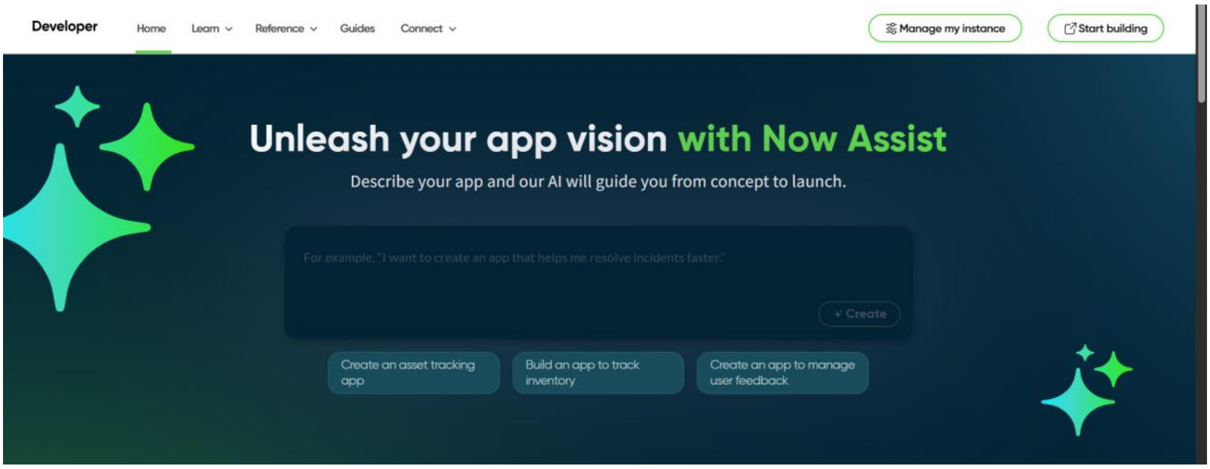
Calculating Family Expenses using Service Now

1. Setting up ServiceNow Instance

Sign up for a developer account at developer.servicenow.com.
Go to Personal Developer Instance → Request Instance and fill in the required details.
Once your instance is ready, log in using the provided credentials to access ServiceNow.



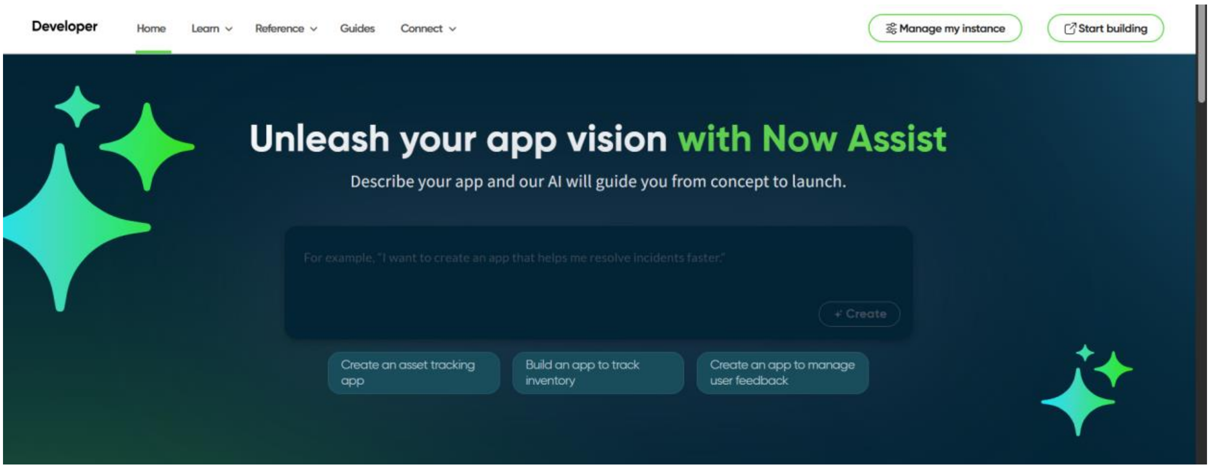
Developer
Home
Learn
Reference
Guides
Connect
Manage my instance
Start building

Unleash your app vision with Now Assist

Describe your app and our AI will guide you from concept to launch.

For example, "I want to create an app that helps me resolve incidents faster."

