Assignment 1

Section 1:

Navigate to System Properties table by typing sys\_properties.LIST in the filter search box

Disable Next experience by looking up for the glide.ui.polaris.experience property and setting the value to false

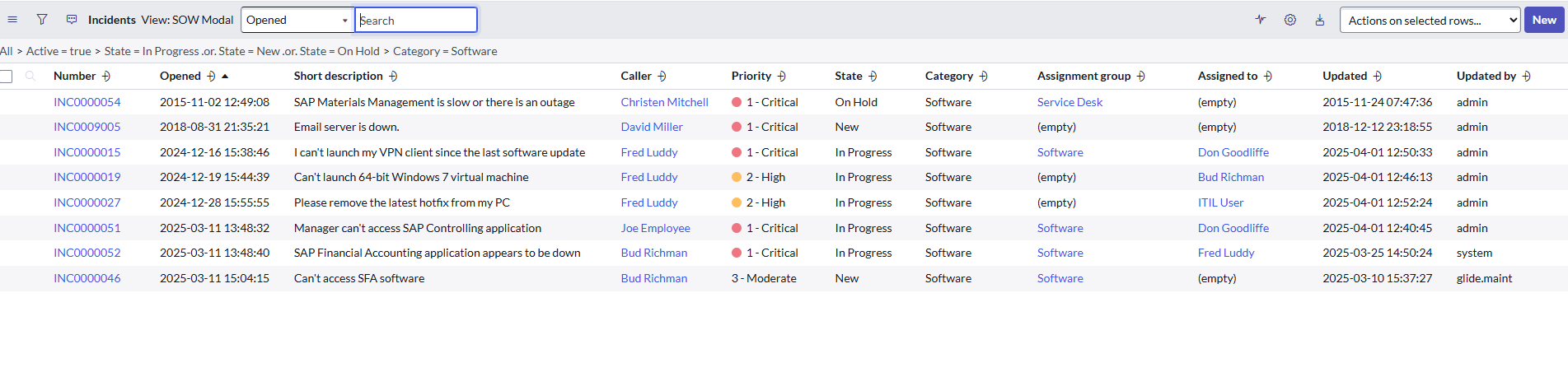
Observe the change in the home page.

If you switch to UI15, switch back to UI16 before re-enabling next experience.

Section 2:

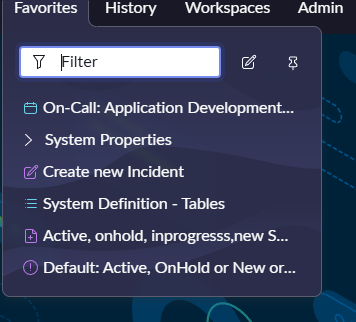
What are list and form views? What are filters?

1. List view – the format in which all the records in a table can be viewed is called a list view. We can create multiple views to view the records for different purposes.
2. Form view – the format in which individual record can be viewed for viewing, editing or new record creation purposes is called a form view. Depending on the necessity, each form view can contain different columns available to the particular form view.
3. Filter breadcrumb: [>](javascript:void(0))Active = true[>](javascript:void(0))State = In Progress .or. State = New .or. State = On Hold[>](javascript:void(0)) Category = Software



Section 3:

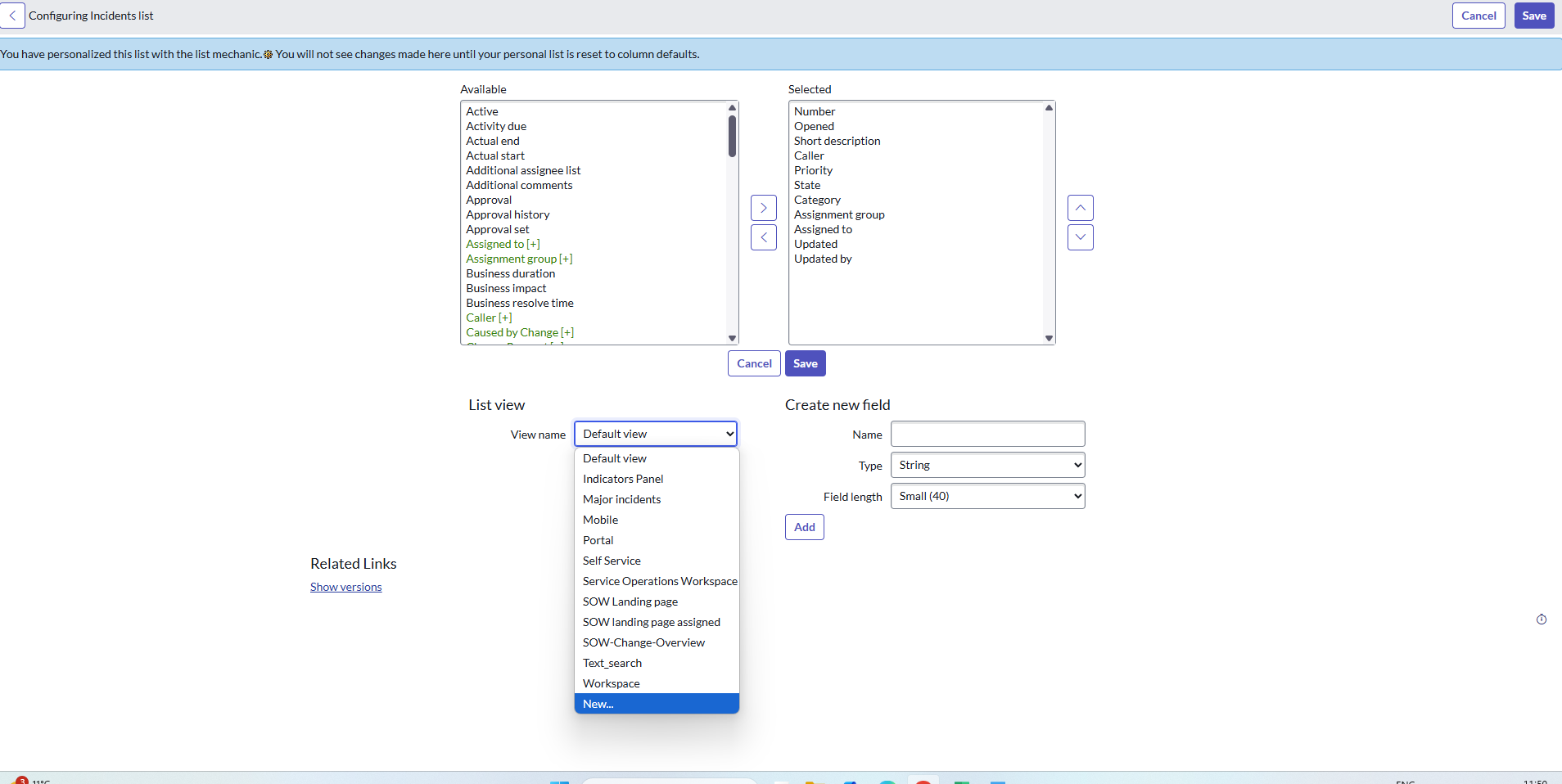
How will you configure it? Create your own filter and mark as favorite



