



**SHREE CHANDRAPRABHU  
JAIN COLLEGE - MINJUR**  
(Affiliated to the University of Madras)  
(Accredited by NAAC)



# **CALCULATING FAMILY EXPENSES USING SERVICE NOW**

**Team ID : NM02025TMID19482**

**Team Size : 4**

**Team Leader : VARSHA.A**

**Team member : SUMALATHA.V**

**Team member : VISITHRA DEVI.A**

**Team member : VISHALI.K**

# **CALCULATING FAMILY EXPENSES**

## **INTRODUCTION**

Managing family expenses is an essential part of financial planning, ensuring that income is properly allocated towards daily needs, savings, and future goals. Traditionally, families depend on manual methods such as notebooks, spreadsheets, or mobile apps to track their spending. However, these methods often lack automation, centralized tracking, and analytical insights.

ServiceNow, a powerful cloud-based platform widely used in enterprises for IT service management and workflow automation, can also be adapted to manage personal and family-related financial tasks. By leveraging ServiceNow's database capabilities, forms, workflows, and reporting tools, families can record, categorize, and calculate their monthly expenses in a structured and automated manner. This approach not only simplifies expense management but also provides transparency, accuracy, and real time insights into financial health.

## **PROBLEM STATEMENT**

Managing family expenses is often challenging due to lack of organization, manual tracking, and limited visibility into spending patterns. Many families rely on paper notes, spreadsheets, or memory to record daily expenses, which leads to errors, missing data, and poor financial planning.

There is a need for a digital, automated, and secure platform that can record, track, and analyze family expenses efficiently. ServiceNow, with its powerful workflow automation, reporting, and cloud capabilities, can provide a structured solution for managing family finances.

## **OBJECTIVE**

- To develop a centralized system for recording and managing family expenses using ServiceNow.
- To categorize expenses (e.g., Food, Transport, Education, Utilities, Entertainment) for better financial visibility.
- To automate calculations of total expenses on a daily, monthly, and yearly basis.
- To set and monitor budgets for each category and notify users when limits are exceeded.
- To generate reports and dashboards that provide insights into spending patterns and trends.
- To ensure data security and role-based access so that only authorized family members can add or view expenses.

## SKILLS

**ServiceNow Platform** – Creating custom applications, tables, and forms. Configuring workflows, business rules, and notifications..

