

Asset Management Portal

Problem Statement:

The Asset Management Portal will streamline the tracking, management, and allocation of both physical and digital assets across an organization. Employees will be able to request and receive assets through an intuitive portal, while administrators can manage the entire asset lifecycle, from procurement to disposal. The portal will also automate asset assignment, ensure accurate record-keeping, and generate real-time reports on asset utilization and condition. Alerts will be triggered for maintenance or replacement needs, ensuring optimal asset performance and reducing downtime. By centralizing asset management, the platform will improve operational efficiency, reduce asset loss, and support informed decision-making.

Create table

1. Open service now.
2. Click on All >> search for tables
3. Open System definition >> tables
4. Click on new
5. Fill in the details as
Name : asset inventory
6. Save the table

- Asset Management Portal
 - Table

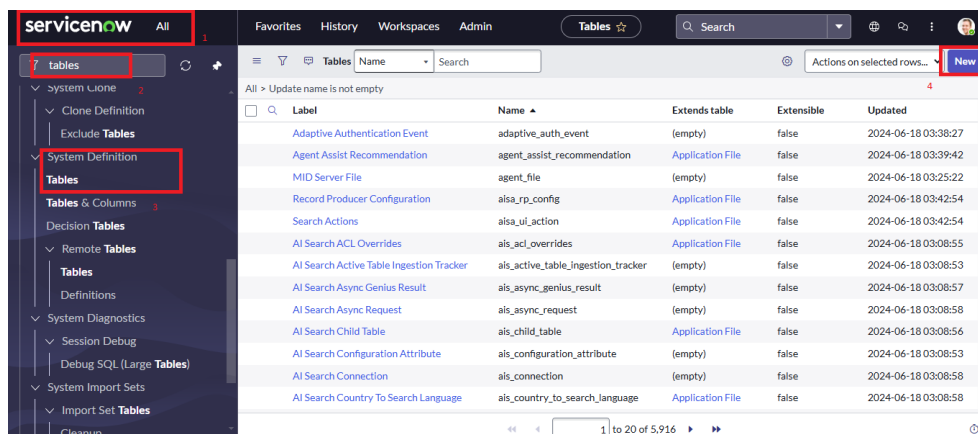
- Create table
- Create Fields
- Create UI Actions
- Scheduled Job
- Report
- Testing
- Conclusion

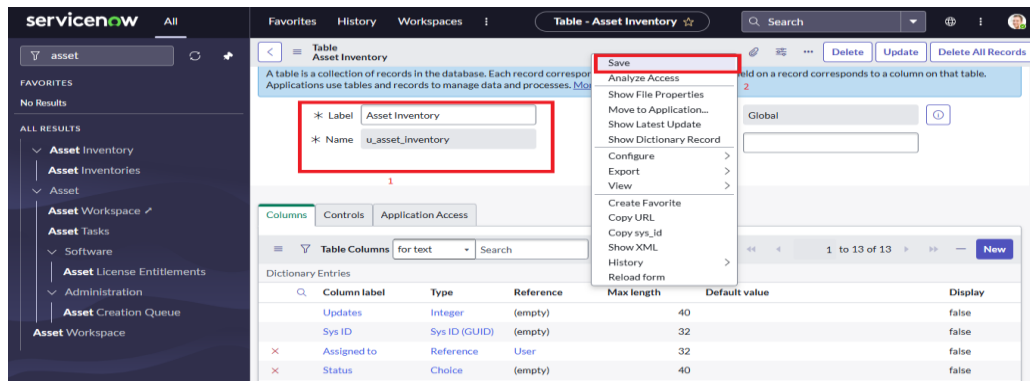
Create table

1. Open service now.
2. Click on All >> search for tables
3. Open System definition >> tables
4. Click on new
5. Fill in the details as

Name : asset inventory

6. Save the table





Create Fields

1)After saving the table scroll down

2)Create fields

- Assigned to : string
- Status : choice
- Purchase date : date
- Warranty Expire : date
- Asset name : string
- Type: choice
- Number : String

3) Click on save

UI Action 1

1. Navigate to System Definition >> UI action

2. Click on New

3. Fill in the details ;

- a. Name : Mark As Lost
- b. Table : Asset Inventory
- c. Action name : mark_as_lost

d. Condition : `current.u_status != 'Lost'`

e. Script :