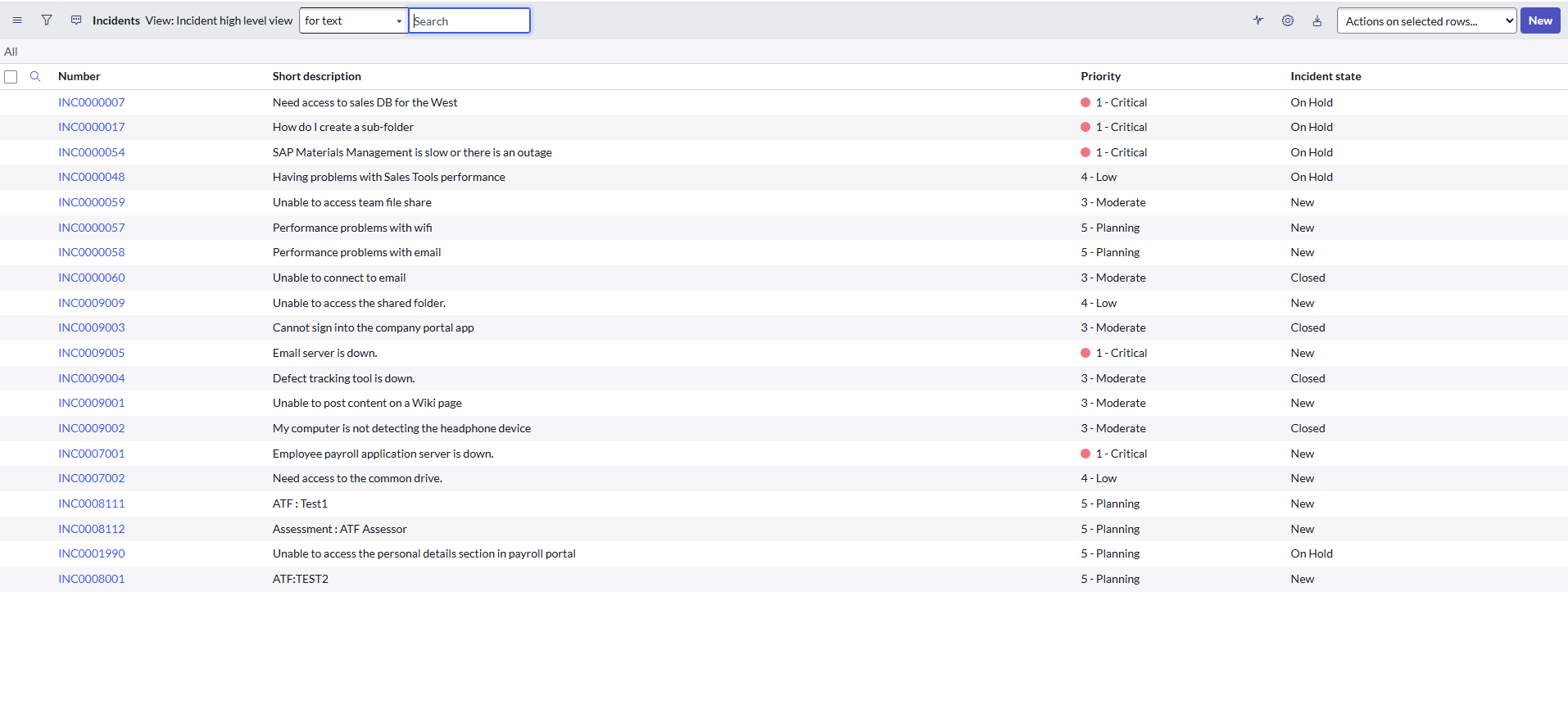
Question 4:

How to create views? Create one sample view in your instance of your choice

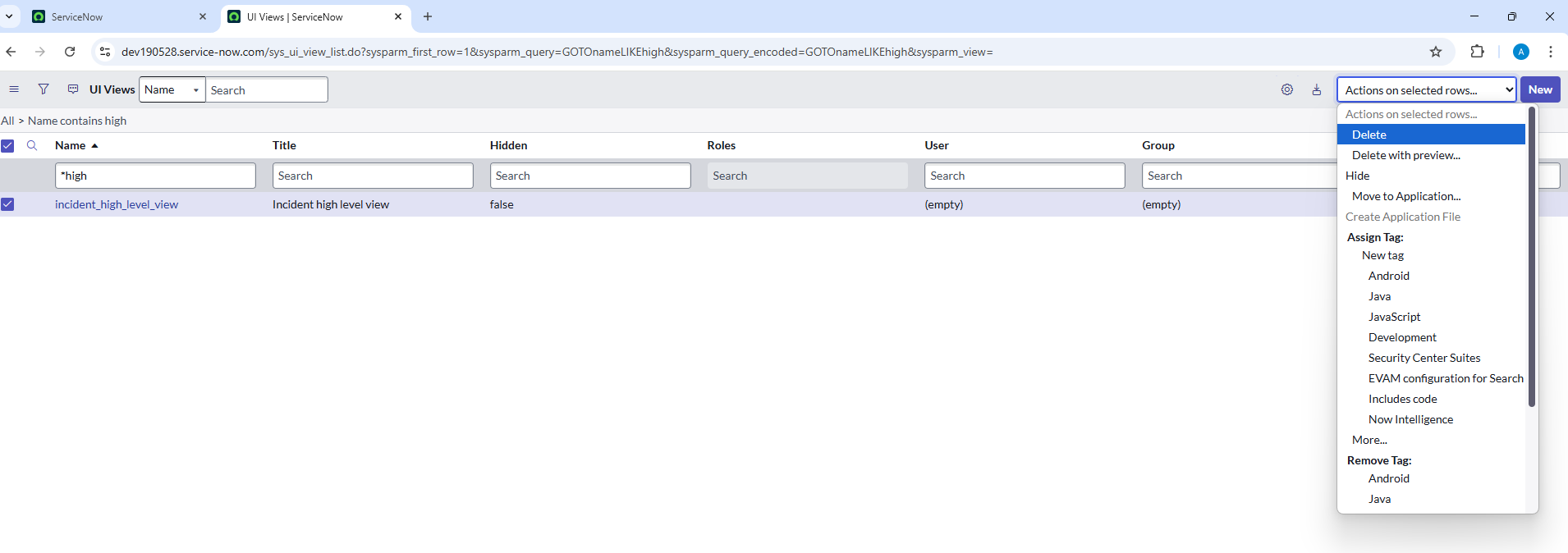
1. Navigate to incident table in a different page by typing incident.LIST in the filter search bar and hitting enter
2. Right click on any field and select Configure 🡪 List Layout
3. Under list view/view name, select “New”



1. Type a name for the new view (for example: Incident high level view)
2. From the “Available” slush bucket, select “Incident State” and double click it
3. From “Selected” slush bucket move all fields except “Number”,” Short Description”,” Priority” and,” Incident State” and move them to the “Available” slush bucket
4. Select “Save”
5. Click on hamburger icon, select view and select “Incident high level view”



To delete the view, from home page, navigate to System UI 🡪 Views and find and delete the view



Section 5:

What type of searches are available in ServiceNow?

Search interface – Global Search, List Search, Knowledge base, Live Feed, UI Pages

Search Operator – Boolean operators, Quotation marks, Wildcard characters

<https://www.servicenow.com/docs/bundle/yokohama-platform-administration/page/administer/search-administration/concept/c_IntroductionToSearching.html>

Section 6:

What is the use of ServiceNow?

