Educational Organisation Using ServiceNow

Team ID : LTVIP2026TMIDS24916

Team Size : 1

Team Leader : MAHIMA JYOTHI A

Team member : MAHIMA JYOTHI A

AIM:

The aim of this project is to **develop an Educational Management System using ServiceNow** that automates and streamlines the administrative processes of educational institutions by managing student and teacher information, simplifying admissions, and providing tools for monitoring academic progress.

ABSTRACT:

The project **“Educational Organisation Using ServiceNow”** focuses on developing an automated Educational Management System on the ServiceNow platform. The system is designed to simplify and streamline various administrative tasks within educational institutions, such as managing student and teacher data, handling admissions, maintaining academic records, and tracking student performance. By using ServiceNow’s powerful tools like table creation, form design, client scripts, and process automation, the project demonstrates how non-IT processes can be efficiently managed on a cloudbased platform. This application enhances data accuracy, reduces manual effort, and improves overall institutional efficiency.

OBJECTIVE:

- The main objective of this project is to develop an Educational Management System using ServiceNow that streamlines the administrative operations of educational institutions.
- It enables efficient management of student and teacher information, simplifies the admission process, automates workflows, and provides tools for monitoring student progress.

METHODOLOGY:

The development of the **Educational Organisation Using ServiceNow** project followed a systematic process using the features and tools provided by the ServiceNow platform. The methodology involves the following steps:

1. **Setting Up ServiceNow Instance**

   - A personal developer instance of ServiceNow was created from the ServiceNow Developer Portal.
   - The instance was configured with admin privileges to allow application creation and customization.

2. **Creating an Update Set**

- An **Update Set** was created to capture all the configurations, scripts, and changes made during development.
- This helps in migrating the developed components between different instances if required.

3. **Creating a Table**

- A custom table named **Educational Organisation** (or similar) was created to store student and teacher information.
- Fields such as *Student ID*, *Student Name*, *Department*, *Course*, *Admission Date*, and *Grade* were added.
- Relationships between related tables were established to manage linked data.

4. **Designing the Form Layout and Form Design**

- The **Form Layout** was organized for a user-friendly interface.
- The **Form Designer** was used to add and arrange fields logically for better data entry and viewing.
- Sections and tabs were customized for clarity and usability.

5. **Number Maintenance Configuration**

- Automatic numbering was configured using **Number Maintenance** to generate unique record IDs for each entry (e.g., STU0001, STU0002).
- This ensures proper record tracking and consistency.

6. **Process Flow Creation**

- Using **Flow Designer**, workflows were created to automate processes like admission approval, grade updates, and status tracking.
- Notifications and task assignments were added to enhance communication and efficiency.

7. **Client Script Implementation**

- **Client Scripts** were developed using JavaScript to add validations, dynamic field behaviors, and form interactivity.
- Example: Auto-filling fields, restricting invalid data entries, and displaying alerts for missing information.

8. **Testing and Result Verification**

- ➢ The developed application was tested by creating, updating, and viewing student records.
- ➢ Automated workflows and scripts were verified for accuracy and functionality.
- ➢ The final output was an efficient, user-friendly system for educational data management.

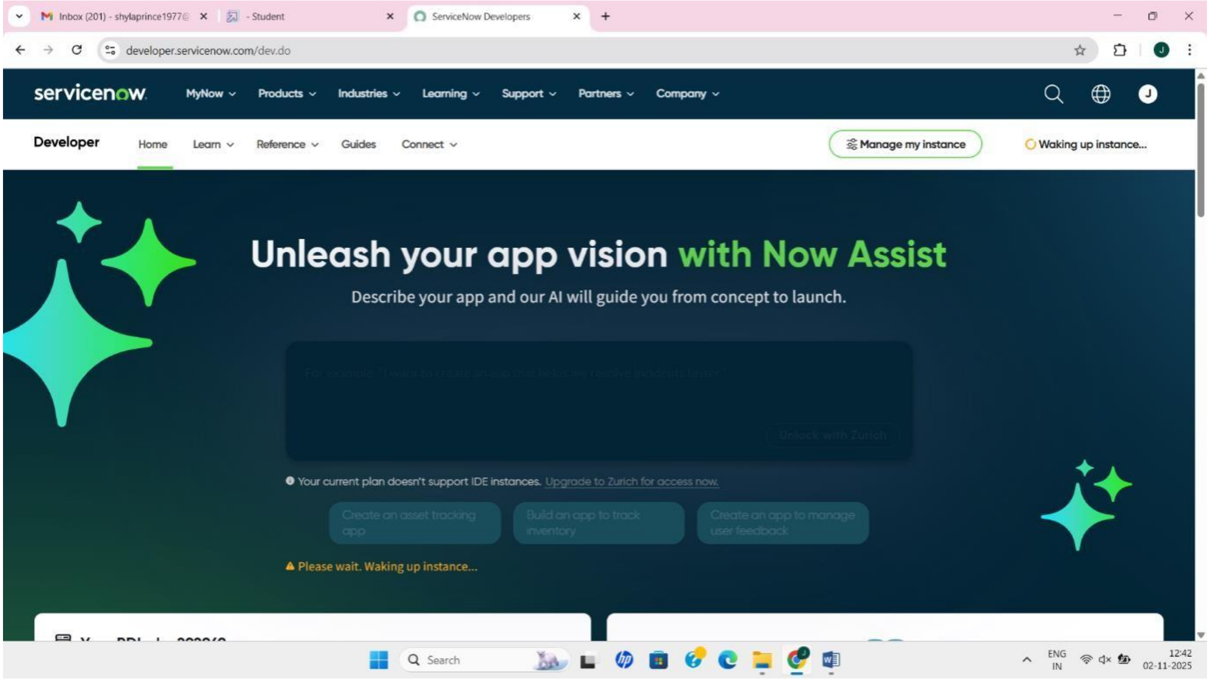
TOOLS AND TECHNOLOGIES USED:

- ❖ Service Now Platform
- ❖ Flow Designer
- ❖ Form Designer
- ❖ Client Scripts (JavaScript)
- ❖ Update Set Management

STEPS TO IMPLEMENT:

Step 1

Set up a service now instance



developer.servicenow.com/dev.do

Unleash your app vision with Now Assist

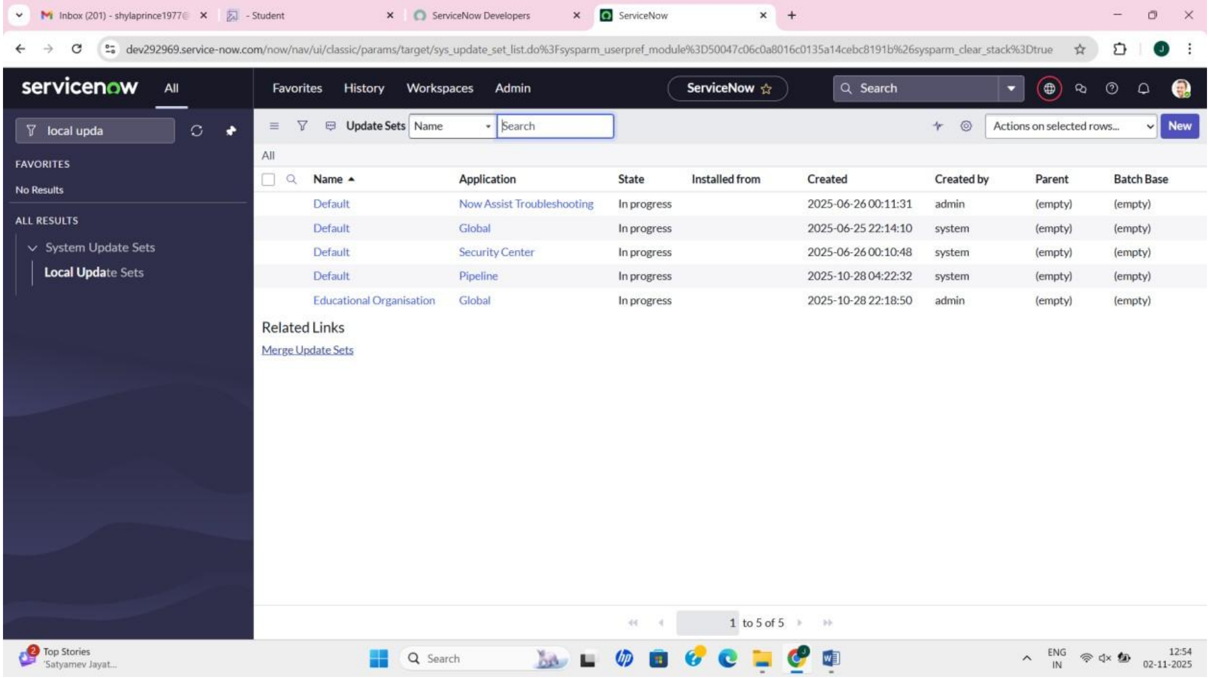
Describe your app and our AI will guide you from concept to launch.

Your current plan doesn't support IDE instances. Upgrade to Zurich for access now.

⚠ Please wait. Waking up instance...

Step 2

Create a update set on the local update set



dev292969.service-now.com/now/nav/ui/classic/params/target/sys_update_set_list.do%3Fsysparm_userpref_module%3D50047c06c0a8016c0135a14cebc8191b%26sysparm_clear_stack%3Dtrue

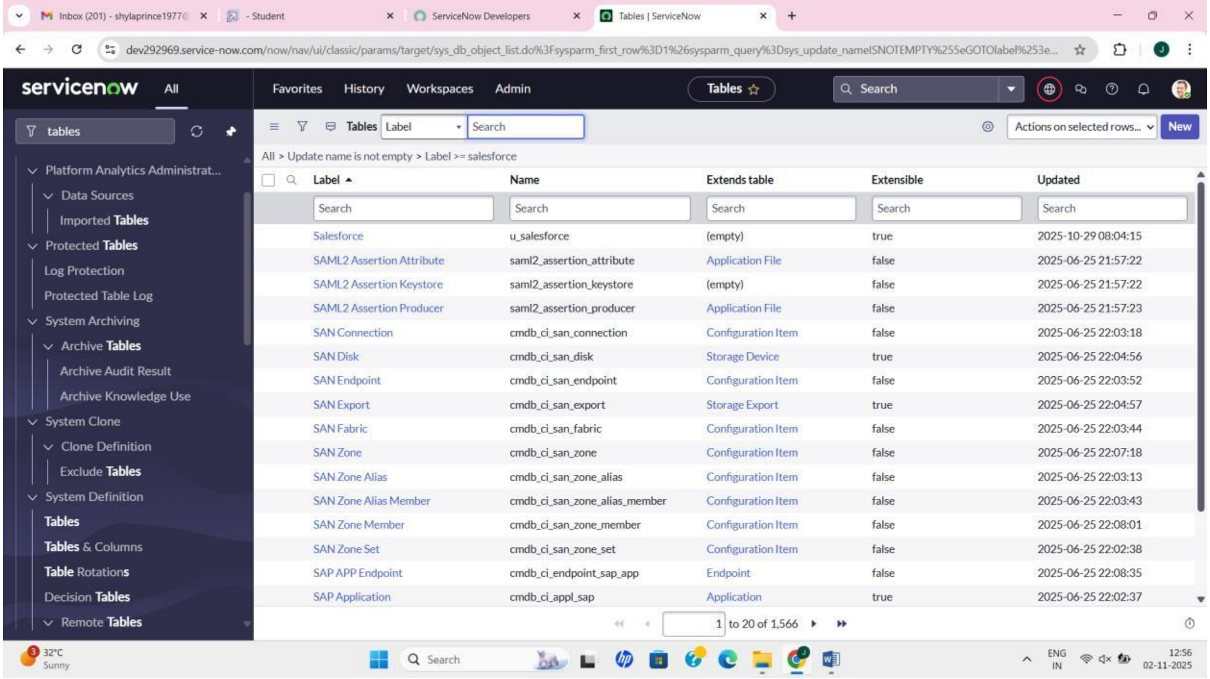
All

Name	Application	State	Installed from	Created	Created by	Parent	Batch Base
Default	Now Assist Troubleshooting	In progress		2025-06-26 00:11:31	admin	(empty)	(empty)
Default	Global	In progress		2025-06-25 22:14:10	system	(empty)	(empty)
Default	Security Center	In progress		2025-06-26 00:10:48	system	(empty)	(empty)
Default	Pipeline	In progress		2025-10-28 04:22:32	system	(empty)	(empty)
Educational Organisation	Global	In progress		2025-10-28 22:18:50	admin	(empty)	(empty)

1 to 5 of 5

Step 3

Create a tables in system definition name the tables as Admission,Salesforce and Student Progress

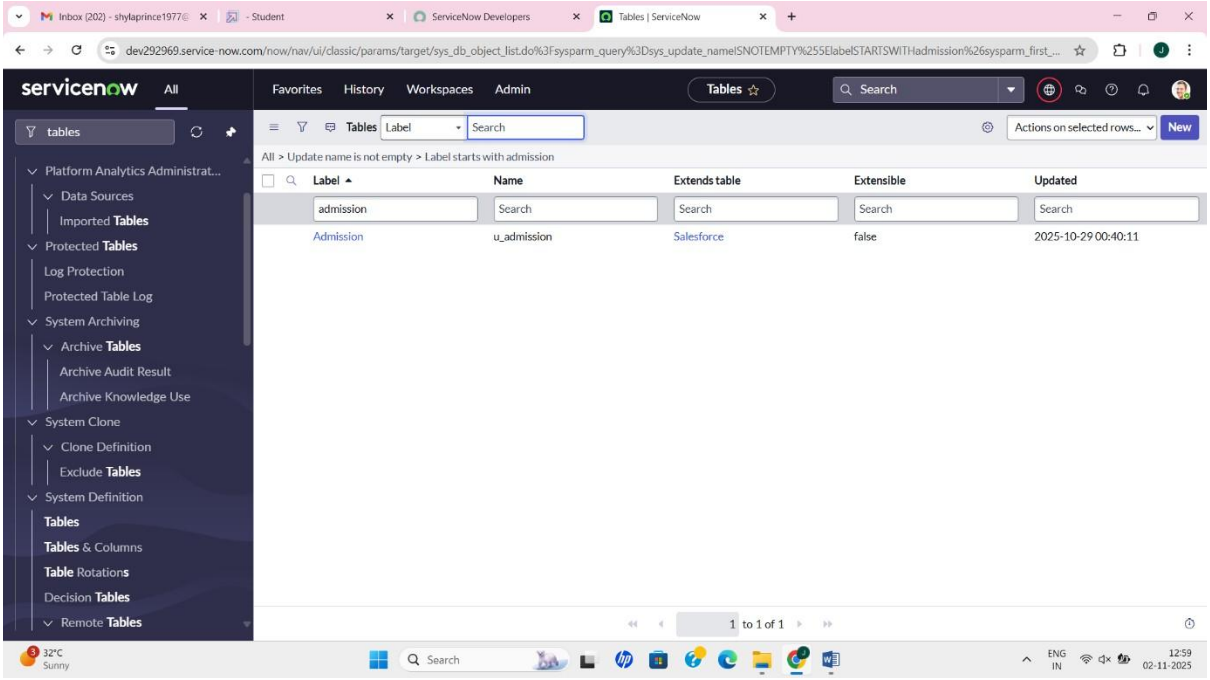


dev292969.service-now.com/now/nav/ui/classic/params/target/sys_db_object_list.do%3Fsysparm_first_row%3D1%26sysparm_query%3Dsys_update_nameISNOTEMPTY%255eGOTOlabel%253e...