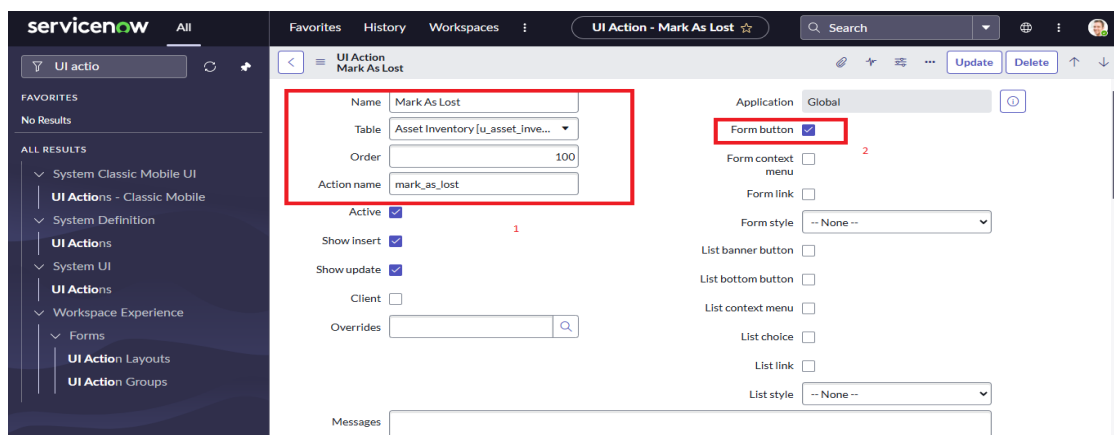
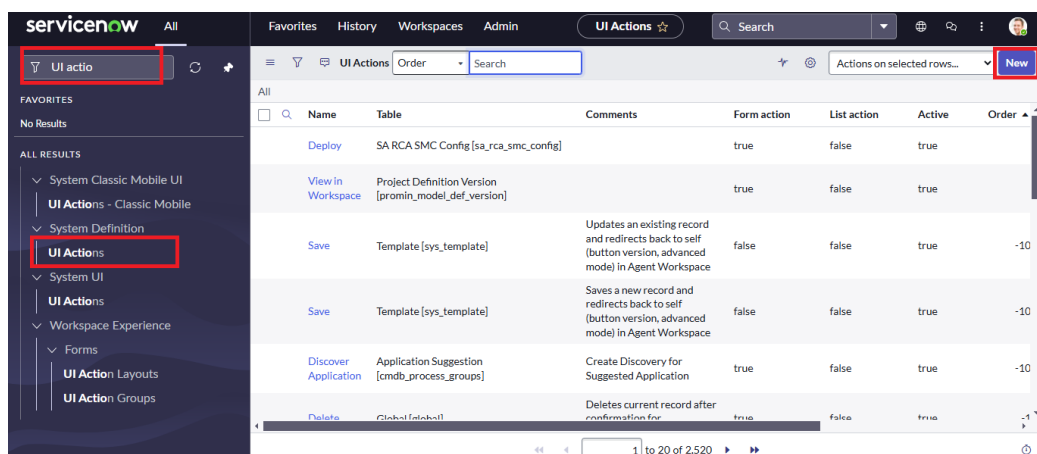
`current.u_status = 'Lost';`

