

Project Design

Team ID	LTVIP2025TMID30907
Project Name	Educational Organization Using ServiceNow

Proposed solution :

S. No	Parameters	Description
1.	Problem Statement	The Educational Management System is a comprehensive digital platform developed to streamline and automate various administrative tasks within educational institutions. It enables efficient and secure management of both student and teacher data, ensuring accuracy and easy accessibility.
2.	Solution Description	This is enables real-time tracking and service automation, allowing educational institutions to respond quickly to requests and manage resources more efficiently.
3.	Uniqueness	The uniqueness of the Educational Organization using ServiceNow project lies in its ability to ensure a secure digital transformation, significantly reducing the administrative burden while enhancing accuracy, operational efficiency, and transparency in academic processes.
4.	Social Impact	The Educational Organization using ServiceNow project creates a meaningful social impact by ensuring equal and timely access to academic resources and services for both students and staff, thereby promoting inclusivity and strengthening support across diverse educational communities.
5.	Business Model	Business model focused on enhancing institutional efficiency and user experience through digital transformation. The model leverages the capabilities of the ServiceNow platform to deliver smart, automated solutions for core educational services.
6.	Scalability Of The Solution	This project is built on a scalable architecture that can easily adapt to the growing needs of institutions of all sizes—from small colleges to large universities or education groups with multiple campuses.

Milestone-1: Setting up Instance

Purpose :

The ServiceNow instance is used to digitize and streamline key administrative processes in an educational organization. It helps manage student admissions, attendance, faculty data, and service requests through automated workflows. This improves efficiency, data accuracy, and collaboration while reducing manual work and paperwork.

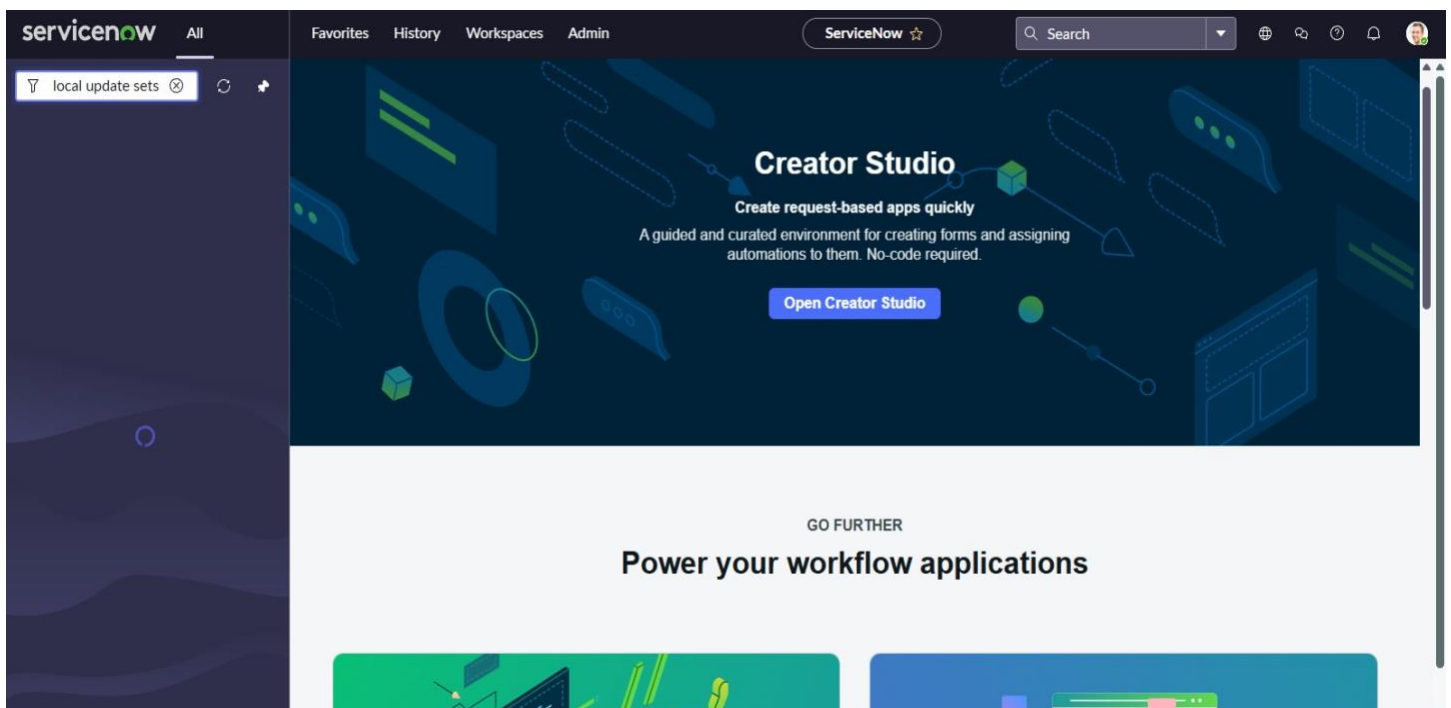
Uses :