**UI Policies & UI Actions** –Designing user-friendly forms and dashboards for expense tracking.

**Scripting** – Basic JavaScript for client scripts and server- side scripts in ServiceNow.

**Workflow Design** – Understanding data structures, relationships, and queries in ServiceNow tables.

**Testing & Validation** – Building charts, dashboards, and scheduled reports in ServiceNow.

**Problem-Solving** – Identifying errors and resolving issues during development.

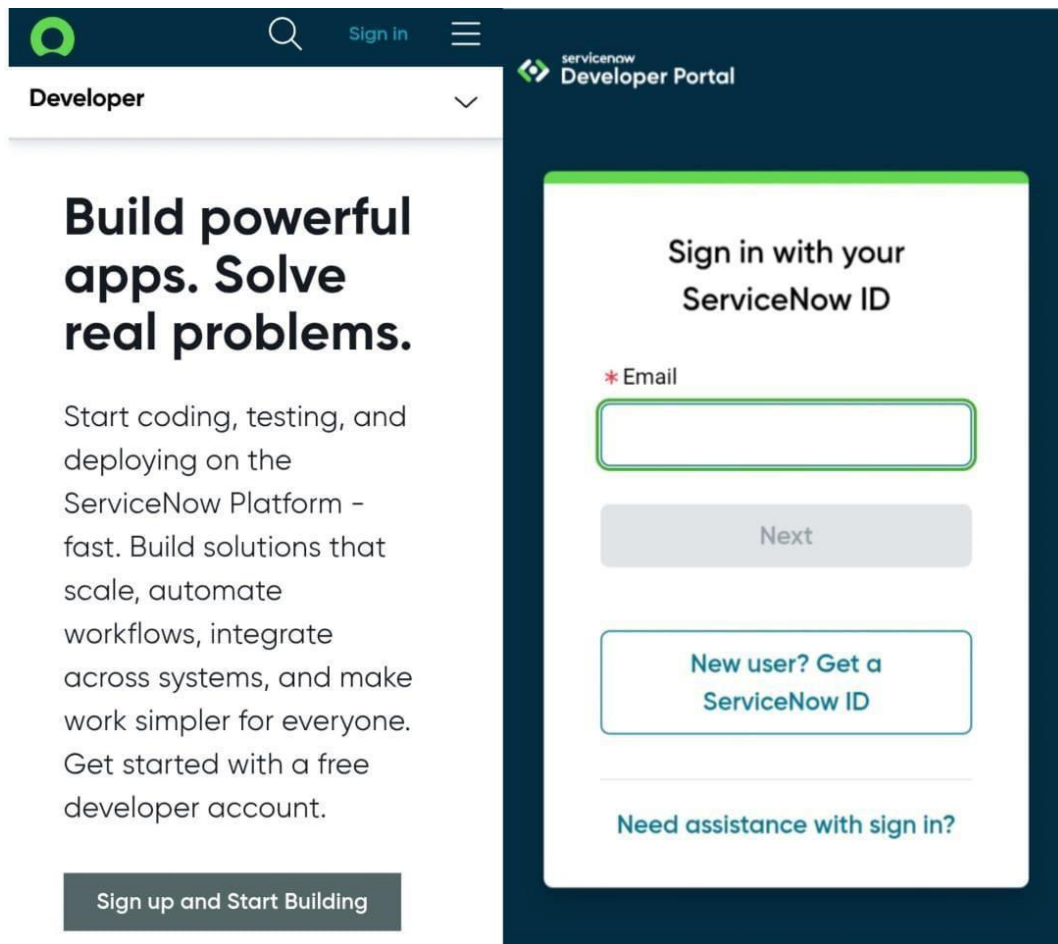
**Communication & Documentation** – Writing clear steps, capturing screenshots, and reporting results.

## TASK INITIATION

### **Milestone 1: ServiceNow Instance**

#### **Activity 1: Setting up ServiceNow Instance**

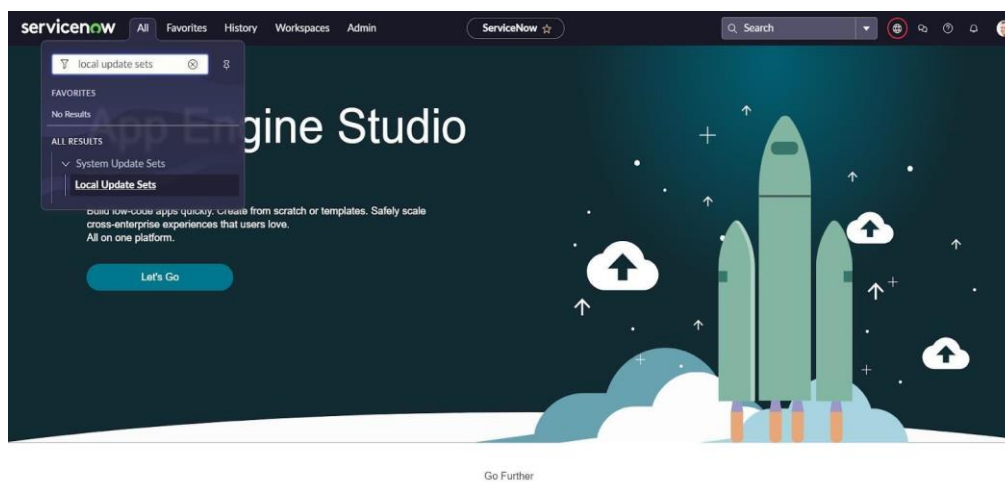
- 1. Sign up for a developer account on the ServiceNow Developer site “<https://developer.servicenow.com>”.**
- 2. Once logged in, navigate to the "Personal Developer Instance" section.**
- 3. Click on "Request Instance" to create a new ServiceNow instance.**
- 4. Fill out the required information and submit the request.**
- 5. You'll receive an email with the instance details once it's ready.**
- 6. Log in to your ServiceNow instance using the provided credentials.**
- 7. Now you will navigate to the ServiceNow.**



## Milestone 2: Update Set

### Activity 2: Creation of update set

1. Go to All >> In the filter search for Local Update set > click on New.



2. Enter the Details as: Name : Family Expenses
3. Then click on Submit and Make current.

The screenshot shows the ServiceNow interface for creating a new update set. The left sidebar shows the navigation menu with 'Local Update Sets' selected. The main form has the following fields:

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

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

### Milestone 3: Family expenses

#### Activity 3: Creation of Family Expenses 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

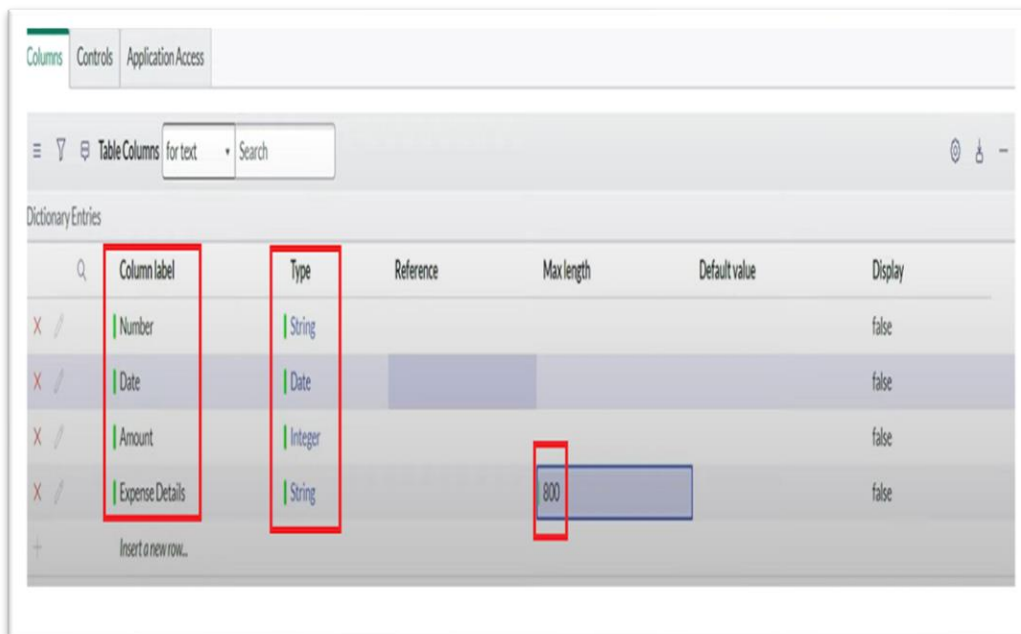
The screenshot shows the ServiceNow interface for configuring a new table. The 'Label' is 'Family Expenses' and the 'Name' is 'u\_st\_family\_expenses'. The 'New menu name' is 'Family Expenditure'. The 'Table Columns' section shows three columns:

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

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

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



## Milestone 5: Number field

### Activity 5: 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.

The screenshot shows the 'Default Value' configuration window. At the top, there are three tabs: 'Choice List Specification', 'Calculated Value', and 'Default Value'. The 'Default Value' tab is selected and highlighted with a red box and a red arrow labeled '1'. Below the tabs, a blue banner states: 'The Default value specifies what value the field has when first displayed.' Under this banner, there is a section with two items: 'Use dynamic default' with a checked checkbox (highlighted with a red box and a red arrow labeled '2'), and 'Dynamic default value' with a text input field containing 'Get Next Padded Number' (highlighted with a red box and a red arrow labeled '3'). At the bottom of the window, there are two buttons: 'Delete Column' and 'Update'. The 'Update' button is highlighted with a red box and a red arrow labeled '4'.

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

The screenshot shows the 'New' form for a Number field. At the top, there is a header bar with a back arrow, a search icon, and buttons for 'Update' and 'Delete'. Below the header, there are several input fields: 'Table' with the value 'Family Expenses' (highlighted with a red box), 'Prefix' with the value 'MFE' (highlighted with a red box), 'Number' with the value '1,000', 'Application' with the value 'Global', and 'Number of digits' with the value '7'. At the bottom of the form, there are two buttons: 'Update' and 'Delete'.

8. Click on Submit.

## Milestone 6: Drag and Drop

### Activity 6: 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 1 Column

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 7: Daily expenses

### Activity 7: Creation of Daily Expenses Table

- a. Go to All > In the filter search for Tables > click on New.
- b. Enter the Details:  
Label : Daily Expenses  
Name : Auto-Populated  
Add Module to menu : Family Expenditure

\* Label Daily Expenses 1

\* Name u\_daily\_expenses 2

Extends table

Application Global

Create module ☒

Create mobile module ☒

Application Menu

Add module to menu Family Expenditure 3

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



## Milestone 8: Column

### Activity 8: 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 : 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.

