

FAMILY EXPENSES

Team Id: NM2025TMID20243

Team Members: 4

Team Leader: MANIKANDAN.J

Member 1: MOHAMMED ALFATHA.F

Member 2: RAKESH.T

Member 3: NIRANJAN.L

Problem Statement: The project aims to develop a comprehensive expense calculation system using ServiceNow. This system will enable users to track and manage family expenses efficiently. It will include features such as expense categorization, budget setting, real-time tracking, and reporting capabilities. Utilizing ServiceNow's robust platform, the project will ensure seamless integration, user-friendly interface, and scalability to accommodate varying family sizes and financial complexities. The end goal is to empower users with the tools they need to make informed financial decisions and promote financial well-being within the family unit.

Objective: The "family expensive project" phrase seems to be a misunderstanding, but if referring to ServiceNow's Strategic Portfolio Management (SPM), also known as Project Portfolio Management (PPM), its objective is to strategically manage and optimize a portfolio of projects by aligning them with business goals, providing real-time visibility into performance, and improving agility and resource allocation to deliver maximum customer value.

Skills: Service now Family expenses IOT Open Hardware platforms, Data Structures

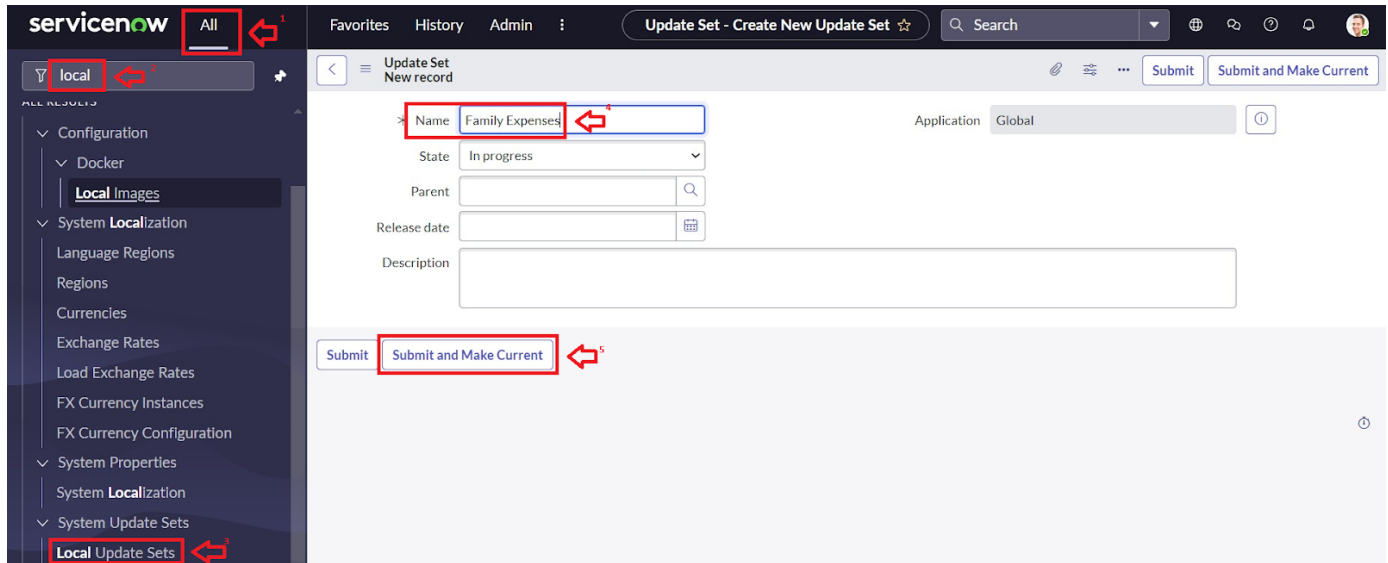
TASK INITIATION

Milestone 1: update set

Activity 1: Local update set

1. Open service now

2. Click on All >> search for Local update
3. Click on new
4. Name: Family Expenses
5. Then click on submit and make current.