All > Update name is not empty > Label >= salesforce

Label	Name	Extends table	Extensible	Updated
Salesforce	u_salesforce	(empty)	true	2025-10-29 08:04:15
SAML2 Assertion Attribute	saml2_assertion_attribute	Application File	false	2025-06-25 21:57:22
SAML2 Assertion Keystore	saml2_assertion_keystore	(empty)	false	2025-06-25 21:57:22
SAML2 Assertion Producer	saml2_assertion_producer	Application File	false	2025-06-25 21:57:23
SAN Connection	cmdb_ci_san_connection	Configuration Item	false	2025-06-25 22:03:18
SAN Disk	cmdb_ci_san_disk	Storage Device	true	2025-06-25 22:04:56
SAN Endpoint	cmdb_ci_san_endpoint	Configuration Item	false	2025-06-25 22:03:52
SAN Export	cmdb_ci_san_export	Storage Export	true	2025-06-25 22:04:57
SAN Fabric	cmdb_ci_san_fabric	Configuration Item	false	2025-06-25 22:03:44
SAN Zone	cmdb_ci_san_zone	Configuration Item	false	2025-06-25 22:07:18
SAN Zone Alias	cmdb_ci_san_zone_alias	Configuration Item	false	2025-06-25 22:03:13
SAN Zone Alias Member	cmdb_ci_san_zone_alias_member	Configuration Item	false	2025-06-25 22:03:43
SAN Zone Member	cmdb_ci_san_zone_member	Configuration Item	false	2025-06-25 22:08:01
SAN Zone Set	cmdb_ci_san_zone_set	Configuration Item	false	2025-06-25 22:02:38
SAP APP Endpoint	cmdb_ci_endpoint_sap_app	Endpoint	false	2025-06-25 22:08:35
SAP Application	cmdb_ci_appl_sap	Application	true	2025-06-25 22:02:37

1 to 20 of 1,566



Tables Label Search

All > Update name is not empty > Label starts with admission

Label	Name	Extends table	Extensible	Updated
admission	Search	Search	Search	Search
Admission	u_admission	Salesforce	false	2025-10-29 00:40:11

1 to 1 of 1

Step 4

Create the label for the tables

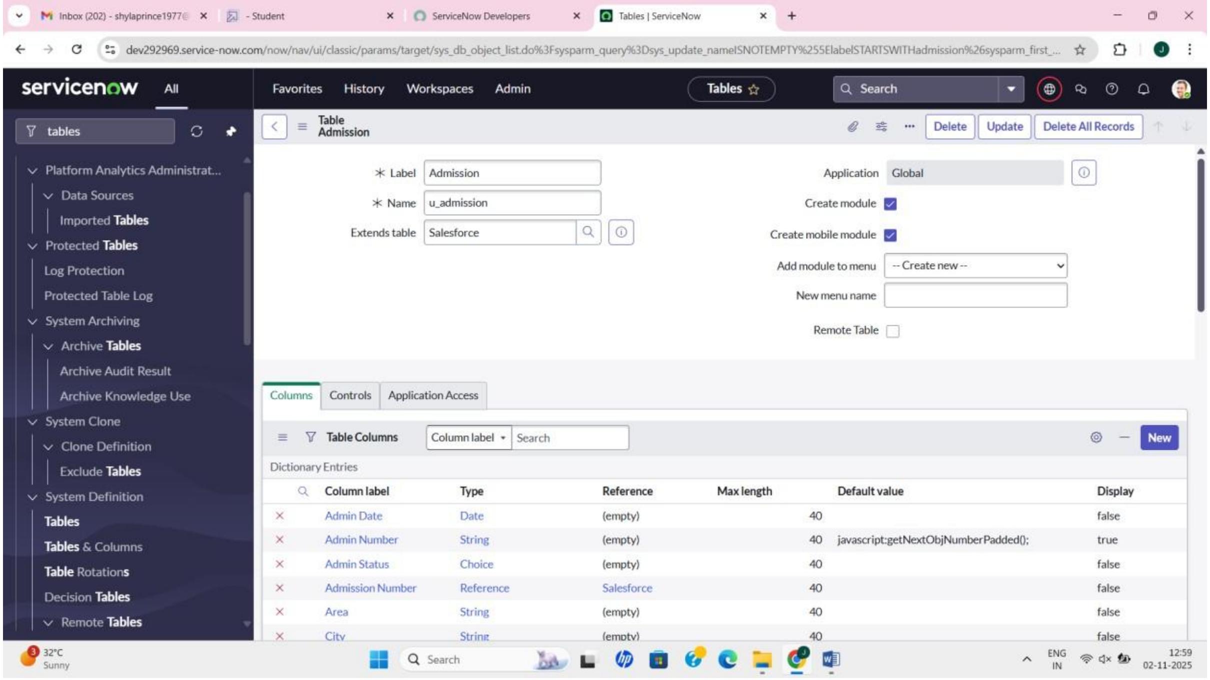


Table
Admission

* Label	Admission	Application	Global
* Name	u_admission	Create module	☑
Extends table	Salesforce	Create mobile module	☑
		Add module to menu	-- Create new --
		New menu name	
		Remote Table	□

