

Calculating Family Expenses Using Service Now

Team Id: NM2025TMID13125

Team Members:4

Team Leader : MARUTHUPANDI.M

Team Member 1 : GOKUL.M

Team Member 2 : AKASH.M

Team Member 3 : SURYA.D

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:

Enhance User Experience:

Provide personalized and intuitive experiences for students, faculty, and staff, ensuring they have the tools and support needed to excel.

Improve Operational Efficiency:

Streamline core institutional processes by automating administrative tasks, which improves productivity across departments.

Foster Digital Transformation:

Modernize the institution by adopting digital tools and processes to adapt to the evolving demands of the digital economy and prepare students for the future.

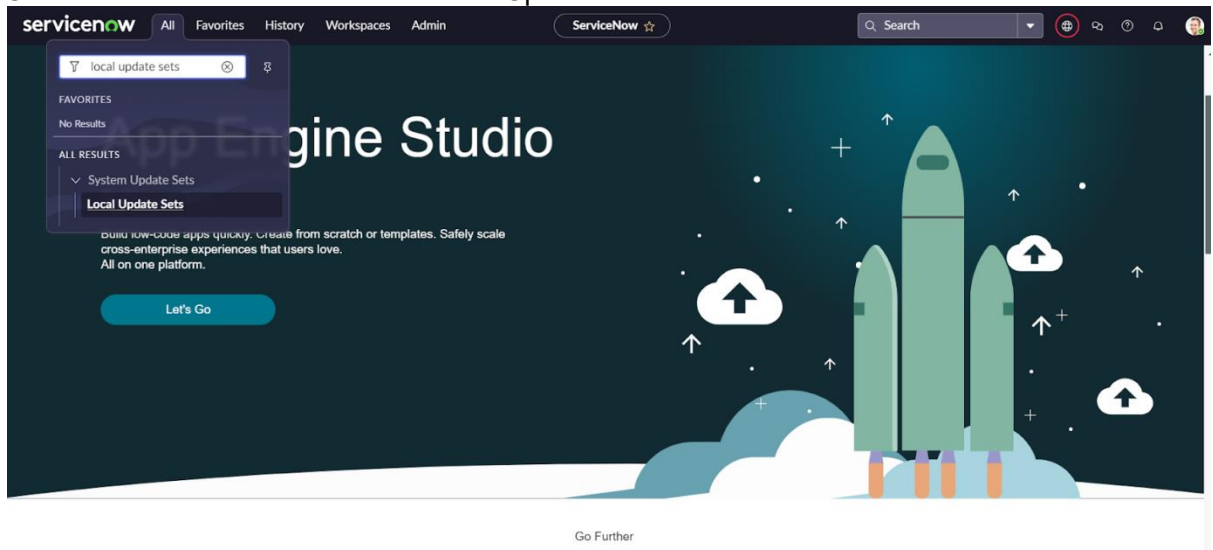
TASK INITIATION

Setting Up ServiceNow Instance

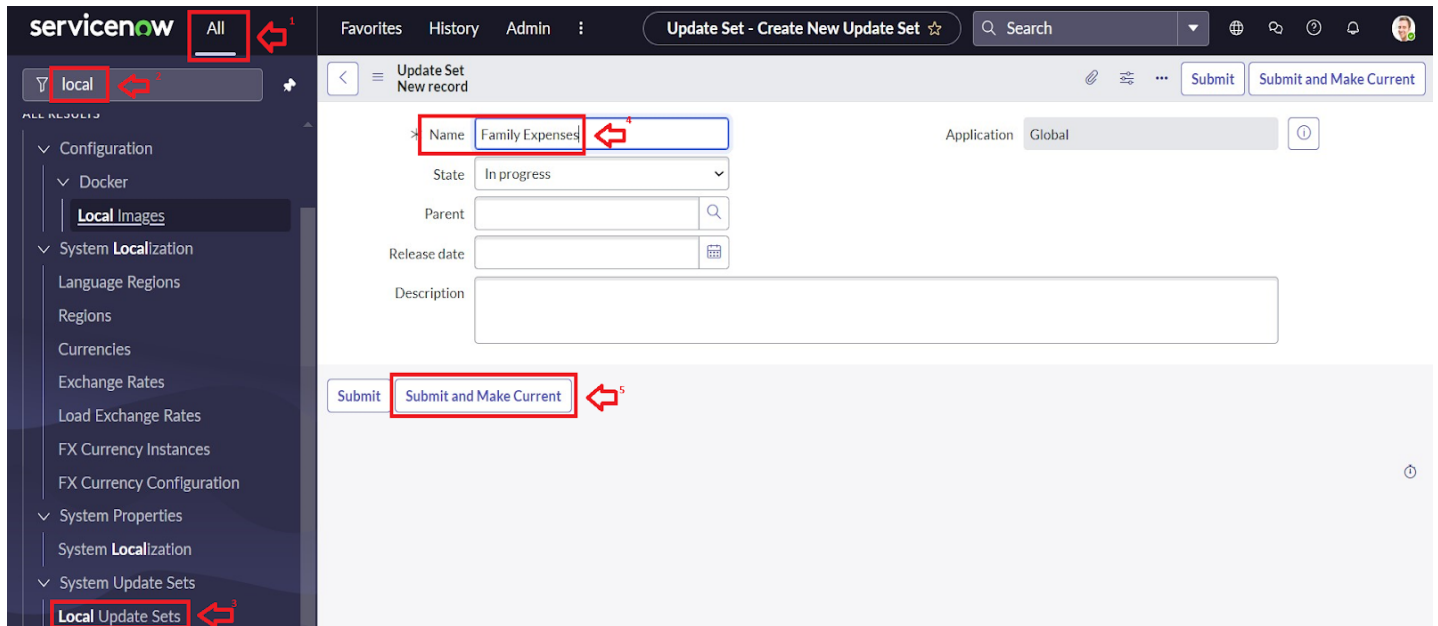
1. Sign up for a developer account on the ServiceNow Developer site "https://developer.servicenow.com".
2. Once logged in, navigate to the "Personal Developer Instance" section.
3. Click on "Request Instance" to create a new ServiceNow instance.
4. Fill out the required information and submit the request.
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.