`current.update();`

`action.setRedirectURL(current);`

4. Check the form button box

5. Click on save



UI Action 2

1. Navigate to System Definition >> UI action

2. Click on New

3. Fill in the details ;

1. Name : Mark As Repaired

2. Table : Asset Inventory

3. Action name : mark_as_repaired

4. Condition : `current.u_status == 'Damaged' || current.u_status == 'Lost'`

5. Script :

```
current.u_status = 'Available';
```

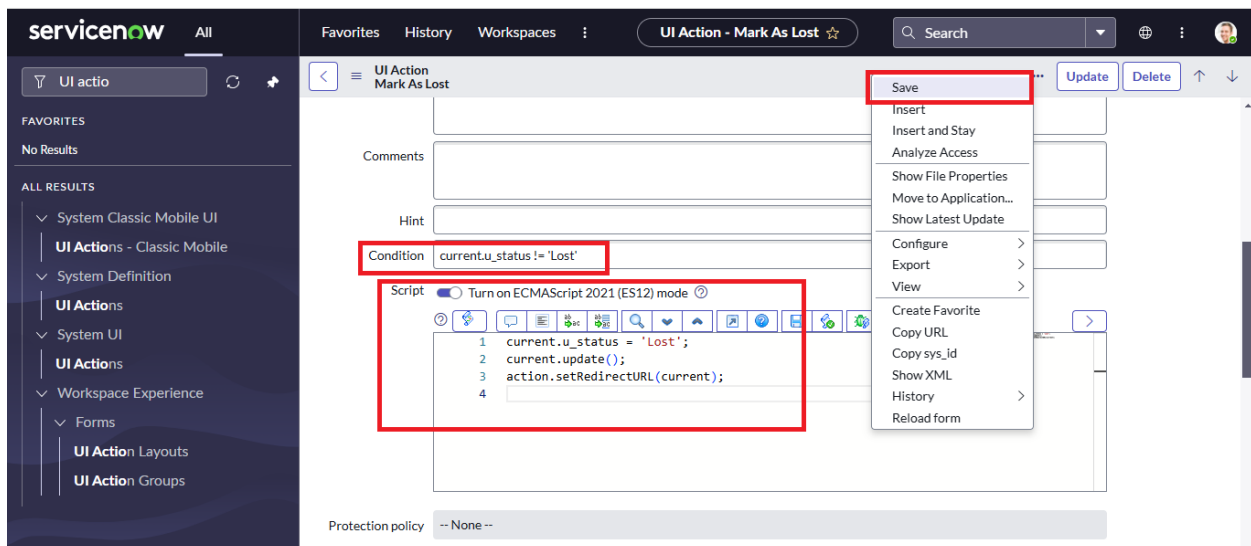
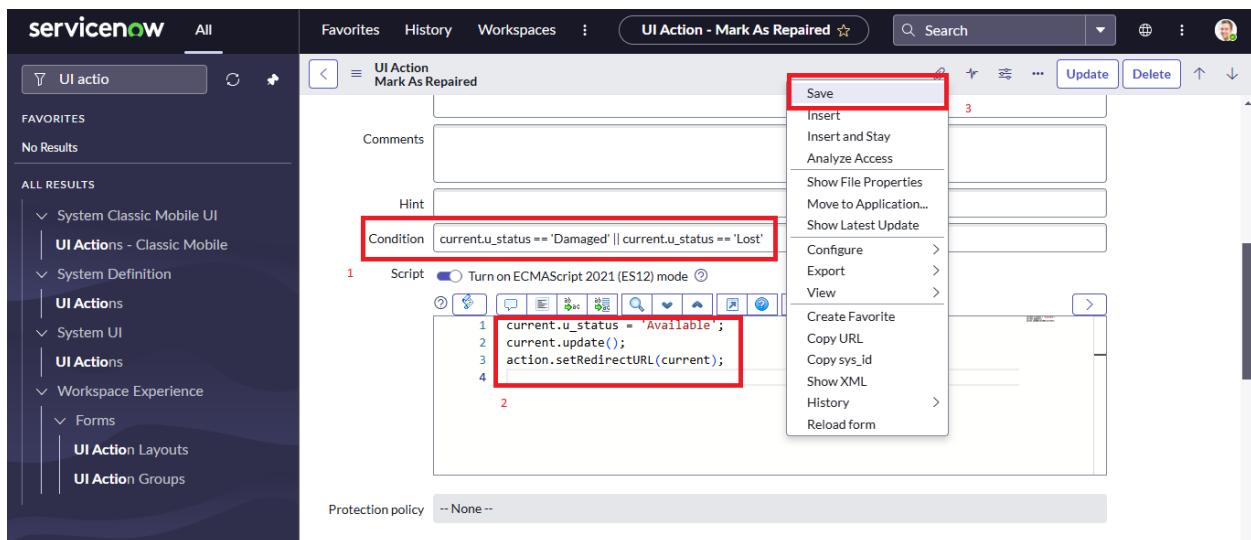
```
current.update();
```

```
action.setRedirectURL(current);
```

4. Check the form button box

5. Click on save

The screenshot shows the ServiceNow UI Action configuration interface. The left sidebar contains a navigation menu with 'UI action' selected. The main area is titled 'UI Action - Mark As Repaired'. A red box highlights the configuration fields: Name (Mark As Repaired), Table (Asset Inventory [u_asset_inve...]), Order (100), and Action name (mark_as_repaired). The 'Active' checkbox is checked, and a red '1' is next to it. The 'Form button' checkbox is also checked, and a red '2' is next to it. Other options like 'Form context menu', 'Form link', 'Form style', 'List banner button', 'List bottom button', 'List context menu', 'List choice', 'List link', and 'List style' are all unchecked or set to 'None'. The 'Messages' section is empty.



