

## CSA SCENARIO-CASES

- Import your batch members as 20 users into your personal developer environment through import set

- Import data – Excel Format

	A	B	C	D	E	F	G	H	I
1	user\_name	first\_name	last\_name	email	phone				
2	ishan.bannerjee	Ishan	Bannerjee	[ishan@example.com](mailto:ishan@example.com)	9000000				
3	hariraj.p	Hariraj	P	[hari@example.com](mailto:hari@example.com)	9000001				
4	jithendra.s	Jithendra	S	[jithendra@example.com](mailto:jithendra@example.com)	9000002				
5	dheeraj.selva.a	Dheeraj	Selva A	[dheeraj@example.com](mailto:dheeraj@example.com)	9000003				
6	yeshwant.reddy	Yeshwant Reddy		[yesh@example.com](mailto:yesh@example.com)	9000004				
7	kalyan.kumar	Kalyan	Kumar	[kalyan@example.com](mailto:kalyan@example.com)	9000005				
8	sarvagnya.p	Sarvagnya P		[sarvagnya@example.com](mailto:sarvagnya@example.com)	9000006				
9	jayabaaskar.baskaran	Jayabaaskar Baskaran		[jb@example.com](mailto:jb@example.com)	9000007				
10	nisha.d	Nisha	D	[nisha@example.com](mailto:nisha@example.com)	9000008				
11	jeeva.prakasam	Jeeva	Prakasam	[jeeva@example.com](mailto:jeeva@example.com)	9000009				
12	harsha.manoj	Harsha	Manoj	[harsha@example.com](mailto:harsha@example.com)	9000010				
13	koturi.tarun	Koturi	Tarun	[tarun@example.com](mailto:tarun@example.com)	9000011				
14	kollati.yuvraj	Kollati	Yuvraj	[yuvraj@example.com](mailto:yuvraj@example.com)	9000012				
15	raam.vignesh	Ram	Vignesh	[ramvignesh@example.com](mailto:ramvignesh@example.com)	9000013				
16	bonthalakoti.revathi	Bonthalakoti	Revathi	[revathi@example.com](mailto:revathi@example.com)	9000014				
17	hari.ganesh	Hari	Ganesh	[hari@example.com](mailto:hari@example.com)	9000015				
18	kalva.praneeth	Kalva	Praneeth	[praneeth@example.com](mailto:praneeth@example.com)	9000016				
19	veluri.rishikesh	Veluri	Rishikesh	[rishikesh@example.com](mailto:rishikesh@example.com)	9000017				
20	naga.harsha	Naga	Harsha	[nagarsha@example.com](mailto:nagarsha@example.com)	9000018				
21	bhaskar.reddy	Bhaskar	Reddy	[baskarreddy@example.com](mailto:baskarreddy@example.com)	9000019				

- To import data from excel sheet : All > Import Sets > Load Data

Import set table  Create table  Existing table  
Import set table

Source of the import  File  Data source  
File  2...lsx  
Sheet number   
Header row

The screenshot shows a ServiceNow browser interface. The top navigation bar includes icons for back, forward, search, and user profile. The address bar shows the URL: dev321122.service-now.com/n... . The main content area displays a "Progress" section for an import job named "ImportProcessor". The job is marked as "Complete" with a "Success" completion code. A message at the bottom states: "Processed: 20, inserts 20, updates 0, errors 0, empty and ignored 0, ignored errors 0 (0:00:01.173)".

**Progress**

Name	ImportProcessor
State	Complete
Completion code	Success
Message	Processed: 20, inserts 20, updates 0, errors 0, empty and ignored 0, ignored errors 0 (0:00:01.173)

**Next steps...**

- [Import sets](#) Go to the import sets for this data load
- [Loaded data](#) Go to the newly imported data inside the staging table: u\_custom\_users\_import
- [Create transform map](#) Create a transform map for the newly staged data
- [Import log](#) View the import log

- Create Transform Map for Mapping fields

The screenshot shows a ServiceNow browser interface for creating a "Table Transform Map". The title bar says "Table Transform M...". The main form has the following fields:

* Name: UserTransformMap	Application: Global
* Source table: custom users import [...]	Created: 2025-11-26 22:20:53
Active: <input checked="" type="checkbox"/>	* Target table: User [sys_user]
Run business rules: <input checked="" type="checkbox"/>	Order: 100
Enforce mandatory fields: No	Run script: <input type="checkbox"/>
Copy empty fields: <input type="checkbox"/>	
Create new record on empty coalesce fields: <input type="checkbox"/>	

At the bottom, there are "Copy", "Update", and "Delete" buttons.

**Related Links**

- [Auto Map Matching Fields](#)
- [Mapping Assist](#)
- [Validate Coalesce Fields](#)
- [Run Point Scan](#)

- Using Mapping Assist Map fields properly

The screenshot shows the ServiceNow Mapping Assist interface. On the left, under 'Source: custom users import', there is a list of fields: Comment, Created, Created by, Error, Import set run, Row, Set, State, Sys ID, and Tags. In the center, the 'Field Map' section contains a grid mapping source fields to target fields:

Source field	Target field
last_name	Last name
user_name	User ID
phone	Business phone
first_name	First name
email	Email

On the right, under 'Target: User', there is a list of fields: Active, Avatar, Building, Calendar integration, City, Company, Cost center, Country code, Date format, and Default perspective. At the bottom, there are 'Save' and 'Cancel' buttons.

## Data Viewer

Show  All Fields  Mapped Fields

custom users import		User	
Field	Value	Field	Value
last_name	Baskaran	Last name	user
user_name	jayabaaskar.baskaran	User ID	survey.user
phone	9,000,007	Business phone	
first_name	Jayabaaskar	First name	survey
email	jb@example.com	Email	survey.user@email.com

- Check the Coalesce Field true for the Unique fields to make sure there are no duplicates in the future and to update

The screenshot shows the ServiceNow Field Maps interface. At the top, there are buttons for 'Copy', 'Update', and 'Delete'. Below is a 'Related Links' section with links to Auto Map Matching Fields, Mapping Assist, Validate Coalesce Fields, Transform, Index Coalesce Fields, and Run Point Scan. The main area has tabs for 'Field Maps (5)' and 'Transform Scripts'. The 'Field Maps (5)' tab displays a table of mappings:

Source field	Target field	Coalesce
u_last_name	last_name	false
u_user_name	user_name	true
u_phone	phone	false
u_first_name	first_name	false
u_email	email	false

At the bottom, there are navigation buttons for page 1 of 5.

- Run the Transform Map Created

The screenshot shows two ServiceNow browser windows. The top window is titled 'Specify Import set and Transform map' and displays the 'Import set' dropdown set to 'ISET0010002 - u\_custom\_users\_import (...'. It shows 'Available maps' on the left and 'Selected maps, run in order' on the right, which contains a single item: 'UserTransformMap - sys\_user'. Below this is a 'Transform' button. The bottom window is titled 'Transform' and shows 'Progress' details: Name: Transforming: ISET0010002, State: Complete, Completion code: Success, Message: Transformation complete. It also lists 'Next steps...' including links to 'ISET0010002', 'Transform history', and 'Import log'.

- Check the User table for imported Users and details.

The screenshot shows a 'Users' table in ServiceNow. The table has columns: User ID, Name, Email, Active, Created, and Updated. The data includes:

User ID	Name	Email	Active	Created	Updated
ishan.bannerjee	Ishan Bannerjee	ishan@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
nisha.d	Nisha D	nisha@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
jithendra.s	Jithendra S	jithendra@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
kollati.yuvaraj	Kollati Yuvaraj	yuvraj@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
sarvagnya.p	Sarvagnya P	sravagnya@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
kalva.praneeth	Kalva Praneeth	praneeth@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
jayabaaskar.baskaran	Jayabaaskar Baskaran	jb@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31

Screenshot of a web browser displaying a list of users from the ServiceNow platform. The URL in the address bar is dev321122.service-now.com/now... . The page title is "Users".

The table has the following columns:

- User ID
- Name
- Email
- Active
- Created
- Updated

The data in the table is as follows:

User ID	Name	Email	Active	Created	Updated
kalva.praneeth	Kalva Praneeth	praneeth@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
jayabaaskar.baskaran	Jayabaaskar Baskaran	jb@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
veluri.rishikesh	Veluri Rishikesh	rishikesh@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
jeeva.prakasam	Jeeva Prakasam	jeeva@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
dheeraj.selva.a	Dheeraj Selva A	dheeraj@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
raam.vignesh	Ram Vignesh	ramvignesh@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
yeshwant.reddy	Yeshwanth Reddy	yesh@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
bonthalakoti.revathi	Bonthalakoti Revathi	revathi@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
naga.harsha	Naga Harsha	nagaharsha@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
harsha.manoj	Harsha Manoj	harsha@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
hariraj.p	Hariraj P	hari@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
koturi.tarun	Koturi Tarun	tarun@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
kalyan.kumar	Kalyan Kumar	kalyan@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
hari.ganesh	Hari Ganesh	hari@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
bhaskar.reddy	Bhaskar Reddy	baskarreddy@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31

Pagination at the bottom shows "1 to 20 of 652".

2. Create 3 groups called Batch-1 and Batch-2 and Batch-3.

- Go to Groups under User Administration – All > User Administration > Groups > New
- Table Name: sys\_user\_group

**Group - Batch 1**

Name	Batch 1	Group email	batch1@example.com
Manager		Parent	
Description			

**Roles**    **Group Members**    **Groups**

Created    Search    Edit...

Group = Batch 1

Created	Role	Granted by	Inherits
 No records to display			

**Groups**

Groups    Updated    Search    Actions on selected rows...    New

	Name	Description	Active	Manager	Parent	Updated
<input checked="" type="checkbox"/>	Search	Search	Search	Search	Search	Search
<input checked="" type="checkbox"/>	Batch-3	CSA B3	true	(empty)	(empty)	2025-11-26 22:58:02
<input checked="" type="checkbox"/>	Batch-2	CSA B2	true	(empty)	(empty)	2025-11-26 22:57:52
<input checked="" type="checkbox"/>	Batch-1	CSA B1	true	(empty)	(empty)	2025-11-26 22:57:47
		Batch Members	true	(empty)	(empty)	2025-11-26 22:00:04
		Help Desk	true	ATF Change Management	(empty)	2025-09-15 02:50:32
		RMA Approvers	Responsible for Return Material Authoriz...	true	(empty)	2025-09-08 14:12:50