Assignment 2

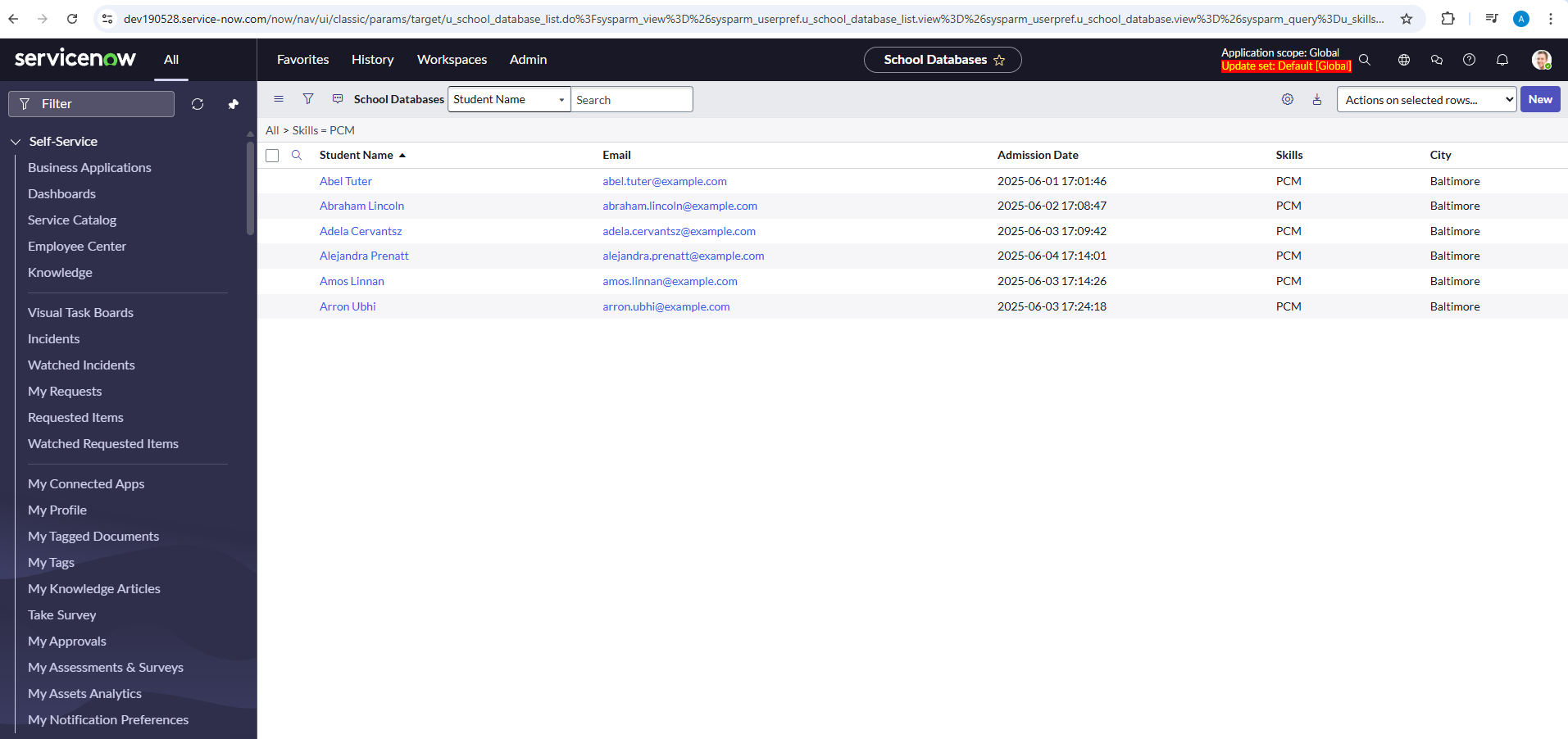
Section 1: Create a custom table

Create a custom table “School Database”

|  |  |
| --- | --- |
| Column Name | Column Type |
| Student Name | Reference (user) |
| Location | Reference (location) |
| Skills | Choice (PCM, PCMB, Commerce) |
| Admission Date | Date Time |
| Fee Paid | True/false |

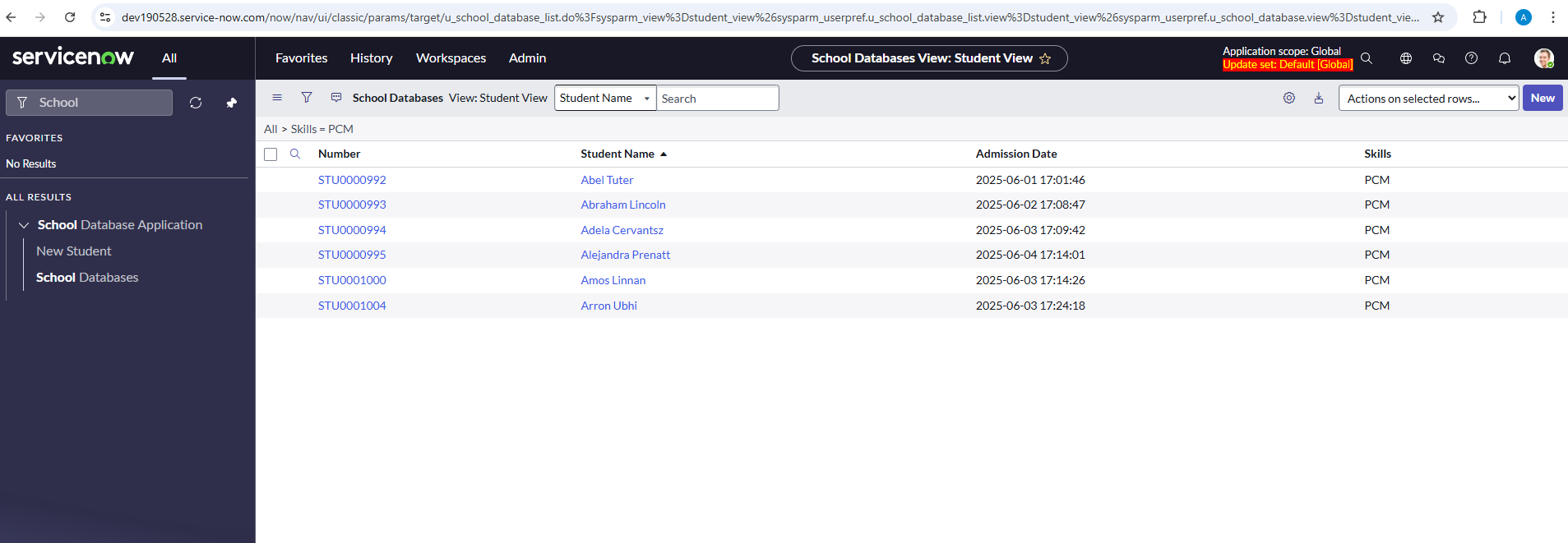
Section 2: Create a Filter for PCM students

Filter: Only PCM Students



Section 2: Create a “Student View” without “Fee Paid” column

Student View – without “Fee Paid” column



Section 3: Create a “Teacher View” with “Fee Paid” column

Teacher View – with “Fee Paid” column

A screenshot of a computer

AI-generated content may be incorrect.