UI Action 3

1. Navigate to System Definition >> UI action
2. Click on New
3. Fill in the details ;
 - a. Name : Mark As Damedged

- b. Table : Asset Inventory
- c. Action name : mark_as_damaged
- d. Condition : current.u_status != 'Damaged'
- e. Script :

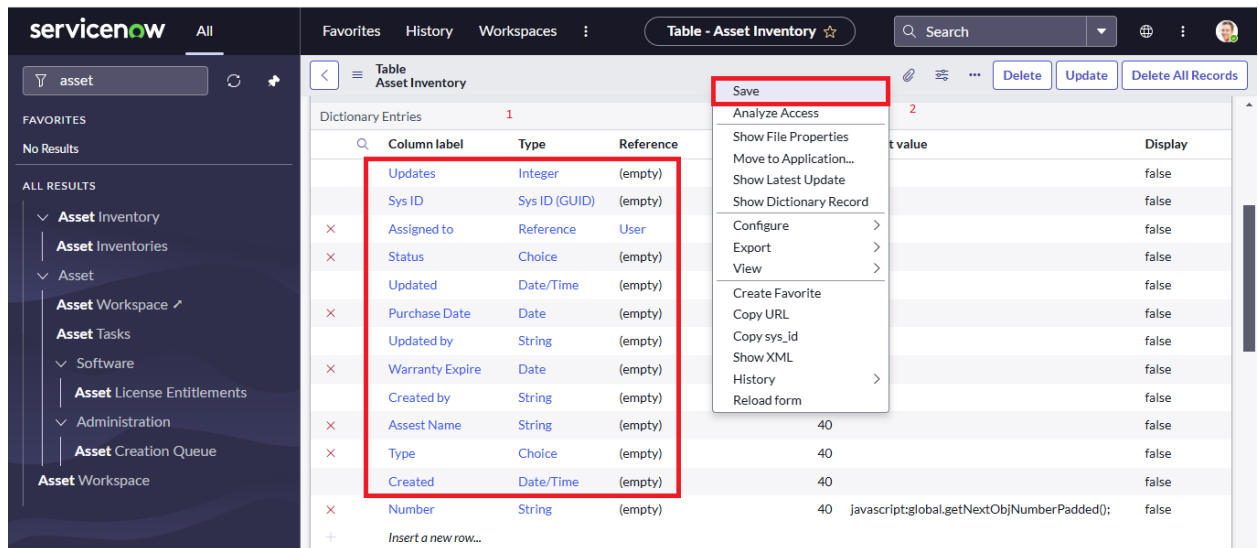
```
current.u_status = 'Damaged';  
current.update();  
action.setRedirectURL(current);
```
- 4. Check the form button box
- 5. Click on save

The screenshot shows the ServiceNow UI Action configuration page for 'Mark As Damaged'. The left sidebar contains a navigation menu with 'UI action' selected. The main area is divided into two columns. The left column contains fields for 'Name' (Mark As Damaged), 'Table' (Asset Inventory [u_asset_inve...]), 'Order' (100), 'Action name' (mark_as_damaged), 'Active' (checked), 'Show insert' (checked), 'Show update' (checked), 'Client' (unchecked), and 'Overrides'. The right column contains fields for 'Application' (Global), 'Form button' (checked), 'Form context menu' (unchecked), 'Form link' (unchecked), 'Form style' (-- None --), 'List banner button' (unchecked), 'List bottom button' (unchecked), 'List context menu' (unchecked), 'List choice' (unchecked), 'List link' (unchecked), and 'List style' (-- None --). A red box highlights the 'Name', 'Table', 'Order', and 'Action name' fields. Another red box highlights the 'Form button' field. A third red box highlights the 'Active' field.

The screenshot shows the ServiceNow UI Action configuration page for 'Mark As Damaged' with the 'Script' tab selected. The left sidebar is the same as the previous screenshot. The main area shows the 'Condition' field with the value 'current.u_status != 'Damaged'', the 'Script' field with the code:

```
1 current.u_status = 'Damaged';  
2 current.update();  
3 action.setRedirectURL(current);  
4
```

, and the 'Protection policy' field with the value '-- None --'. A red box highlights the 'Condition' field. Another red box highlights the 'Script' field. A third red box highlights the 'Save' button in the top right corner. A fourth red box highlights the 'Form button' field in the right column.



