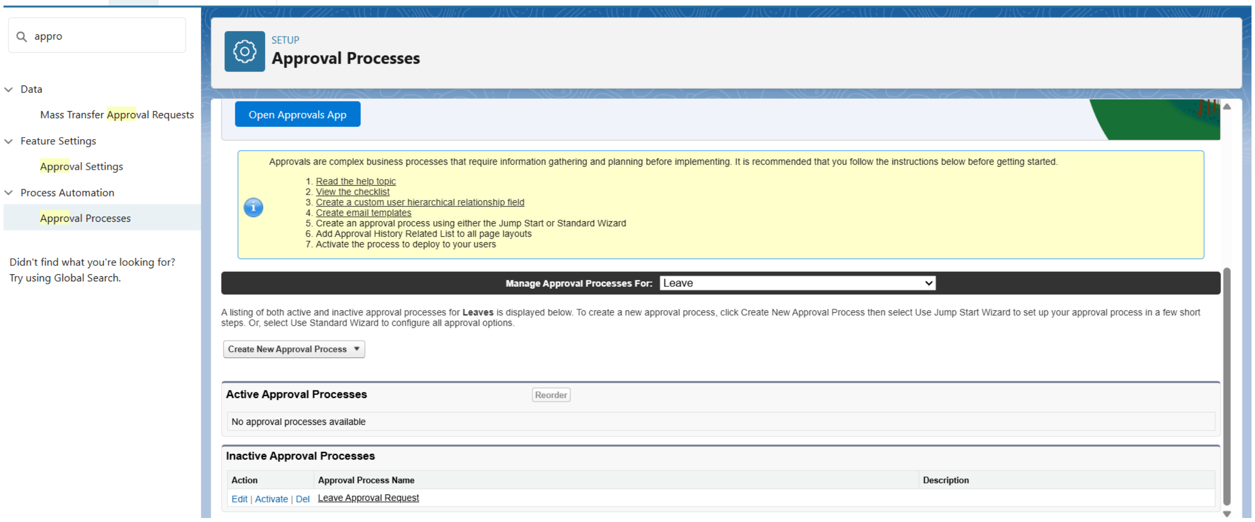
1. Dashboards visualize data from reports.
2. Steps: Dashboard Tab → New Dashboard → Add Components → Select Report → Save & Done.



Approval Process

1. Use case: Leave approval.

2. Steps: Setup → Approval Processes → Create → Assign Approvers (Manager, HR) → Define Submission, Approval & Rejection actions → Save.



Apex Trigger

Prevent duplicate employee names.

Code:

```
trigger EmpInsert on Employee__c (before insert) {
  for(Employee__c pass : Trigger.New){
     List<Employee__c> mynew = [SELECT Id, Name FROM Employee__c WHERE Employee_Name__c
=: pass.Employee_Name__c];
     if(mynew.size() > 0){
        pass.Name.addError('Employee with same name is existing');
     }
  }
```

}

Conclusion

This project enables organizations to maintain accurate employee records, track projects, monitor performance, and manage assets effectively. By automating administrative tasks, it reduces manual effort, improves transparency, ensures efficient resource utilization, and supports better decision-making.

THE END…