Creation Of New Update Set

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

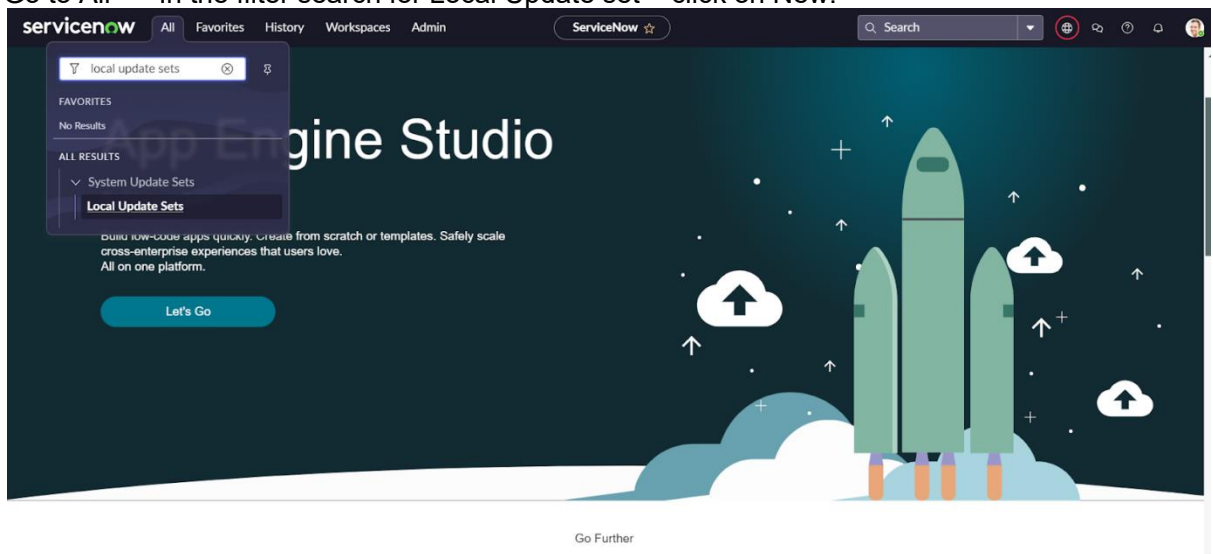


2. Enter the Details as:
Name : Family Expenses
3. Then click on Submit and Make current.

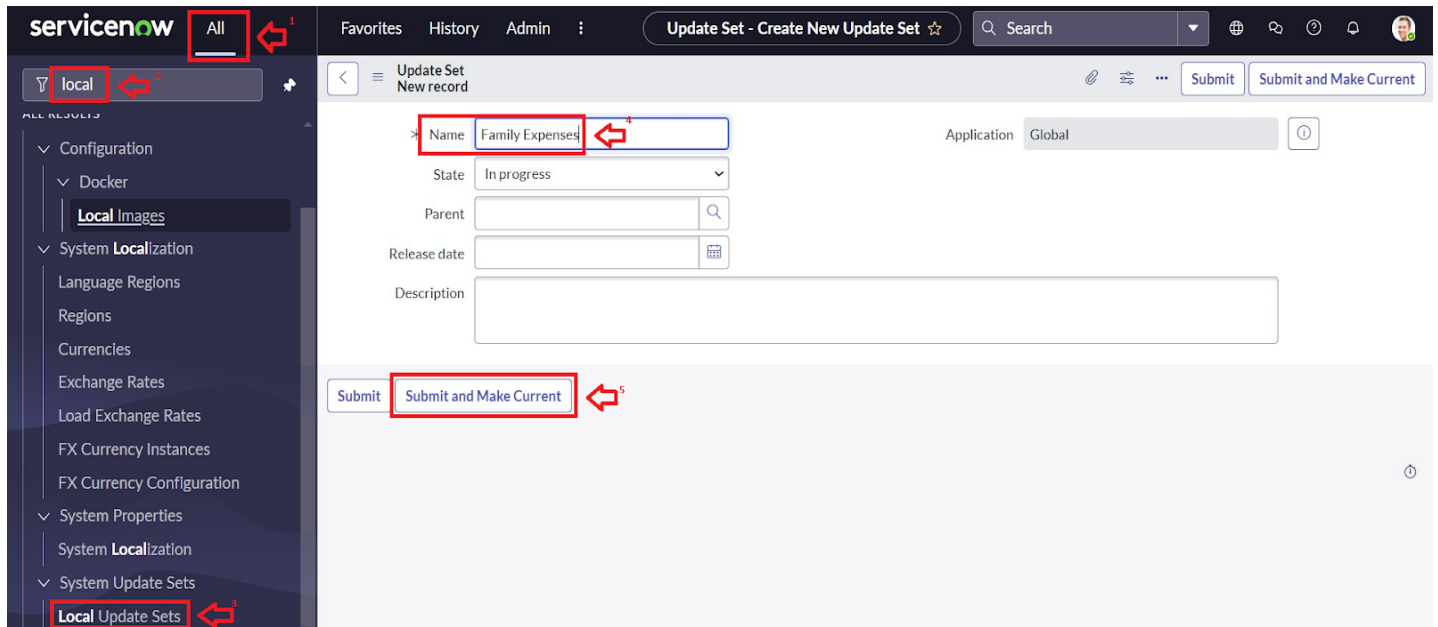


Creation Of New Update Set

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.



servicenow All

Update Set - Create New Update Set

local

Configuration

Docker

Local Images

System Localization

Language Regions

Regions

Currencies

Exchange Rates

Load Exchange Rates

FX Currency Instances

FX Currency Configuration

System Properties

System Localization

System Update Sets

Local Update Sets

Name: Family Expenses

State: In progress

Parent:

Release date:

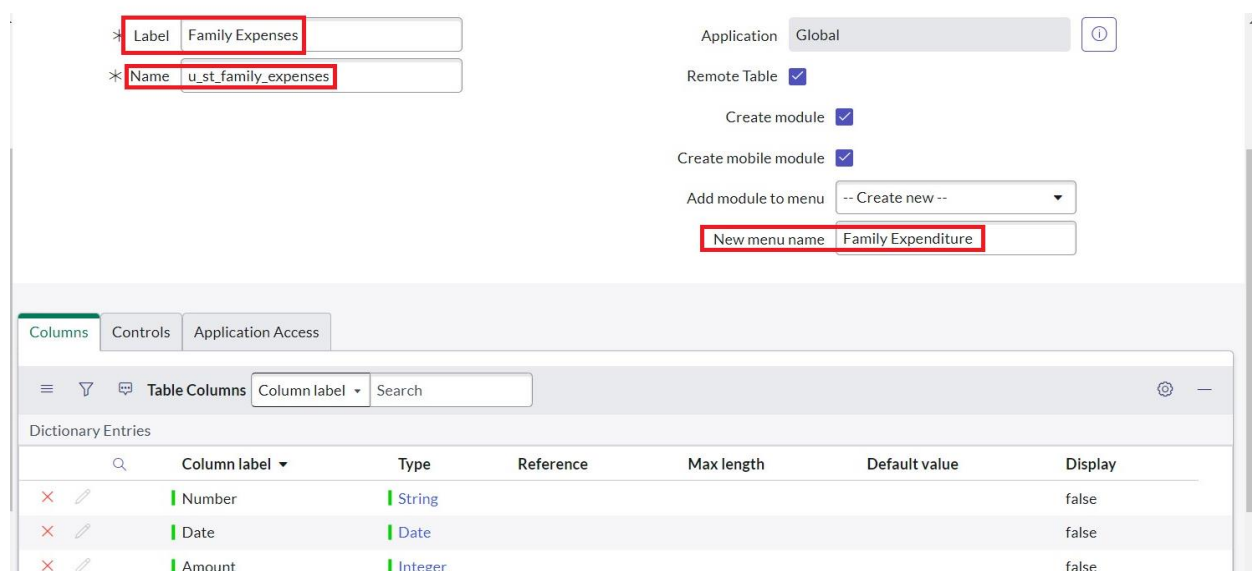
Description:

Application: Global

Submit Submit and Make Current

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



Label: Family Expenses

Name: u_st_family_expenses

Application: Global

Remote Table: ☒

Create module: ☒

Create mobile module: ☒

Add module to menu: -- Create new --

New menu name: Family Expenditure

Columns Controls Application Access

Table Columns

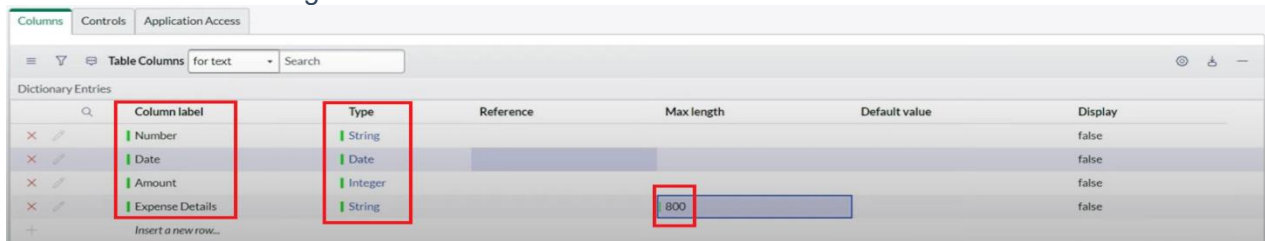
Dictionary Entries

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




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

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

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** 

 **Update** 

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

< Number MFE    **Update** **Delete**

* **Table** **Family Expenses**  

Prefix **MFE**

* Number 1,000

Application Global 

Number of digits 7

Update **Delete**

9. Click on Submit.

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.

Creation Of Daily Expenses Table

1. Go to All > In the filter search for Tables > click on New.
2. 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

☒

Add module to menu

Family Expenditure

3

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

Creation Of Columns(Fields)

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:

Delete Column | **Update**

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



9. Click on Submit.

Configure 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.

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.