To create columns, select the text or area you want to format in programs like Microsoft Word and then go to the Layout or Page Layout tab and click Columns to choose the number of columns.

For software like Power Apps, open the form designer, and use the Add column button or drag fields from the Columns pane onto the form preview to create them.

In a database context, like Beaver, you create new columns by accessing the table's properties or the Database Navigator, right-clicking, and selecting an option to add a new column to the table structure.

### Activity 9: 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  
b. Click on Update.

The screenshot shows the 'Default Value' configuration window. At the top, there are three tabs: 'Choice List Specification', 'Calculated Value', and 'Default Value'. The 'Default Value' tab is selected and highlighted with a red box and arrow 1. Below the tabs, a blue banner states: 'The Default value specifies what value the field has when first displayed.' Underneath, there is a section with a checkbox labeled 'Use dynamic default' which is checked, highlighted with a red box and arrow 2. Below this checkbox is a text input field labeled 'Dynamic default value' containing the text 'Get Next Padded Number', also highlighted with a red box and arrow 3. At the bottom of the window, there are two buttons: 'Delete Column' and 'Update'. The 'Update' button is highlighted with a red box and arrow 4.

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

The screenshot shows the 'New record' form for 'Number Maintenance'. At the top, there is a header bar with a back arrow, a menu icon, the text 'Number New record', and a 'Submit' button. Below the header, there are several input fields. The first field is labeled '\* Table' and contains the text 'Daily Expenses', highlighted with a red box and arrow 1. Below it is a field labeled 'Prefix' containing the text 'DFE', highlighted with a red box and arrow 2. Below that is a field labeled '\* Number' with the value '1,000'. Below that is a field labeled 'Application' with the value 'Global'. Below that is a field labeled 'Number of digits' with the value '7'. At the bottom of the form, there is a 'Submit' button highlighted with a red box and arrow 3.

8. Click on Submit.

### Milestone 10: Drag and Drop

#### Activity 10: Configure the Form

- i. Go to All >> In the filter search for Daily Expenses >> Open Daily Expenses
- ii. Click on New
- iii. Go to the Header and right click there >> click on Configure >> Select Form Design
- iv. Customize or Drag Drop the form as per your requirement.

The screenshot shows the 'Form Design' configuration window for 'Daily Expenses'. At the top, there is a header bar with the text 'Daily Expenses [u\_daily\_expenses]' and a '2 Column' dropdown menu. Below the header, there are two rows of fields. The first row contains 'Number' and 'Family Member Name'. The second row contains 'Date' and 'Expense'. Each field has a settings icon (gear) and a close icon (X). Below the first row of fields, there is a section labeled '1 Column' with a 'Comments' field, also with a settings icon and a close icon.

- v. Make Number Read-Only Field by clicking on the gear icon and checking Read-Only
- vi. Make Date, Family Member Name Mandatory Field by clicking on the gear icon and checking Mandatory
- vii. Click on Save.