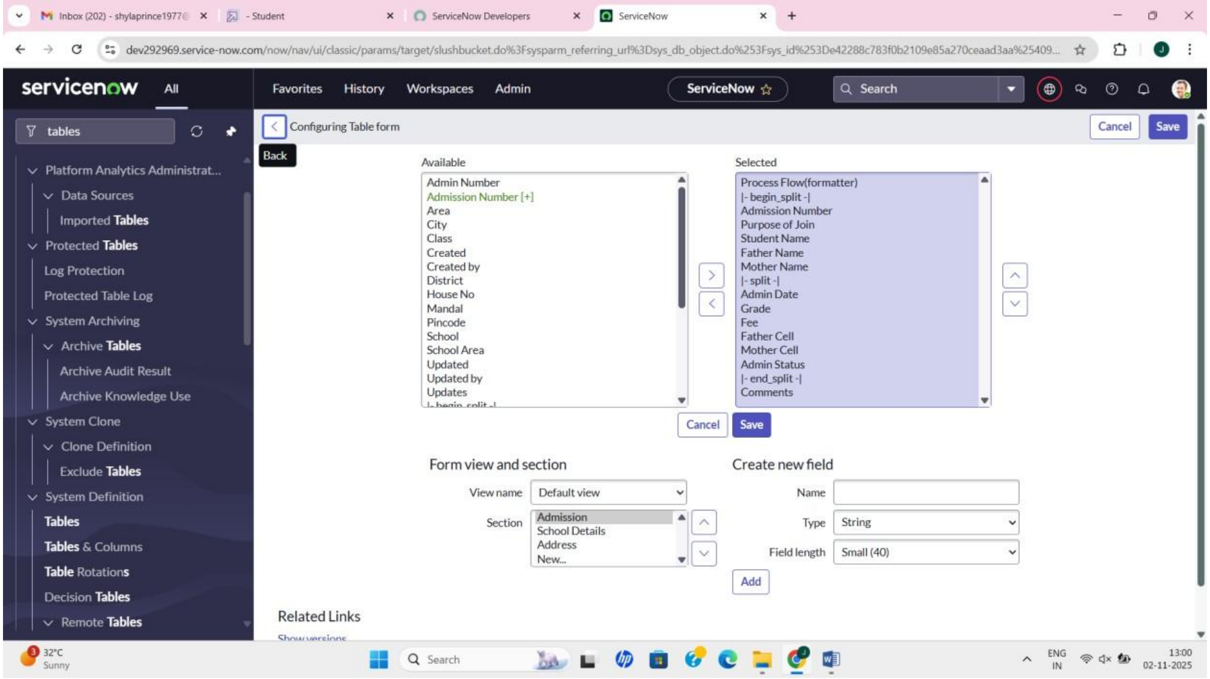
Table Columns Column label Search

Dictionary Entries

Column label	Type	Reference	Max length	Default value	Display
Admin Date	Date	(empty)	40		false
Admin Number	String	(empty)	40	javascript:getNextObjNumberPadded();	true
Admin Status	Choice	(empty)	40		false
Admission Number	Reference	Salesforce	40		false
Area	String	(empty)	40		false
City	String	(empty)	40		false

Step 5

Design the form for the table content then configure the table in the service now administrator



Configuring Table form

Available

- Admin Number
- Admission Number [+]
- Area
- City
- Class
- Created
- Created by
- District
- House No
- Mandal
- Pincode
- School
- School Area
- Updated
- Updated by
- Updates
- |- begin_split -|

Selected

- Process Flow(formatter)
- |- begin_split -|
- Admission Number
- Purpose of Join
- Student Name
- Father Name
- Mother Name
- |- split -|
- Admin Date
- Grade
- Fee
- Father Cell
- Mother Cell
- Admin Status
- |- end_split -|
- Comments

Form view and section

View name: Default view

Section: Admission, School Details, Address, New...

Create new field

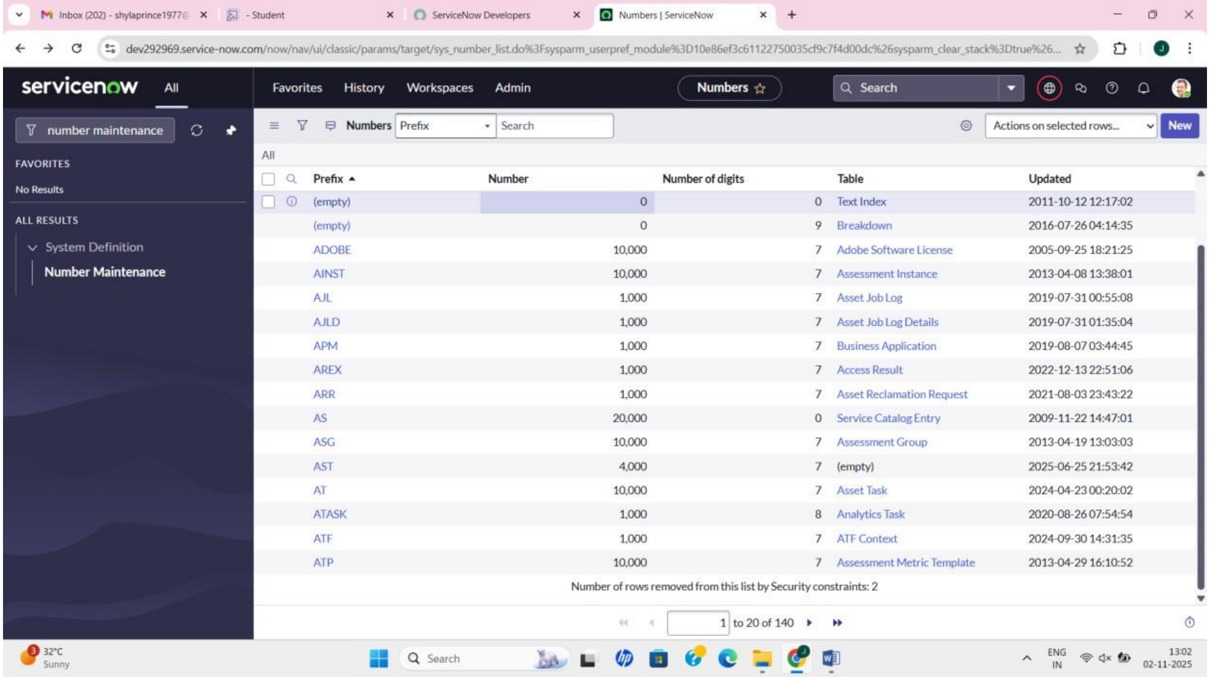
Name:

Type: String

Field length: Small (40)

Step 6

In the Service Now Administrator,Number Maintenance Configuration



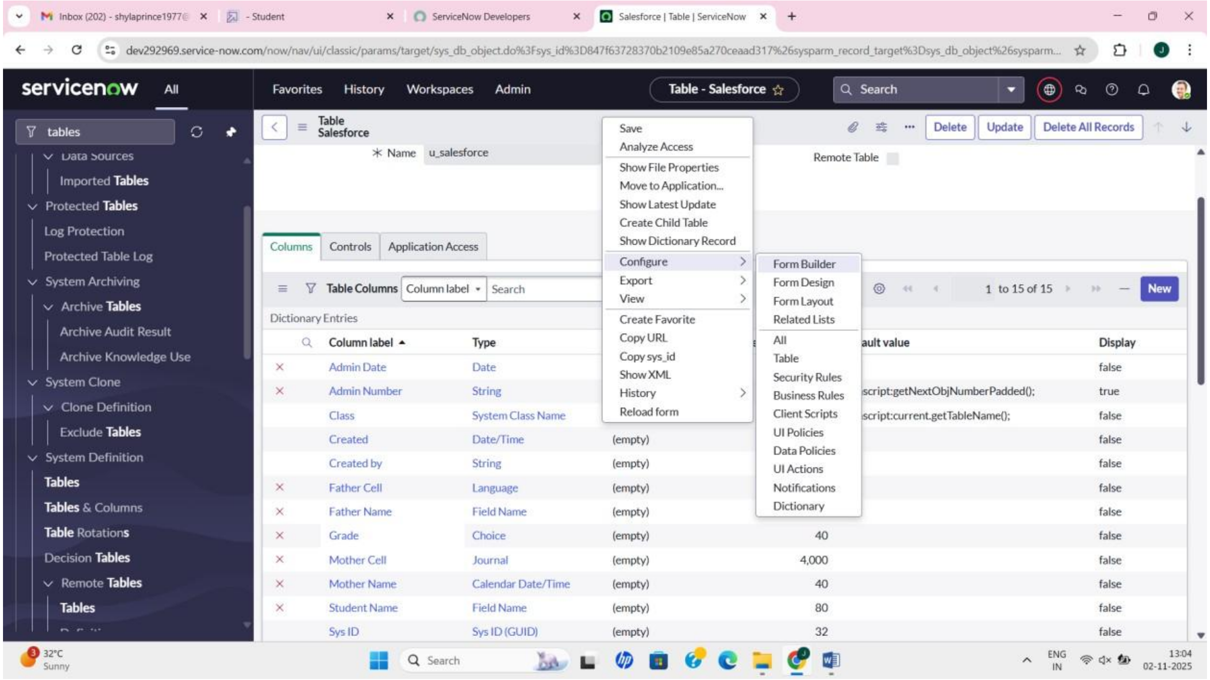
Prefix ▲	Number	Number of digits	Table	Updated
(empty)	0	0	Text Index	2011-10-12 12:17:02
(empty)	0	9	Breakdown	2016-07-26 04:14:35
ADOBE	10,000	7	Adobe Software License	2005-09-25 18:21:25
AINST	10,000	7	Assessment Instance	2013-04-08 13:38:01
AJL	1,000	7	Asset Job Log	2019-07-31 00:55:08
AJLD	1,000	7	Asset Job Log Details	2019-07-31 01:35:04
APM	1,000	7	Business Application	2019-08-07 03:44:45
AREX	1,000	7	Access Result	2022-12-13 22:51:06
ARR	1,000	7	Asset Reclamation Request	2021-08-03 23:43:22
AS	20,000	0	Service Catalog Entry	2009-11-22 14:47:01
ASG	10,000	7	Assessment Group	2013-04-19 13:03:03
AST	4,000	7	(empty)	2025-06-25 21:53:42
AT	10,000	7	Asset Task	2024-04-23 00:20:02
ATASK	1,000	8	Analytics Task	2020-08-26 07:54:54
ATF	1,000	7	ATF Context	2024-09-30 14:31:35
ATP	10,000	7	Assessment Metric Template	2013-04-29 16:10:52