3. Create Assignment lookup rules when category is software the assignment group should be software.

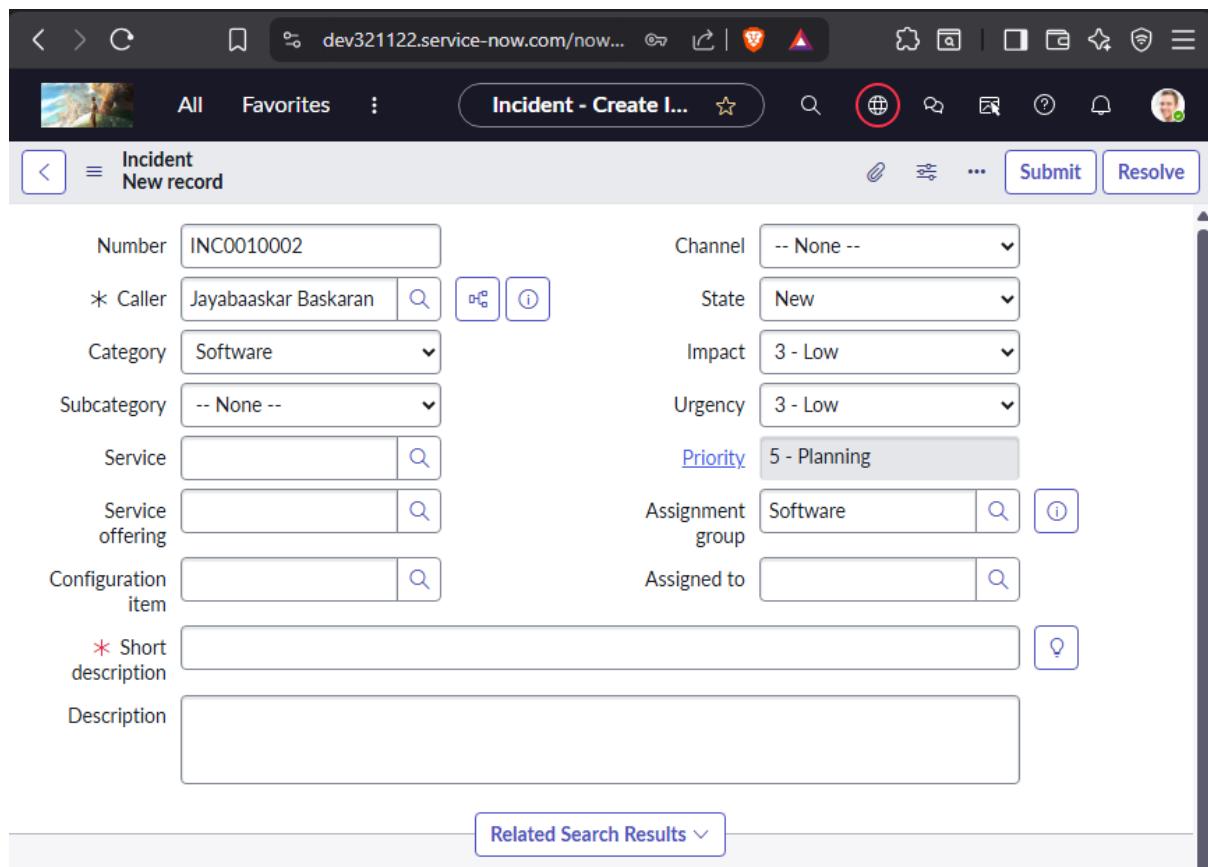
- Go to Assignment Data Lookup table – table name – dl\_u\_assignment
- Type dl\_u\_assignment.list on All Filters and Enter to open the table.

Order	Category	Subcategory	Configuration Item	Location	Assigned To	Assignment Group
100	Software	(empty)	(empty)	(empty)	Fred Luddy	(empty)

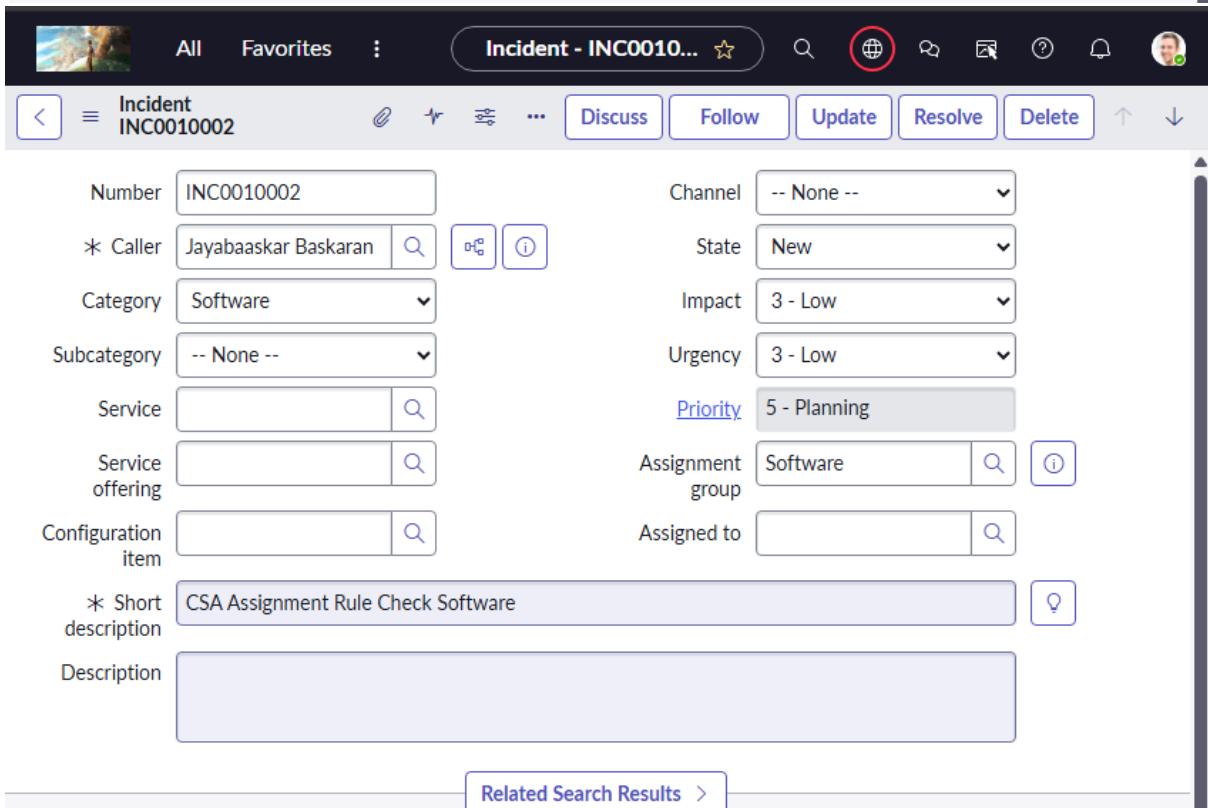
- Create New Data Lookup for Software

Order	Category	Subcategory	Configuration Item	Location	Assigned To	Assignment Group
100	Software	(empty)	(empty)	(empty)	Fred Luddy	(empty)
200	Software	(empty)	(empty)	(empty)	(empty)	Software

- Check If the assignment work in Incident Table



The screenshot shows the 'Incident - Create I...' page. The 'Number' field is set to 'INC0010002'. Other fields include 'Channel' (None), 'State' (New), 'Impact' (3 - Low), 'Urgency' (3 - Low), 'Priority' (5 - Planning), 'Assignment group' (Software), and 'Assigned to' (empty). The 'Short description' field contains 'CSA Assignment Rule Check Software'.

The screenshot shows the 'Incident - INC0010002' page. The 'Number' field is 'INC0010002'. The 'Short description' field also contains 'CSA Assignment Rule Check Software'.

#### 4. Create a custom table to store Credit card request details.

- custom table name – u\_credit\_card\_request\_details.

Credit Card Request Details

Created

No records to display

- Along with an application with the same name is created

Title	Table	Active	Filter	Order	Link type	Def
All Requests	Credit Card Request Details [u_credit_card_request_details]	true			List of Records	
Create New	Credit Card Request Details [u_credit_card_request_details]	true		10	URL (from Arguments:)	

- Module –Create New & All Request & My Request

Application menu = Credit Card Request Details

<input type="checkbox"/>	Title	Table	Active	Filter	Order ▲	Link type	De
	All Requests	Credit Card Request Details [u_credit_card_request_details]	true			List of Records	
	Create New	Credit Card Request Details [u_credit_card_request_details]	true		10	URL (from Arguments:)	

1 to 2 of 2

Module - All Request... ★

All Favorites : Module - All Request... ★ 🔍 网站地图 🔍 网站地图 ... Update Delete

Title: All Requests Application: Global  ⓘ

Application menu: Credit Card Request I 🔍  ⓘ

Order:

Hint:

Display name: All Requests

Visibility Link Type

Roles  ⓘ  
u\_credit\_card\_request\_details\_user

Override application menu roles

Active

Update Delete

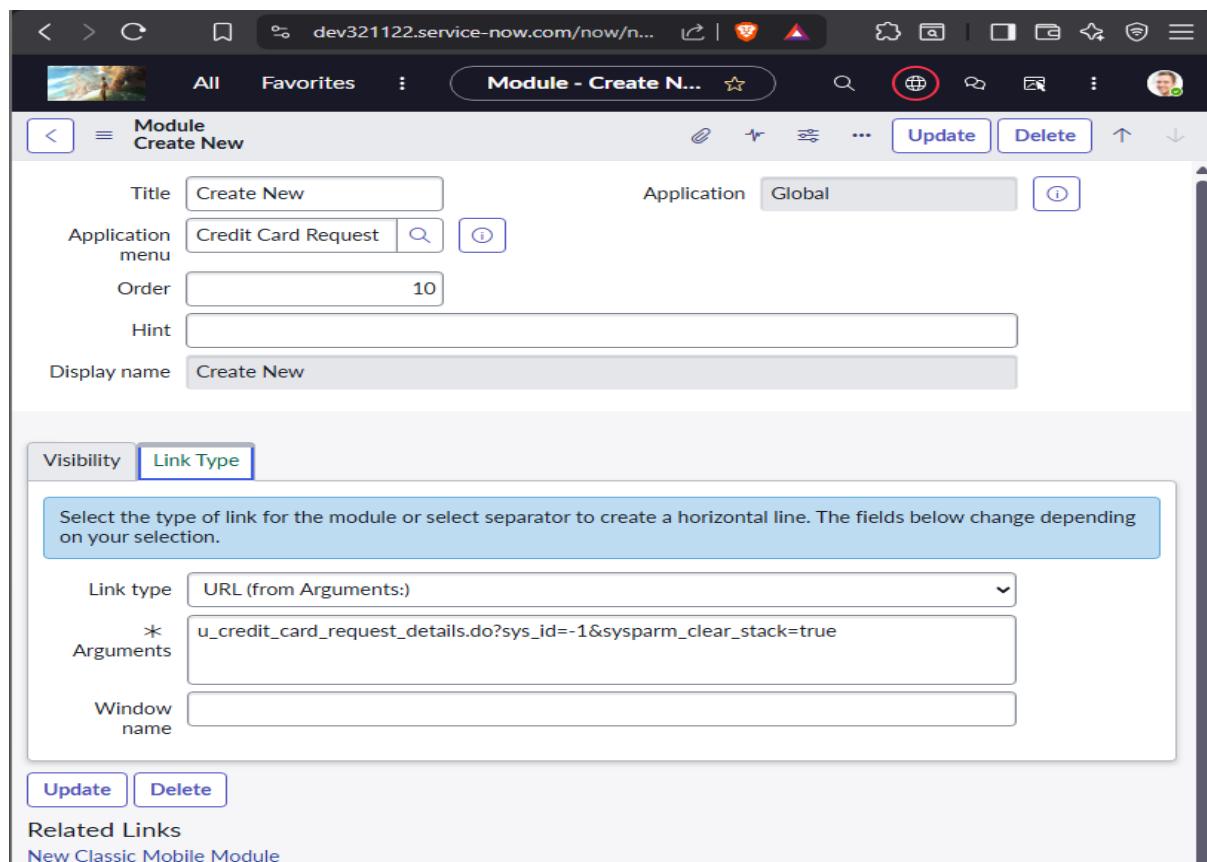
Related Links

[New Classic Mobile Module](#)

Classic Mobile Modules

<input type="checkbox"/>	Name	Active	Application menu	Filter	Order ▲	Path	Path Relative To Root
							No records to display

