

Project Design

Team ID	LTVIP2026TMIDS36010
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 solution 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.

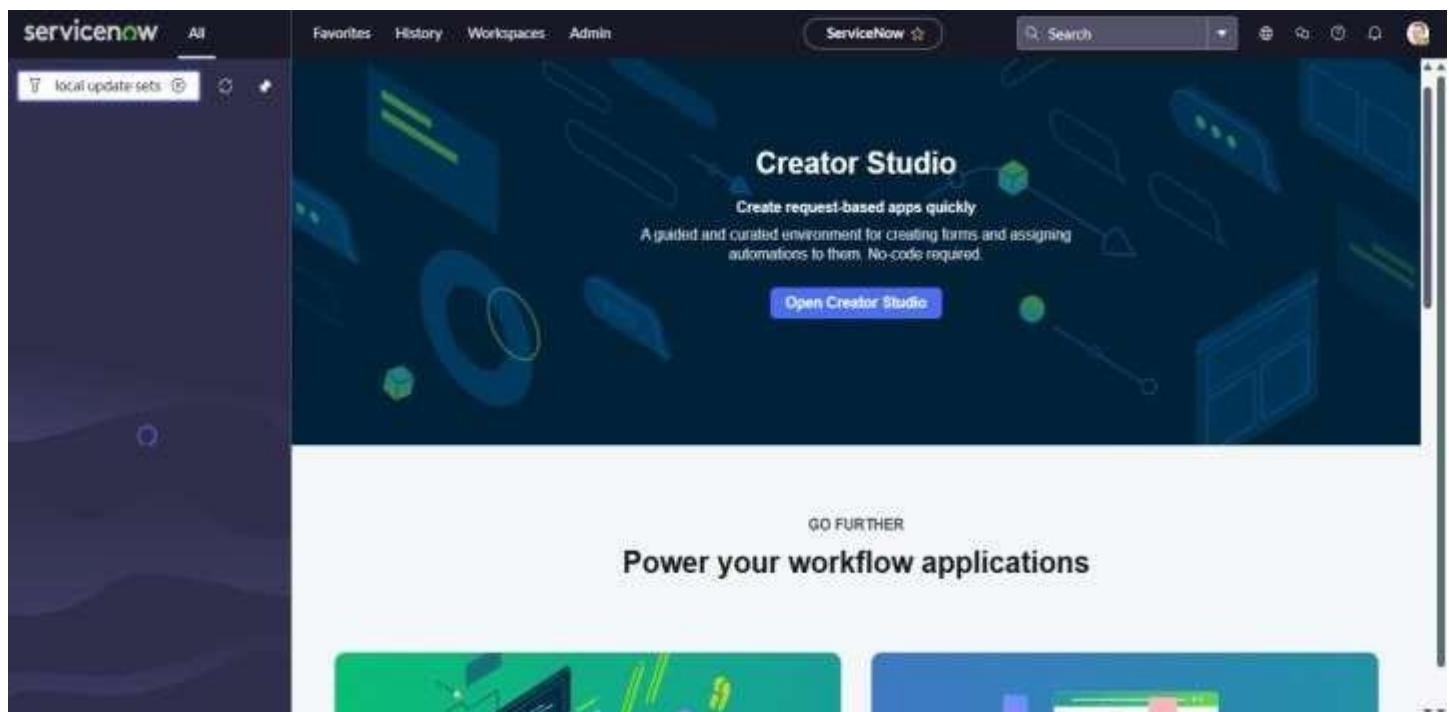
Uses :

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 a software interface for creating a new update set. At the top, there's a navigation bar with 'Favorites', 'History', 'Workspaces', and 'Admin'. The title bar says 'Update Set - Create New Update Set'. Below the title bar, there's a breadcrumb trail 'Update Set' and a link 'New record'. On the right side of the title bar are several icons for search, refresh, and user profile. The main form area has the following fields:

- * Name: Educational Organisation
- State: In progress
- Parent: (empty)
- Release date (dd/MM/yyyy hh:mm:ss): (empty)
- Description: (empty)

At the bottom of the form, there are two buttons: 'Submit' and 'Submit and Make Current'. The entire interface has a clean, modern design with a light blue header and white background for the form area.

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 .