The screenshot shows the ServiceNow 'Update Set - Create New Update Set' form. The sidebar on the left has 'Local Update Sets' highlighted. The main form has the following fields:

- Name:** Family Expenses
- State:** In progress
- Parent:** (empty)
- Release date:** (empty)
- Description:** (empty)
- Application:** Global

At the bottom, there are two buttons: 'Submit' and 'Submit and Make Current'.

Milestone 2: CREATION OF TABLE

Activity 1: TABLE

1. Open service now.
2. Click on All >> search for Table
3. Click on new.
4. Label: Family Expenses, Name: Auto-Populated,
New menu name: Family Expenditure.
5. Go to the Header and right click there>> click on Save.

* Label

* Name

Application ⓘ

Remote Table ☒

Create module ☒

Create mobile module ☒

Add module to menu

New menu name

Columns Controls Application Access









Table Columns Column label Search ⓘ

Dictionary Entries

	Column label ▼	Type	Reference	Max length	Default value	Display
✕ ✎	Number	String				false
✕ ✎	Date	Date				false
✕ ✎	Amount	Integer				false

Activity 2: Creation of Columns (Fields)

1. Near Columns Double click near insert a new row.
2. Give the details as:
 Column label: Number
 Type: String
3. Double click on insert a new row again
4. Give the details as:
 Column label: Date
 Type: Date
5. Double click on insert a new row again
6. Give the details as:
 Column label: Amount
 Type: Integer
7. Double click on insert a new row again
8. Give the details as:
 Column label: Expense Details
 Type: String
 Max length: 800

Columns						
Table Columns for text						
Dictionary Entries						
	Column label	Type	Reference	Max length	Default value	Display
 	Number	String				false
 	Date	Date				false
 	Amount	Integer				false
 	Expense Details	String		800		false
Insert a new row...						

ACTIVITY3: Making Number Field an Auto-Number

1. Double click on the Number Field/Column.
2. Go down and double click on Advanced view
3. In Default Value:
 - Use dynamic default: check the box
 - Dynamic default value: Get Next Padded Number
4. Click on Update.

Choice List Specification
 Calculated Value
 Default Value

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

Use dynamic default: ☒

Dynamic default value: Get Next Padded Number

Delete Column
 Update

- 5 Go to All>> In the filter search for Number Maintenance >> select Number Maintenance
6. Click on New.
7. Enter the below Details:
 - Table: Family Expenses
 - Prefix: MFE
- 8 Click on Submit.

<
≡
Number MFE
🔗
⚙️
⋮
Update
Delete

* Table Family Expenses 🔍 ⓘ

Prefix MFE

* Number 1,000

Application Global ⓘ

Number of digits 7

Update
Delete

ACTIVITY 4: CONFIGURE THE FORM

1. Go to All >> In the filter search for Family Expenses >> Open Family Expenses
2. Click on New
3. Go to the Header and right click there>> click on Configure >> Select Form Design
4. Customize or Drag Drop the form as per your requirement.

Family Expenses [u_family_expenses]
2 Column
+
×

Number
⚙️
×

Date
⚙️
×

Amount
⚙️
×

Expense Details
⚙️
×

5. Make Number Read-Only Field by clicking on the gear icon and checking Read-Only
6. Make Date, Amount Mandatory Field by clicking on the gear icon and checking Mandatory
7. Click on Save.

Milestone 3: CREATION OF TABLE(DAILY EXPENSES)

Activity 1: Creation of Daily Exepenses Table.

1. Go to All > In the filter search for Tables > click on New.
2. Enter the Details:
 Label: Family Expenses
 Name: Auto-Populated
 New menu name: Family Expenditure
3. Go to the Header and right click there>> click on Save.