The ServiceNow instance is used to automate student admissions, track attendance, manage faculty and student records, handle service requests, and monitor academic progress.

Activity-1: ServiceNow instance

Steps :

1. Sign in to your ServiceNow instance with the given credentials.
2. Go to the ServiceNow Developer site: developer.servicenow.com.
3. Sign up and log in to your developer account.
4. Navigate to Personal Developer Instance.
5. Click Request Instance and submit the required details.
6. Wait for an email with your instance access link



Milestone-2: Update Set

Purpose :

Update set is used to track, manage, and migrate customizations like forms, tables, and workflows related to student admissions, attendance, and academic records. It ensures safe deployment, error-free updates, and helps maintain consistency across development and production environments.

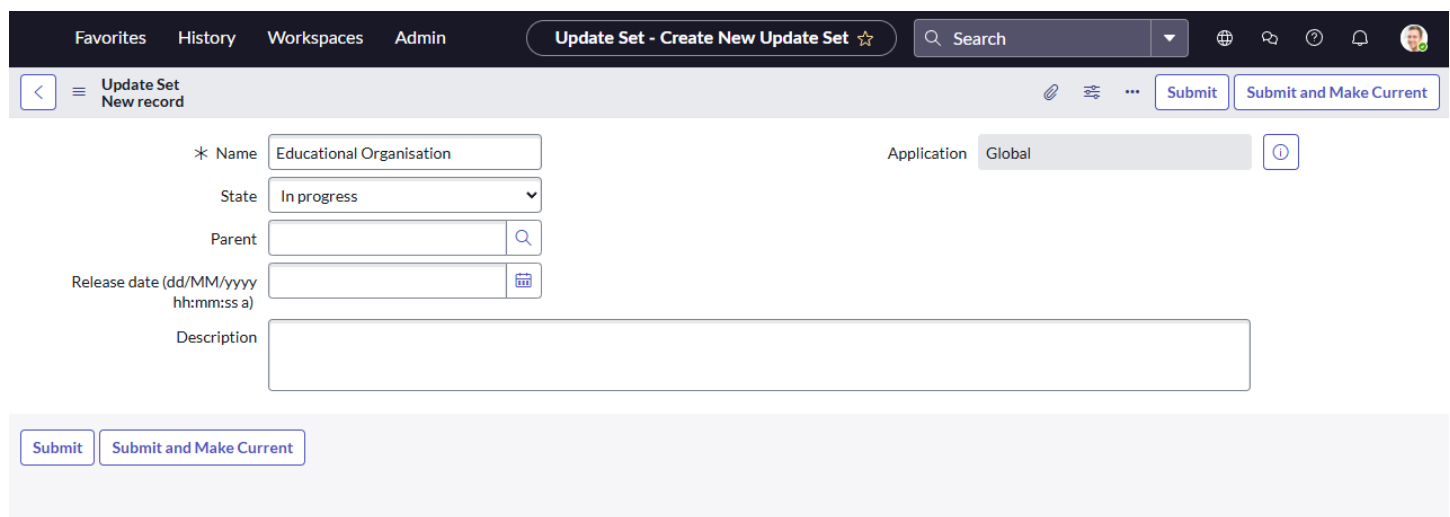
Uses :

Tracks modifications made to applications (like student admission or attendance modules), allowing rollback if needed. Once an update set is created for features like course registration or faculty evaluation, it can be reused or cloned for other departments or campuses.