Create
Create an asset tracking app
Build an app to track inventory
Create an app to manage user feedback


Developer
Home
Learn
Reference
Guides
Connect
Manage my instance
Start building
+ Create
Create an asset tracking app
Build an app to track inventory
Create an app to manage user feedback

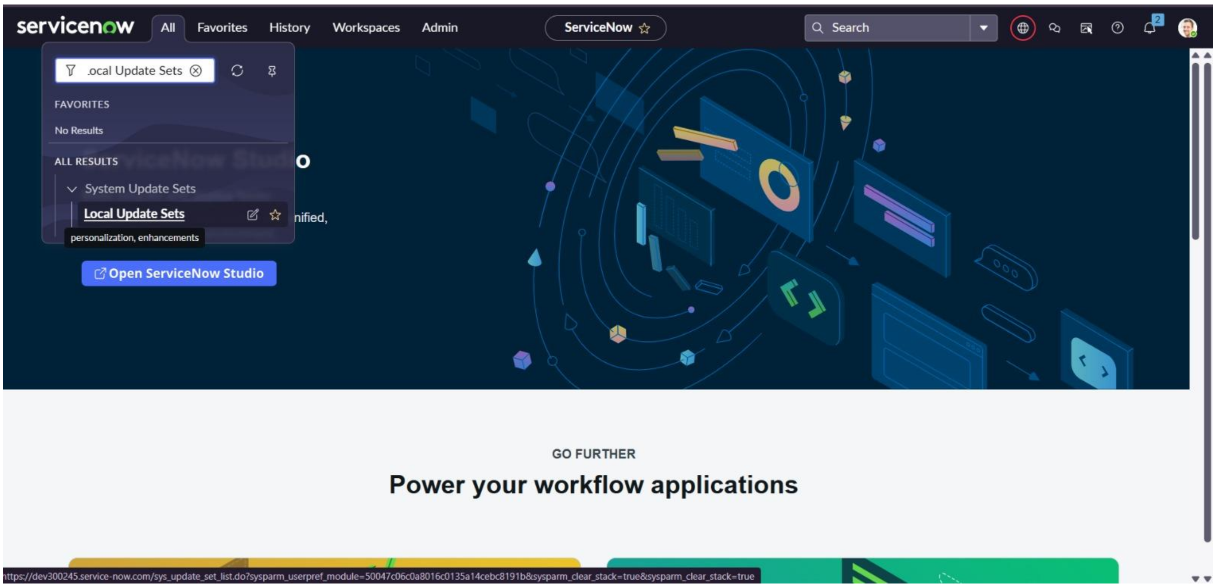
Unleash your app vision with Now Assist

Describe your app and our AI will guide you from concept to launch.

For example, "I want to create an app that helps me resolve incidents faster."

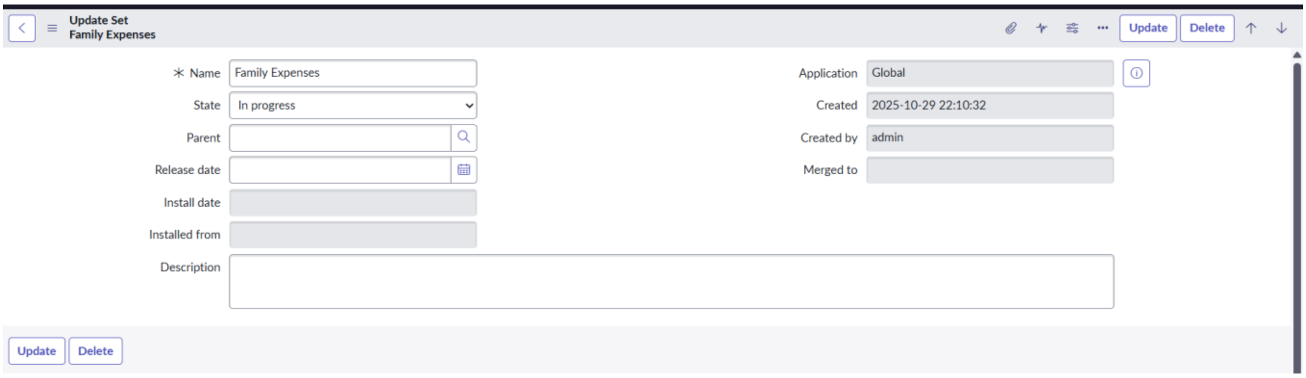
2. Creation of New Update Set

In the filter navigator, search Local Update Set → click New.



https://dev300245.service-now.com/sys_update_set_list.do?sysparm_userpref_module=50047c06c0a8016c0135a14cebc8191b&sysparm_clear_stack=true&sysparm_clear_stack=true

Enter Name: Family Expenses, then click Submit and Make Current.