* Label
 * Name

Application
 Remote Table ☒
 Create module ☒
 Create mobile module ☒
 Add module to menu
 New menu name

Columns Controls Application Access

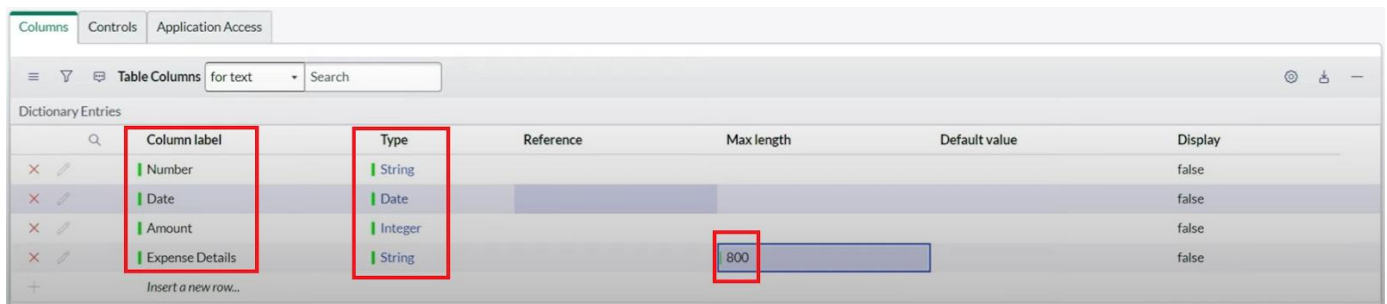
Table Columns

	Column label	Type	Reference	Max length	Default value	Display
<input type="checkbox"/> <input type="checkbox"/>	Number	String				false
<input type="checkbox"/> <input type="checkbox"/>	Date	Date				false
<input type="checkbox"/> <input type="checkbox"/>	Amount	Integer				false

Activity 2: Creation of columns(Field)

1. Near Columns Double click near insert a new row.
2. Give the details as:
 Column label: Number
 Type: String
3. Double click on insert a new row again
4. Give the details as:
 Column label: Date
 Type: Date

5. Double click on insert a new row again
6. Give the details as:
 Column label: Expense
 Type: Integer
7. Double click on insert a new row again
8. Give the details as:
 Column label: Family Member Name
 Type: Reference
 Max length: 800
9. Double click on insert a new row again
10. Give the details as:
 Column label: Comments
 Type: String
 Max length: 800
11. Go to the Header and right click there>> click on Save.




	Column label	Type	Reference	Max length	Default value	Display
	Number	String				false
	Date	Date				false
	Amount	Integer				false
	Expense Details	String		800		false


Insert a new row...




Activity 3: Making Number Field an Auto-Number


1. Double click on the Number Field/Column.
2. Go down and double click on Advanced view
3. In Default Value:
 Use dynamic default: check the box
 Dynamic default value: Get Next Padded Number
4. Click on Update.

Choice List Specification Calculated Value **Default Value** 

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

Use dynamic default ☒ 

Dynamic default value Get Next Padded Number   

Delete Column **Update** 


5.Go to All >> In the filter search for Number Maintenance >> select Number Maintenance

6.Click on New.

7.Enter the below Details:

Table: Family Expenses

Prefix: MFE

< ≡ Number
New record     **Submit**

* Table **Daily Expenses**   

Prefix **DFE** 

* Number

Application Global 

Number of digits

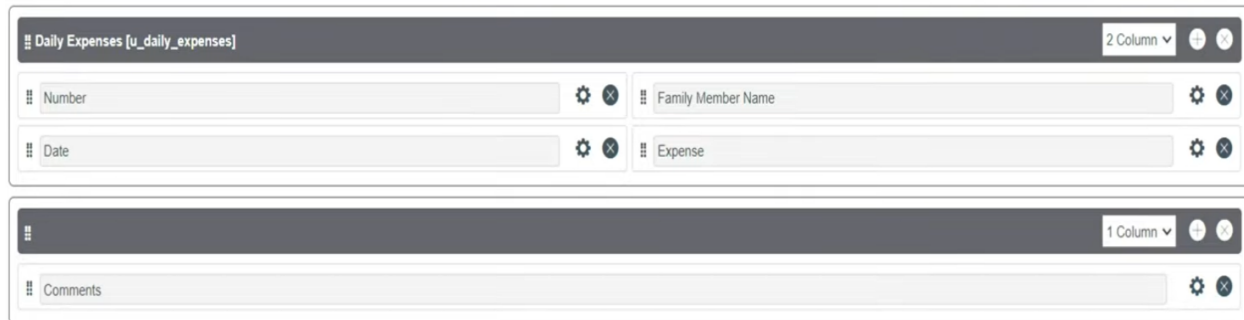
Submit 

8.Click on Submit.