Activity-1: Create update set

Steps :

1. Click on All >> Local update sets
2. Click on new



The screenshot shows the 'Update Set - Create New Update Set' form in ServiceNow. The top navigation bar includes 'Favorites', 'History', 'Workspaces', and 'Admin'. The main header is 'Update Set - Create New Update Set' with a star icon. Below the header is a search bar and a user profile icon. The form itself has a left sidebar with a menu icon and the text 'Update Set New record'. The main content area contains the following fields: 'Name' (required, with an asterisk) set to 'Educational Organisation', 'Application' set to 'Global', 'State' set to 'In progress', 'Parent' (with a search icon), 'Release date (dd/MM/yyyy hh:mm:ss a)' (with a calendar icon), and 'Description' (a large text area). At the bottom of the form are two buttons: 'Submit' and 'Submit and Make Current'.

3. Enter the Details Name: Educational Organisation
4. Click on Submit and make Current.

Milestone-3: Table

Purpose :

Tables are used to store and organize data such as student records, faculty details, attendance, admissions, and academic progress. They help in centralizing information, enabling easy access, retrieval, and management of educational data within the ServiceNow platform.

Uses :

Store and organize structured data ,Allow easy retrieval and filtering , Enable data relationships across systems , Support report generation , Help in process automation , Maintain data accuracy and integrity


Activity-1: Creating Salesforce Table.

1. All >> Tables.
2. Click on new
3. Enter the Label Salesforce double Click on Name it will Automatically generate Api name.
4. Create columns as given below, Double Click on Column label and Enter the Column labels and click on the tick mark <<Give Type as given .



Admin Date	Date	(empty)	40	false
Admin Number	String	(empty)	40 javascript:getNextObj(NumberPadded)];	true
Father Cell	String	(empty)	40	false
Father Name	String	(empty)	40	false
Grade	Choice	(empty)	40	false
Mother Cell	String	(empty)	40	false
Mother Name	String	(empty)	40	false
Student Name	String	(empty)	40	false

5. Click on controls >> Enable Extensible.
6. Click on “Admin Number” column, In Related Links Click on Advanced View >> Default View (Enable Use dynamic default) >> select Get Next Padded Number in Dynamic default value >> Update .



Choice List Specification Calculated Value **Default Value**

The **Default value** specifies what value the field has when first displayed.

Use dynamic default ☒

Dynamic default value

7. Click on “Grade” Column >> Click on Choices and give Label, Value and Sequence as given below.

Dictionary Entry - Grade

Access Controls Choices Attributes Labels (1)

Label Search

Choices

	Label	Value	Language	Sequence	Inactive	Updated
①	Prept	Prept	en	1	false	24/06 10:33:19 AM
①	Nursery	Nursery	en	2	false	24/06 10:33:49 AM
①	UKG	UKG	en	3	false	24/06 10:34:27 AM
①	I	1st	en	4	false	24/06 10:35:03 AM
①	II	2nd	en	5	false	24/06 10:35:33 AM
①	III	3rd	en	6	false	24/06 10:35:55 AM
①	IV	4th	en	7	false	24/06 10:36:14 AM
①	V	5th	en	8	false	24/06 10:36:36 AM
①	VI	6th	en	9	false	24/06 10:37:01 AM
①	VII	7th	en	10	false	24/06 10:37:25 AM
①	VIII	8th	en	11	false	24/06 10:38:04 AM
①	IX	9th	en	12	false	24/06 10:38:37 AM
①	X	10th	en	13	false	24/06 10:38:59 AM

Insert a new row...

Activity-2: Creating Admission Table

1. Create an Admission Table with Columns given.
2. Select Extends Table >> Salesforce and also Select Add module to menu >> Salesforce.
3. Create Fields as shown
4. Create choice for Admin Status as

Dictionary Entry - Admin Status

Choice Dropdown with -- None --

Create Choice List Delete Column Update

Related Links

Show Table Run Point Scan Advanced view

Access Controls Choices Attributes Labels (1)

Label Search

Choices

	Label	Value	Language	Sequence	Inactive	Updated
①	New	New	en	1	false	24/06 11:26:18 AM
①	Join in progress	In progress	en	2	false	24/06 11:27:07 AM
①	Joined	Joined	en	3	false	24/06 11:27:42 AM
①	Rejected	Rejected	en	4	false	24/06 11:30:13 AM
①	Closed	Closed	en	5	false	24/06 11:30:13 AM
①	Rejoined	Rejoined	en	6	false	24/06 11:32:14 AM
①	Cancelled	Cancelled	en	7	false	24/06 11:32:50 AM

Insert a new row...