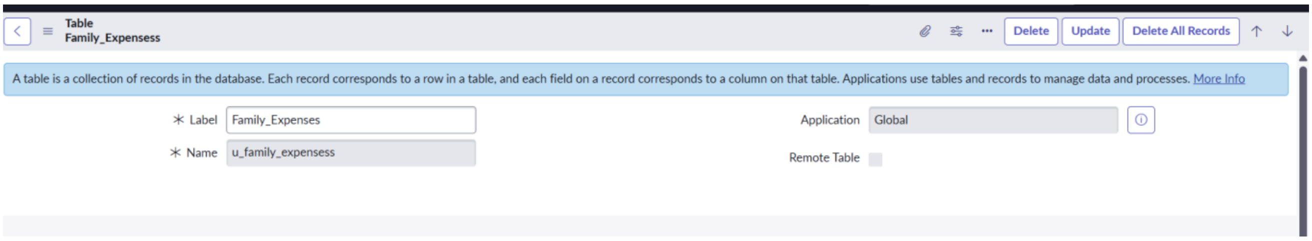


Update Set
Family Expenses

* Name	Family Expenses	Application	Global
State	In progress	Created	2025-10-29 22:10:32
Parent		Created by	admin
Release date		Merged to	
Install date			
Installed from			
Description			

3.Creation of Family Expenses Table

In the filter navigator, search Tables → click New.
Enter Label: Family Expenses, Menu Name: Family Expenditure, then right-click the header and select Save.



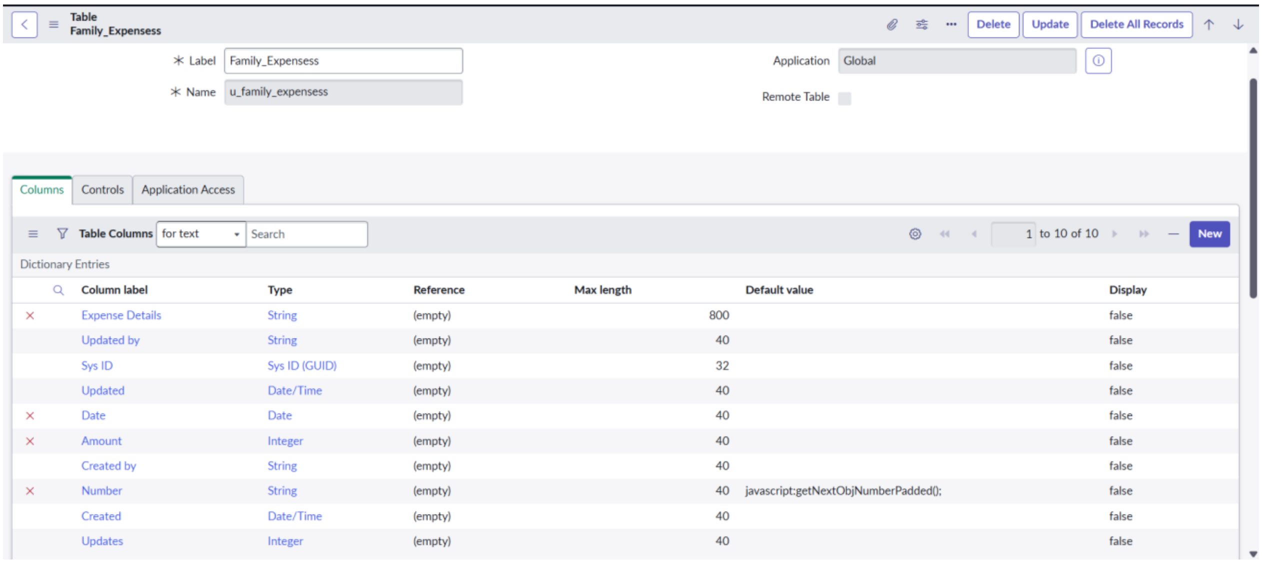
A table is a collection of records in the database. Each record corresponds to a row in a table, and each field on a record corresponds to a column on that table. Applications use tables and records to manage data and processes. More Info