ACTIVITY4: Configr the form.

1. Go to All >> In the filter search for Daily Expenses >> Open Daily Expenses
2. Click on New

3. Go to the Header and right click there>> click on Configure >> Select Form Design
4. Customize or Drag Drop the form as per your requirement.

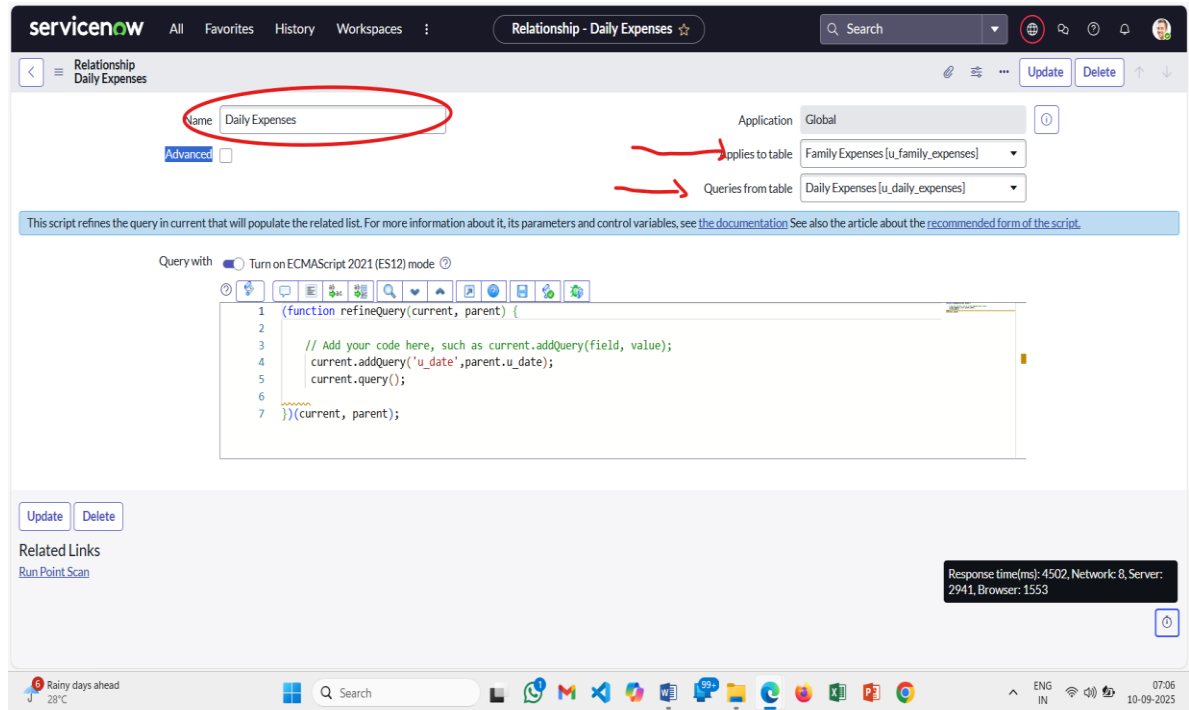


5. Make Number Read-Only Field by clicking on the gear icon and checking Read-Only
6. Make Date, Family Member Name Mandatory Field by clicking on the gear icon and checking Mandatory
7. Click on Save.