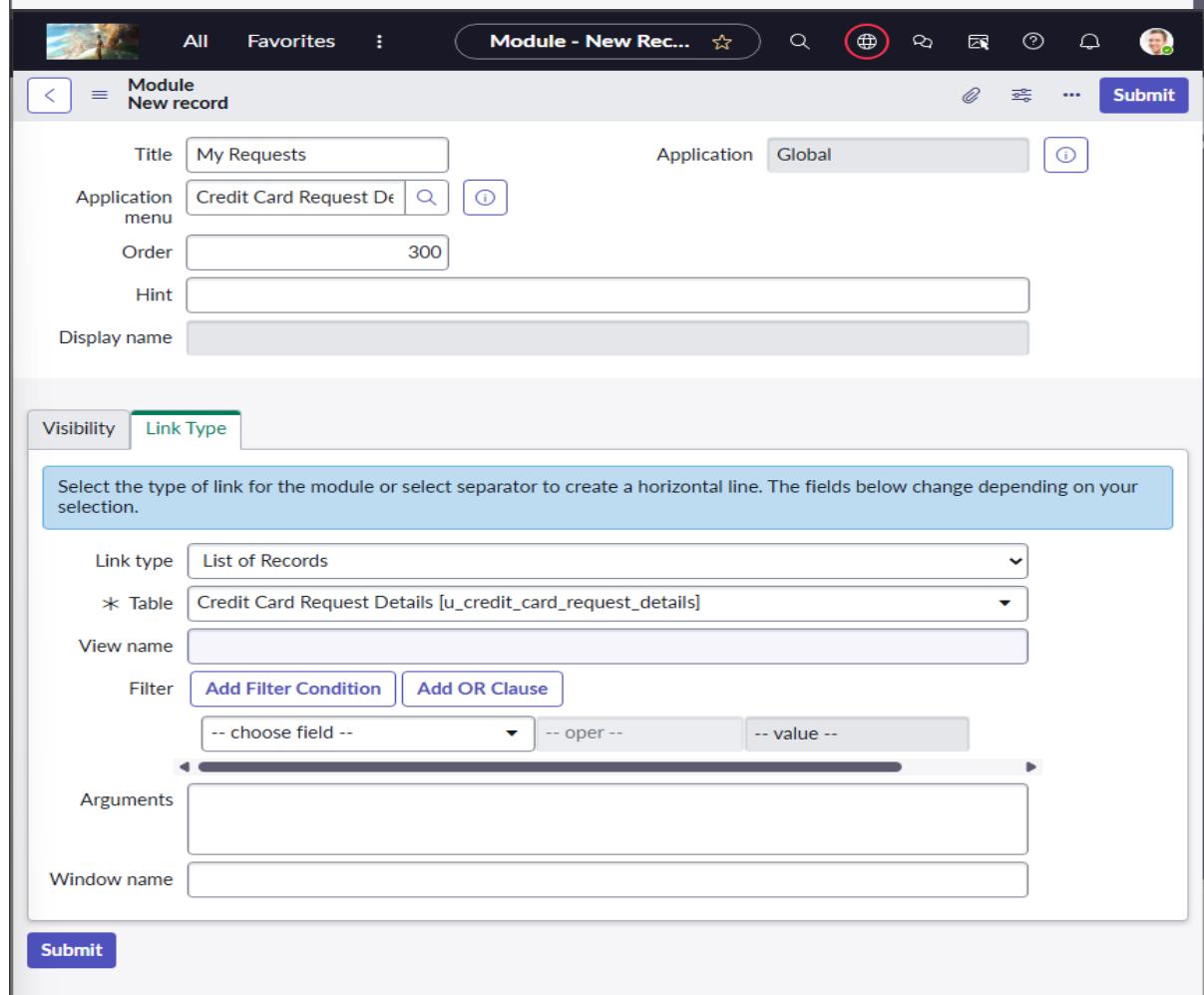
- Add Link type URL and in Arguments add the Form link Suffix for Create New



The screenshot shows the 'Module - Create New' configuration page. The 'Link Type' tab is selected. Under 'Link type', it is set to 'URL (from Arguments:)'. The 'Arguments' field contains the value 'u\_credit\_card\_request\_details.do?sys\_id=-1&sysparm\_clear\_stack=true'. There are also fields for 'Window name' and other module settings like Title, Application menu, Order, Hint, and Display name.

**Related Links:**

- New Classic Mobile Module

The screenshot shows the 'Module - New Record' configuration page. The 'Link Type' tab is selected. Under 'Link type', it is set to 'List of Records'. The 'Table' dropdown is set to 'Credit Card Request Details [u\_credit\_card\_request\_details]'. There are fields for 'View name', 'Filter' (with options 'Add Filter Condition' and 'Add OR Clause'), 'Arguments', and 'Window name'. A 'Submit' button is visible at the bottom.

Module	Title	Table	Active	Filter	Order ▲	Link type	Device type
Application menu = Credit Card Request Details							
<input checked="" type="checkbox"/>	All Requests	Credit Card Request Details [u_credit_card_request_details]	true		10	List of Records	
<input checked="" type="checkbox"/>	Create New	Credit Card Request Details [u_credit_card_request_details]	true		20	URL (from Arguments:)	
<input checked="" type="checkbox"/>	My Requests	Credit Card Request Details [u_credit_card_request_details]	true		30	List of Records	

- o Custom Role Creation credit\_admin for application

Role - credit\_admin

Name	credit_admin	Application	Global
		Elevated privilege	<input type="checkbox"/>
Description			

Application Menu - ...

Roles

Available

- workflow\_creator
- workflow\_publisher
- workflow\_report\_viewer
- workspace.activity\_stream\_restrict\_c
- workspace\_admin
- workspace\_configure\_form\_field
- workspace\_list\_admin
- workspace\_user
- u\_credit\_card\_request\_details\_user

Selected

- credit\_admin

**Done**

Specifies the menu category, which defines the navigation menu style. The default value is Custom Applications.

Application Menu - ...

An application menu is a group of modules in the application navigator. Choose the roles that are required to access the application and add or remove modules in the related list below. [More Info](#)

* Title	Credit Card Request Details	Application	Global
Active <input checked="" type="checkbox"/>			
Restricts access to the specified roles. Otherwise, all users can view the application menu when it is active.			
Roles	<input type="button" value="edit"/>		
credit_admin			

Specifies the menu category, which defines the navigation menu style. The default value is Custom Applications.

- Required fields: 1) Name: refer to user table. 2) Phone Number: String 3) Email: String 4) City: Choice type –All metro cities. 5) Credit card Type: any 4 choices as your wish. 6) Accept Terms and Condition: check box
- Using Either Form Builder or Form Layout create Fields add choices

Credit Card Request Details  
New record

Name

Phone Number

Email

City

Credit card Type

Accept Terms  and Condition

**Submit**

- Create catalog to request credit card \*Required Variables 1) Name : refer to user table.-→mandatory 2) Phone Number : String 3) Email : String 4) City : Choice type –All metro cities. 5) Credit card Type: any 4 choices as your wish. 6) Accept Terms and Condition : check box→if not accepted then catalog should not be submitted and need to alert user “please accept the T&C”

Catalog Item - Req...

Name: Request Credit Card

Application: Global

Active:

Category: Credit Card Related

Fulfillment automation level: Unspecified

Item Details: Request Credit Card

Process Engine: Global

Picture: Global

Pricing: Global

Portal Settings:

- Request method: Request
- Hide 'Save as Draft':
- Hide 'Add to Cart':
- Hide Quantity:
- Hide Delivery time:
- Mandatory Attachment:
- Hide Attachment:

**Copy Try It Update Edit in Catalog Builder Delete**

Variables (7)	Variable Sets	Catalog UI Policies	Catalog Client Scripts (1)	Available For	Not Available For	Categories (1)
Catalogs (1)	Catalog Data Lookup Definitions	Related Articles	Related Catalog Items	Assigned Topics		
<input type="button" value="Order"/> <input type="text" value="Search"/> <input type="button" value="Actions on selected rows..."/> <input type="button" value="New"/>						
Catalog item = Request Credit Card						
Type	Question	Order	Mandatory			
Reference	Name	10	true			
Single Line Text	Phone Number	20	false			
Single Line Text	Email	30	false			
Multiple Choice	City	40	false			
Multiple Choice	Credit Card Type:	50	false			
Label	Terms and Conditions	60	false			
CheckBox	Accept Terms and Conditions	70	false			

dev321122.service-now.com/now/nav/ui...

**Catalog Client Script...**

**Catalog Client Scripts**  
Terms and Conditions

**Name:** Terms and Conditions  
**Applies to:** A Catalog Item  
**Active:**   
**UI Type:** All

**Application:** Global  
**Type:** onSubmit  
**Catalog item:** Request Credit Card  
**Applies on a Catalog Item view:**   
**Applies on Requested Items:**   
**Applies on Catalog Tasks:**

**Script:**

```

1 function onSubmit() {
2     if (_form.getValue('accept_terms_and_conditions') != 'true'){
3         alert('please accept the T&C');
4         return false;
5     }
6 }
```

Request Credit Card

\*Name  
Abraham Lincoln

Phone Number

Email  
abraham.lincoln@example.com

City  
 Chennai  
 Bangalore  
 Hyderabad  
 Mumbai  
 Kolkata  
 Delhi

Credit Card Type:  
 Visa Card  
 RuPay  
 MasterCard  
 Gold Card

Terms and Conditions  
 Accept Terms and Conditions

**Delivery Time:** 2 Days

**Request**

- Create Group Credit req approval group. Add the few user to this group.

All Favorites : Group - Credit Req... ☆

Group - Credit Request Approval

Name: Credit Request Approval

Manager: [Search]

Parent: [Search]

Description:

**Update Delete**

**Group Members (5)**

User: Adela Cervantsz, Abel Tuter, Aileen Mottern, Abraham Lincoln, Alejandra Prenatt

Actions on selected rows... New Edit...

1 to 5 of 5

5. Once user requested the credit card service it should go approval for Credit req approval group.

- 1) If approval approved then need to create a document verification task.
- 2) if rejected or completed then notify the user with rejected content or approved content.
- 3) if task completed successfully then create a delivery task.
- 4) if delivery task completed then need to close RITM and Req records.
- 5) if delivery task closed incomplete, then notify user req has not been processed and incomplete the req and ritm.
- 6) The Assignment group for above tasks should be i.e., Credit req approval group.

The screenshot shows the ServiceNow Workflow Studio interface with three tabs open: 'Workflow Studio', 'CreditCardReq Flow \* Global', and 'CreditCardReq Flow execution \* None'. The main view displays the 'EXECUTION DETAILS' for the 'CreditCardReq' flow. The table lists 16 steps:

Step	Action	Type	Status	Start time	Duration
1	Get Catalog Variables from Request Credit Card	Core Action	Completed	2025-11-27 04:04:02	13ms
2	Ask For Approval	Core Action	Completed	2025-11-27 04:04:02	722ms
3	If Request Approved	Flow Logic	Evaluated - True	2025-11-27 04:04:25	969ms
4	Send Notification	Core Action	Completed	2025-11-27 04:04:25	4ms
5	Create Catalog Task	Core Action	Completed	2025-11-27 04:04:25	819ms
6	If Document Verification Completed	Flow Logic	Evaluated - True	2025-11-27 04:06:16	146ms
7	Create Catalog Task	Core Action	Completed	2025-11-27 04:06:16	100ms
8	Wait For Condition	Core Action	Completed	2025-11-27 04:07:21	1ms
9	If Closed Complete	Flow Logic	Evaluated - False	2025-11-27 04:07:21	1ms
12	Else If Closed Incomplete	Flow Logic	Evaluated - True	2025-11-27 04:07:21	43ms
13	Send Notification	Core Action	Completed	2025-11-27 04:07:21	3ms
14	Update Record	Core Action	Completed	2025-11-27 04:07:21	39ms
15	Else If Request Rejected	Flow Logic	Not Run		
16	Send Notification	Core Action	Not Run		

## ■ Attaching the flow as Process Engine

Catalog Item - Request Credit Card

Name: Request Credit Card

Application: Global

Active:

Catalogs: Credit Card

Category: Credit Card Related

Fulfillment automation level: Unspecified

State: -- None --

Checked out: -- None --

Owner: System Administrator

Process Engine Tab selected.