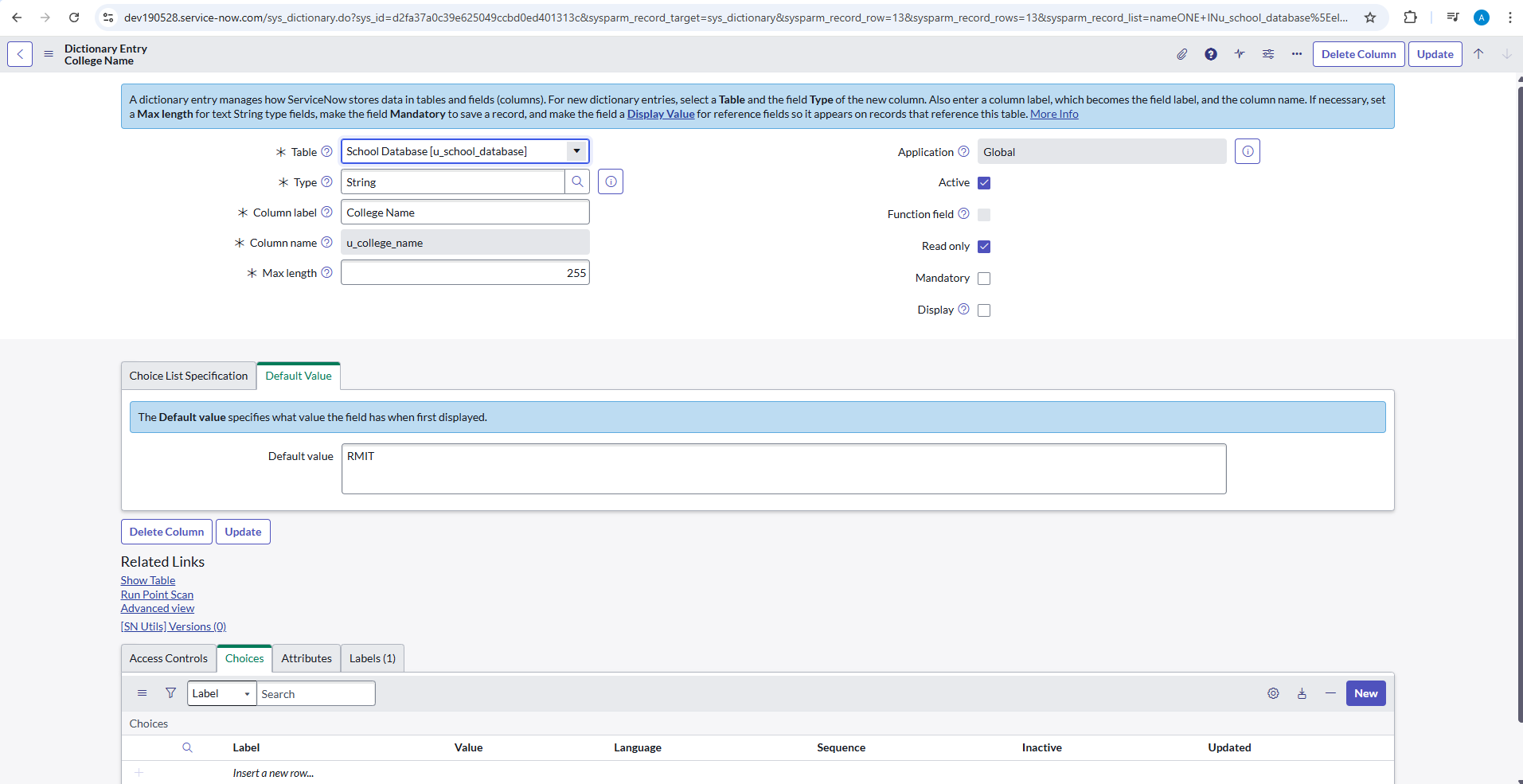
Default view – without filter

A screenshot of a computer

AI-generated content may be incorrect.

Assignment 3

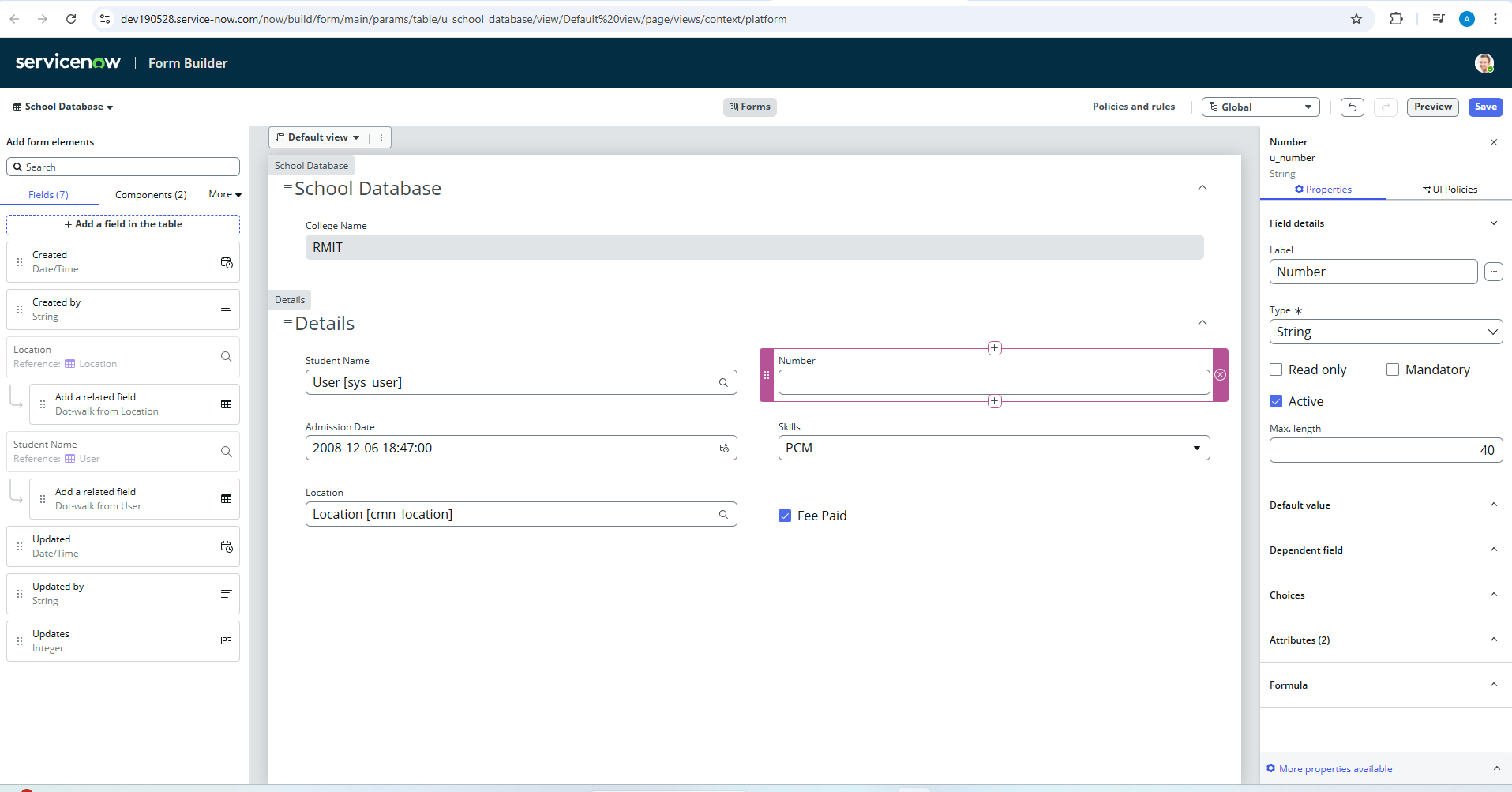
Section 1: Create a read only column “College Name” of type “String” on “School Database” table



A screenshot of a computer

AI-generated content may be incorrect.

Section 2: Create form structure for the table with “College Name” on top



Form Builder Preview:

A screenshot of a computer

AI-generated content may be incorrect.

Section 3: Configure form structure for to be split into 2 columns for all fields except “College Name”

A screenshot of a computer

AI-generated content may be incorrect.

Assignment 4

Section 1: Importing employee data with department mapping

Employees are stored in sys\_user table. Departments are stored in cmn\_department table. Ensure all employees are imported with correct department references while avoiding duplicate records.

Excel file structure

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Employee Id | First Name | Last Name | Email | Phone Number | Department Name | Location |

Fields:

Employee Id

First Name

Last Name

Email

Phone Number

Department Name

Location

Following are the data import goals

Import employees into sys\_user table

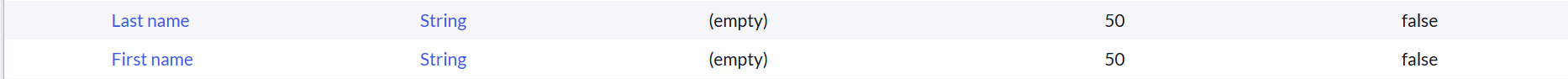
Import departments into cmn\_department table

Link employees to their corresponding departments via Department reference field

Skip importing employees if an employee ID already exists

**Skip importing departments if the department name already exists**

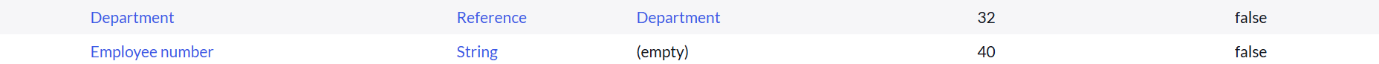
Following are the relevant fields in sys\_user table









Following are the relevant fields in Department table



Solution:

Create excel sheet Assignment4.xlsx

In developer instance, navigate to All 🡪System Import Sets 🡪 Load Data and create import set by name Employee Details. Select the Assignment4.xslx for input file.