Milestone 4: CREATION OF RELATIONSHIP

Activity 1: Creation of Relationship between Family Expenses and Daily Expenses tables

1. Go to All >> In the filter search for Relationships >> Open Relationships
2. Click on New.
3. Enter the details:
 - Name: Daily Expenses
 - Applies to table: Select Family Expenses
 - Daily Expenses: Select Daily Expenses
4. Click Save.
6. Fill the following details to create a new ACL
7. Scroll down under requires role
8. Double click on insert a new row
9. Give task table and team member role
10. Click on submit
11. Similarly create 4 acl for the following fields



servicenow All Favorites History Workspaces Relationship - Daily Expenses

Name: Daily Expenses

Advanced ☐

Applies to table: Family Expenses [u_family_expenses]

Queries from table: Daily Expenses [u_daily_expenses]

This script refines the query in current that will populate the related list. For more information about it, its parameters and control variables, see [the documentation](#). See also the article about the [recommended form of the script](#).

Query with ☐ Turn on ECMAScript 2021 (ES12) mode

```

1 (function refineQuery(current, parent) {
2
3   // Add your code here, such as current.addQuery(field, value);
4   current.addQuery('u_date', parent.u_date);
5   current.query();
6
7 })(current, parent);
  
```

Update Delete

Related Links

[Run Point Scan](#)