5. Create choice for Pincode as

Dictionary Entry - Pincode

Displays a list of suggested values in a Choice list. In the Advanced view you can select the Choice table and the Choice field to take choice values from, plus a Dependent field.

Choice: -- None --

Delete Column Update

Related Links

Show Table
Run Point Scan
Advanced view

Access Controls Choices (2) Attributes Labels (1)

Label Search

Actions on selected rows... New

Label	Value	Language	Sequence	Inactive	Updated
509358	509358	en	1	false	24/06 11:34:58 AM
500079	500079	en	2	false	24/06 11:37:57 AM
500081	500081	en	3	false	24/06 11:40:07 AM
Insert a new row...					

1 to 2 of 2

6. Create choice for purpose of join

Dictionary Entry - Purpose of join

Displays a list of suggested values in a Choice list. In the Advanced view you can select the Choice table and the Choice field to take choice values from, plus a Dependent field.

Choice: -- None --

Delete Column Update

Related Links

Show Table
Run Point Scan
Advanced view

Access Controls Choices (1) Attributes Labels (1)

Label Search

Actions on selected rows... New

Label	Value	Language	Sequence	Inactive	Updated
Tuition	Tuition	en	1	false	24/06 11:45:13 AM
Coaching	Coaching	en	2	false	24/06 11:50:57 AM
Teacher	Teacher	en	3	false	24/06 11:51:21 AM
Insert a new row...					

1 to 1 of 1

7. Create choice for school

<

≡

Dictionary Entry
School

✎ ✕ ⚙ ⋮

Delete Column

Update

↑

↓

* Max length

40

Mandatory ☐

Display ☐

Choice List Specification

Default Value

Displays a list of suggested values in a Choice list. In the Advanced view you can select the Choice table and the Choice field to take choice values from, plus a Dependent field.

Choice

-- None --

Delete Column

Update

Related Links

[Show Table](#)

[Run Point Scan](#)

[Advanced view](#)

Access Controls

Choices

Attributes

Labels (1)

≡

🔍

Label

Search

⚙

—

New

Choices

	Label ▲	Value	Language	Sequence	Inactive	Updated
①	Stanley	Stanley	en	1	false	24/06 12:05:03 PM
①	Naresh it	Naresh it	en	2	false	24/06 12:06:03 PM
+	Insert a new row...					

8. Create choice for school area

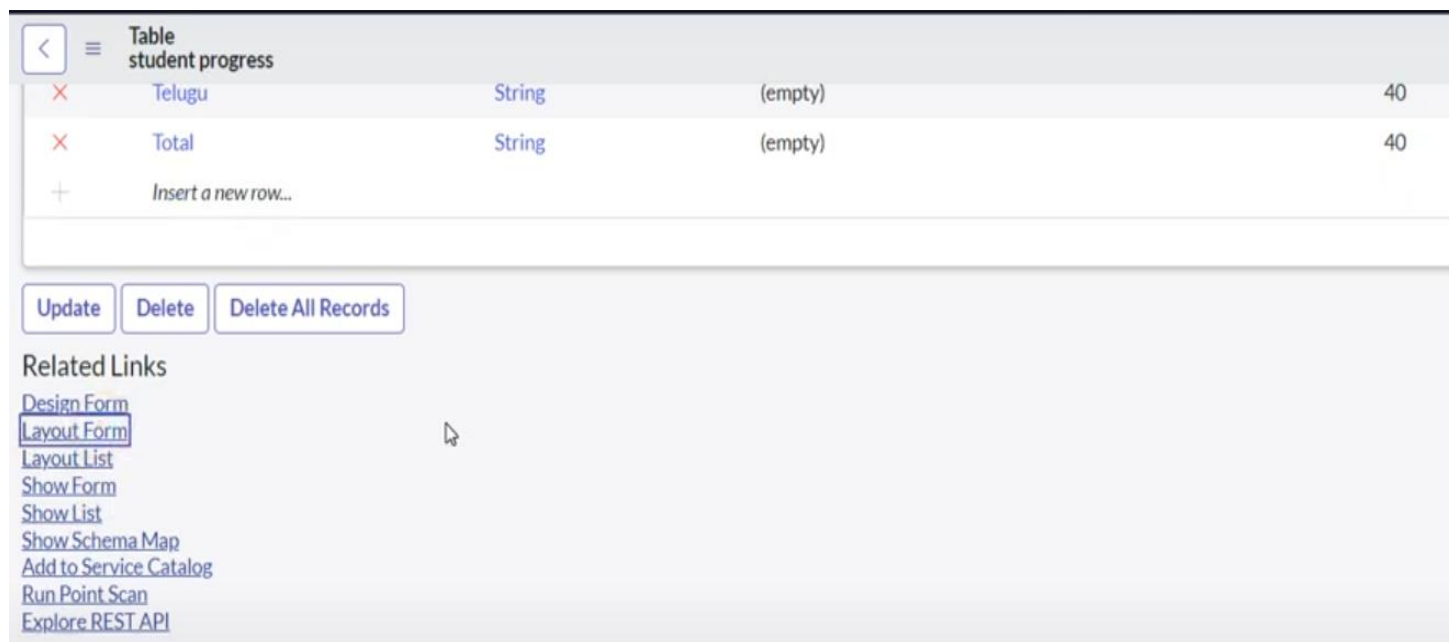
Form Layout is used to organize and display fields in a structured way on forms. It helps educational institutions separate student and faculty data, show important details clearly, and improve data entry and user experience.