Click on Import Sets

A screenshot of a computer

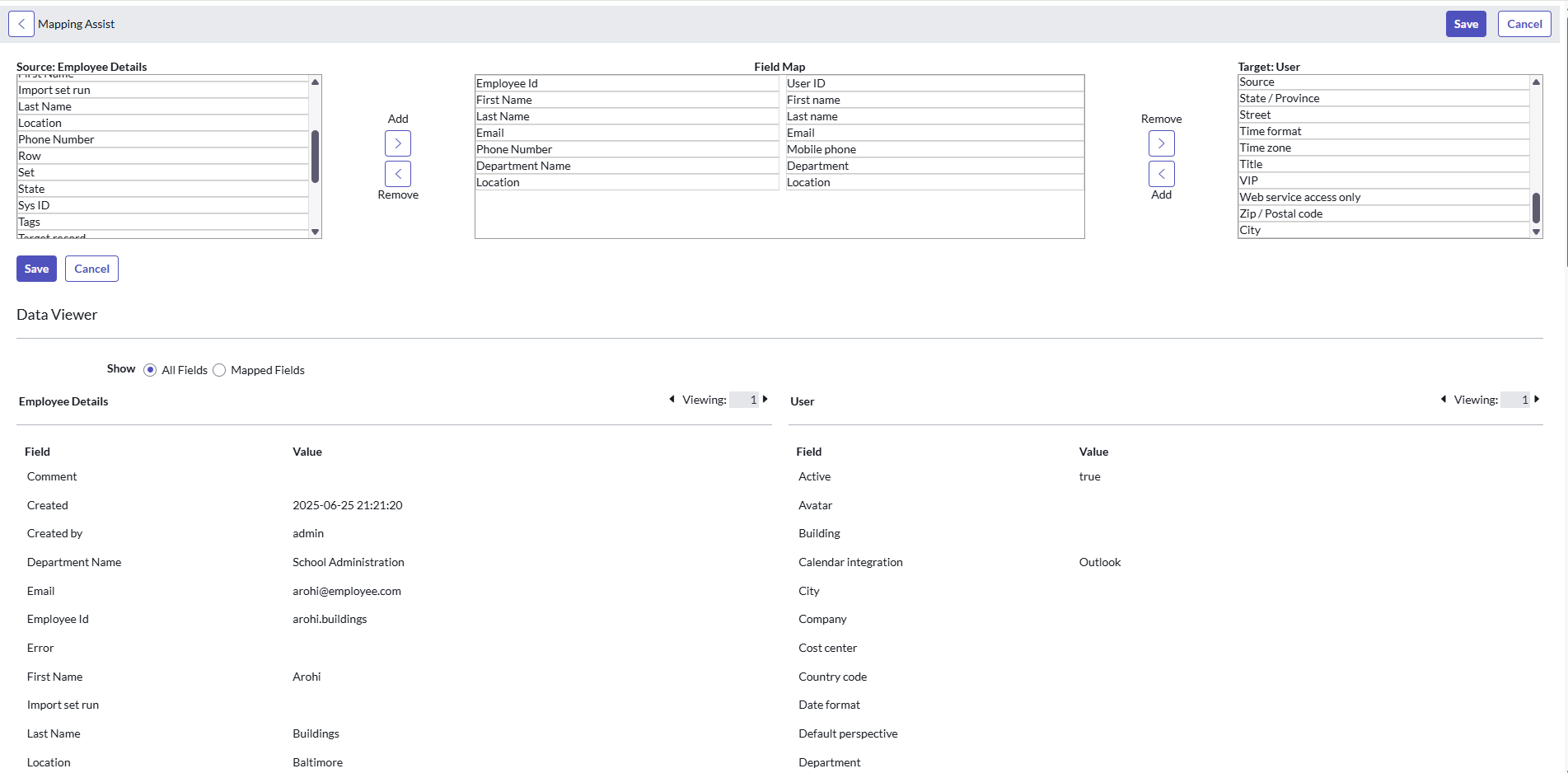
AI-generated content may be incorrect.

Transform Map

A screenshot of a computer

AI-generated content may be incorrect.

Use Mapping Assist to map fields appropriately and select save



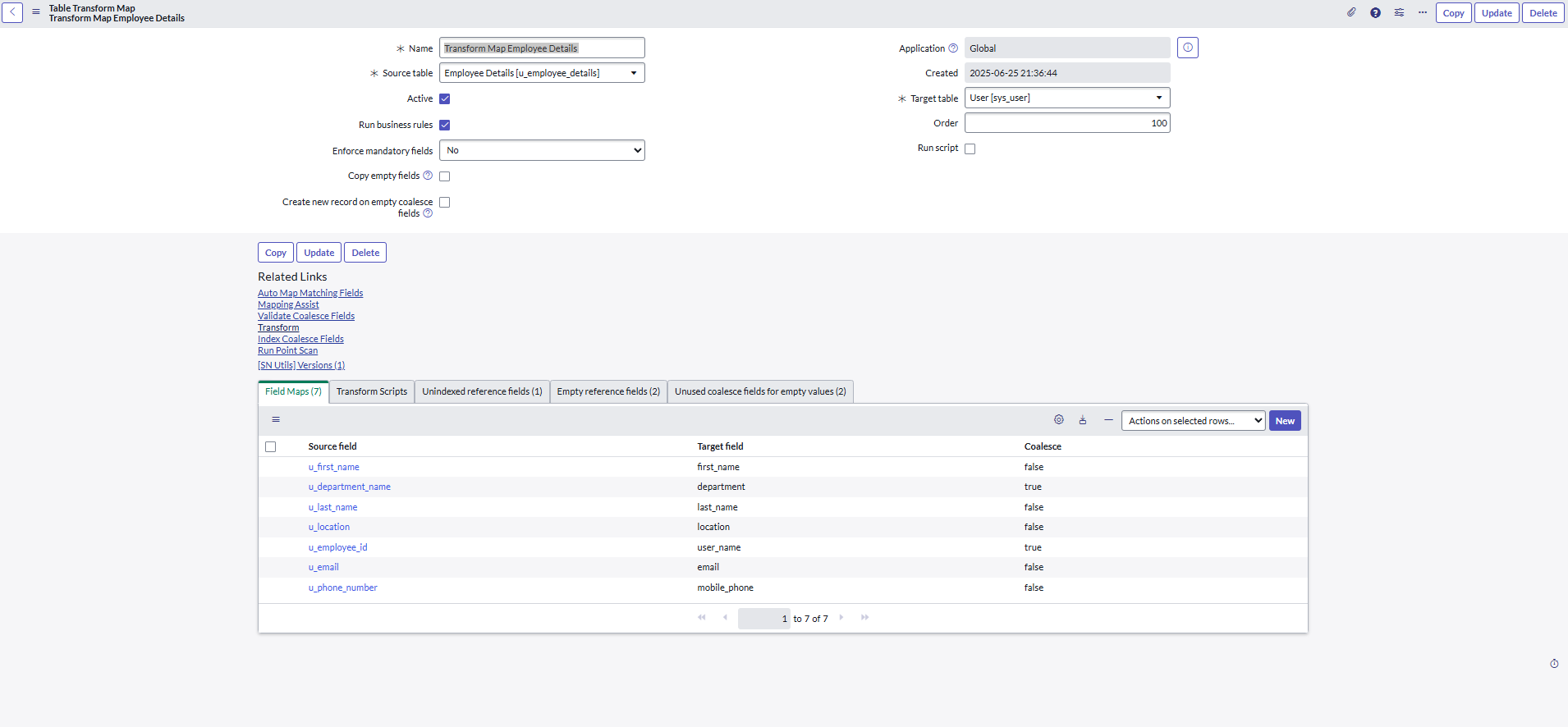
In the “Field Maps” tab, set **Coalesce** property as **true** for the Target Fields “department” and “user\_name”

A screenshot of a computer

AI-generated content may be incorrect.

Save the transform map.

Under “Related Links” click on “Transform”





A screenshot of a computer

AI-generated content may be incorrect.

Click on the button that says “Transform”.