Response time(ms): 4502, Network: 8, Server: 2941, Browser: 1553

Rainy days ahead 28°C

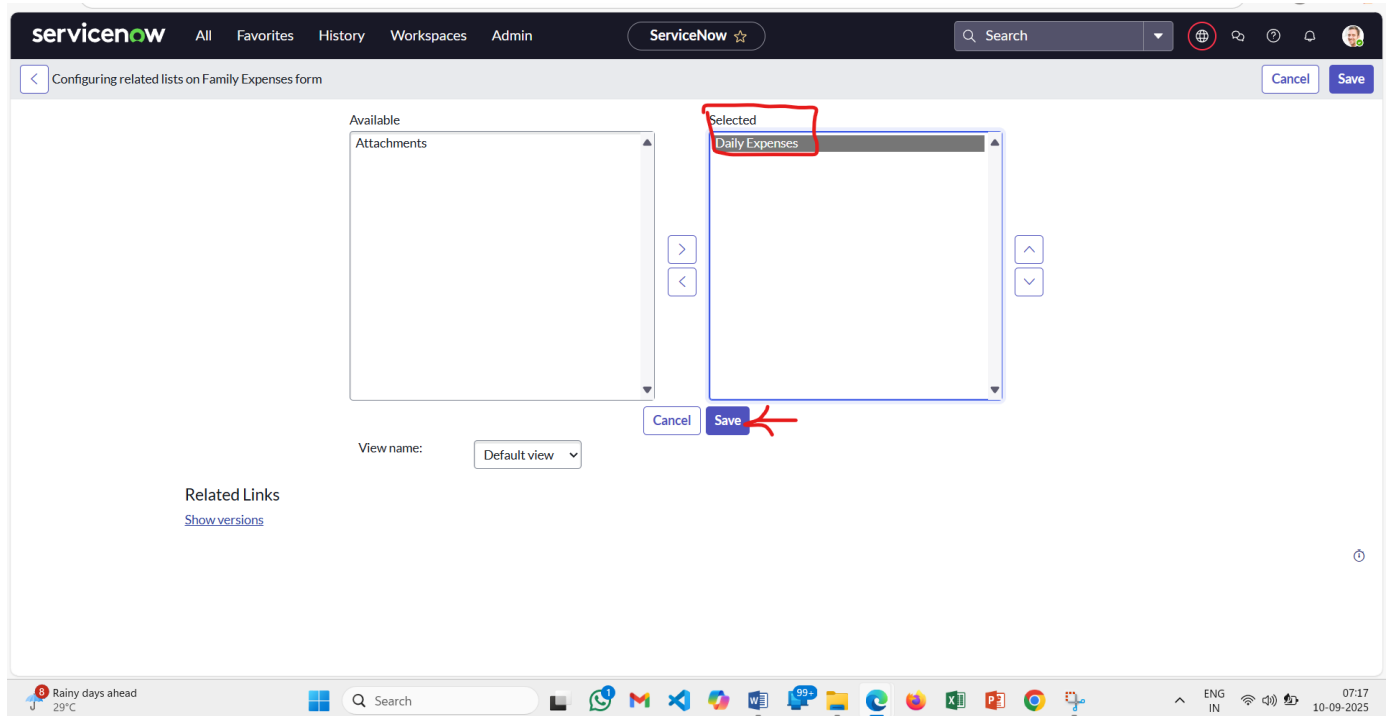
Search

ENG IN 07:06 10-09-2025

Milestone 5: Configuring Related List on Family Expenses

Activity 1: Create a Flow to Assign operations ticket to group

1. Go to All >> In the filter search for Family Expenses >> Open Family Expenses
2. Click on New
3. Go to the Header and right click there >> click on Configure >> Select Related Lists
4. Add Daily Expenses to the Selected Area.
5. Click on Save



servicenow All Favorites History Workspaces Admin ServiceNow Search

< Configuring related lists on Family Expenses form Cancel Save

Available
Attachments

Selected
Daily Expenses

> < ^ v

View name: Default view

Cancel Save

Related Links
[Show versions](#)

Rainy days ahead 29°C

Search

ENG IN 07:17 10-09-2025

Milestone 6: Creation of Business Rules

Activity1: Creation of Business Rules

1. Go to All >> In the filter search for Business Rules.
2. Under System Definition Select Business Rules then click on New.
3. Enter the Details:
 Name: Family Expenses BR
 Table: Select Daily Expenses
 Check Advanced

Business Rule
New record

Business rule is a server-side script that runs when a record is displayed, inserted, deleted, or when a table is queried. Use business rules to automatically change values in form fields when the specified conditions are met





Name 


Table 

Application 

Active ☒

Advanced ☒ 


4. In when to run Check Insert and Update

When to run  **Actions** **Advanced**

Specify whether the business rule should run on **Insert** or **Update**. Use **Filter Conditions** to specify under which conditions