Number of rows removed from this list by Security constraints: 2

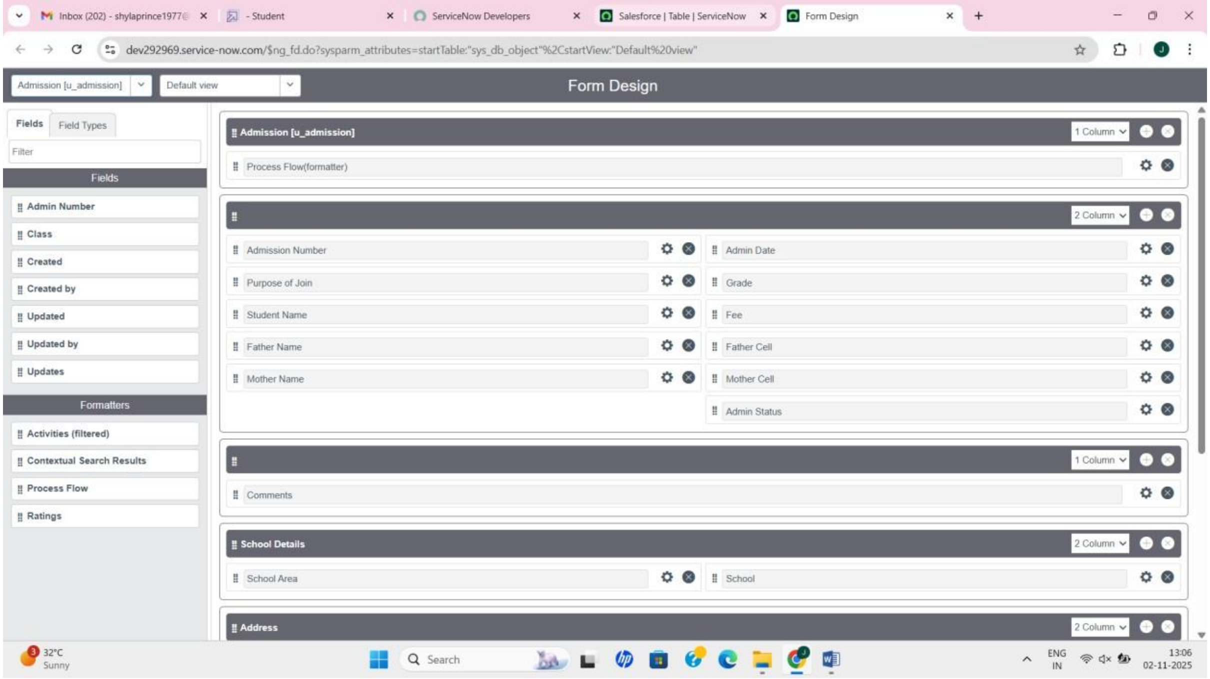
1 to 20 of 140

Step 7

The process flow was created for designing the form and layout



Column label	Type			ault value	Display
Admin Date	Date				false
Admin Number	String			script:getNextObjNumberPadded();	true
Class	System Class Name			script:current.getTableName();	false
Created	Date/Time	(empty)			false
Created by	String	(empty)			false
Father Cell	Language	(empty)			false
Father Name	Field Name	(empty)			false
Grade	Choice	(empty)	40		false
Mother Cell	Journal	(empty)	4,000		false
Mother Name	Calendar Date/Time	(empty)	40		false
Student Name	Field Name	(empty)	80		false
Sys ID	Sys ID (GUID)	(empty)	32		false



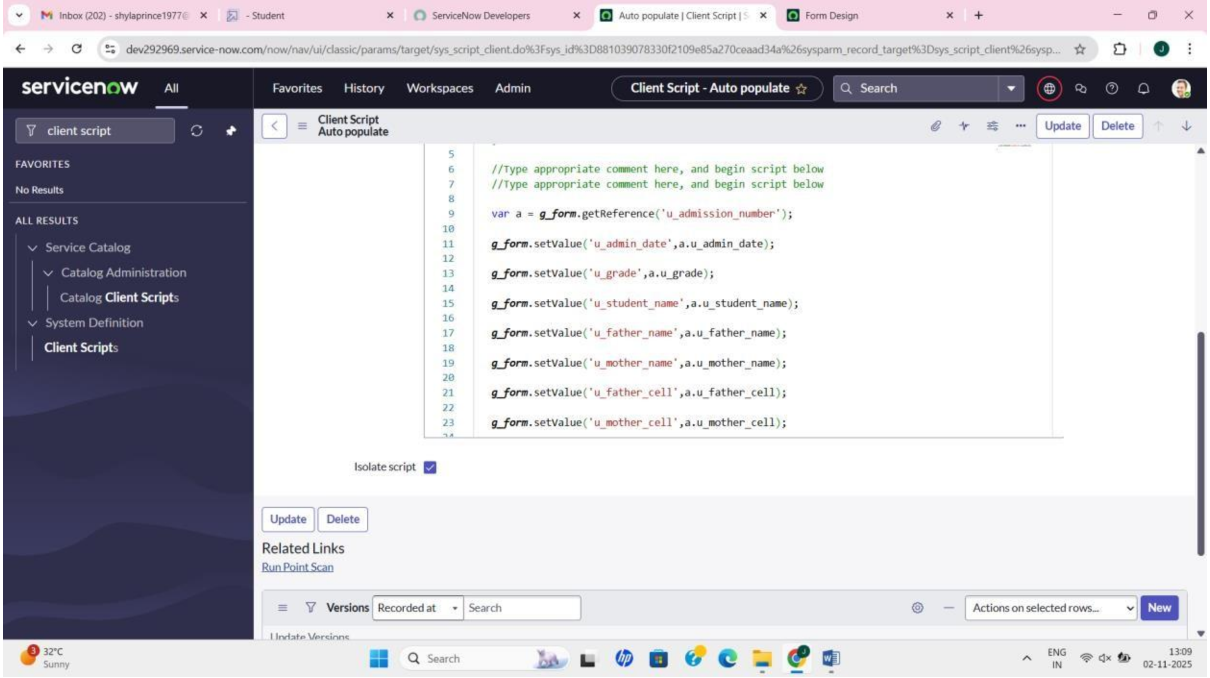
Step 8

Write the script for the client

```
functiononChange(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); }
```