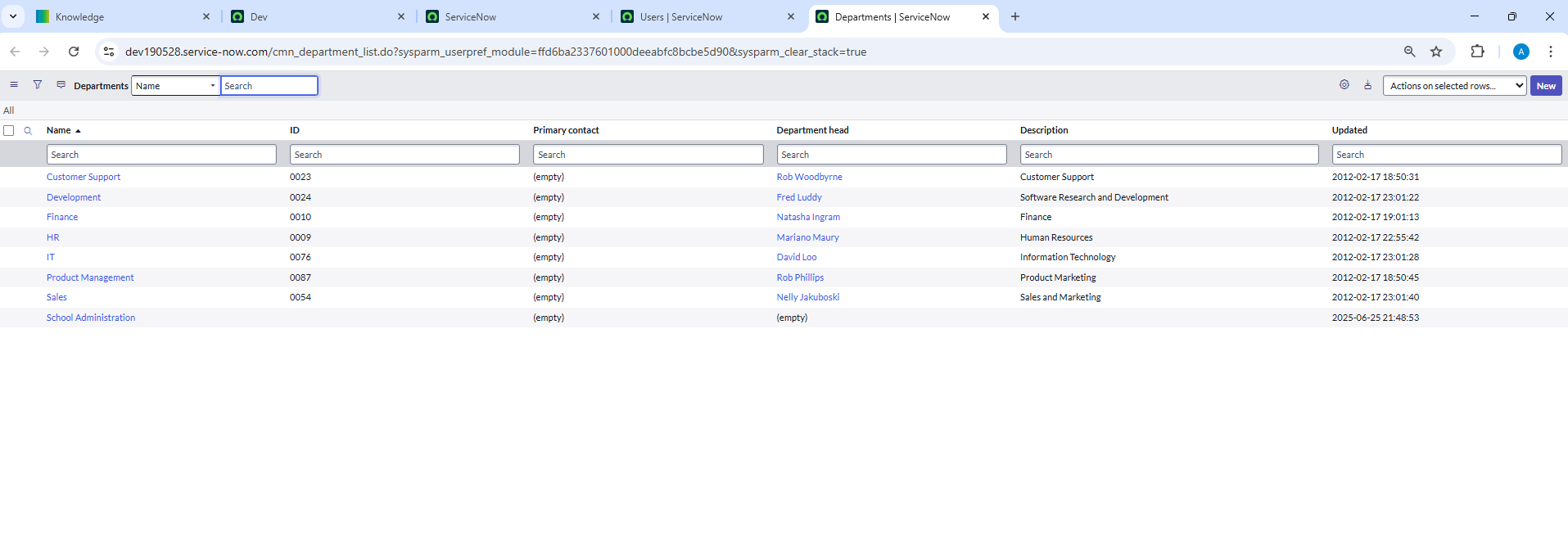


Click on the import set **ISET0010001** to see the actual data imported

A screenshot of a computer

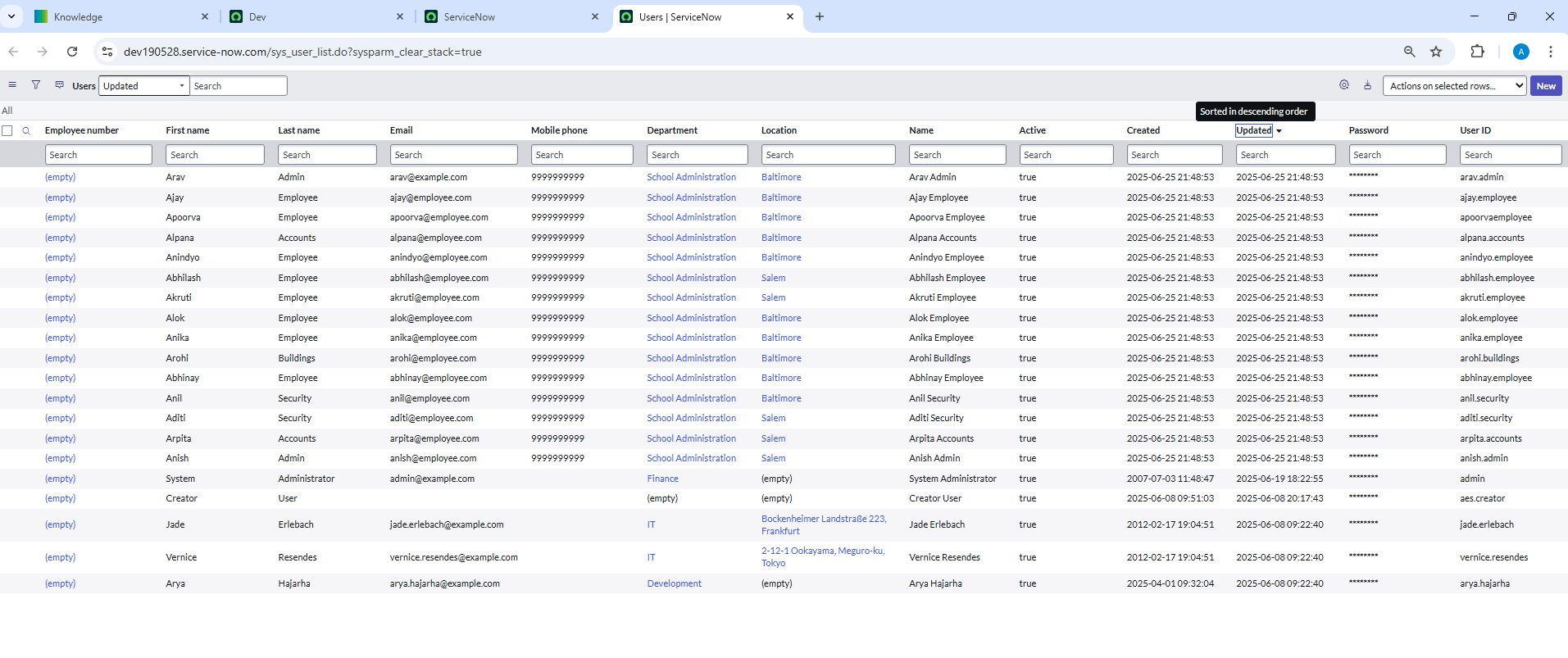
AI-generated content may be incorrect.

Department “School Administration” is added to the Departments (cmn\_department) table





