types of sandboxes

diff between sales cloud and service cloud

community users

Trigger execution sequence

Tasks and events -- whoId and whatId

@future annotation

Queuable

Batch apex //Heap size exceeded

Process builder - delete records

Custom setting - when it is implement

Things to be taken care of when using controller on VF page

Remoting feature of Vf: JavaScript remoting allows you to run asynchronous actions by decoupling the page from the controller and to perform tasks on the page without having to reload the entire page.

actionFunction:

Vf page execution order:

1. Evaluate Constructors on controllers and extension

2. Evaluate component --- evaluate constructors on

**Notes and Attachments Object: -------------//remember**

SAML:

The Security Assertion Markup Language (SAML), is an open standard that allows security credentials to be shared by multiple computers across a network. It describes a framework that allows one computer to perform some security functions on behalf of one or more other computers:

Authentication: Determining that the users are who they claim to be

Authorization: Determining if users have the right to access certain systems or content

Blob

outbound actions

Objects:-

Identity, system, and name fields are standard on every object in Salesforce

LookUp Relationships:-

A lookup relationship essentially links two objects together so that you can “look up” one object from the **related items** on another object. Lookup relationships can be one-to-one or one-to-many.

Ex: One-to-many: Contact and Account

you use lookup relationships when objects are only related in some cases. Sometimes a contact is associated with a specific account, but sometimes it’s just a contact. Objects in lookup relationships usually work as stand-alone objects and have their own tabs in the user interface.

* Lookups are generally for use where you may or may need to have a relationship between two objects (but not always).
* Lookups are generally used to reference commonly shared data, such as reference data.
* Lookups are used to link two objects together when you don't want the behaviour of the master-detail - particularly around sharing rules, profile permissions and cascade delete.
* Lookups are used when you need to relate multiple 'parents' to the detail record.

Master-Detail Relationships:-

The master object controls certain behaviors of the detail object, like who can view the detail’s data.

In a master-detail relationship, the detail object doesn’t work as a stand-alone.

if a record on the master object is deleted, all its related detail records are deleted as well. When you’re creating master-detail relationships, you always create the relationship field on the detail object.

The key difference is master-detail has a direct dependency between the objects:

* You cannot have a detail record without a master.
* The detail record inherits sharing rules from the master.
* You cannot update the relationship to the master in a master-detail relationship.
* The number of master-detail relationships you can use are limited.
* You cannot set profile object permissions for a detail record
* Master-detail relationships are automatically included in report record types

Hierarchical relationship:-

Hierarchical relationships are a special type of lookup relationship. The main difference between the two is that hierarchical relationships are only available on the User object. You can use them for things like creating management chains between users.

**Data Import Wizard:**  It can import up to 50,000 records at a time.

**Data Loader**—this is a client application that can import up to five million records at a time, of any data type, either from files or a database connection

**Data Export Wizard**—this is an in-browser wizard, accessible through the Setup menu. It allows you to export data manually once every six days (for weekly export) or 28 days (for monthly export).

**Compact Layouts:**

Compact layouts control which fields your users see in the highlights panel at the top of a record. They also control the fields that appear in the expanded lookup card you see when you hover over a link in record details, and in the details section when you expand an activity in the activity timeline.

Compact layouts also control how records display in the Salesforce mobile app.

Lightning pages are a collection of Lightning components arranged in regions on the page. You can customize the structure of the page and the position of its components with the **Lightning App Builder**

The page layout editor, also known as **page layouts**, helps you manage the content of pages.

Custom Button and Links:

Custom links can link to an external URL, such as www.google.com, a Visualforce page, or your company’s intranet. Custom buttons can connect users to external applications, such as web pages, and launch custom links.

There are three primary types of custom buttons and links that you can create.

* List button—Appears on a related list on an object record page.
* Detail page link—Appears in the Links section of the record details on an object record page.
* Detail page button—Appears in the action menu in the highlights panel of a record page

REPORTS:

a report is a list of records (like opportunities or accounts) that meet the criteria you define.

Every report is stored in a folder. Report folders determine how reports are accessed, and who can access them to view, edit, or manage. Folders can be public, hidden, or shared. You control who has access to the contents of the folder based on roles, permissions, public groups, and license types. You can make a folder available to your entire organization, or make it private so that only the owner has access.

DASHBOARD:

A dashboard is a visual display of key metrics and trends for records in your org. The relationship between a dashboard component and report is 1:1; for each dashboard component, there is a single source report. ou can use the same report in multiple dashboard components on a single dashboard. Each dashboard has a running user, whose security settings determine which data to display in a dashboard.