When

Order

 **Insert** ☒


Update ☒

Delete ☐

Query ☐

Filter Conditions

-- choose field --

Role conditions 

5. In Advance (we write the code): Write the below code >>

```
(function execute Rule (current, previous /*null when async*/) {

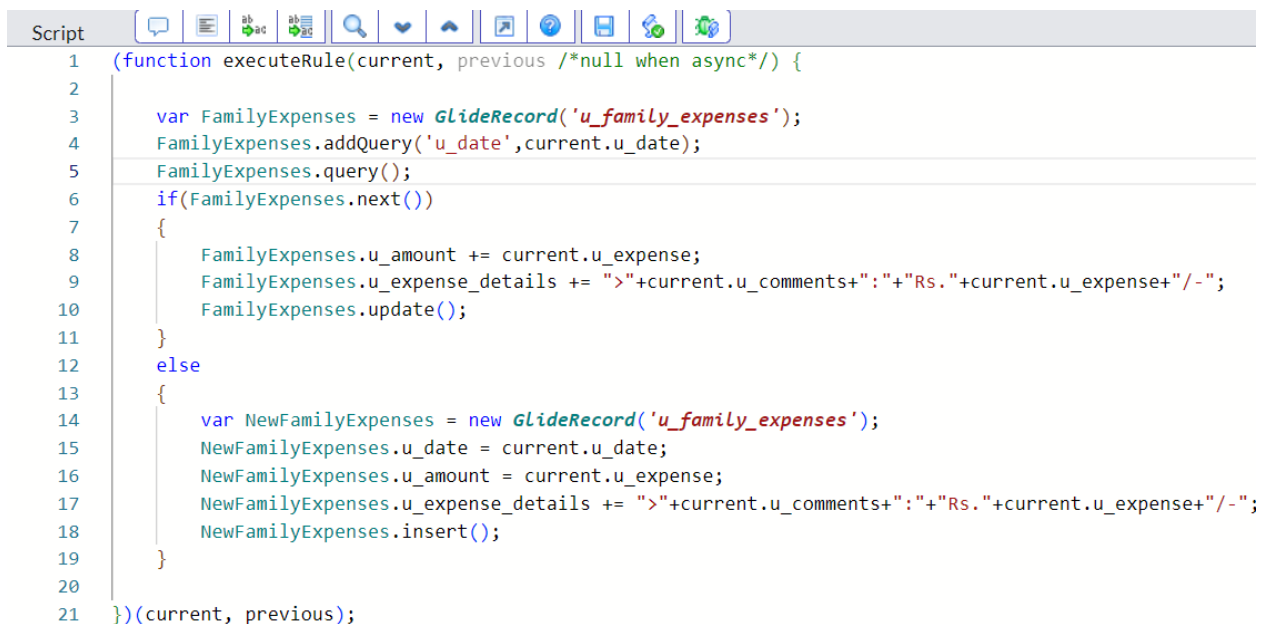
var Family Expenses = new Glide Record (' u_family_expenses ');
Family Expense. Add Query('u_date',current.u _date);
Family Expenses. query();
If (Family Expenses. next())
{
Family Expenses .u _amount += current .u _expense;
}
```

```

Family Expenses .u_expense_details += ">" + current.u_comments + ":" + "Rs." + current.u_expense + "/" -
";
Family Expenses. update();
}
else
{
var New Family Expenses = new GlideRecord('u_family_expenses');
New Family Expenses. u_date = current. u_date;
New Family Expenses. u_amount = current. u_expense;
New Family Expenses .u_expense_details +=
">" + current.u_comments + ":" + "Rs." + current.u_expense + "/" -
";
New Family Expenses. insert();
}

})(current, previous);

```



```

Script
1  (function executeRule(current, previous /*null when async*/) {
2
3      var FamilyExpenses = new GlideRecord('u_family_expenses');
4      FamilyExpenses.addQuery('u_date', current.u_date);
5      FamilyExpenses.query();
6      if(FamilyExpenses.next())
7      {
8          FamilyExpenses.u_amount += current.u_expense;
9          FamilyExpenses.u_expense_details += ">" + current.u_comments + ":" + "Rs." + current.u_expense + "/" -
10         ";
11         FamilyExpenses.update();
12     }
13     else
14     {
15         var NewFamilyExpenses = new GlideRecord('u_family_expenses');
16         NewFamilyExpenses.u_date = current.u_date;
17         NewFamilyExpenses.u_amount = current.u_expense;
18         NewFamilyExpenses.u_expense_details += ">" + current.u_comments + ":" + "Rs." + current.u_expense + "/" -
19         ";
20         NewFamilyExpenses.insert();
21     }
22 })(current, previous);

```

6. Go to the Header and right click there>> click on Save.

Milestone 7: Configure the Relationship

Activity1: Configure the Relationship

1. Go to All >> In the filter search for Relationships >> Open Relationships.
2. In that, open Daily Expenses Relationship.