New employees are added to sys\_user table



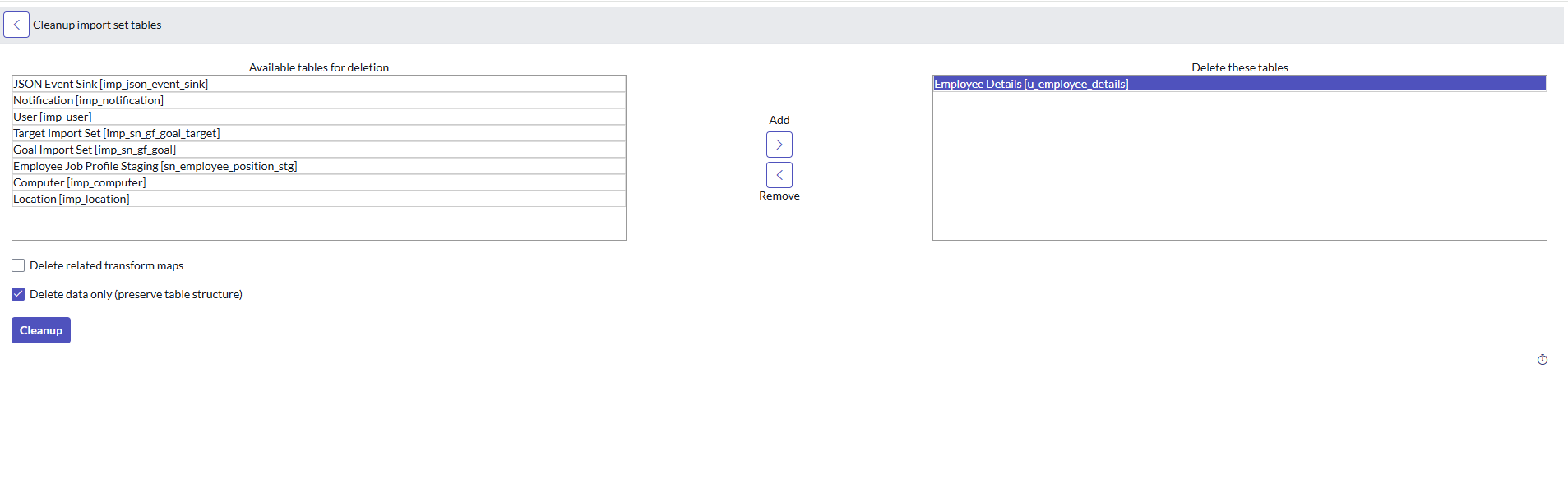


Edit the excel sheet to contain fewer employees

Add one new department, and two new employees. Keep the rest same as before.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Employee Id | First Name | Last Name | Email | Phone Number | Department Name | Location |
| arav.admin | Arav | Admin | [arav@example.com](mailto:arav@example.com) | 9999999999 | School Administration | Baltimore |
| ajay.employee | Ajay | Employee | [ajay@employee.com](mailto:ajay@employee.com) | 9999999999 | School Administration | Baltimore |
| anil.security | Anil | Security | [anil@employee.com](mailto:anil@employee.com) | 9999999999 | School Administration | Baltimore |
| apoorvaemployee | Apoorva | Employee | [apoorva@employee.com](mailto:apoorva@employee.com) | 9999999999 | School Administration | Baltimore |
| anika.employee | Anika | Employee | [anika@employee.com](mailto:anika@employee.com) | 9999999999 | School Administration | Baltimore |
| alpana.accounts | Alpana | Accounts | [alpana@employee.com](mailto:alpana@employee.com) | 9999999999 | School Administration | Baltimore |
| arohi.buildings | Arohi | Buildings | [arohi@employee.com](mailto:arohi@employee.com) | 9999999999 | School Administration | Baltimore |
| anindyo.employee | Anindyo | Employee | [anindyo@employee.com](mailto:anindyo@employee.com) | 9999999999 | School Administration | Baltimore |
| abhinay.employee | Abhinay | Employee | [abhinay@employee.com](mailto:abhinay@employee.com) | 9999999999 | School Administration | Baltimore |
| alok.employee | Alok | Employee | [alok@employee.com](mailto:alok@employee.com) | 9999999999 | School Administration | Baltimore |
| anish.admin | Anish | Admin | [anish@employee.com](mailto:anish@employee.com) | 9999999999 | School Administration | Salem |
| abhilash.employee | Abhilash | Employee | [abhilash@employee.com](mailto:abhilash@employee.com) | 9999999999 | School Administration | Salem |
| aditi.security | Aditi | Security | [aditi@employee.com](mailto:aditi@employee.com) | 9999999999 | School Administration | Salem |
| akruti.employee | Akruti | Employee | [akruti@employee.com](mailto:akruti@employee.com) | 9999999999 | School Administration | Salem |
| arpita.accounts | Arpita | Accounts | [arpita@employee.com](mailto:arpita@employee.com) | 9999999999 | School Administration | Salem |
| anirudh.accounts | Anirudh | Accounts | [anirudh@employee.com](mailto:anirudh@employee.com) | 9999999999 | Group Accounts | Baltimore |
| aparajita.finance | Aparajita | Finance | [aparajita@employee.com](mailto:aparajita@employee.com) | 9999999999 | School Administration | Salem |
|  |  |  |  |  |  |  |

Clean up the “Employee Details” import set table. Click on the “Cleanup” button. Cleanup only data.



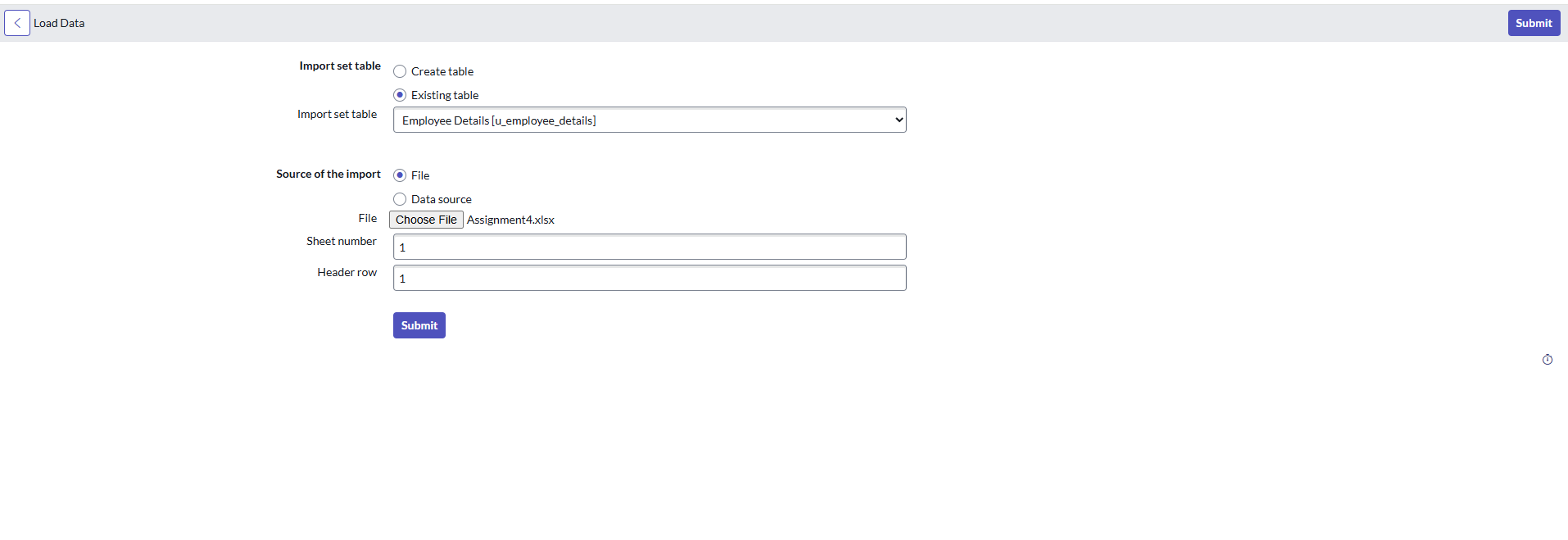
Go back to “Load Data”

A screenshot of a computer

AI-generated content may be incorrect.

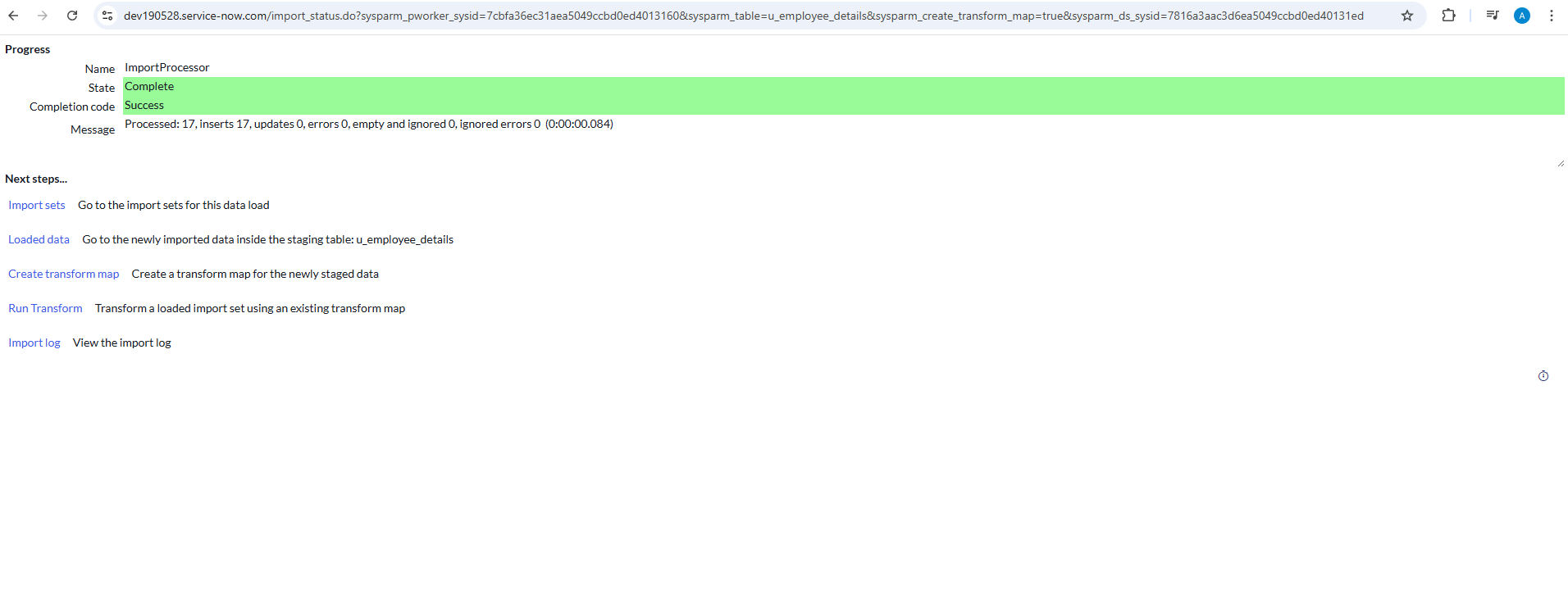


Select Existing Table and use the u\_employee\_details import set table. Select the edited excel sheet and click on “Submit”.