Select the appropriate process engine for the catalog item. Only one engine can be selected.

Flow: CreditCardReq

Workflow:

Execution Plan:

## ■ email trigger for the user

Created	Recipients	Subject	Type	Notification type	User ID	Updated
2025-11-27 04:07:22	jb@example.com	Request Closed	send-ready	SMTP	(empty)	2025-11-27 04:07:22
2025-11-27 04:06:21	creditreqapproval@example.com	Catalog Task SCTASK0010004 has been assigned to group Credit Request Approval	send-ready	SMTP	(empty)	2025-11-27 04:06:21
2025-11-27 04:04:31	jb@example.com	Request Approved	send-ready	SMTP	(empty)	2025-11-27 04:04:31
2025-11-27 04:04:31	creditreqapproval@example.com	Catalog Task SCTASK0010003 has been assigned to group Credit Request Approval	send-ready	SMTP	(empty)	2025-11-27 04:04:31
2025-11-27 04:04:11	aileen.mottern@example.com	Requested Item RITM0010003 Approval Request	send-ready	SMTP	(empty)	2025-11-27 04:04:11
2025-11-27 04:04:11	abraham.lincoln@example.com	Requested Item RITM0010003 Approval Request	send-ready	SMTP	(empty)	2025-11-27 04:04:11
2025-11-27 03:57:21	jb@example.com	Request Rejected	send-ready	SMTP	(empty)	2025-11-27 03:57:21
2025-11-27 03:57:21	adela.cervantsz@example.com	Requested Item RITM0010003 has been rejected by	send-ready	SMTP	(empty)	2025-11-27 03:57:21

6. Create A dashboard called Incident Priorities. ADD a P1 P2 P3 report of count type to respective dashboard.

**P1**

Report Title : P1

Group by  
Active ▾ Additional group by

Stack by  
-- None --

Display data table  
Configure function field

Aggregation  
Count

[Set Value Formatting](#)

Max number of groups  
System Default  
 Show Other

Active	Incident Count	Percentage of Count
true	17	62.96%
false	10	37.04%
Total	27	100%

**P2**

Report Title : P2

Configure function field

Aggregation  
Count

Ask another question

Table: Incident [incident]

All > Priority = 2 - High

**P3**

Report Title : P3

Group by: Active

Stack by: -- None --

Display data table

Aggregation: Count

Set Value Formatting

Max number of groups: System Default

Show Other

Table: Incident [incident]

All > Priority = 3 - Moderate

Incident Count

Active	Incident Count	Percentage of Count
true	7	58.33
false	5	41.67
<b>Total</b>	<b>12</b>	<b>100%</b>

## ○ DashBoard

P2

P1

P3

Active	Incident Count	Percentage of Count
true	7	58.33%
false	5	41.67%
<b>Total</b>	<b>12</b>	<b>100%</b>

Configuration

Tab

Create Tab

Dashboard background color

Use custom background color

Apply Quick Layout

7. Create a P1 , P2 , P3 reports , schedule it and send to one user

- o Scheduling Report

The image consists of two vertically stacked screenshots of the ServiceNow web interface, both titled "Schedule an email containing this report".

**Screenshot 1 (Top): Configuration for Report P2**

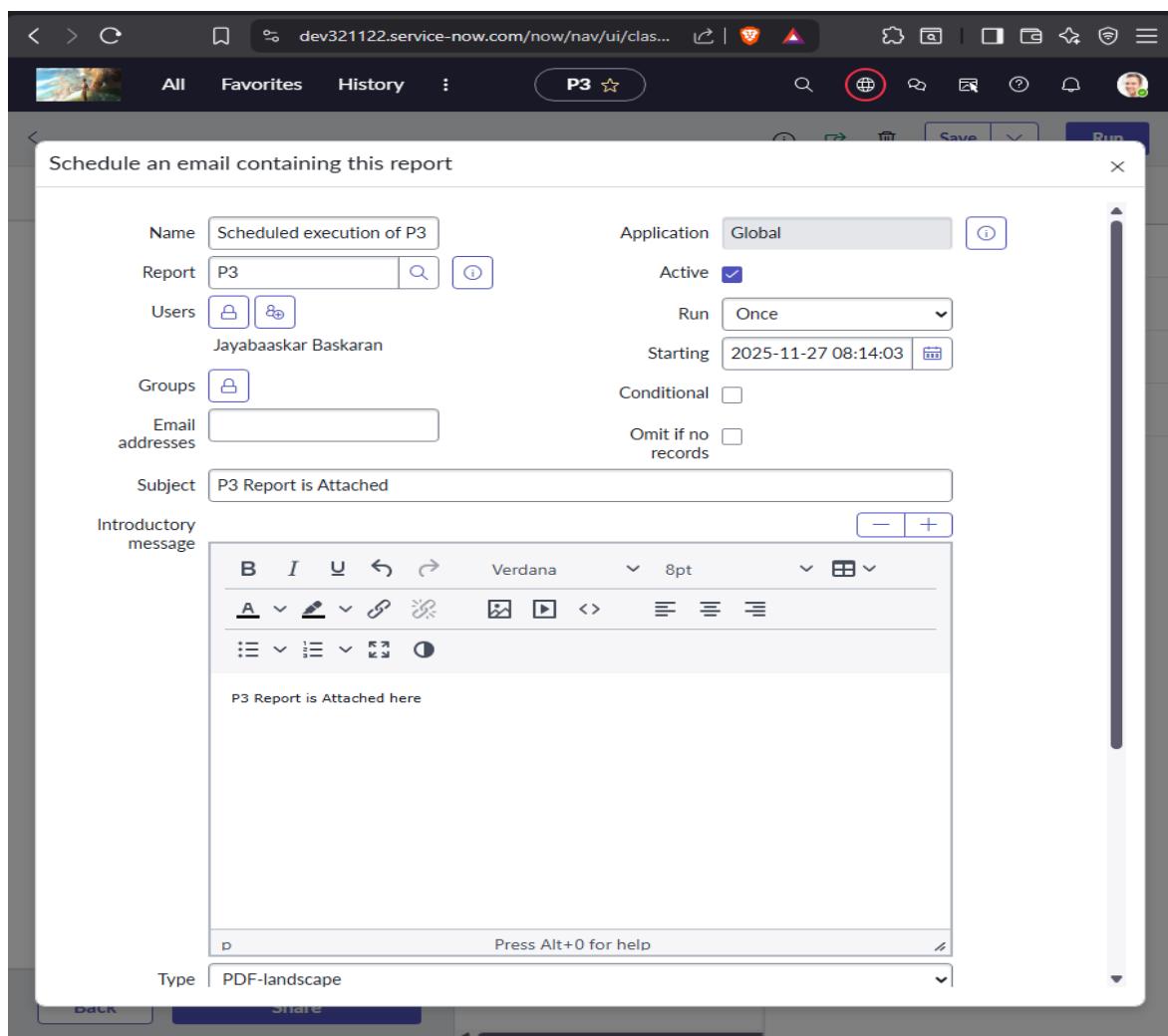
- Name:** Scheduled execution of P2
- Report:** P2
- Application:** Global
- Active:**
- Run:** about 3 minutes from now
- Starting:** 2025-11-27 08:09:49
- Conditional:**
- Omit if no records:**
- Subject:** Report P2 Attached
- Introductory message:** (Rich text editor)
 

The report is attached P2

**Screenshot 2 (Bottom): Configuration for Report P1**

- Name:** Scheduled execution of P1
- Report:** P1
- Application:** Global
- Active:**
- Run:** Once
- Starting:** 2025-11-27 08:11:59
- Conditional:**
- Omit if no records:**
- Subject:** P1 Report is Attached
- Introductory message:** (Rich text editor)
 

P1 Report is Attached
- Type:** PDF-landscape



- Check if triggered on scheduled time

All	Favorites	History	Emails	Created	Search	Actions on selected rows...	New
All > Created on Today							
<input type="checkbox"/>	<input type="checkbox"/>	Created	Recipients	Subject	Type	Notification type	User ID
			Search	Search	Search	Search	Search
<input checked="" type="checkbox"/>	2025-11-27 08:14:09	jb@example.com	P3 Report is Attached	send-ready	SMTP	(empty)	2025-11-27 08:14:09
<input checked="" type="checkbox"/>	2025-11-27 08:12:00	jb@example.com	P1 Report is Attached	send-ready	SMTP	(empty)	2025-11-27 08:12:00
<input checked="" type="checkbox"/>	2025-11-27 08:09:52	jb@example.com	Report P2 Attached	send-ready	SMTP	(empty)	2025-11-27 08:09:52
	2025-11-27 04:07:22	jb@example.com	Request Closed	send-ready	SMTP	(empty)	2025-11-27 04:07:22
	2025-11-27 04:06:21	creditreqapproval@example.com	Catalog Task SCTASK0010004 has been assigned to group Credit Request Approval	send-ready	SMTP	(empty)	2025-11-27 04:06:21
	2025-11-27 04:04:31	jb@example.com	Request Approved	send-ready	SMTP	(empty)	2025-11-27 04:04:31
	2025-11-27 04:04:31	creditreqapproval@example.com	Catalog Task SCTASK0010003 has been assigned to group Credit	send-ready	SMTP	(empty)	2025-11-27 04:04:31

**8. Create a report to Find a software category incident in bar graph and share it and publish it to groups**

The screenshot shows the ServiceNow report configuration interface. On the left, the 'Style' tab is selected under 'Configure'. In the 'CONDITIONS' section, a condition is defined: 'Category is Software'. A bar chart titled 'Software Category Incidents' is displayed, showing two bars: one blue bar for 'true' with a value of 9, and one orange bar for 'false' with a value of 5. Below the chart is a summary table:

Active	Incident Count	Percentage of Count
true	9	64.29%
false	5	35.71%
<b>Total</b>	<b>14</b>	<b>100%</b>

On the right, a 'Sharing settings' dialog is open, showing 'Groups and Users' selected under 'Visible to'. The 'Groups' tab is active, displaying a list containing 'Software'. An 'OK' button is at the bottom right of the dialog.

9. Create Notification to opened by user when incident is inserted.
- Sub : IncXXXX123 has been created.
  - Body : Hi XXXX, The inc –number has been created by you. Feel free to reach to system admin on related to Incident. Thanks!

The screenshot shows the 'Notification - NotifiOpenedbyUser' configuration page in ServiceNow. The top navigation bar includes 'All', 'Favorites', 'History', 'Notification - NotifiOpenedbyUser', 'Search', 'Preview Notification', 'Update', and 'Delete' buttons.

**General Settings:**

- Name: NotifiOpenedbyUser
- Application: Global
- Table: Incident [incident]
- Active:
- Mandatory:
- Category: Uncategorized

A note states: "A digest combines and summarises multiple updates for a single or multiple target records within a set time interval." There is a checkbox for "Allow Digest".

**When to send:** Set to "Record inserted or updated" (Selected) and "Updated" (unchecked). Conditions: "Inserted" (Selected), "Add Filter Condition" (button), "Add OR Clause" (button).

**Who will receive:** This tab is selected. It shows options for "Users" (with icons for lock and user), "Groups" (with icon for lock), and "Subscribable" (checkbox). Below these are "Users/Groups in fields" and "Opened by" (both with icons for lock).