A **report type** is like a template that makes reporting easier. The report type determines which fields and records are available for use when creating a report. This is based on the relationships between a primary object and its related objects. For example, with the ‘Contacts & Accounts’ report type, ‘Contacts’ is the primary object and ‘Accounts’ is the related object.

report type’s object relationship:

* **Primary object with related object**—Records returned are only those where the primary object has at least one related object record. In our example of Opportunities with Products, the only records that would be displayed on the report would be opportunities that have at least one related product record.
* **Primary object with or without related object**—Records returned are those where the primary object may or may not have a related object record. If we were to create a custom report type, Opportunities with or without Products, then opportunities would be displayed whether or not they have a related product record.

To create, edit, or review report types, from Setup, enter Report Types in the Quick Findbox, then select **Report Types**.

There are three report formats available: Tabular, Summary, and Matrix. Tabular is the default format.

**Tabular Reports**

Similar to a spreadsheet, they consist simply of an ordered set of fields in columns, with each matching record listed in a row.

SUMMARY REPORTS

Summary reports are similar to tabular reports, but also allow you to group rows of data, view subtotals, and create charts. Summary reports give us many more options for organizing the data, and are great for use in dashboards.

MATRIX REPORTS

Matrix reports allow you to group records both by row and by column. These reports are the most time-consuming to set up, but they also provide the most detailed view of our data.

DASHBOARDS:

With dynamic dashboards, each user sees the data they have access to without needing to create separate dashboards for each user.

Take note of these dynamic dashboard limitations:

* Dynamic dashboards don’t support following components.
* You can’t save dynamic dashboards in private folders.
* You can’t schedule refreshes for dynamic dashboards. They must be refreshed manually.

**DATA SECURITY:**

**Control Access to the Organization : Y**ou do this by managing authorized users, setting password policies, and limiting when and where users can log in.

* **Set Password policy:** You can configure several settings to ensure that your users’ passwords are strong and secure.
* **Password policies**
* **User password expiration**
* **User password resets**
* **Login attempts and lockout periods**
* **Whitelist Trusted IP Ranges for the Org**
* **Restrict Login Access by IP Address Using Profiles**
* **Restrict Login Access by Time**

**Control Access to Objects :** Use profiles to grant the minimum permissions and settings that all users of a particular type need. Then use permission sets to grant more permissions as needed. The combination of profiles and permission sets gives you a great deal of flexibility in specifying object-level access.

You can set object permissions with profiles or permission sets. A user can have one profile and many permission sets.

* A user’s profile determines the objects they can access and the things they can do with any object record (such as create, read, edit, or delete).
* Permission sets grant *additional* permissions and access settings to a user.

**Standard profile:**

* Read Only
* Standard User
* Marketing User
* Contract Manager
* System Administrator

The System Administrator profile also includes two special permissions:

* View All Data
* Modify All Data

**These permissions override all other sharing settings**

**CONTROL FLS:**

Field settings can be applied either by modifying profiles or permission sets or from the Field Accessibility menu in Setup.s

You control record-level access in four ways. They’re listed in order of increasing access. You use org-wide defaults to lock down your data to the most restrictive level, and then use the other record-level security tools to grant access to selected users, as required.

* **Org-wide defaults**specify the default level of access users have to each other’s records.
* **Role hierarchies** ensure managers have access to the same records as their subordinates. Each role in the hierarchy represents a level of data access that a user or group of users needs.
* **Sharing rules** are automatic exceptions to org-wide defaults for particular groups of users, to give them access to records they don’t own or can’t normally see.
* **Manual sharing** lets record owners give read and edit permissions to users who might not have access to the record any other way.

## Org-Wide Sharing

Org-wide defaults specify the baseline level of access that the most restricted user should have. Use org-wide defaults to lock down your data, and then use the other record-level security and sharing tools

**PICKLISTS:**

We have three types of picklists:

1. Standard
2. Custom
3. Custom Multi-Select

And picklist fields can have the following properties:

* Restricted
* Dependent or Controlling

## Restricted Picklists

Restricted picklists keep users from adding new values (either through the API or other apps). This restriction is useful for keeping your data consistent.

## Dependent Picklists

Guide users, save UI space, and further improve data integrity with a dependent picklist. A dependent picklist filters values for one picklist based on a selection from another picklist or a checkbox (the controlling value) on the same record.

s

**GIT hub testings.**