

**PageLayout** : It is a drag and drop tool to edit the layout of object. Layout is where the object's data will be displayed to user's profile. In general,

1. This controls how an object data should be displayed to the profile does.
2. It controls which fields should be displayed.
3. In which order fields should be displayed.
4. In which format fields should be displayed.
5. Which buttons should be displayed on the detail page?
6. It controls the related list and fields in the related list.
7. More than 1 page layout can be created for an object.
8. One profile will have only one page layout on an object.
9. It has either 1 or maximum 2 column layout.
10. It is used to make the field as visible, read only, or required to the user's profile.

#### **Standard Navigation:**

Steps to create PageLayout:

```
Setup
  |--Build
    |--Create
      |--Object
        |--Object Name
          |--PageLayouts
            |--New | Edit
```

Using above navigation, choose any of your custom object (Eg., Training) and do the below use cases to understand the use of page layout.

- 1: Add the Section to the pageLayout
- 2: Choose no Columns and Tab order
- 3: Choose the Fields
- 4: Specify the format of the fields on the Layout (Read | Write | Required)
- 5: Add the buttons to the layout.
- 6: Add the related list to the layout.
- 7: Specify the fields that need to be displayed on related list and then save.

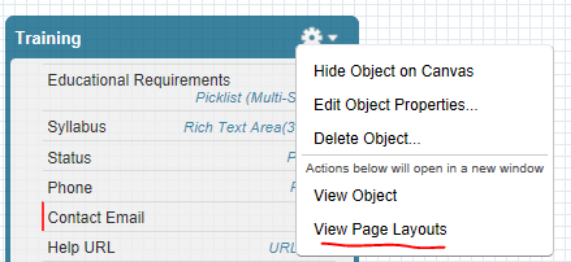
#### **Different Cases of Pagelayout**

1. If a field is universally required, we can NOT remove it from the pagelayout.
2. If a field is marked as read only in Field Level Security, then we cannot mark that field as required in Pagelayout. Reason, FLS here has restricted the edit access for the field.
3. The field is not required in edit page of field and marked as required in pagelayout.  
This case, User can still skip that field value when updates the object data through API (coding).
4. The field is optional in Pagelayout and is not required in edit page but made required through validation rules.  
This case, the field became strictly a required field (similar to universally required).
5. If the field is marked as required in page layout, then this required will be enforced only when user enters record through pagelayout.

Note : We can reassign the visibility of the field at page Layout

Field Level		PageLayout		Final View
Visible	Read	Visible	Read	
NO	NO	----Hidden----		Hidden
OK	NO	NO	NO	Hidden
OK	Ok	OK	NO	Read   Write
Ok	Ok	NO	NO	Hidden
Ok	Ok	OK	OK	Read   Write

Navigation in Schema builder to go page layout tool: Click on gear symbol at top right of training object and select view pagelayouts.



**MINI-Pagelayout :**

1. It is a sub set of pagelayout. It means this will have fields that are part of the main pagelayout.
2. It will appear as pop up when you keep mouse hover on the record name field appears
  - i. in sider bar items.
  - ii. in console app layout
 This popup window is called mini-pagelayout .
3. Every Pagelayout has corresponding mini-page =layout.
4. When we assign the pagelayout to a profile, corresponding mini page layout will be assigned to the Profile

**Navigation to modify Mini-Pagelayout**

Setup -> Build -> Create -> Object-> Object Name -> PageLayouts -> Choose the PageLayout -> Edit

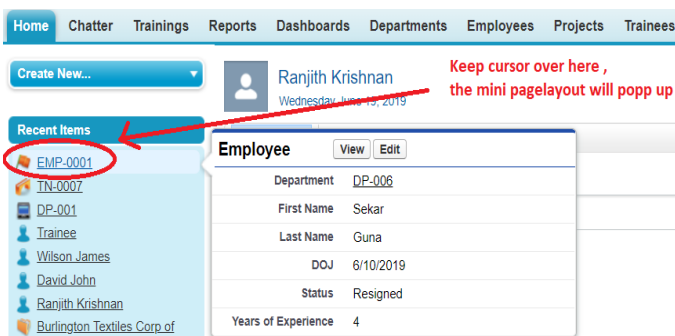
**Step 1.** Choose the Mini Page Layout from menu bar of page layout

**Example:** From the edit page of Page Layout of Employee\_\_c object



**Step 2.** Choose the fields.

**Step 3.** Save the mini page layout and Save the pagelayout. Now verify the same in side bar (click on Home Tab)



**Record Type**

This helps to expose different picklist values and pagelayout to different users.

### Scenarios when do we need record type?

An organization has team of HR people who will create job positions as record in a custom object called Position to recruit candidates for different departments. The object 'Position' has picklist field 'Functional Area' with below values .

#### Functional\_Area\_\_c

Information Tehnology

Sales

Support

Banking

Engineering

BPO

There are two different HR people. One group will work only to technical related job vacancies and other will work for non –technical position. Hence the requirement is to display the values such as Information Tech and Engineering to Technical HRs and other options for Non-Technical HRs for the same picklist field.

This can be implemented using the feature called Record Types.

#### Steps to implement Record Type

Navigation: Setup | Build | Create | Object

Then select the required object (here choose training object)

**Step 1:** Go to Record Type Section and click New button

**Step2:** Enter the name of the record type

**Step 3:** Assign to which profile needs access to this record type.

**Step 4:** Once the record type is created, you will find the entire picklist fields belong to the object would be available under this record type. Here, the Department picklist values will be available to select the options to be displayed.

**Step 5:** Save.

In this scenario, we need to create record types. One can be named as "Technical Position" and another one as "Non-Technical Position". Then customize the picklist field 'Functional Area' accordingly. This will display the record types as below when we click 'New' button under the tab to create the position record as below.

Record Type Name	Description
Non-Technical	Position related to Departments - Sales, Finance and Support
Technical Position	Position related to Departments - IT and Engineering

When record type 'Technical Position' is chosen and click continue, then the picklist list field 'Functional Area' will show only the options Information and Engineering as defined. Similarly for the Non-Technical Record Type.