**What it will contain:** This tab is also present but not selected.

**Email Template Configuration:**

- Email template: Unsubscribe and Preferences
- Subject: \${number} has been Created

**Message HTML Editor:**

Rich text editor controls are available. The message body contains:

```

Hi ${opened_by},  

The inc - ${number} has been created by you. Feel free to reach to system admin on related to Incident.  

Thanks!

```

On the right, a "Select variables:" panel shows a "Fields" section with a plus sign icon.

- Email Trigger check – All > System Logs > Emails.

The screenshot shows a web application interface for managing system logs, specifically focusing on emails. At the top, there's a navigation bar with links for 'All', 'Favorites', 'History', and a search bar. Below this is a toolbar with icons for 'Emails' (highlighted), search, and other actions. A dropdown menu shows 'Created' and a 'Search' field. On the right, there are buttons for 'Actions on selected rows...' and 'New'.

The main area displays a table of emails. The columns are: Created (dropdown with 'Search'), Recipients (with 'Search' button), Subject (with 'Search' button), Type (with 'Search' button), Notification type (with 'Search' button), User ID (with 'Search' button), and Updated (with 'Search' button). There are six rows of data:

	Created	Recipients	Subject	Type	Notification type	User ID	Updated
<input checked="" type="checkbox"/>	2025-11-27 09:02:42	admin@example.com	INC0010003 has been Created	send-ready	SMTP	(empty)	2025-11-27 09:02:42
	2025-11-27 09:02:42	jb@example.com	Incident INC0010003 was created	send-ready	SMTP	(empty)	2025-11-27 09:02:42
	2025-11-27 08:14:09	jb@example.com	P3 Report is Attached	send-ready	SMTP	(empty)	2025-11-27 08:14:09
	2025-11-27 08:12:00	jb@example.com	P1 Report is Attached	send-ready	SMTP	(empty)	2025-11-27 08:12:00
	2025-11-27 08:09:52	jb@example.com	Report P2 Attached	send-ready	SMTP	(empty)	2025-11-27 08:09:52
	2025-11-27 04:07:22	jb@example.com	Request Closed	send-ready	SMTP	(empty)	2025-11-27 04:07:22
			Catalog Task SCTASK0010004				

Below the table, a modal window titled 'Email - INC001000...' shows the details of the first email. The subject is 'INC0010003 has been Created'. The preview email body contains:

```

Hi System Administrator,  

The inc – INC0010003 has been created by you. Feel free to reach to system admin on related to Incident.  

Thanks!

```

At the bottom of the modal, there are links for 'Unsubscribe | Notification Preferences' and 'Ref:MSG0000454\_1aUvhgeLCMjNzUktqbnv'. The URL in the address bar is partially visible at the bottom.

10. Create a notification on update of incident to opened by. with the same body content as above.

- Sub : IncXXXX123 has been updated.
- Body : Hi XXXX, The inc –number has been commented out. Feel free to reach out Assigned to {assigned to person name} Thanks!

The screenshot shows the configuration of a notification rule named "NotifiOnUpdate" for the "Incident [incident]" table. The rule is set to "Global" application and is active. It is categorized as "Uncategorized".

**When to send:** Set to "Record inserted or updated" and "Updated" is checked. There is an "Inserted" checkbox which is unchecked. Below this, there are buttons for "Add Filter Condition" and "Add OR Clause", along with dropdown menus for selecting fields, operators, and values.

**Who will receive:** The email template is set to "Unsubscribe and Preferences". The subject is "\${number} has been updated.".

**What it will contain:** The message content is defined as follows:

```

Hi XXXX,
The inc: ${number} has been commented out.
Feel free to reach out Assigned to ${assigned_to}
Thanks!

```

A rich text editor toolbar is available for editing the message content. To the right of the message area, a "Select variables:" dropdown menu is open, listing various system variables such as Knowledge, Last, reopened at, Last, reopened by, Location, Made SLA, Notify, Number, On hold, reason, Opened, Order, Origin, Origin table, Parent, Parent Incident, and Priority. The variable "Last" is currently selected.

- Mail Log Check

The screenshot shows the ServiceNow classic UI interface. At the top, there's a navigation bar with links for All, Favorites, History, Workspaces, and a search bar. Below the navigation is a toolbar with icons for Email, Search, and Actions on selected rows.

The main area displays a table titled "Emails" with the following columns: Created, Recipients, Subject, Type, Notification type, User ID, and Updated. The table lists several email entries, all of which were created on November 27, 2025, at various times between 09:36:51 and 08:14:09. Most emails are sent to admin@example.com or jb@example.com and relate to incidents (INC0010003).

A modal window titled "Email - INC0010003 has been updated..." is open in the center. It shows the subject of the email: "INC0010003 has been updated." and the preview content:

```

Hi System Administrator,  

The inc -INC0010003 has been commented out  

Feel free to reach out Assigned to Andrew Och  

Thanks!

```

At the bottom of the modal, there are "Update" and "Delete" buttons, along with a "Close" button.

Below the main table, there's a section for "Related Links" and "Email Log (1)". The "Email Log" table has columns for Created, Level, Message, and Source. It shows one entry from November 27, 2025, at 09:47:31, categorized as "Information". The message details a notification update, and the source is EMAIL.6873631483213610e99ce4d0deaad3e5.

11. Create a Basic Knowledge article and publish them and retire them. Content: off your own choice.

The screenshot shows the 'Knowledge - New record' page in ServiceNow. The page has several input fields:

- Number:** KB0010001
- \* Knowledge base:** Knowledge
- Category:** (empty)
- Scheduled publish date:** 2025-11-28 19:59:27
- Valid to:** 2025-12-31
- Version:** (empty)
- Article type:** HTML
- Workflow:** Draft
- Source task:** (empty)
- Attachment link:**
- Display attachments:**
- \* Short description:** Basic Knowledge Article
- Article body:** A rich text editor containing the following text:
 

A Knowledge Article is a documented piece of information stored in the Knowledge Base (KB) of ServiceNow. It helps users find answers, solve issues, and follow procedures without needing support teams.

Think of it as a self-help document for employees and customers.

At the bottom are 'Submit' and 'Search for Duplicates' buttons.

The screenshot shows the 'Knowledge - KB0010001 v0.02' page in ServiceNow. The page displays the following information:

- Number:** KB0010001
- Knowledge base:** Knowledge
- Category:** (empty)
- Scheduled publish date:** 2025-11-28 19:59:27
- Valid to:** 2025-12-31
- Version:** 0.02
- Article type:** HTML
- Workflow:** Scheduled for publish
- Source task:** (empty)
- Attachment link:**
- Display attachments:**
- Short description:** Basic Knowledge Article
- Article body:** A Knowledge Article is a documented piece of information stored in the Knowledge Base (KB) of ServiceNow. It helps users find answers, solve issues, and follow procedures without needing support teams.

At the top right are 'Recall' and 'Delete' buttons. The browser's address bar shows the URL as dev321122.service-now.com/now/nav/ui/classic/...

The screenshot shows a ServiceNow Knowledge Article page. The URL in the address bar is dev321122.service-now.com/now/nav/ui... . The page title is "Knowledge - Basic ...". The breadcrumb navigation shows "Home / General / Basic Knowledge Article". On the right side of the header, there are buttons for "Flag Article" and "Create Incident". Below the header, there is a section titled "Basic Knowledge Article" with the ID KB0010001 (Scheduled for publish). It has a rating of 5 stars and 0 views. A descriptive text explains what a Knowledge Article is and its purpose. Below this, there is a "Helpful?" section with "Yes" and "No" buttons. A "Leave a comment" section with a text input field and a "Comment" button. At the bottom, there is a "Copy Permalink" link and a "Edit" button.

## Basic Knowledge Article

KB0010001 (Scheduled for publish)

0 views

A Knowledge Article is a documented piece of information stored in the Knowledge Base (KB) of ServiceNow. It helps users find answers, solve issues, and follow procedures without needing support teams.

Think of it as a self-help document for employees and customers.



Authored by System Administrator  
Last modified 1 minute ago

Helpful?

**Yes**

**No**

Leave a comment

**Comment**

[Copy Permalink](#)



All

Favorites

Knowledge - Basic ...

Subscribed

[Flag Article](#)

[Create Incident](#)

**Edit**

## Basic Knowledge Article

KB0010001 (draft)

0 views

A Knowledge Article is a documented piece of information stored in the Knowledge Base (KB) of ServiceNow. It helps users find answers, solve issues, and follow procedures without needing support teams.

Think of it as a self-help document for employees and customers.



Authored by System Administrator  
Last modified Just now

Helpful?

**Yes**

**No**

12. On incident form create a custom field called manager, phone number and Email. On select of caller the caller details need to be auto populated.

- Dot walk caller manager phone and email

The screenshot shows the ServiceNow Configuration Manager interface. A modal window titled "Configuring Incident form" is open, displaying two lists: "Available" and "Selected".

**Available:**

- Active
- Activity due
- Actual end
- Actual start
- Additional assignee list
- Approval
- Approval history
- Approval set
- Assigned to [+]
- Assignment group [+]
- Business duration
- Business impact
- Business resolve time
- Caller [+]
- Caused by Change [+]
- Change Request [+]

**Selected:**

- Number
- Caller
- Caller.Manager
- Caller.Business phone
- Caller.Email
- Category
- Subcategory
- Service
- Service offering
- Configuration item
- Universal Request
- Transfer reason
- | - split - |
- Channel
- State
- On hold reason

At the bottom of the modal are "Cancel" and "Save" buttons.

The screenshot shows the ServiceNow Incident Create screen. The "Number" field is populated with "INC0010009". The "Caller" field is populated with "Abel Tuter". The "Manager" field is populated with "Abraham Lincoln". Other fields shown include "Business phone" (1234567890), "Email" (abel.tuter@example.com), "Category" (Inquiry / Help), "Subcategory" (-- None --), "Service" (dropdown), "Service offering" (dropdown), "Configuration item" (dropdown), "Short description" (empty), and "Description" (empty). To the right, dropdowns show "Channel" (-- None --), "State" (New), "Impact" (3 - Low), "Urgency" (3 - Low), "Priority" (5 - Planning), "Assignment group" (dropdown), and "Assigned to" (dropdown). At the bottom, there is a "Related Search Results" section with a search bar and a "Knowledge & Catalog" dropdown.

13. Trigger a notification to opened by user through event when priority changes to P1.. Body content of your own.

The screenshot shows the ServiceNow web interface for creating a new notification. The URL is <https://dev321122.service-now.com/nav/nav/ui/cl...>. The title bar says "Notification - Priori...". The main form fields are:

- Name:** PrioritychangeTo1
- Type:** EMAIL
- Table:** Incident [incident]
- Active:**
- \* Category:** Uncategorized
- Description:** (empty)

Below the form, there is a note about a digest and an "Allow Digest" checkbox. The "When to send" tab is selected, showing the following configuration:

- Send when:** Event is fired
- Event name:** incident.priority.1
- Conditions:** Add Filter Condition, Add OR Clause, -- choose field --, -- oper --, -- value --
- Weight:** 0

The "Who will receive" tab is selected, showing:

- Notifications can be sent to specific **Users** and **Groups** or to **Users/Groups** in fields on the record.
- The Notification will not be sent to the User who caused the Notification to be triggered unless **Send to event creator** is checked.
- The Notification will also be sent to Delegates of users unless **Exclude Delegates** is checked.

Configuration options under "Who will receive":

- Users:**
- Groups:**
- Users/Groups in fields:**  Opened by
- Event parm 1:**  contains recipient
- Event parm 2:**  contains recipient
- Send to event creator:**
- Exclude delegates:**
- Subscribable:**

When to send Who will receive **What it will contain**

When using an **Email Template**, the **Subject** and **Message** content is picked from that template unless a different **Subject** or **Message** content is specified in this form.