Uses :

Organizes form fields for better clarity , Separates student, faculty, and admin data sections , Highlights mandatory fields for accurate data entry , Improves user experience and efficiency, Supports customized views for different users.

Activity-1: Table form for Student Progress Table

1. In the Student Progress Table Page , Click on Layout form.



The screenshot displays a web interface for a 'Table student progress'. At the top, there is a header bar with a back arrow, a menu icon, and the title 'Table student progress'. Below this is a table with two rows. The first row has columns for 'Telugu', 'String', '(empty)', and '40'. The second row has columns for 'Total', 'String', '(empty)', and '40'. Below the table is a button with a plus sign and the text 'Insert a new row...'. Underneath the table are three buttons: 'Update', 'Delete', and 'Delete All Records'. At the bottom, there is a 'Related Links' section with several links: 'Design Form', 'Layout Form', 'Layout List', 'Show Form', 'Show List', 'Show Schema Map', 'Add to Service Catalog', 'Run Point Scan', and 'Explore REST API'.

2. Click on Admission Number [+].
3. Select below Admission Number fields in Available side and send it to selected side as below >> save.

Milestone-5: Form Design

Purpose :

Form Design is used to customize the appearance and structure of forms to suit the institution’s needs. It helps in displaying relevant fields, improving data accuracy, and creating user-friendly interfaces for managing student, faculty, and academic records.

Activity-1: Form design for salesforce table

1. All >> System Definition >> Tables .
2. In Label Search for Salesforce and open .
3. Right Click on top Toggle >> Configure >> Form Design.
4. In drop down select Salesforce(u_salesforce)
5. Drag and drop the fields to the left side as below.

The screenshot shows the 'Form Design' window for the 'Salesforce [u_salesforce]' table. The interface includes a top bar with the table name and a 'Default view' dropdown. Below this, there are tabs for 'Fields' and 'Field Types'. The 'Fields' tab is active, showing a list of fields on the left and a design canvas on the right. The design canvas displays a table with two columns and six rows of fields. The fields are: Admin Number, Admin Date, Grade, Student Name, Father Name, Mother Name, Father Cell, and Mother Cell. Each field has a small icon to its right, likely for configuration or deletion.

Activity-2: Form Design for Admission Table

1. Follow the same steps as Activity1,Configure the fields as below and Save.

The screenshot shows the 'Form Design' window for the 'Admission [u_admission]' table. The interface includes a top bar with the table name and a 'Default view' dropdown. Below this, there are tabs for 'Fields' and 'Field Types'. The 'Fields' tab is active, showing a list of fields on the left and a design canvas on the right. The design canvas displays a form with multiple sections. The sections are: Admission Number, Admission Date, Purpose of join, Grade, Fee, Student Name, Father Name, Mother Name, Father Cell, Mother Cell, Admin Status, Comments, School Details, School Area, School, Address, Pincode, Area, Mandal, City, District, and House No. Each field has a small icon to its right, likely for configuration or deletion.