Create Scheduled Job

1. Navigate to System Definition >> Scheduled Job
2. Click on New
3. Name : Warranty Expiry Alert ,
4. Run : Daily
5. Time : 12:00
6. Write the script
7. And click on save

Create Report

1. Navigate To Reports
2. Click on Create New
3. Report Name : Available vs assigned assets , Source Type : Table ,
Table : Asset Inventory
4. Type : Pie Chart
5. Group By : Status , Aggregation : Count
6. Click on save

7. And then click on Run

servicenow All Favorites History Workspaces Admin ServiceNow Search

< Create a report

Data > Type > Configure > Style * Report Title :

* Report name

* Source type Table

* Table No table selected

Create your report with Analytics Q&A

Ask for information. You can give simple filtering conditions. You get the answer with an appropriate visualization.

What do you want to see? Ask

How can I improve my results?

Next

servicenow All Favorites History Workspaces Available vs. Assigned Assets Search

< Edit report Save Run

Data > Type > Configure > Style * Report Title : Available vs. Assigned Assets

* Report name Available vs. Assigned Assets

* Source type Table

* Table Asset Inventory [u_asset_inventory]

Description There is no description for this table. To add a description, please contact your admin.

Next

Type a question about your data

What do you want to see? Ask

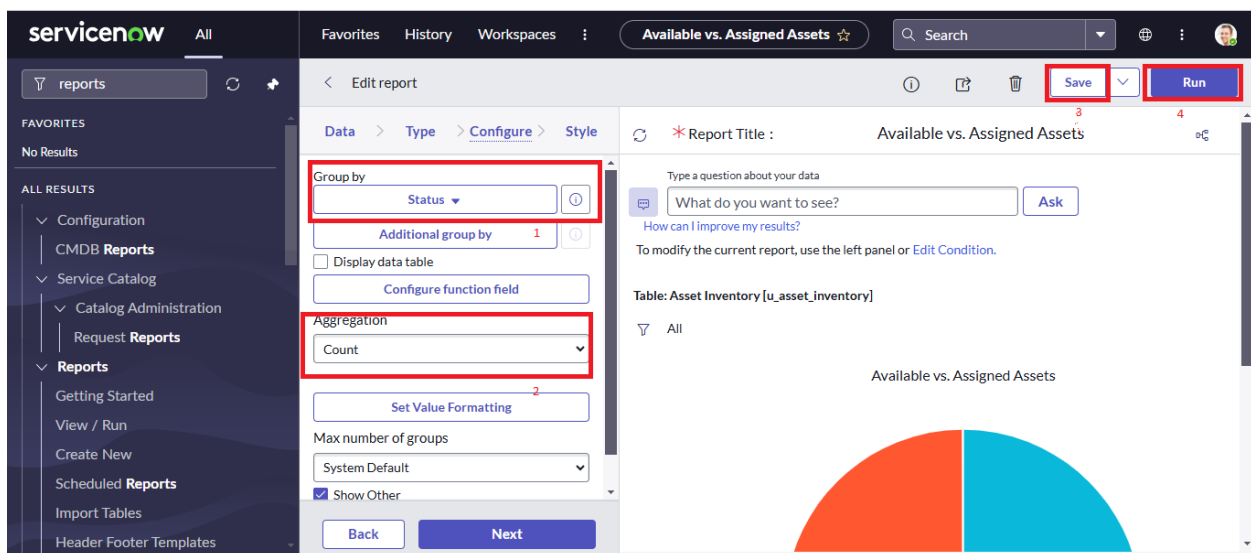
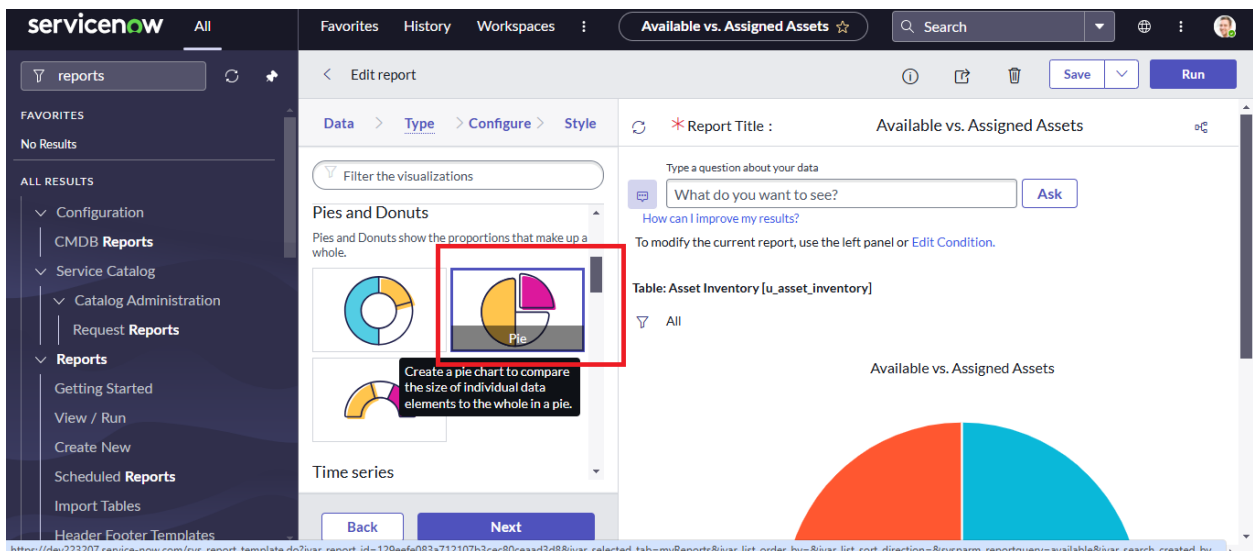
How can I improve my results?