Step 9

The client script was isolated and saved

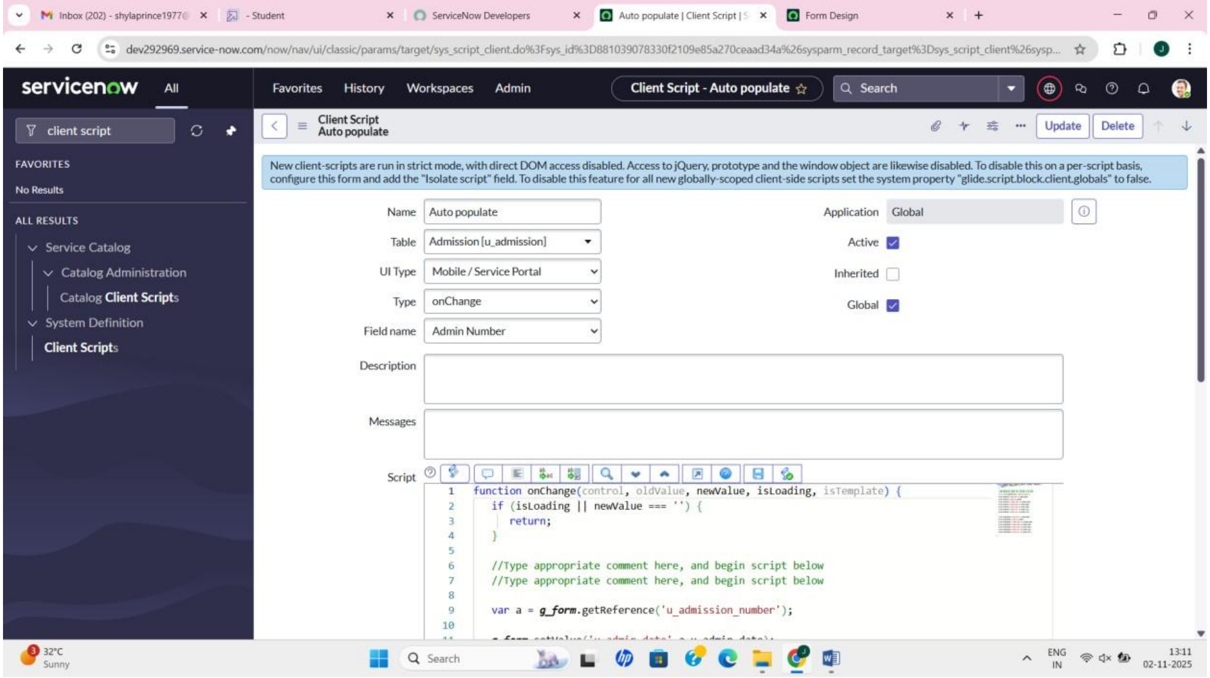


```
//Type appropriate comment here, and begin script below
//Type appropriate comment here, and begin script below

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);
```

Isolate script ☑

Step 10

The testing of the client script for the autopopulate in the admission table



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.

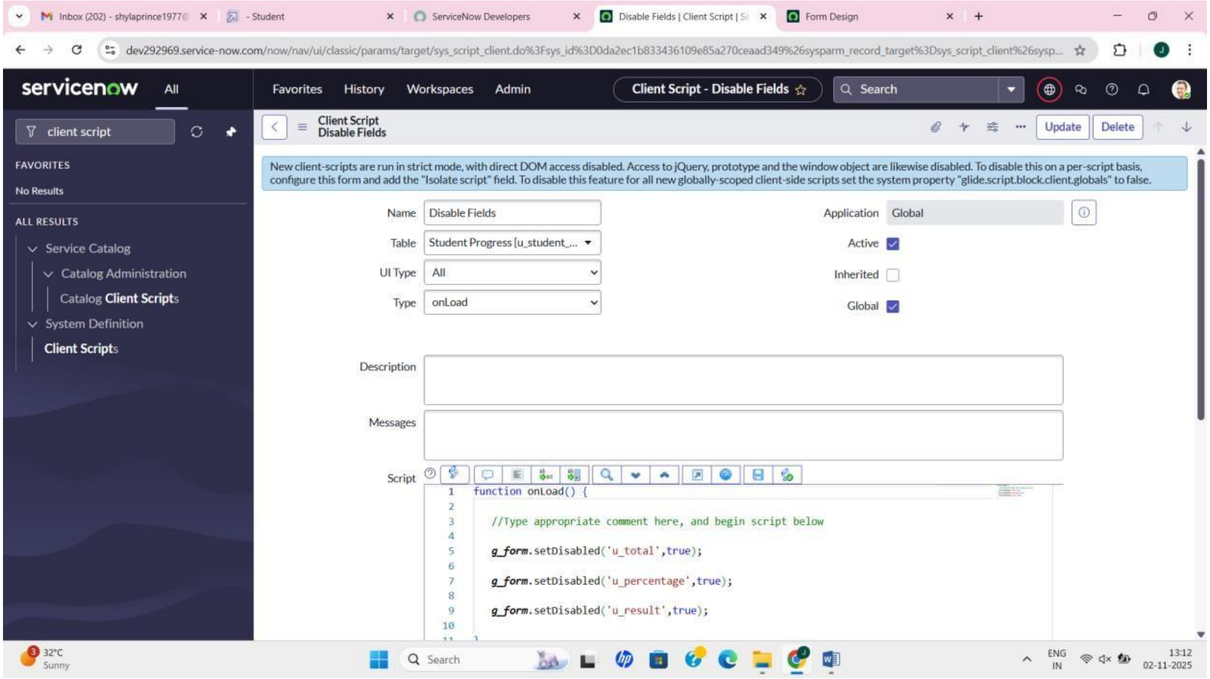
Name	Auto populate	Application	Global
Table	Admission [u_admission]	Active	☑
UI Type	Mobile / Service Portal	Inherited	☐
Type	onChange	Global	☑
Field name	Admin Number		
Description			
Messages			

Script

```
function onChange(control, oldValue, newValue, isLoading, isTemplate) {
  if (isLoading || newValue === '') {
    return;
  }

  //Type appropriate comment here, and begin script below
  //Type appropriate comment here, and begin script below

  var a = g_form.getReference('u_admission_number');
```

Step 11

Create the disable fields for student progress table



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.