Activity-3: Form Design for Student progress Table

1. Follow the same steps as Activity1,Configure the fields as below and Save.

The screenshot shows the 'Form Design' interface for a table named 'Student Progress'. The interface is divided into three main sections: 'Fields', 'Formatters', and 'Table Structure'.

- Fields:** A list of fields on the left includes 'Class', 'Created', 'Created by', 'Social', 'Updated', 'Updated by', and 'Updates'. A 'Filter' button is also present.
- Formatters:** A list of formatters on the left includes 'Activities (filtered)', 'Contextual Search Results', and 'Ratings'.
- Table Structure:** The main area shows the table structure with three sections:
 - New Section:** Contains one field: 'Admission Number'.
 - Admission Number:** A section with two columns. The first column contains 'Admission Number Grade' and 'Admission Number Student Name'. The second column contains 'Admission Number Father Name', 'Admission Number Mother Name', 'Admission Number Father Cell', and 'Admission Number Mother Cell'.
 - Student Progress:** A section with two columns. The first column contains 'Telugu', 'Hindi', 'English', 'Maths', and 'Science'. The second column contains 'Total', 'Percentage', and 'Result'.

Milestone-6: Number Maintenance

Purpose :

Number Maintenance is used to automatically generate unique identification numbers for records like student IDs, admission forms, and attendance entries. It ensures consistency, avoids duplication, and helps in easily tracking and managing records.

Uses :

Generates unique IDs for students, staff, and records, Prevents duplication of entries, Helps in tracking records efficiently, Maintains consistency across forms and module, Supports automated record creation for processes like admissions and attendance

Activity-1: Number Maintenance for Admin Number

1. All >> Number Maintenance >> New

2. Fill the details >> Submit

The screenshot shows the Salesforce 'Number' configuration page for a table named 'Salesforce'. The interface includes a top navigation bar with 'Favorites', 'History', 'Workspaces', and 'Admin'. Below this is a search bar and a 'Number - SAL' header. The main form contains the following fields:

- * Table: Salesforce
- Prefix: SAL
- * Number: 1,000
- Application: Global
- Number of digits: 7

At the bottom of the form, there are 'Update' and 'Delete' buttons. Below the form, there is a 'Related Links' section with a link to 'Show Counter'.

Milestone-7 : Process Flow

Activity-1 : Create flow

1. All >> Process Flow>> New
2. Fill the Details as given Below
3. Right Click on toggle and click on the save .
4. Replace the Name and Label as below and click on Insert on stay.
5. Replace the Name and Label in order and click on Insert on stay. Joined >> Rejected >> Rejoined >> Closed >> Cancelled.
6. Order should be New >> InProgress >> Joined >> Rejected >> Rejoined >> Closed >> Cancelled.

Milestone-8: Client Scripts

Purpose :

Client Scripts are used to run custom code on the client side (browser) to enhance form behavior. They help in validating data, auto-filling fields, and improving user interaction in real-time without needing a server request.

Uses :

Validate form data before submission, Show/hide fields dynamically , Auto-fill or modify fields based on user input , Improve user experience with real-time responses , Reduce server load by handling logic on the client side

Activity-1: Auto populate Client Scripts for Admission Table

1. All >> Client Scripts >> New.
2. Fill the Details as given
3. Write the Code as given, Enable Isolate script and Save.

```
Function onChange(control, oldValue, newValue, isLoading, isTemplate) {  
  
    If (isLoading || newValue === '') {  
  
        Return;  
  
    }  
  
    //Type appropriate comment here, and begin script below  
  
    Var a = g_form.getReference('u_admission_number');  
  
    G_form.setValue('u_admin_date',a.u_admin_date);  
  
    G_form.setValue('u_grade',a.u_grade);  
  
    G_form.setValue('u_student_name',a.u_student_name);  
  
    G_form.setValue('u_father_name',a.u_father_name);  
  
    G_form.setValue('u_mother_name',a.u_mother_name);
```

Client Script - Auto populate

Search

UpdateDelete

New client-scripts are run in strict mode, with direct DOM access disabled. Access to jQuery, prototype and the window object are likewise disabled. To disable this on a per-script basis, configure this form and add the "Isolate script" field. To disable this feature for all new globally-scoped client-side scripts set the system property "glide.script.block.client_globals" to false.