To modify the current report, use the left panel or Edit Condition.

Table: Asset Inventory [u_asset_inventory]

All

Available vs. Assigned Assets



Testing UI action

1. Go to Asset Inventory table
2. Click on New
3. Fill in the details
 - a) Asset name : Laptop
 - b) Type : laptop

- c) Assigned to : Abel Tutor
- d) Status : Available
- e) select some purchase and expiry date
- 4. Click on submit
- 5. Open the record again
- 6. Click on mark as lost button and save
- 7. Check the status is changed to lost.

servicenow

All

asset inv

↺

↻

FAVORITES

No Results

ALL RESULTS

Asset Inventory

Asset Inventories

1

Favorites

History

Workspaces

:

Asset Inventories

☆

Search

▼

🌐

🔍

⋮

≡

🔍

💬

Asset Inventories

Number

▼

Search

⚙️

Actions on selected rows...

▼

New

All

2

<input type="checkbox"/>	🔍	Number	Asset Name	Assigned to	Purchase Date	Status	Type	Warranty Expire	Updates
		AST001011	Printer	Abel Tutor	2025-01-01	Available	printer	2025-02-02	0
		AST001006	Mobile	Abel Tutor	2025-01-15	Available	mobile	2025-02-03	0
		AST001005	Dell Laptop XPS	Abel Tutor	2025-01-01	Available	laptop	2025-02-02	2
		AST001009	Monitor	Abel Tutor	2025-01-02	Available	monitor	2025-02-02	0

⏪

⏩

1 to 4 of 4

⏪

⏩

🔄

servicenow

All

asset inv

FAVORITES
No Results

ALL RESULTS
Asset Inventory
Asset Inventories

Favorites History Admin

Asset Inventory - Create AST001013

Search

Submit Mark As Damaged Mark As Lost

Asset Inventory
New record

1

NumberAST001013

Asset NameLaptop

Typelaptop

Assigned toAbel Tuter

StatusAvailable

Purchase Date2025-01-01

Warranty Expire2025-02-02

2

SubmitMark As DamagedMark As Lost

servicenow

All

asset inv

FAVORITES
No Results

ALL RESULTS
Asset Inventory
Asset Inventories

Favorites History Workspaces

Asset Inventory - AST001014

Search

Update Mark As Damaged Mark As Lost Delete

Asset Inventory
AST001014

NumberAST001014

Asset NameLaptop

Typelaptop

Assigned toAbel Tuter

StatusAvailable

Purchase Date2025-01-01

Warranty Expire2025-02-13

1

UpdateMark As DamagedMark As LostDelete

2

servicenow

All

asset inv

FAVORITES
No Results

ALL RESULTS
Asset Inventory
Asset Inventories

Favorites History Workspaces

Asset Inventory - AST001014

Search

Update Mark As Damaged Mark As Repaired Delete

Asset Inventory
AST001014

NumberAST001014

Asset NameLaptop

Typelaptop

Assigned toAbel Tuter

StatusLost

Purchase Date2025-01-01

Warranty Expire2025-02-13

1

UpdateMark As DamagedMark As RepairedDelete

Testing Schduled Job

1. Navigate to background scripts
2. Write the Scheduled job script in the background scripts
3. Click on Run Script
4. Check the result