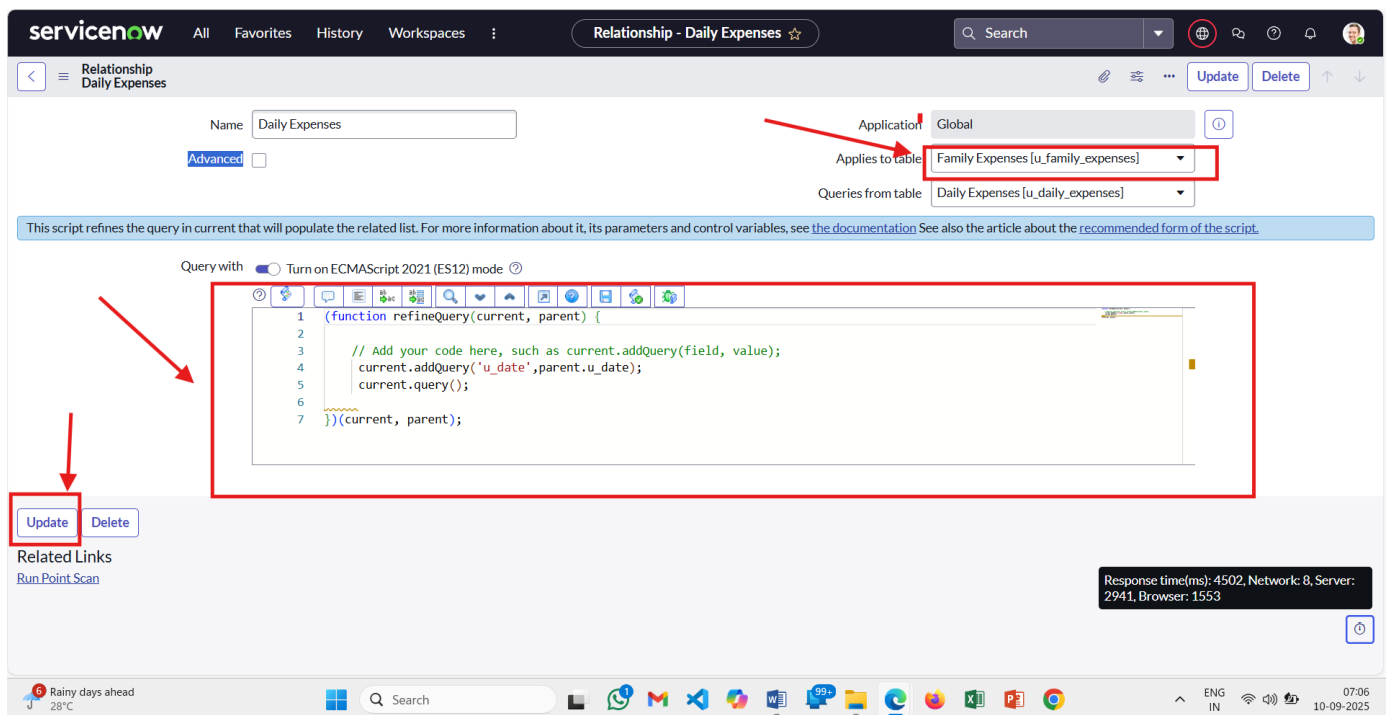
3. For Applies to table: Select Family Expenses.
4. In Query with: write the below Query.

```
(function refine Query (current, parent) {
```

```
// Add your code here, such as current. add Query (field, value);
current. add Query ('u_ date', parent. u_ date);
current. Query ();
```

```
})();
```

5. Click on Update.



servicenow All Favorites History Workspaces Relationship - Daily Expenses Search

Name: Daily Expenses Advanced ☐

Application: Global

Applies to table: Family Expenses [u_family_expenses]

Queries from table: Daily Expenses [u_daily_expenses]

This script refines the query in current that will populate the related list. For more information about it, its parameters and control variables, see [the documentation](#). See also the article about the [recommended form of the script](#).

Query with ☐ Turn on ECMAScript 2021 (ES12) mode

```

1 (function refineQuery(current, parent) {
2
3     // Add your code here, such as current.addQuery(field, value);
4     current.addQuery('u_date',parent.u_date);
5     current.query();
6
7 })(current, parent);

```

Update Delete

Related Links
[Run Point Scan](#)

Response time(ms): 4502, Network: 8, Server: 2941, Browser: 1553

Rainy days ahead 28°C

Search

ENG IN 07:06 10-09-2025

Conclusion: A successful ServiceNow "family" (project or platform) implementation, even a costly one, is concluded by demonstrating tangible business value through strategic alignment, improved operational efficiency, and positive user satisfaction. Achieving these outcomes requires senior expertise, comprehensive process mapping, a strong focus on configuration over customization, and ongoing transparent communication to manage risks, ensuring the project delivers its intended benefits and a positive return on investment.