

With sharing and without sharing in Apex Class

With sharing and without sharing in Apex Class determines whether the class should enforce all the Organization sharing settings.

Scenario 1:

```
<apex:page controller="Sample">
  The count is {!intCount}
</apex:page>
```

```
public with sharing class Sample {

    public Integer intCount {get;set;}

    public Sample() {
        intCount = [ SELECT COUNT() FROM Employee__c ];
    }

}
```

If OWD is Private, the VF will show no of Employee records the user have access to.

If OWD is Public Read Only or Public Read/Write, the VF will show no of Employee records in the employee object.

Scenario 2:

```
<apex:page controller="Sample">
  The count is {!intCount}
</apex:page>
```

```
public without sharing class Sample {

    public Integer intCount {get;set;}

    public Sample() {
        intCount = [ SELECT COUNT() FROM Employee__c ];
    }

}
```

```
}
```

If OWD is Private or Public Read Only or Public Read/Write, the VF will show no of Employee records in the employee object.

Scenario 3:

```
<apex:page controller="Sample">
  <apex:form >
    <apex:commandButton value="Create Employee" action="{!createEmp}"/>
  </apex:form>
</apex:page>
```

```
public with sharing class Sample {
```

```
    public Sample() {}
```

```
    public void createEmp() {
        Employee__c emp = new Employee__c();
        insert emp;
    }
```

```
}
```

If an user don't have CRUD access to employee object and clicks Create Employee button, it will create employee record.

Scenario 4:

```
<apex:page controller="Sample">
  <apex:form >
    <apex:commandButton value="Create Employee" action="{!createEmp}"/>
  </apex:form>
</apex:page>
```

```
public without sharing class Sample {
```

```
    public Sample() {}
```

```
    public void createEmp() {
        Employee__c emp = new Employee__c();
```

```
        insert emp;
    }
}
```

If an user don't have CRUD access to employee object and clicks Create Employee button, it will create employee record.

Scenario 5:

Example:

Visualforce page:

```
<apex:page sidebar="false" controller="Sample">
    <apex:form >
        <apex:pageblock >
            <apex:pageblockTable value="{!var}" var="a">
                <apex:column value="{!a.Name}"/>
            </apex:pageblockTable>
        </apex:pageblock>
    </apex:form>
</apex:page>
```

Apex Class:

```
public with Sharing class Sample
{
    public List<Account> getVar()
    {
        return [SELECT Name FROM Account];
    }
}
```

OWD for Account is Private.

Force.com Home

Sharing Settings

This page displays your organization's sharing settings. These

System Overview

Personal Setup

- My Personal Information
- Email
- Import
- Desktop Integration
- My Chatter Settings
- My Social Accounts and Contacts

App Setup

- Customize
- Create
- Develop
- Deploy
 - Schema Builder
 - Canvas App Previewer **New!**
 - Installed Packages
 - AppExchange Marketplace
 - Critical Updates

Administration Setup

- Manage Users
- Manage Apps
- Company Profile
- ☒ Security Controls
 - Sharing Settings**
 - Field Accessibility
 - Password Policies
 - Session Settings

Manage sharing settings for: All Objects ▼

Default Sharing Settings

Organization-Wide Defaults

[Edit](#)

Object	Default Acc
Lead	Public Rea
Account, Contract and Asset	Private
Contact	Controlled
Opportunity	Private
Case	Private
Campaign	Public Full
Activity	Private
Calendar	Hide Detail
Price Book	Use
Blog	Public Rea
Interest	Controlled
Member	Public Rea
Photos	Controlled

Sharing Rules

Lead Sharing Rules

[New](#)[Rec](#)

No sharing rules specified.

Account Sharing Rules

[New](#)[Rec](#)

No sharing rules specified.

Role Hierarchy:



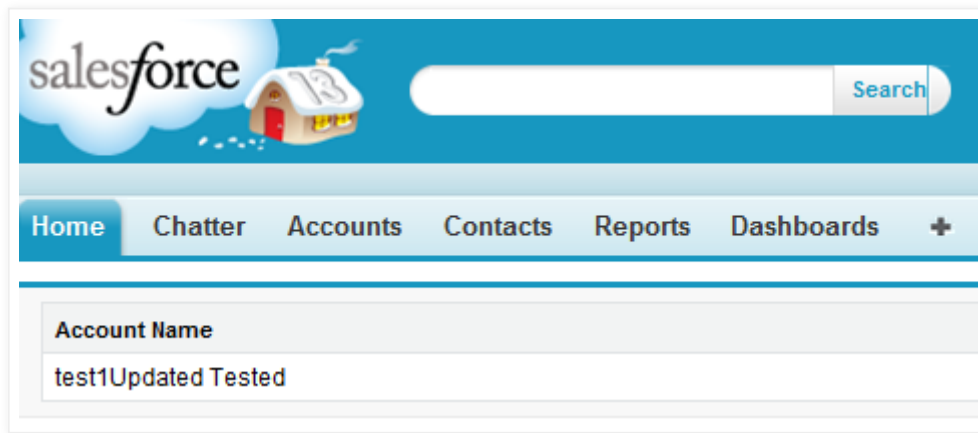
Output of the page for Managing Director Role:

A screenshot of the Salesforce user interface. The top header includes the Salesforce logo, a house icon with the number 13, and a search bar. Below the header is a navigation bar with tabs for Home, Chatter, Accounts, Contacts, Reports, and Dashboards. The main content area displays a table of account names. A red arrow points from the text "Manager's Record" to the last row of the table.

Account Name
test accountUpdated Tested
test
tested
test111
test1Updated Tested

Manager's Record

Output of the page for Manager Role:



The 'without sharing' keyword is to ensure that the sharing rules (not permissions) for the current user are not enforced.

The with sharing keyword allows you to specify that the sharing rules (and not permissions) for the current user be taken into account for a class. You have to explicitly set this keyword for the class because Apex code runs in system mode.