The screenshot displays the ServiceNow user interface. On the left sidebar, under 'ALL RESULTS', the 'Scripts - Background' option is highlighted with a red box. The main content area shows a 'Run script (JavaScript executed on server)' window. A red box highlights the following JavaScript code:

```
1 var grAsset = new GlideRecord('u_asset_inventory'); // Replace with your table name
2 var today = new GlideDateTime();
3 var futureDate = new GlideDateTime();
4 futureDate.addDays(30); // Get date 30 days from now
5
6
7 grAsset.addQuery('u_warranty_expire', '<=', futureDate); // Warranty expiring within the next 30 days
8 grAsset.addQuery('u_warranty_expire', '>=', today); // Warranty expiring after today
9 grAsset.query();
10
11
12 while (grAsset.next()) {
13     var email = new GlideEmailOutbound();
14     email.setSubject("Warranty Expiry Alert: " + grAsset.getValue('u_assest_name')); // Use getValue
15     email.setBody("The warranty for " + grAsset.getValue('u_assest_name') + " (Type: " + grAsset.
16         grAsset.getValue('u_asset_type') +
17         ") is expiring soon on " + grAsset.getValue('u_warranty_expiry') + ". Please take
18         action."); // Get values dynamically
19 }
```

servicenow All Favorites History Workspaces Admin ServiceNow Search

backgr

FAVORITES
No Results

ALL RESULTS

- Employee Profile
- Background Banner
- Process Mining
- System
- Background Jobs
- System Definition
- Scripts - Background
- System Logs
- Transactions (Background)

```
16      ") is expiring soon on " + grAsset.getValue('u_warranty_expiry') + ". Please take
17      action."); // Get values dynamically
18
19
20
21
22  email.setTo('it-support@company.com'); // Change to your IT support email
23  email.send();
24
25
26  gs.info("Email sent for assest: " + grAsset.getValue('u_assest_name')); // Log for confirmation
27  }
28
29
30
31
32
```

Run Script in scope global Record for rollback? ☒ Execute in sandbox? ☐ Execute as scriptlet? ☐ Cancel after 4 hours ☒