Label	Type	API Name	Default Value	Extensible
Admin Date	Date	(empty)	40	false
Admin Number	String	(empty)	40 [javascript:getNextObjNumberPadded();]	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 .

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

Use dynamic default

Dynamic default value: Get Next Padded Number

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

The screenshot shows a table titled "Choices" under the "Dictionary Entry - Grade" section. The table has columns: Label, Value, Language, Sequence, Inactive, and Updated. The data includes various grade levels from Prept to X, each with a unique sequence number, language set to en, and updated at different times between 24/06/10:33:19 AM and 24/06/10:38:59 AM.

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

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

The screenshot shows a table titled "Choices" under the "Dictionary Entry - Admin Status" section. The table has columns: Label, Value, Language, Sequence, Inactive, and Updated. The data includes various status levels from New to Cancelled, each with a unique sequence number, language set to en, and updated at different times between 24/06/11:26:18 AM and 24/06/11:32:50 AM.

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

5. Create choice for Pincode as

Favorites History Workspaces Admin Dictionary Entry - Pincode Search Delete Column Update

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 (3) Attributes Labels (1)

	Label	Value	Language	Sequence	Inactive	Updated
1	S09358	S09358	en	1	false	24/06/11:34:56 AM
2	S00079	S00079	en	2	false	24/06/11:37:57 AM
3	S00081	S00081	en	3	false	24/06/11:40:07 AM

Insert a new row...

1 to 3 of 3

New

This screenshot shows the 'Dictionary Entry - Pincode' page. It displays a table of choices for pincode, with columns for Label, Value, Language, Sequence, Inactive status, and Updated timestamp. The sequence is set to 1, 2, and 3 respectively. The language is en for all entries. The updated times are 24/06/11:34:56 AM, 24/06/11:37:57 AM, and 24/06/11:40:07 AM. There are buttons for 'Delete Column', 'Update', and a 'New' button at the top right of the table.

6. Create choice for purpose of join

Favorites History Workspaces Admin Dictionary Entry - Purpose of join Search Delete Column Update

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 (3) Attributes Labels (1)

	Label	Value	Language	Sequence	Inactive	Updated
1	Tutor	Tutor	en	1	false	24/06/11:45:13 AM
2	Coaching	Coaching	en	2	false	24/06/11:50:57 AM
3	Teacher	Teacher	en	3	false	24/06/11:51:21 AM

Insert a new row...

1 to 1 of 1

New

This screenshot shows the 'Dictionary Entry - Purpose of join' page. It displays a table of choices for purpose of join, with columns for Label, Value, Language, Sequence, Inactive status, and Updated timestamp. The sequence is set to 1, 2, and 3 respectively. The language is en for all entries. The updated times are 24/06/11:45:13 AM, 24/06/11:50:57 AM, and 24/06/11:51:21 AM. There are buttons for 'Delete Column', 'Update', and a 'New' button at the top right of the table.

7. Create choice for school as

Dictionary Entry School

Max length: 50

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)

Choices						
	Label	Value	Language	Sequence	Inactive	Updated
1	Stanley	Stanley	en	1	false	24/06 12:05:03 PM
2	Naresh It	Naresh It	en	2	false	24/06 12:06:03 PM

Insert a new row...

8. Create choice for school area as

Dictionary Entry - School Area

Search

Dictionary Entry School Area

Delete Column Update

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 (2) Attributes Labels (1)

Choices						
	Label	Value	Language	Sequence	Inactive	Updated
1	Near Bus Stand	Near Bus Stand	en	2	false	24/06 12:21:33 PM
2	Near Market	Near Market	en	1	false	24/06 12:20:09 PM

Actions on selected rows... New

Activity-3: Creating Student Progress Table

1. Create a Student Progress Table with Columns given.
2. Select Add module to menu >> Salesforce.
3. Create Fields as show

	AttributeNumber	Reference	Sanctions	32	false
X	English	String	(empty)	40	false
X	Hindi	String	(empty)	40	false
X	Maths	String	(empty)	40	false
X	Percentage	String	(empty)	40	false
X	Result	String	(empty)	40	false
X	Science	String	(empty)	40	false
X	Social	String	(empty)	40	false
X	Telugu	String	(empty)	40	false
X	Total	String	(empty)	40	false
Insert a new record...					

Milestone-4: Form Layout

Purpose:

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.

Telugu	String	(empty)	40
Total	String	(empty)	40
Insert a new row...			