Configuring Related List On Family Expenses

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



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

ss 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  ¹


Table  ²

Application 

Active ☒

Advanced ☒  ³


4. In when to run Check Insert and Update

When to run  ¹

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


When

Order

 ² ☒ Insert ☒ Update ☐ Delete ☐ Query

Filter Conditions

-- choose field --

Role conditions 

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

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

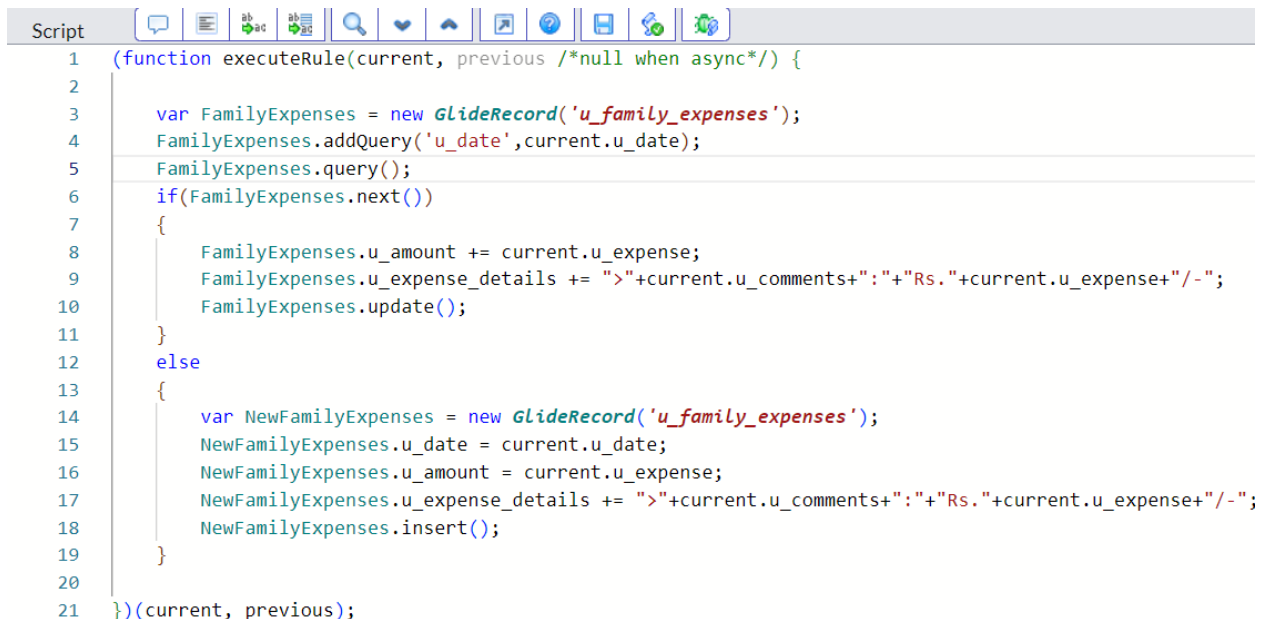
```
var FamilyExpenses = new GlideRecord('u_family_expenses');
FamilyExpenses.addQuery('u_date',current.u_date);
```

```

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);

```

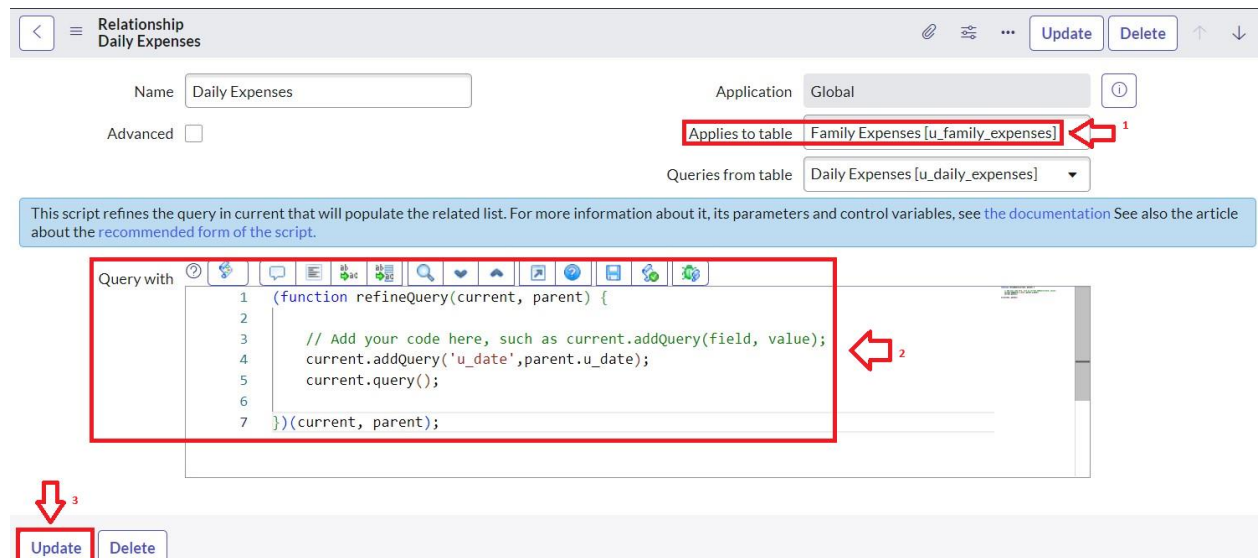
6. Go to the Header and right click there>> click on Save.

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

```
(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);
```

Update Delete