+ Instance Scripts

servicenow All Favorites History Workspaces Admin ServiceNow Search

backgr

FAVORITES
No Results

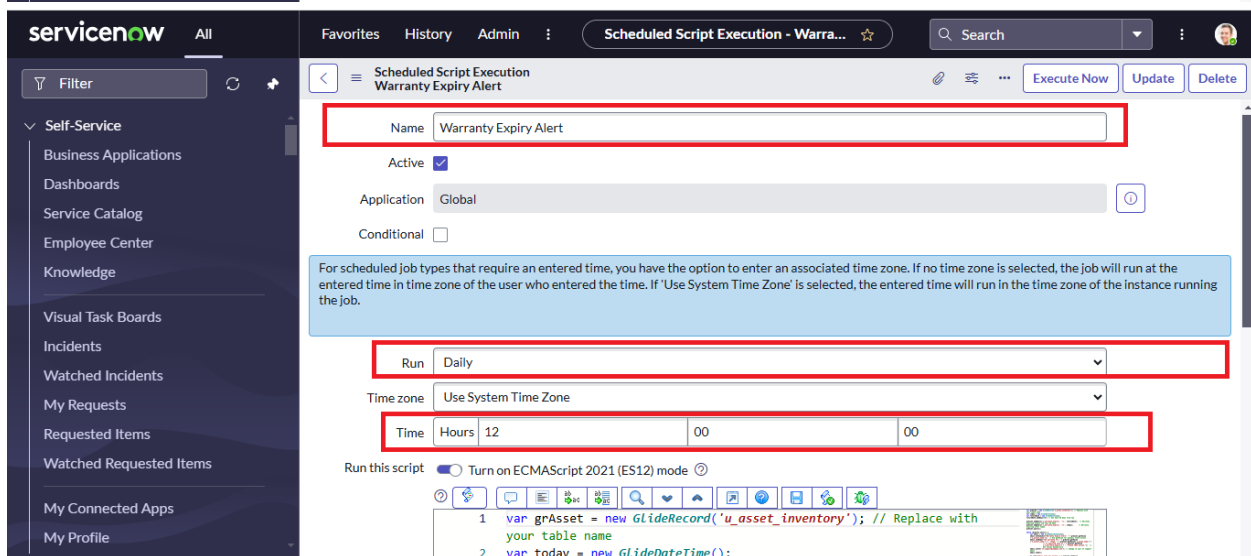
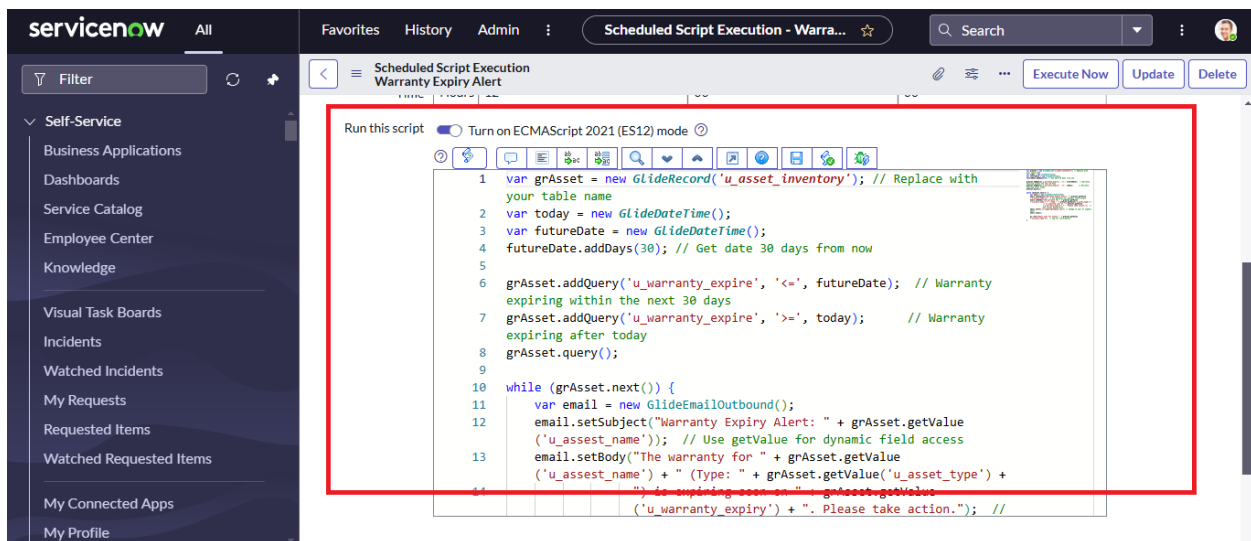
ALL RESULTS

- Employee Profile
- Background Banner
- Process Mining
- System
- Background Jobs
- System Definition
- Scripts - Background
- System Logs
- Transactions (Background)

[0:00:00.252] Script completed in scope global: script

Script execution history and recovery [available here](#)

*** Script: Email sent for assest: Mobile	
*** Script: Email sent for assest: Laptop	1



SCRIPT :

var grAsset = new GlideRecord('u_asset_inventory'); // Replace with your table name

var today = new GlideDateTime();

var futureDate = new GlideDateTime();

futureDate.addDays(30); // Get date 30 days from now

```

        grAsset.addQuery('u_warranty_expire', '<=', futureDate); //
Warranty expiring within the next 30 days

        grAsset.addQuery('u_warranty_expire', '>=', today);    // Warranty
expiring after today

        grAsset.query();

        while (grAsset.next()) {

            var email = new GlideEmailOutbound();

            email.setSubject("Warranty Expiry Alert: " +
grAsset.getValue('u_assest_name')); // Use getValue for dynamic field
access

            email.setBody("The warranty for " +
grAsset.getValue('u_assest_name') + " (Type: " +
grAsset.getValue('u_asset_type') +

                ") is expiring soon on " +
grAsset.getValue('u_warranty_expiry') + ". Please take action."); // Get
values dynamically

            email.setTo('it-support@company.com'); // Change to your IT
support email

            email.send();

            gs.info("Email sent for assest: " +
grAsset.getValue('u_assest_name')); // Log for confirmation

        }

```

Conclusion

The Asset Management Portal provides a comprehensive solution for tracking, managing, and optimizing physical and digital assets throughout their lifecycle. By leveraging automation and real-time data updates, the platform ensures efficient asset allocation, minimizes discrepancies, and enhances operational visibility. Automated workflows for asset tracking, maintenance alerts, and reporting enable organizations to make data-driven decisions, reduce asset downtime, and optimize resource utilization. This project demonstrates the power of ServiceNow's capabilities in integrating asset tracking, automation, and reporting tools to create a streamlined asset management system. By improving asset accountability and operational efficiency, the platform helps organizations maximize asset value, reduce costs, and enhance overall productivity.