[Update](#) [Delete](#) [Delete All Records](#)

Related Links

- [Design Form](#)
- [Layout Form](#)
- [Layout List](#)
- [Show Form](#)
- [Show List](#)
- [Show Schema Map](#)
- [Add to Service Catalog](#)
- [Run Point Scan](#)
- [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.

Form Design

Fields

Fields

Formulas

Activities (Related)

Content Search Results

Ratings

Student Number

Admin Date

Grade

Student Name

Father Name

Mother Name

Mobile No.

Student ID

Activity-2: Form Design for Admission Table

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

Form Design

Fields

Admission Number

Student Name

Father Name

Mother Name

Address

Contact

Activity-3: Form Design for Student progress Table

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

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 interface. At the top, there are navigation links for Favorites, History, Workspaces, and Admin. The title bar says "Number - SAL". Below the title, there's a search bar and various toolbar icons. The main form has fields for "Table" (set to "Salesforce"), "Prefix" (set to "SAL"), "Number" (set to "1,000"), "Application" (set to "Global"), and "Number of digits" (set to "7"). At the bottom left of the form area, there are "Update" and "Delete" buttons. On the far left, under "Related Links", there are "Show Counter" and other links.

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);  
G_form.setValue('u_father_cell',a.u_father_cell);  
G_form.setValue('u_mother_cell',a.u_mother_cell);
```

```

G_form.setDisabled('u_admin_date',a.u_admin_date);

G_form.setDisabled('u_grade',a.u_grade);

G_form.setDisabled('u_student_name',a.u_student_name);

G_form.setDisabled('u_father_name',a.u_father_name);

G_form.setDisabled('u_mother_name',a.u_mother_name);

G_form.setDisabled('u_father_cell',a.u_father_cell);

G_form.setDisabled('u_mother_cell',a.u_mother_cell);

}

```

The screenshot shows the configuration page for a 'Client Script - Auto populate' entry. The top navigation bar includes 'Favorites', 'History', 'Workspaces', 'Admin', and a search bar. The main area displays the script code and various configuration options.

Script Configuration:

- Name:** Auto populate
- Table:** Admission [u_admission]
- UI Type:** Mobile / Service Portal
- Type:** onChange
- Field name:** Admin Number
- Application:** Global
- Active:**
- Inherited:**
- Global:**

Description: (Empty)

Messages: (Empty)

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  
}
```

The screenshot shows the configuration for a client script named "Pincode Update". The script is set to run on the "Admission [u_admission]" table, using the "onChange" type on the "Pincode" field. It is marked as "Active" and "Global". The "Description" and "Messages" sections are empty. The "Script" editor contains the following code:

```
17 var a = g_form.getValue('u_pincode');
18
19 if(a == '509958'){
20
21
22 g_form.setValue('u_mandal', 'kadthal');
23
24 g_form.setValue('u_city', 'kadthal');
25
26 g_form.setValue('u_district', 'Rangareddy');
```

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

1. Fill the Details as given.
 2. 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);  
}
```

Activity-4: Total Update 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_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);

}

}

```

The screenshot shows the configuration of a client script named "Total Update". The script is set to run on the "Student Progress" table, specifically on the "Social" field, using the "onChange" type. It is marked as "Global" and "Active". The script code is as follows:

```

function onChange(control, oldValue, newValue, isLoading, isTemplate) {
    if (isLoading || newValue === '') {
        return;
    }
    //Type appropriate comment here, and begin script below
}
function onChange(control, oldValue, newValue, isLoading, isTemplate) {
    if (isLoading || newValue === '') {

```

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.');//  
g_form.clearValue('u_result');  
}  
}  
}  
}
```

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: Result
Table: Student Progress [u_student_pr...]
UI Type: All
Type: onChange
Field name: Percentage

Description:

Messages:

Script:

```
0  Function onChange(control, oldValue, newValue, isLoading, isTemplate) {  
1  
11:  If (isLoading || newValue === '') {  
12:  
13:    Return;  
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+'%');
```

```
}
```

Favorites History Workspaces Admin Client Script - Percentage Search

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: Percentage Application: Global

Table: Student Progress [u_student_pr... UI Type: All Active:

Type: onChange Inherited:

Field name: Total Global:

Description:

Messages:

Script:

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