### Milestone 11: creation of relationship

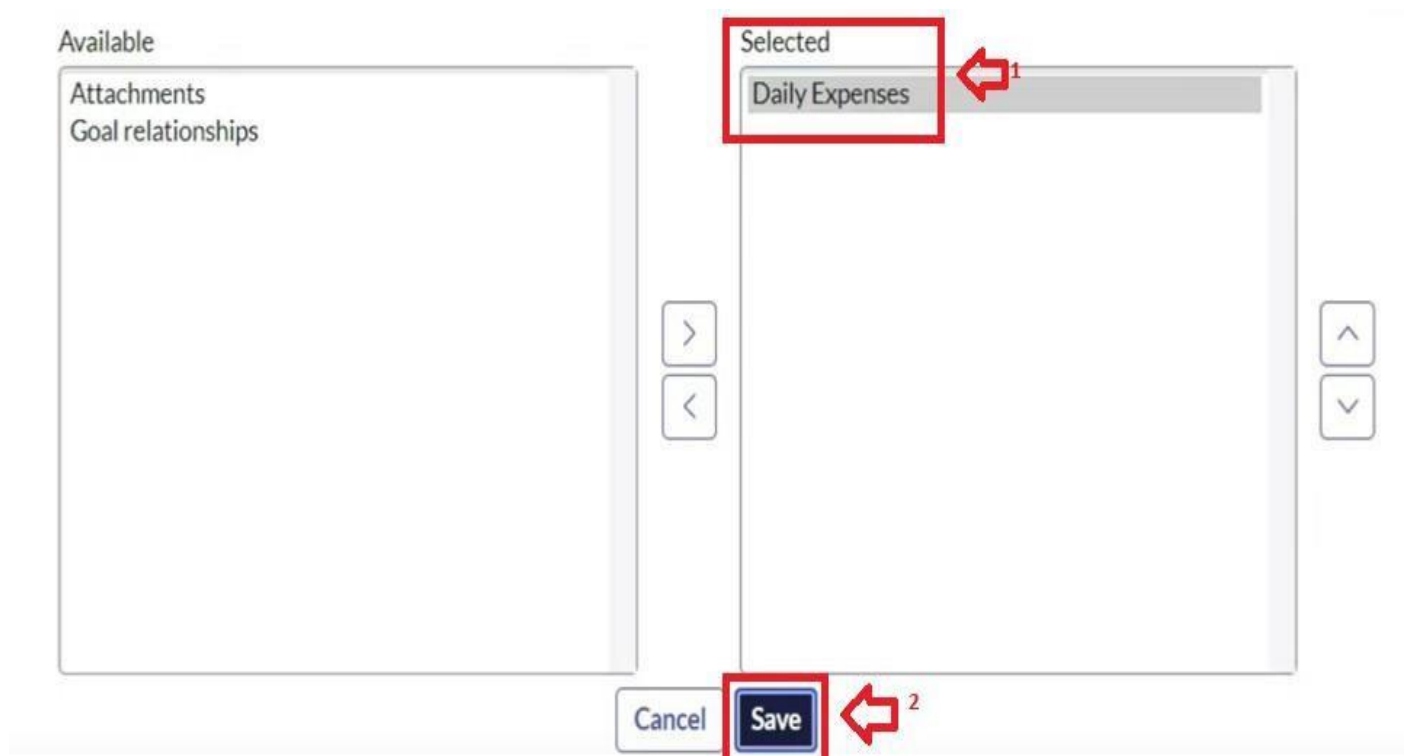
**Activity 11:** Creation of Relationship between Family Expenses and Daily Expenses tables

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.

### Milestone 12: Related list of family expenses

**Activity 12:** Configuring Related List on Family Expenses

- ii. Go to All >> In the filter search for Family Expenses >> Open Family Expenses
- iii. Click on New
- iv. Go to the Header and right click there >> click on Configure >> Select Related Lists
- v. Add Daily Expenses to the Selected Area.
- vi. Click on Save



## Milestone 13: Business rules

### Activity 13: 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

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: Family Expenses BR

Table: Daily Expenses (u.daily\_expenses)