Name

Auto populate

Table

Admission [u_admission]

UI Type

Mobile / Service Portal

Type

onChange

Field name

Admin Number

Application

Global

Active

☒

Inherited

☐

Global

☒

Description

Messages

Script

```
14
15
16
17
18 //Type appropriate comment here, and begin script below
19
20 var a = g_form.getReference('u_admission_number');
21
22 g_form.setValue('u_admin_date',a.u_admin_date);
23
24 g_form.setValue('u_grade',a.u_grade);
25
```

Activity-2: Pincode Update Client Scripts for Admission Table

1. Fill the Details as given.
2. Write the Code as given, Enable Isolate script and Save.

```
Function onChange(control, oldValue, newValue, isLoading, isTemplate) {  
    If (isLoading || newValue === '') {  
        Return ;  
    }  
    Var a = g_form.getValue('u_pincode');  
    If(a == '509358')  
    {  
        G_form.setValue('u_mandal', 'kadthal');  
        G_form.setValue('u_city', 'kadthal')  
        G_form.setValue('u_district', 'RangaReddy');  
    }  
    Else if(a == '500081')  
    {  
        G_form.setValue('u_mandal', 'karmanghat');  
        G_form.setValue('u_city', 'karmanghat');  
        G_form.setValue('u_district', 'RangaReddy');  
    }  
    Else if(a == '500079')
```

```

{
G_form.setValue('u_mandal', 'Abids');

G_form.setValue('u_city', 'AsifNagar');

G_form.setValue('u_district', 'Hyderabad');

}

//Type appropriate comment here, and begin script below

}

```

Client Script - Pincode Update

New client-scripts are run in strict mode, with direct DOM access disabled. Access to jQuery, prototype and the window object are likewise disabled. To disable this on a per-script basis, configure this form and add the "Isolate script" field. To disable this feature for all new globally-scoped client-side scripts set the system property "glide.script.block.client.globals" to false.

Name	Pincode Update	Application	Global
Table	Admission [u_admission]	Active	<input checked="" type="checkbox"/>
UI Type	Desktop	Inherited	<input type="checkbox"/>
Type	onChange	Global	<input checked="" type="checkbox"/>
Field name	Pincode		
Description			
Messages			

Script

```

17   var a = g_form.getValue('u_pincode');
18
19   if(a == '509358')
20
21   {
22
23       g_form.setValue('u_mandal', 'kadthal');
24       g_form.setValue('u_city', 'kadthal');
25       g_form.setValue('u_district', 'RangaReddy');
26
27   }
28

```

Activity-3 :Disable Fields Client Scripts for Student progress Table

1. Fill the Details as given.

- Write the Code as given, Enable Isolate script and Save.

```
Function onLoad() {  
  
    //Type appropriate comment here, and begin script below  
  
    G_form.setDisabled('u_total',true);  
  
    G_form.setDisabled('u_percentage',true);  
  
    G_form.setDisabled('u_result',true);  
  
}
```

Client Script - Disable Fields

New client-scripts are run in strict mode, with direct DOM access disabled. Access to jQuery, prototype and the window object are likewise disabled. To disable this on a per-script basis, configure this form and add the "Isolate script" field. To disable this feature for all new globally-scoped client-side scripts set the system property "glide.script.block.client.globals" to false.

Name: Disable Fields

Table: Student Progress [u_student_pr...]

UI Type: All

Type: onLoad

Application: Global

Active: ☒

Inherited: ☐

Global: ☒

Description:

Messages:

Script:

```
1 function onLoad() {  
2     //Type appropriate comment here, and begin script below  
3  
4 }  
5 function onLoad() {  
6  
7     //Type appropriate comment here, and begin script below  
8  
9     g_form.setDisabled('u_total',true);  
10  
11     g_form.setDisabled('u_percentage',true);  
12
```

Activity-4: Total Update Client Scripts for Student progress Table

- Fill the Details as given.
- Write the Code as given, Enable Isolate script and Save.

```
Function onChange(control, oldValue, newValue, isLoading, isTemplate) {
```

```
If (isLoading || newValue === '') {  
  
    Return;  
  
}  
  
//Type appropriate comment here, and begin script below  
  
If (newValue){  
  
    Var a = parseInt(g_form.getValue('u_telugu'));  
  
    Var b = parseInt(g_form.getValue('u_hindi'));  
  
    Var c = parseInt(g_form.getValue('u_english'));  
  
    Var d = parseInt(g_form.getValue('u_maths'));  
  
    Var e = parseInt(g_form.getValue('u_science'));  
  
    Var f = parseInt(g_form.getValue('u_social'));  
  
    Var Total = parseInt(a+b+c+d+e+f);  
  
    G_form.setValue('u_total', Total);  
  
}  
  
}
```