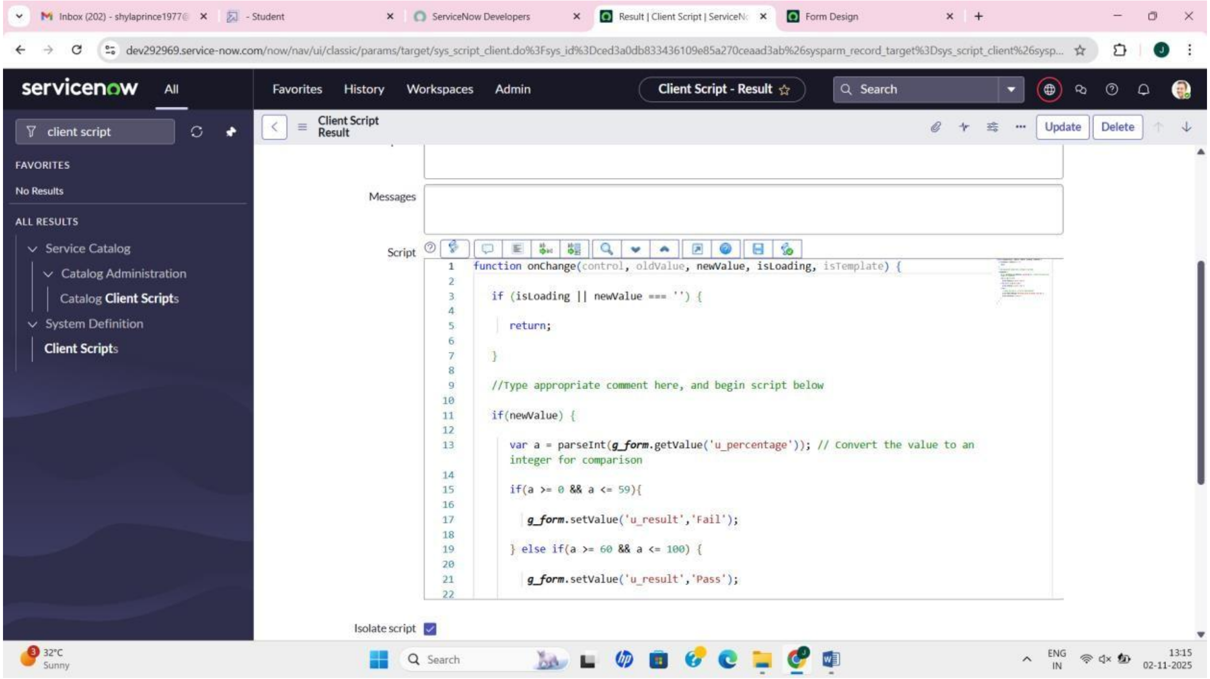
Name: Disable Fields
Table: Student Progress [u_student_...
UI Type: All
Type: onLoad
Application: Global
Active: ☑
Inherited: □
Global: ☑

Script:

```
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);

}
```

Step 12

Create result and percentage script for the student progress table



Client Script - Result

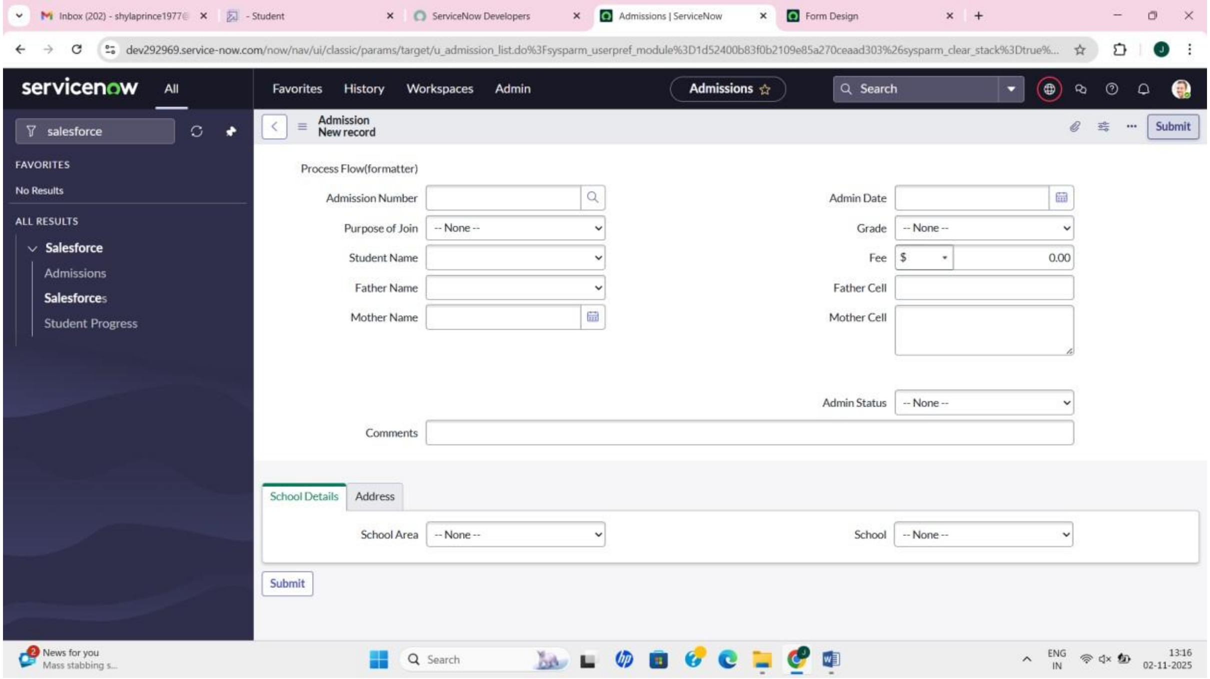
Script:

```
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');