Content type	HTML only	Importance	-- None --
Include attachments	<input type="checkbox"/>	From	<input type="text"/>
Omit watermark	<input checked="" type="checkbox"/>	Reply to	<input type="text"/>
Push Message Only	<input type="checkbox"/>	Push Messages	<input checked="" type="checkbox"/>
Email template	Unsubscribe and Preferences <input type="button" value="Search"/> <input type="button" value="Help"/>		
Subject	\$[number] Priority Changed to 1		
Message HTML	<div style="border: 1px solid #ccc; padding: 5px;"> <b>B</b> <b>I</b> <b>U</b> <input type="button" value="Left"/> <input type="button" value="Right"/>          Verdana 8pt <input type="button" value="Font"/> <input type="button" value="Size"/>  <input type="button" value="A"/> <input type="button" value="Pencil"/> <input type="button" value="Link"/> <input type="button" value="Image"/> <input type="button" value="Video"/> <input type="button" value="Code"/> <input type="button" value="List"/> <input type="button" value="Table"/> <input type="button" value="Text"/>  <input type="button" value="List"/> <input type="button" value="Table"/> <input type="button" value="Text"/>           Number: \${number}          Opened by: \${opened_by}          Trigger On Event Fired Priority Change to 1       </div> <span style="border: 1px solid #ccc; padding: 2px;">-</span> <span style="border: 1px solid #ccc; padding: 2px;">+</span> Select variables: <input type="button" value="Fields"/>		

All Favorites History : Email - INC001000...

**INC0010003 Priority Changed to 1**

Type	send-ready	UID	N/A
Target	Incident: INC0010003 <input type="button" value="Search"/> <input type="button" value="Help"/>	Created	2025-11-27 22:02:01
User	<input type="text"/>	Deleted	<input type="checkbox"/>
Notification type	SMTP	Weight	0
		Importance	<input type="text"/>

Originating Event and Notification

Event	Notification
incident.priority.1	PrioritychangeTo1

Subject: INC0010003 Priority Changed to 1  
 Recipients: admin@example.com

All Favorites History : Email - INC001000...

**INC0010003 Priority Changed to 1**

Preview Email

Number: INC0010003  
 Opened by: System Administrator  
 Trigger On Event Fired Priority Change to 1

[Unsubscribe | Notification Preferences](#)  
 Ref:MSG0000473\_eG7RkLGMrsAtjtdEUYi

## 14. Capture below client script on a update set and move to other pdi

- o ex: update set name: Client script on load example

The screenshot shows the 'Update Set' details page for 'Client Script on Load Example'. The application scope is set to Global. The update set was created on 2025-11-27 at 22:28:28 by admin.

Field	Value
Name	Client Script on Load Example
State	In progress
Parent	
Release date	
Install date	
Installed from	
Description	

- o Create an on change client script on incident When user change the caller the:
  - 1) Email of that changed caller will be displayed in the email field.
  - 2) CI field become mandatory.
  - 3) State field value will change to In-Progress
  - 4) Short description will populated with some text.

The screenshot shows the 'Client Script' details page for 'Onchange on Incident'. The script is active and global, targeting the Incident table with an onChange type on the Caller field.

Field	Value
Name	Onchange on Incident
Table	Incident [incident]
UI Type	All
Type	onChange
Field name	Caller

**Description:**

Create an onchnage client script on incident  
When user change the caller the:  
1) Email of that changed caller will be displayed in the email field.  
2) CI field become mandatory.  
3) State field value will change to In-Progress  
4) Short description will populated with some text.

**Script:**

```

1  function onChange(control, oldValue, newValue, isLoading,
2    isTemplate) {
3    if (isLoading || newValue === '') {
4      return;
5    }
6    g_form.getReference('caller_id', function(caller) {
7      var email = caller.email;
8      g_form.setValue('u_email',email);
9    });
10   g_form.setMandatory('cmdb_ci',true);
11   g_form.setValue('state',2);
12   g_form.setValue('short_description','On change Script Run
Check');

```

- Complete Update set and Move to Other PDI
  - Export this Update set to XML
  - Go to Retrieved Update Sets and Import this Update set

Importing records from an XML file will not run Business Rules

Step 1: Choose file to upload

\* XML file  sys\_remote\_u...4013103.xml

Step 2: Upload the file

**Upload**

Retrieved Update Set  
Client Script on Load Example

After committing this update set, ensure you map any unmapped custom tables to a subscription so that you can track your entitlements accurately in Subscription Management.

Name	Client Script on Load Example	Committed	
Application	Global	Inserted	
Update source		Deleted	
Parent			
State	Loaded		
Loaded	2025-11-27 23:48:05		
Description			
Application name	Global		

**Update** **Delete** **Preview Update Set**

Related Links  
[Export to XML](#)

- Preview Update Set and Commit Update Set

Retrieved Update Set  
Client Script on Load Example

Update Set Commit

Succeeded 100%

Update set committed - Succeeded in 1 Second

**Close**

**Update** **Delete** **Run Preview Again** **Commit Update Set**

- Check if the Client script is imported and Functional

The screenshot shows the 'Client Script - Onc...' page in the ServiceNow interface. The script is named 'Onchange on Incident' and is associated with the 'Incident [incident]' table. It is set to 'All' UI Type, 'onChange' Type, and 'Caller' Field name. The 'Application' is 'Global' and 'Active'. The 'Description' field contains a detailed note about the script's purpose: it creates an onchange client script on incident, changes the caller's email, makes the CI field mandatory, sets the state to In-Progress, and populates the short description. The 'Script' section displays the following code:

```

1 function onChange(control, oldValue, newValue, isLoading,
2   isTemplate) {
3   if (isLoading || newValue === '') {
4     return;
5   }
6   g_form.getReference('caller_id', function(caller) {
7     var email = caller.email;
8     g_form.setValue('u_email',email);
9   });
10  g_form.setMandatory('cmdb_ci',true);
11  g_form.setValue('state',2);
12  g_form.setValue('short_description','On change Script Run
13  Check');

```

The screenshot shows the 'Incident - Create I...' page in the ServiceNow interface. A new record is being created with the following details:

- Number:** INC0010015
- Email:** abel.tuter@example.com
- \* Caller:** Abel Tuter
- Manager:** Abraham Lincoln
- Business phone:** 1234567890
- Email:** abel.tuter@example.com
- Category:** Inquiry / Help
- Subcategory:** -- None --
- Service:** (empty)
- Service offering:** (empty)
- \* Configuration item:** (empty)
- \* Short description:** On change Script Run Check
- Channel:** -- None --
- State:** In Progress
- Impact:** 3 - Low
- Urgency:** 3 - Low
- Priority:** 5 - Planning
- Assignment group:** (empty)
- Assigned to:** (empty)

## 15. Create a On Submit client script on incident

- Create custom fields on form called Issue dated on and Issue resolved on
- We have to compare two dates on Submit Client script.
- 1) If "issue resolved date" field date is greater or equal to "issue dated on" "filed date then form will get submitted
- 2) If "issue resolved date" field date is smaller to" issue dated on "filed date then form will not get submitted

The screenshot shows the configuration of a Client Script named "Issue resolved date". The script is set to run on the "onSubmit" event for the "Incident [incident]" table. It is global and active. The description provides instructions for creating a client script to validate the "Issue resolved date" field against the "Issue dated on" field. The script itself compares the two dates and alerts if the resolved date is lower than the dated on.

```

1 function onSubmit() {
2     var created_date = g_form.getValue('u_issue_dated');
3     var resolved_date = g_form.getValue('u_issue_resolved');
4
5     if (resolved_date >= created_date){
6         return true;
7     }else{
8         alert("Issue Resolved on cannot be lower than Issue Dated");
9         return false;
10    }
11
12 }

```

The screenshot shows an "Incident New record" form. The "Issue Resolved on" field has a value of "2025-11-28 00:59:45" and the "Issue Dated on" field has a value of "2025-11-30 00:59:32". A modal dialog box displays the error message "Issue Resolved on cannot be lower than Issue Dated".

## 16.Create a On Submit client script on incident

- 1) will make email filed mandatory read only.
- 2) CI field become mandatory.
- 3) will add a status choice “pending with admin” for users having admin role
- 4) Short description will populated with some text.

The screenshot shows the configuration of a client script named "StatusChoicePendingAdmin".

**General Settings:**

- Name: StatusChoicePendingAdmin
- Application: Global
- Table: Incident [incident]
- UI Type: All
- Type: onLoad
- Active: checked
- Inherited: unchecked
- Global: checked

**Description:**

16. Create a On Load client script on incident

- 1) will make email filed mandatory read only.
- 2) CI field become mandatory.
- 3) will add a status choice “pending with admin” for users having admin role
- 4) Short description will populated with some text.

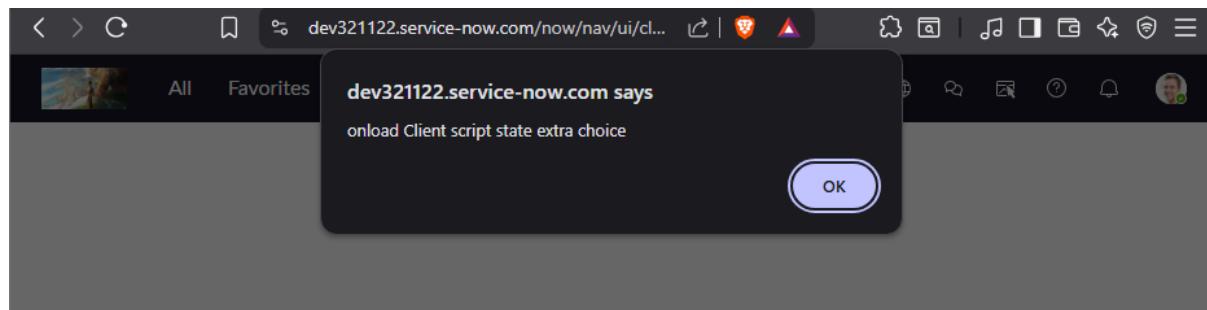
**Script:**

```

1 function onLoad() {
2     g_form.setMandatory('email',true);
3     g_form.setReadOnly('email',true);
4     g_form.setMandatory('cmdb_ci',true);
5
6     if (g_user.hasRoleExactly('admin')){
7         g_form.addOption('state','9','Pending with Admin');
8     }
9     g_form.setValue('short_description','On Load Check Admin
state choice');
10    alert('onload Client script state extra choice');
}

```

**Isolate script:** checked



The screenshot shows the 'Incident - Create' form in ServiceNow. The form includes fields for Number (INC0010027), Email, Caller, Manager, Business phone, Category (Inquiry / Help), Subcategory, Service, Service offering, Configuration item, Short description (containing 'On Load Check Admin state choice'), and Description. On the right side, there are dropdown menus for Channel (None), State (New, highlighted in blue), Impact (New, In Progress, On Hold, Resolved, Closed, Canceled), Urgency (Priority), Assignment group (Pending with Admin, highlighted in blue), Assigned to, Issue Dated, and Issue Resolved on. A 'Related Search Results' button is at the bottom.

## 17. Create a UI Policy

- When the caller is system administrator then email mandatory, phone number hidden and manager is read only

The screenshot shows the 'UI Policy - caller is ...' configuration screen. It lists the Table (Incident [incident]), Application (Global), and Active status (checked). Below these are fields for Short description (caller is system administrator) and Order (100).