Select “Run Transform” since Transform Map is already in place with appropriate Coalesce fields selected.



Select “Transform”

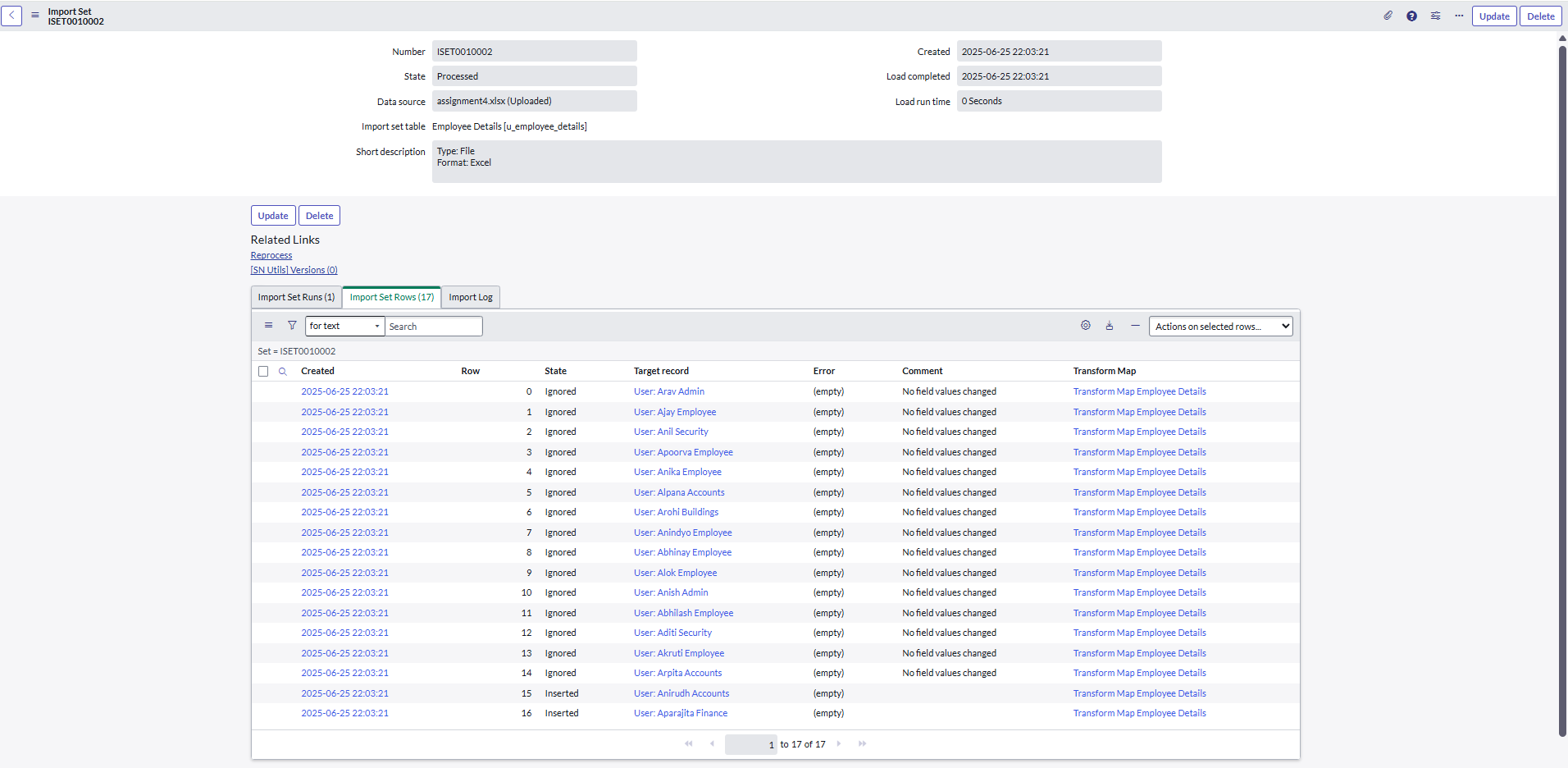
A screenshot of a computer

AI-generated content may be incorrect.

A green and white rectangle

AI-generated content may be incorrect.

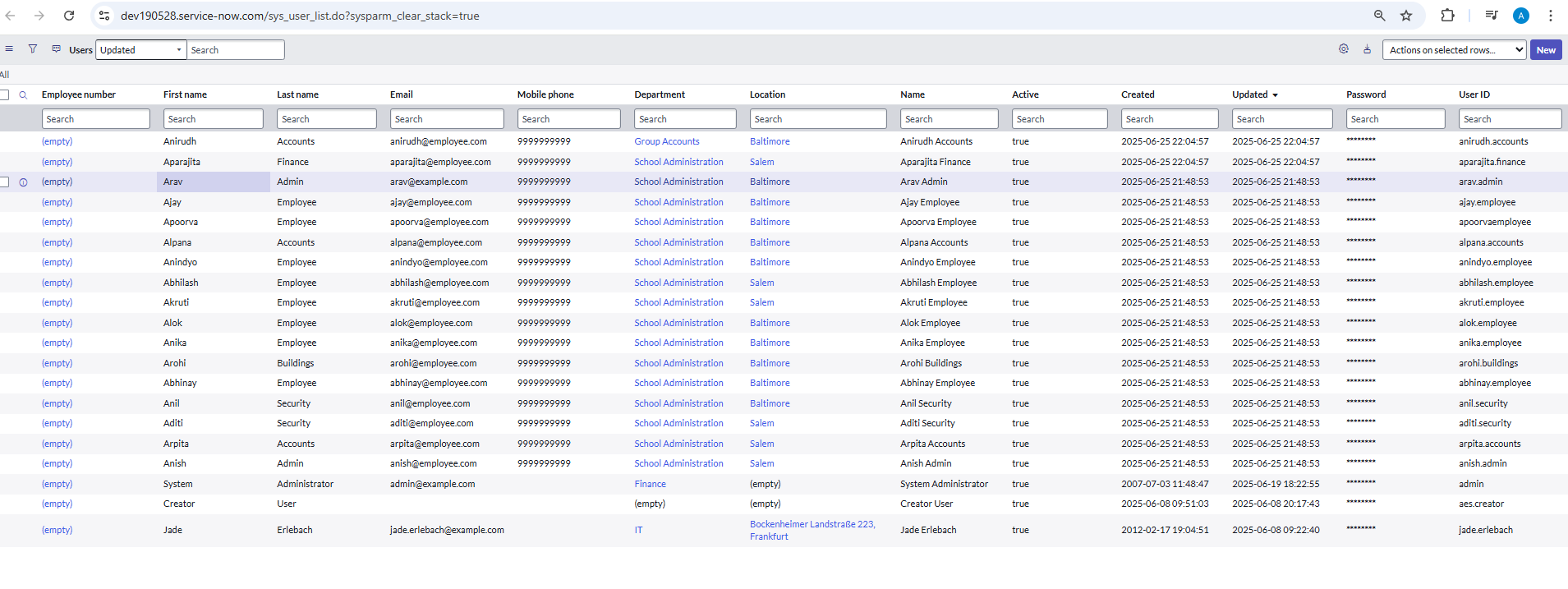
Click on the import set ISET0010002. Notice that duplicate rows are ignored during transform. Only new rows are inserted.





Check User table and note the two new users added

Check Department table and note that only 1 new department is added





A screenshot of a computer

AI-generated content may be incorrect.