Client Script - Total Update

Search

Update Delete

New client-scripts are run in strict mode, with direct DOM access disabled. Access to jQuery, prototype and the window object are likewise disabled. To disable this on a per-script basis, configure this form and add the "Isolate script" field. To disable this feature for all new globally-scoped client-side scripts set the system property "glide.script.block.client.globals" to false.

Name: Total Update

Application: Global

Table: Student Progress [u_student_pr...

UI Type: All

Type: onChange

Field name: Social

Active: ☒

Inherited: ☐

Global: ☒

Description:

Messages:

Script

```

1 function onChange(control, oldValue, newValue, isLoading, isTemplate) {
2     if (isLoading || newValue === '') {
3         return;
4     }
5
6     //Type appropriate comment here, and begin script below
7
8 }
9 function onChange(control, oldValue, newValue, isLoading, isTemplate) {
10
11     if (isLoading || newValue === '') {
12

```

Activity-5: Result Client Scripts for Student progress

Table

1. Fill the Details as given.
2. Write the Code as given, Enable Isolate script and Save.

```

function onChange(control, oldValue, newValue, isLoading, isTemplate) {
    if (isLoading || newValue === '') {
        return;
    }
    //Type appropriate comment here, and begin script below
    if(newValue) {
        var a = parseInt(g_form.getValue('u_percentage')); // Convert the value to
an integer          for comparison
        if(a >= 0 && a <= 59){
            g_form.setValue('u_result','Fail');

```

```
} else if(a >= 60 && a <= 100) {
g_form.setValue('u_result','Pass');
} else {
// Handle the case if a is out of range (optional)
g_form.addErrorMessage('Percentage should be between 0 and 100.');
```

Favorites

History

Workspaces

Admin

Client Script - Result ☆

Search

Update

Delete

New client-scripts are run in strict mode, with direct DOM access disabled. Access to jQuery, prototype and the window object are likewise disabled. To disable this on a per-script basis, configure this form and add the "Isolate script" field. To disable this feature for all new globally-scoped client-side scripts set the system property "glide.script.block.client.globals" to false.

NameResult

ApplicationGlobal ⓘ

TableStudent Progress [u_student_pr... ▼

Active☒

UI TypeAll ▼

Inherited☐

TypeonChange ▼

Global☒

Field namePercentage ▼

Description

Messages

Script

```
9 function onChange(control, oldValue, newValue, isLoading, isTemplate) {  
10  
11   if (isLoading || newValue === '') {  
12     return;  
13   }  
14  
15 }  
16  
17 //Type appropriate comment here, and begin script below  
18  
19 if(newValue) {
```

Activity-6: Percentage Client Scripts for Student progress Table

1. Fill the Details as given.
2. Write the Code as given, Enable Isolate script and Save.

```
Function onChange(control, oldValue, newValue, isLoading, isTemplate) {  
    If (isLoading || newValue === '') {  
  
        Return;  
  
    }  
    //Type appropriate comment here, and begin script below  
    Var Total = g_form.getValue('u_total');  
    Var Percentage = (Total/600)*100;  
    G_form.setValue('u_percentage',Percentage+'%');  
  
}
```

The screenshot shows the Salesforce Client Script configuration page. The script is named 'Percentage' and is associated with the 'Student Progress' table. The configuration includes the following details:

- Name:** Percentage
- Table:** Student Progress [u_student_pr...]
- UI Type:** All
- Type:** onChange
- Field name:** Total
- Application:** Global
- Active:** ☒
- Inherited:** ☐
- Global:** ☒

The Description and Messages fields are empty. The Script section contains the following code:

```
17  
18 //Type appropriate comment here, and begin script below  
19  
20 var Total = g_form.getValue('u_total');  
21  
22 var Percentage = (Total/600)*100;  
23  
24 g_form.setValue('u_percentage',Percentage+'%');  
25  
26 }  
27  
28
```