The main area is divided into two tabs: 'When to Apply' (selected) and 'Script'. Under 'When to Apply', there are sections for Conditions (Add Filter Condition, Add OR Clause) and Actions (Apply the UI policy actions when the form is loaded and when the user changes values on the form). The 'Conditions' section shows a dropdown for Caller (is System Administrator) and an 'AN' button.

Below the 'When to Apply' tab, there are checkboxes for Global (checked), Reverse if false (checked), and Inherit (unchecked). Buttons for Update and Delete are at the bottom.

UI Policy Actions (3) UI Policy Related List Actions

for text Search Actions on selected rows... New

UI policy = caller is system administrator

<input type="checkbox"/>	Field name	Mandatory	Visible	Read only
	u_manager	Leave alone	Leave alone	True
	u_email	True	Leave alone	Leave alone
<input type="checkbox"/>	u_phone_number	Leave alone	False	Leave alone

1 to 3 of 3

### 18. Create a custom table—User details

- Name - Reference type (sys\_user)
- Email Id - String
- Address - String
- Mobile No - String
- User Description - String

All Favorites : Table - User Details

Table User Details Delete Update Delete All Records

A table is a collection of records in the database. Each record corresponds to a row in a table, and each field on a record corresponds to a column on that table. Applications use tables and records to manage data and processes. [More Info](#)

\* Label User Details Application Global

\* Name u\_user\_details Remote Table

Columns Controls Application Access

Table Columns for text Search 1 to 11 of 11 New

Dictionary Entries

<input type="checkbox"/>	Column label	Type	Reference	Max length	Default value	Display
X	Name	Reference	User	32	false	
X	User Description	String	(empty)	40	false	
X	Address	String	(empty)	40	false	
	Sys ID	Sys ID (GUID)	(empty)	32	false	
	Created	Date/Time	(empty)	40	false	
X	Mobile No.	String	(empty)	40	false	
	Updated by	String	(empty)	40	false	
	Updated	Date/Time	(empty)	40	false	
	Updates	Integer	(empty)	40	false	
	Created by	String	(empty)	40	false	
X	Email Id	String	(empty)	40	false	
+	Insert a new row...					

Delete Update Delete All Records

## 19. Scheduled Jobs

- Order the catalog item Evision access by using a schedule job every month on day 1.

The screenshot shows the 'Scheduled Script Execution' page in ServiceNow. The title bar says 'Scheduled Script E...'. The main area displays a table with one row. The first column contains a blue link labeled 'EvisionAccessScheduledJobs'. The second column has three checkboxes: 'Active' (checked), 'Application' (set to 'Global'), and 'Conditional' (unchecked). A note below states: 'For scheduled job types that require an entered time, you have the option to enter an associated time zone. If no time zone is selected, the job will run at the entered time in time zone of the user who entered the time. If 'Use System Time Zone' is selected, the entered time will run in the time zone of the instance running the job.' The 'Run' section shows 'Monthly' as the frequency, '1' as the day, and 'GMT' as the time zone. The 'Time' field is set to 00:00:00. The 'Advanced' section is collapsed. The 'Run this script' section contains a code editor with the following JavaScript:

```

1 // var vision_sysid = '734543f883213210e99ce4d0deaad33a'
2 (function() {
3     var itemID = "734543f883213210e99ce4d0deaad33a";
4     var cart = new Cart();
5     var cartID = cart.addItem(itemID);
6
7     cart.setVariable(cartID,'opened_on_behalf_of',
8         '62826bf03710200044e0bfc8bcbe5df1');
9     cart.setVariable(cartID,'email_requester_info','123@example.
com');
10    cart.setVariable(cartID,'phone_requester_info','1234567890');
11
12    var rc = cart.placeOrder();
13
14    gs.log("Evision Access request ordered automatically. Request
Number: " + rc.number, "AutoOrderEvision");

```

The screenshot shows the 'Log Entry - Created...' page in ServiceNow. The title bar says 'Log Entry - Created...'. The main area displays a table with one row. The first column contains a blue link labeled 'Created 2025-11-28 21:35:24'. The second column has four fields: 'Level' (Information), 'Source' (AutoOrderEvision), 'Created' (2025-11-28 21:35:24), and 'Message' (Evision Access request ordered automatically. Request Number: REQ0010002). Below the table is a 'Related Links' section with a link to 'Show Log Entries'.

- Delete all closed incident tickets every week on friday.

Scheduled Script Execution  
New record

Name: StateclosedDelete  
Active:   
Application: Global  
Conditional:

For scheduled job types that require an entered time, you have the option to enter an associated time zone. If no time zone is selected, the job will run at the entered time in time zone of the user who entered the time. If 'Use System Time Zone' is selected, the entered time will run in the time zone of the instance running the job.

Run: Weekly  
Run days of week:  Mon  Tue  Wed  Thu  Fri  Sat  Sun  
Time zone: GMT  
Time: Hours: 23 Minutes: 00  
Advanced:

Run this script:  Turn on ECMAScript 2021 (ES12) mode (?)

```

1 var gr = new GlideRecord('incident');
2
3 gr.addQuery('state',5);
4 gr.query();
5 gr.deleteMultiple();
6

```

**Submit**

- Send all on hold incidents tickets report to the Manager with Excel format.

Scheduled Email of Report  
New record

Name: OnholdIncidentReport  
Report: OnHold Incidents  
Users: Abel Tuter  
Groups:  
Email addresses:  
Application: Global  
Active:   
Run: about 3 minutes from now  
Starting: 2025-11-28 21:58:59  
Conditional:   
Omit if no records  
Subject:  
Introductory message:

Type: Excel (.xlsx)  
Zip output:

The screenshot shows the 'Email' screen in ServiceNow. A file named 'OnHold Incidents.xlsx' is attached. The form fields include:

- Type: send-ready
- Target: Scheduled Email of Report: Onhold (with a search icon)
- User: (empty)
- Notification type: SMTP
- UID: N/A
- Created: 2025-11-28 22:00:20
- Deleted: (checkbox)
- Weight: -1
- Importance: (empty)
- Subject: (empty)
- Recipients: abel.tuter@example.com
- Body: HTML content (shown as code block)
- Content type: multipart/mixed
- Headers: X-ServiceNow-Source:Notification  
List-Unsubscribe:<mailto:?>  
body=Sending+this+email+with+the+predefined+content+in+the+subject+and+body+will+unsubscribe+you+from+the+notification.  
%0D%0A%0D%0AUnsubscribe%3A%7B%22id%22%3A%22null%22%2C%22token%22%3A%22c797e62132%22%7D  
&subject=Unsubscribe+from+%22%22  
X-ServiceNow-SysEmail-Version:2

Buttons at the bottom: Update, Delete.

- Automatically Create the database cleanup incident ticket on every month day1.

The screenshot shows the 'Scheduled Script Execution' screen for a job named 'DBCleanupMonthly'. The configuration includes:

- Name: DBCleanupMonthly
- Active: checked
- Application: Global
- Conditional: (checkbox)

A note below states: "For scheduled job types that require an entered time, you have the option to enter an associated time zone. If no time zone is selected, the job will run at the entered time in time zone of the user who entered the time. If 'Use System Time Zone' is selected, the entered time will run in the time zone of the instance running the job."

Run settings:

- Run: Monthly
- Day: 1
- Time zone: GMT
- Time: Hours: 05, Minutes: 30, Seconds: 00
- Advanced: (checkbox)

Run this script:

- Turn on ECMAScript 2021 (ES12) mode: checked
- Script content (shown as code block):

```

1 (function() {
2     var inc = new GlideRecord('incident');
3     inc.initialize();
4     inc.short_description = "Monthly Database Cleanup Task";
5     inc.description = "Automatically generated ticket for
6     scheduled monthly database cleanup ";
7     inc.priority = 3;
8     inc.category = "database";
9     var incID = inc.insert();
gs.log("Monthly DB Cleanup Incident created: " + incID,
"AutoDBCleanup");

```

The screenshot shows a ServiceNow Log Entry page with the following details:

- Log Entry - Created...**
- Created:** 2025-11-28 22:39:07
- Level:** Information
- Source:** AutoDBCleanup
- Created:** 2025-11-28 22:39:07
- Message:** Monthly DB Cleanup Incident created: 60ad5ff883e13210e99ce4d0dead374

**Related Links:**

- Show Log Entries

- Automatically load the data into the User table with schedule import by using data source it should load every day 10AM.
- create a data soure with type file and format excel – attach the file

The screenshot shows a ServiceNow Data Source - New record page with the following fields:

- Name:** UserDataSchedule
- Import set table label:** Import Sets Users
- Import set table name:** u\_import\_sets\_users
- Type:** File
- Format:** Excel (.xlsx/.xls)
- Zipped:**
- Sheet number:** 1
- Header row:** 1
- Use Batch Import:**

**File retrieval method:** Attachment

**Submit**

- Load all records

Progress

Name	ImportProcessor
State	Complete
Completion code	Success
Message	Processed: 23, inserts 23, updates 0, errors 0, empty and ignored 0, ignored errors 0 (0:00:01.191)

#### Next steps...

[Import sets](#) Go to the import sets for this data load

[Loaded data](#) Go to the newly imported data inside the staging table: u\_import\_sets\_users

[Create transform map](#) Create a transform map for the newly staged data

[Import log](#) View the import log

- Create proper transform map with assist and set coalesce field

Table Transform Map... [Copy](#) [Update](#) [Delete](#)

custom user import transform

Run business rules  Order

Enforce mandatory fields  Run script

Copy empty fields

Create new record on empty coalesce fields

[Copy](#) [Update](#) [Delete](#)

Related Links

- [Auto Map Matching Fields](#)
- [Mapping Assist](#)
- [Validate Coalesce Fields](#)
- [Transform](#)
- [Index Coalesce Fields](#)
- [Run Point Scan](#)

Field Maps (5) [Transform Scripts](#)

<input type="checkbox"/>	Source field	Target field	Coalesce
<input type="checkbox"/>	u_user_name	user_name	true
<input type="checkbox"/>	u_last_name	last_name	false
<input type="checkbox"/>	u_email	email	false
<input type="checkbox"/>	u_phone	phone	false
<input type="checkbox"/>	u_first_name	first_name	false

Actions on selected rows... [New](#)

1 to 5 of 5

- Go to All > Scheduled Imports > Create New > Select the Data Source created

The screenshot shows the 'Scheduled Data Import' page in ServiceNow. A new scheduled job is being configured:

- Name:** CSA Project Scheduled jobs
- Application:** Global
- Data source:** UserDataSchedule
- Run as:** System Administrator
- Run:** Daily
- Time:** Hours 10 00 00
- Active:** checked
- Conditional:** unchecked
- Concurrent Import:** unchecked
- Execute pre-import script:** unchecked
- Execute post-import script:** unchecked

At the bottom, there are buttons for **Execute Now**, **Update**, and **Delete**. Below the form, there's a section titled **Related Links** with links to **View Dashboard** and **Run Point Scan**.