```

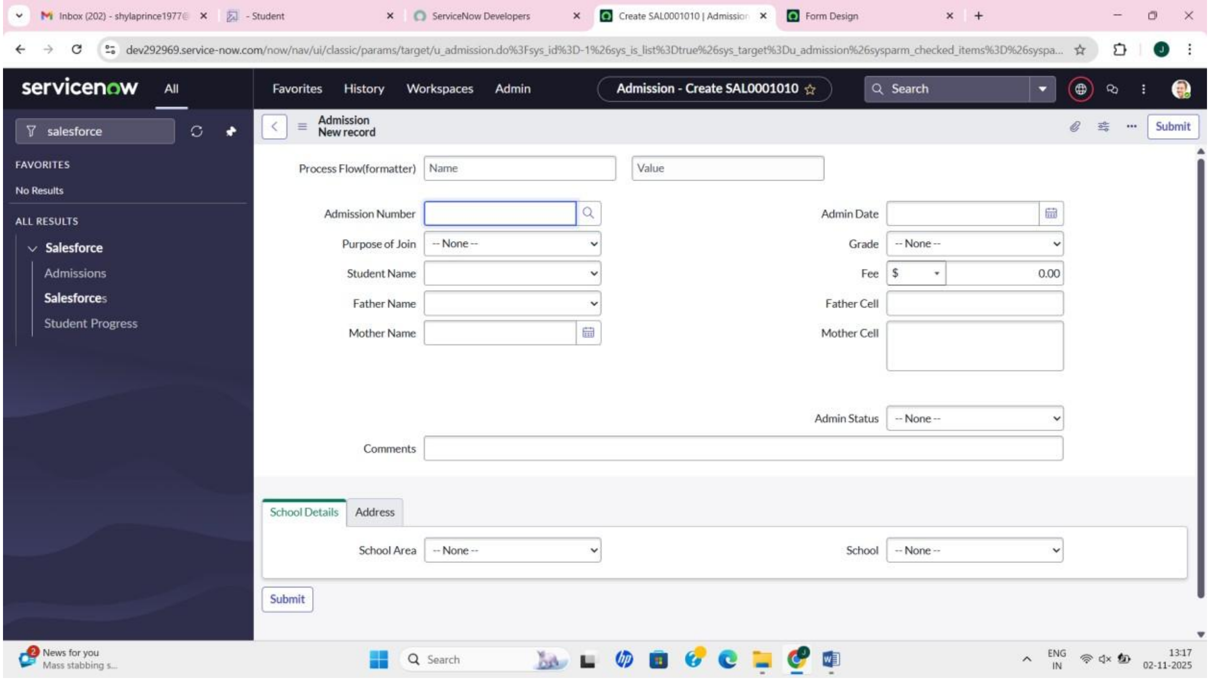
Isolate script: ☑

Step 13

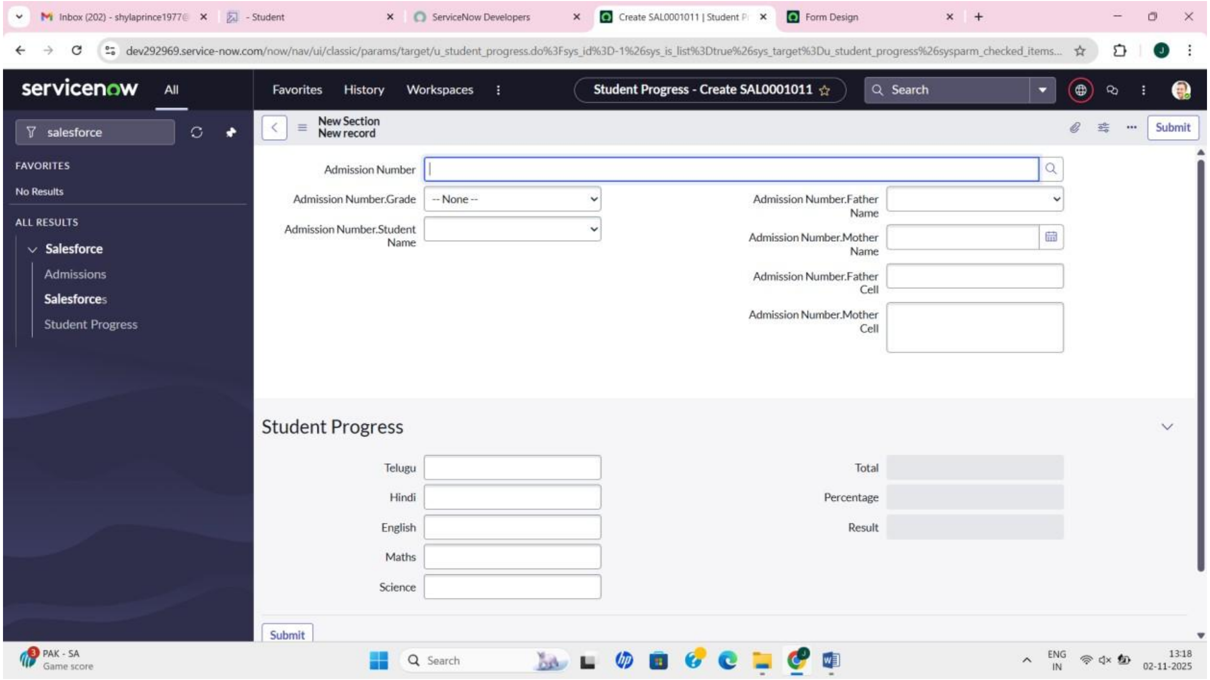
The testing and result verification can be output in the salesforce



Admission:



Student Progress:



dev292969.service-now.com/now/nav/ui/classic/params/target/u_student_progress.do%3Fsys_id%3D-1%26sys_is_list%3Dtrue%26sys_target%3Du_student_progress%26sysparm_checked_items...

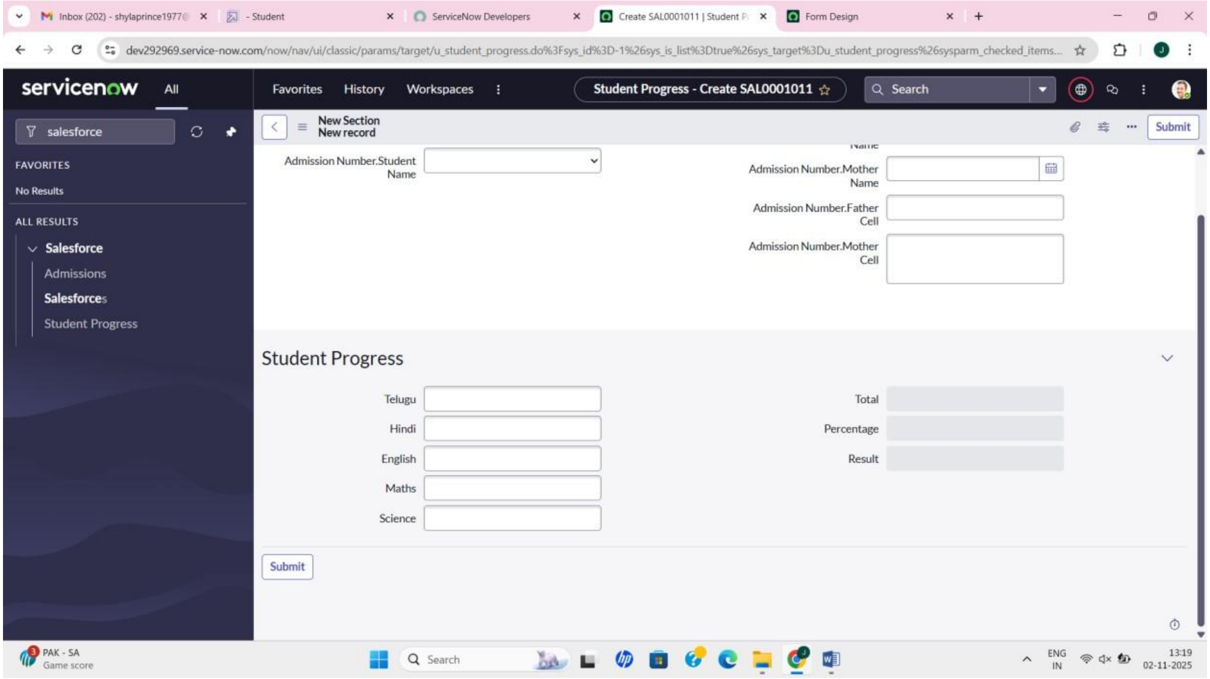
Student Progress - Create SAL0001011

New Section
New record

Admission Number
Admission Number.Grade: -- None --
Admission Number.Student Name
Admission Number.Father Name
Admission Number.Mother Name
Admission Number.Father Cell
Admission Number.Mother Cell

Student Progress

Telugu
Hindi
English
Maths
Science
Total
Percentage
Result



dev292969.service-now.com/now/nav/ui/classic/params/target/u_student_progress.do%3Fsys_id%3D-1%26sys_is_list%3Dtrue%26sys_target%3Du_student_progress%26sysparm_checked_items...

Student Progress - Create SAL0001011

New Section
New record

Admission Number.Student Name
Admission Number.Mother Name
Admission Number.Father Cell
Admission Number.Mother Cell

Student Progress

Telugu
Hindi
English
Maths
Science
Total
Percentage
Result

RESULT:

- A working **Educational Management Application** was successfully developed on ServiceNow. □ It enables:

  - Easy record creation and tracking
  - Workflow automation
  - Real-time data management for educational institutions

- The project demonstrates how ServiceNow can be used beyond IT services — to manage education-related operations efficiently.

CONCLUSION:

This project successfully demonstrates the implementation of a basic Educational Management System using ServiceNow.Through this project, the administrative activities such as maintaining student details, admissions, and grades are automated and streamlined.It shows the flexibility of ServiceNow as a platform for developing custom enterprise applications in different domains.