Application: Global

Active: ☒

Advanced: ☒

4. In when to run Check Insert and Update

When to run

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

When: before

Order: 100

Insert: ☒

Update: ☒

Delete: ☐

Query: ☐

Filter Conditions: Add Filter Condition Add "OR" Clause

-- choose field --

-- oper --

-- value --

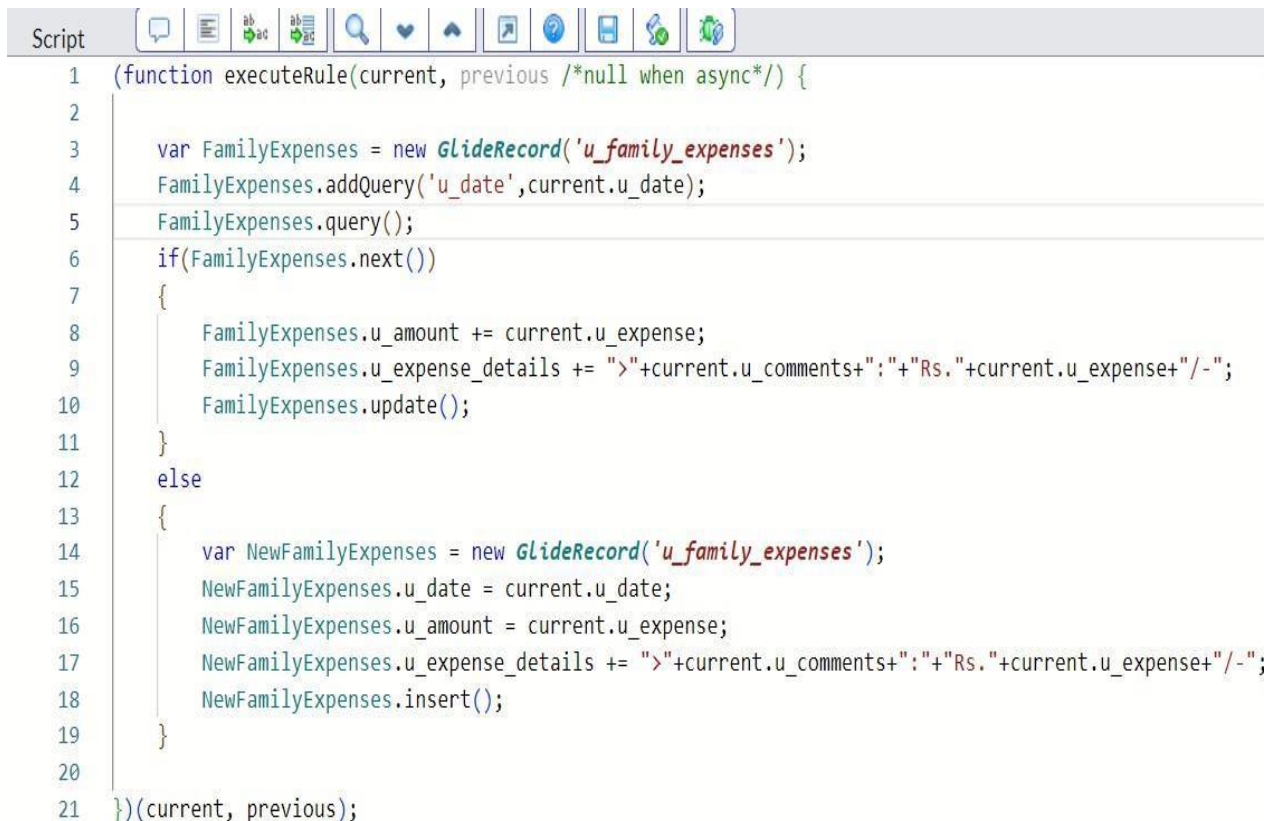
Role conditions

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

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

    var FamilyExpenses = new GlideRecord('u_family_expenses');
    FamilyExpenses.addQuery('u_date',current.u_date);
    FamilyExpenses.query();
    if(FamilyExpenses.next())
    {
        FamilyExpenses.u_amount += current.u_expense;
        FamilyExpenses.u_expense_details +=
        ">" + current.u_comments + ":" + "Rs." + current.u_expense +
        "/-"; FamilyExpenses.update();
    }
    else
    {
        var NewFamilyExpenses = new
        GlideRecord('u_family_expenses'); NewFamilyExpenses.u_date
        = current.u_date; NewFamilyExpenses.u_amount =
        current.u_expense; NewFamilyExpenses.u_expense_details +=
        ">" + current.u_comments + ":" + "Rs." + current.u_expense + "/-";
        NewFamilyExpenses.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         FamilyExpenses.update();
11     }
12     else
13     {
14         var NewFamilyExpenses = new GlideRecord('u_family_expenses');
15         NewFamilyExpenses.u_date = current.u_date;
16         NewFamilyExpenses.u_amount = current.u_expense;
17         NewFamilyExpenses.u_expense_details += ">" + current.u_comments + ":" + "Rs." + current.u_expense + "/-";
18         NewFamilyExpenses.insert();
19     }
20
21     })(current, previous);
```

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

## Milestone 14: Configure the relationship

### Activity 14: 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 refineQuery(current, parent) {  
  
    // Add your code here, such as current.addQuery(field, value);  
    current.addQuery('u_date',parent.u_date);  
    current.query();  
  
})(current, parent);
```

5. Click on Update.

Relationship Daily Expenses

Name: Daily Expenses

Application: Global

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

```
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

## **Conclusion:**

The project “Calculating Family Expenses using ServiceNow” successfully demonstrates how a powerful cloud-based platform can be applied to solve day-to-day

financial management challenges within a family. By

digitizing the expense tracking process, the system eliminates the inefficiencies of manual methods and provides a structured, automated, and secure approach to financial planning.

Overall, the project achieves its objectives of improving financial awareness, promoting savings, and showcasing the real-world application of ServiceNow in household expense management