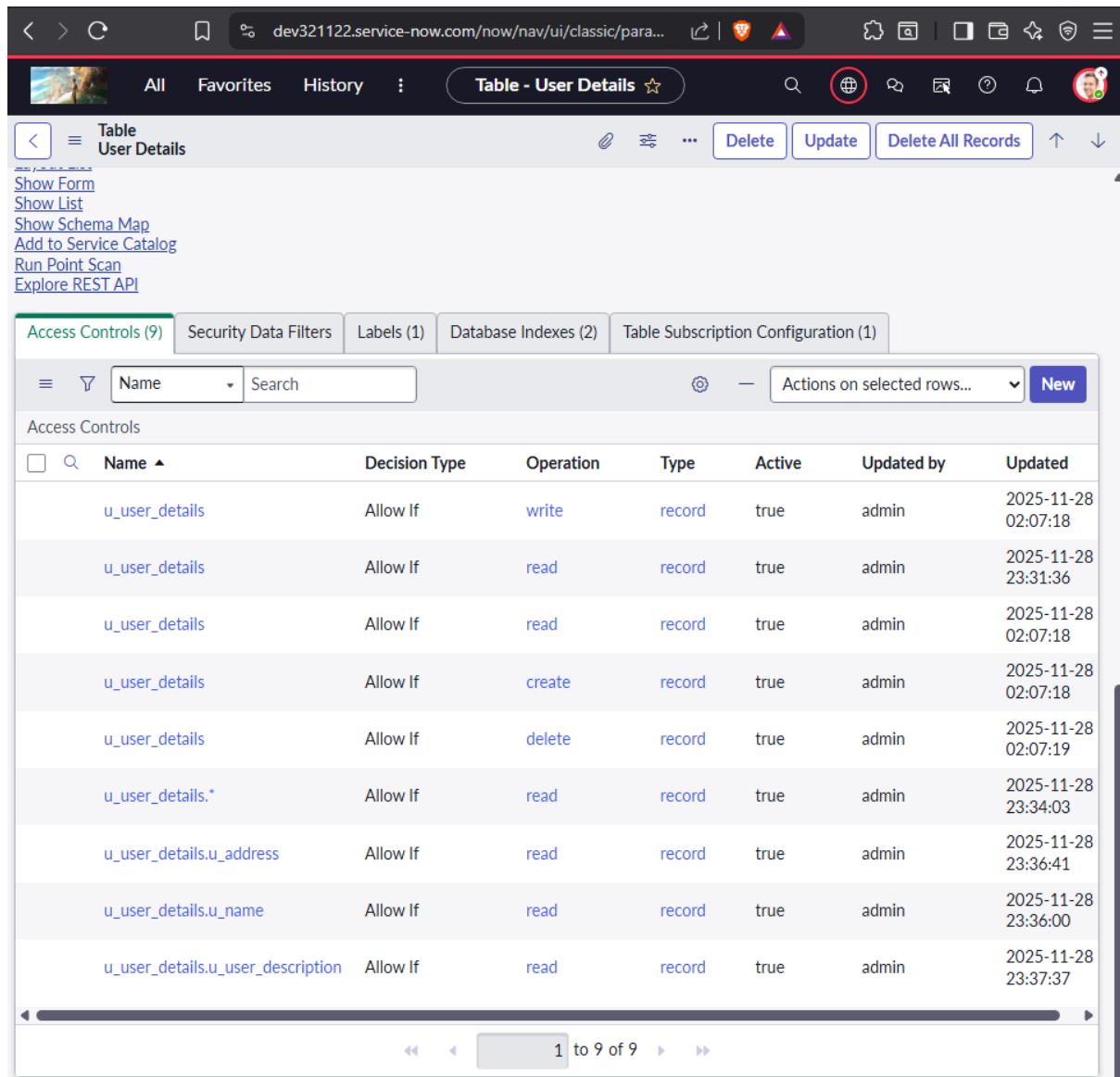
- Execute Now to Verify Process

The screenshot shows the 'Users' list page in ServiceNow, displaying a table of user records:

	User ID	Name	Email	Active	Created	Updated
<input checked="" type="checkbox"/>	Search	Search	Search	Search	Search	Search
<input checked="" type="checkbox"/>	kadam.rahl	Kadam Rahul	kadamRahul@example.com	true	2025-11-28 23:11:54	2025-11-28 23:11:54
<input checked="" type="checkbox"/>	pabbathi.sahithi	Pabbathi Sahithi	sahithi@example.com	true	2025-11-28 23:11:54	2025-11-28 23:11:54
<input checked="" type="checkbox"/>	santosh.raju	Santosh Raju	Santosh@example.com	true	2025-11-28 23:11:54	2025-11-28 23:11:54
	ishan.bannerjee	Ishan Bannerjee	ishan@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
	nisha.d	Nisha D	nisha@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
	jithendra.s	Jithendra S	jithendra@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
	kollatiyuvraj	Kollati Yuvaraj	yuvraj@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
	sarvagnya.p	Sarvagnya P	sravagnya@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
	kalva.praneeth	Kalva Praneeth	praneeth@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
	jayabaaskar.baskaran	Jayabaaskar Baskaran	jb@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
	veluri.rishikesh	Veluri Rishikesh	rishikesh@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
	jeeva.prakasam	Jeeva Prakasam	jeeva@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
	dheeraj.selva.a	Dheeraj Selva A	dheeraj@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
	raam.vignesh	Ram Vignesh	ramvignesh@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
	yeshwant.reddy	Yeshwanth Reddy	yesh@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
	bonthalakoti.revathi	Bonthalakoti Revathi	revathi@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
	naga.harsha	Naga Harsha	nagarharsha@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
	harsha.manoj	Harsha Manoj	harsha@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
<input type="checkbox"/>	hariraj.p	Hariraj P	hari@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31
	koturi.tarun	Koturi Tarun	tarun@example.com	true	2025-11-26 22:43:31	2025-11-26 22:43:31

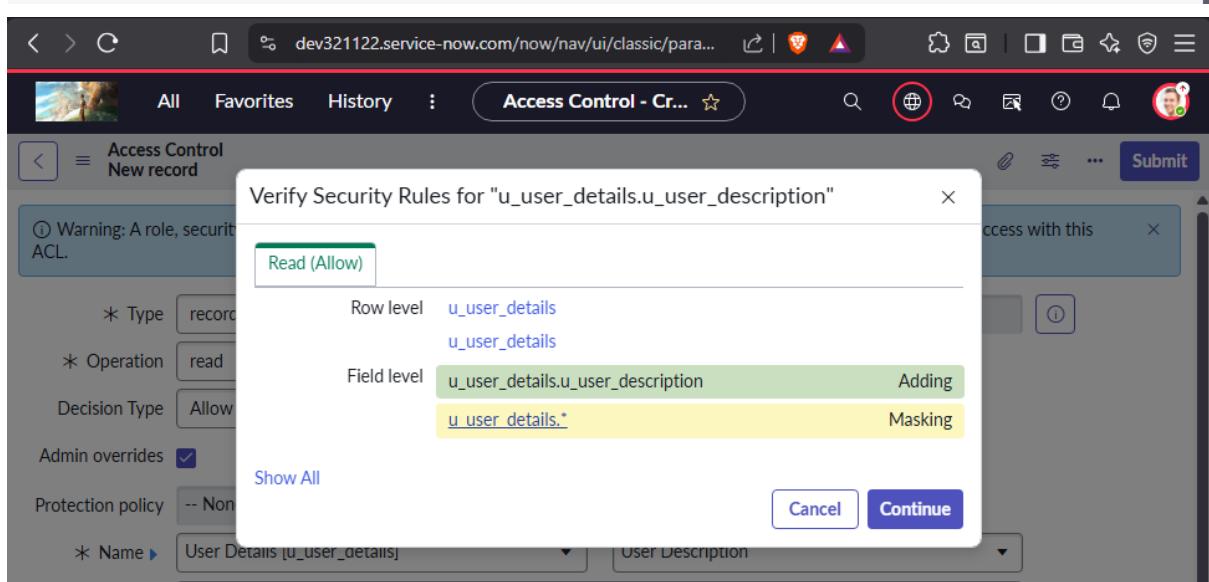
## 20. Role A users—name, address, user description

- Role B role users → access all fields.



The screenshot shows the ServiceNow interface for the 'User Details' table. The top navigation bar includes links for 'All', 'Favorites', 'History', and a search bar. Below the navigation is a toolbar with icons for 'Delete', 'Update', and 'Delete All Records'. A red circle highlights the globe icon in the toolbar. The main content area displays the 'Access Controls' tab of the table configuration. The table lists nine entries, each defining an access rule for the 'u\_user\_details' object. The columns include 'Name', 'Decision Type', 'Operation', 'Type', 'Active', 'Updated by', and 'Updated'. The rules cover various operations like 'read', 'write', 'create', and 'delete' at both row and field levels. The last two rows show specific rules for the 'u\_user\_description' field, including one for 'Masking'. At the bottom of the table view, there is a navigation bar with arrows and a page number indicator '1 to 9 of 9'.

Name	Decision Type	Operation	Type	Active	Updated by	Updated
u_user_details	Allow If	write	record	true	admin	2025-11-28 02:07:18
u_user_details	Allow If	read	record	true	admin	2025-11-28 23:31:36
u_user_details	Allow If	read	record	true	admin	2025-11-28 02:07:18
u_user_details	Allow If	create	record	true	admin	2025-11-28 02:07:18
u_user_details	Allow If	delete	record	true	admin	2025-11-28 02:07:19
u_user_details.*	Allow If	read	record	true	admin	2025-11-28 23:34:03
u_user_details.u_address	Allow If	read	record	true	admin	2025-11-28 23:36:41
u_user_details.u_name	Allow If	read	record	true	admin	2025-11-28 23:36:00
u_user_details.u_user_description	Allow If	read	record	true	admin	2025-11-28 23:37:37



The screenshot shows the 'Access Control - Create' screen. A warning message at the top states: '① Warning: A role, security, or ACL.' The main form has several fields: 'Type' set to 'record', 'Operation' set to 'read', 'Decision Type' set to 'Allow', 'Admin overrides' checked, 'Protection policy' set to '-- None', and 'Name' set to 'User Details (u\_user\_details)'. A modal window titled 'Verify Security Rules for "u\_user\_details.u\_user\_description"' is open, showing a 'Read (Allow)' rule. This rule specifies 'Row level' and 'Field level' access for the 'u\_user\_description' field. The 'Field level' section includes 'Adding' and 'Masking' options. At the bottom of the modal are 'Cancel' and 'Continue' buttons.

- Admin view of the table

The screenshot shows a ServiceNow interface for the 'User Details' table. The table has columns for Name, Address, Email Id, Mobile No., and User Description. The rows contain sample data for various users like Adela Cervantsz, Alene Rabeck, Alissa Mountjoy, Abel Tuter, and Abraham Lincoln.

	Name	Address	Email Id	Mobile No.	User Description
	Adela Cervantsz	432,werty,kjhgfd	werty@example..com	1234567890	ACI test project
	Alene Rabeck	7654,qwejhds,ffdf	qwer@example.com	123787654	ACI test project
	Alissa Mountjoy	111,asdd,ttt	Alisa@example.com	9876543217	ACI test project
	Abel Tuter	123,wded,st	abel@example.com	123456890	Some
	Abraham Lincoln	1234,qwerty,asd	abra@example.com	123567890	ACI test project

- Impersonate with each user and view the table u\_user\_details

- User A view – Role A

- Has Name,Address,User description Read Acl

This screenshot shows the same ServiceNow interface, but it is filtered to show only the Name, Address, and User Description columns. The rows for the other users are not visible.

	Name	Address	User Description
	Adela Cervantsz	432,werty,kjhgfd	ACI test project
	Alene Rabeck	7654,qwejhds,ffdf	ACI test project
	Alissa Mountjoy	111,asdd,ttt	ACI test project
	Abel Tuter	123,wded,st	Some
	Abraham Lincoln	1234,qwerty,asd	ACI test project

- User B view – Role B

- Has Table.\* - Read ACL (All Fields)

This screenshot shows the ServiceNow interface again, but it is filtered to show only the Email Id and Mobile No. columns. The rows for the other users are not visible.

	Email Id	Mobile No.
	werty@example..com	1234567890
	qwer@example.com	123787654
	Alisa@example.com	9876543217
	abel@example.com	123456890
	abra@example.com	123567890

(Both Role A and Role B are provided with Table\*None Read Access)