* Label: Family_Expensess
* Name: u_family_expensess
Application: Global
Remote Table: □

4.Creation of Columns (Fields)

Add new rows under **Columns** with the following details:

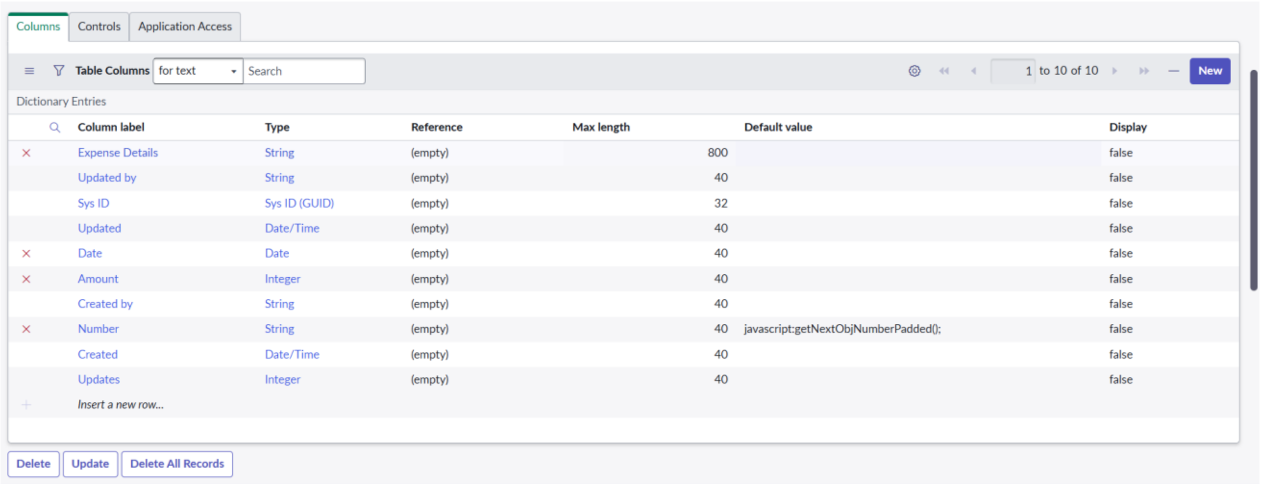
- **Number** – String
- **Date** – Date
- **Amount** – Integer
- **Expense Details** – String (Max length: 800)



* Label: Family_Expensess
* Name: u_family_expensess
Application: Global
Remote Table: □

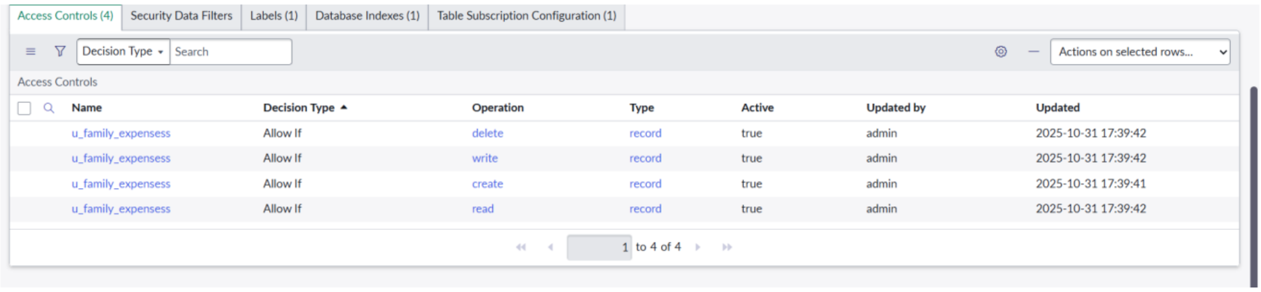
Dictionary Entries

	Column label	Type	Reference	Max length	Default value	Display
×	Expense Details	String	(empty)	800		false
	Updated by	String	(empty)	40		false
	Sys ID	Sys ID (GUID)	(empty)	32		false
	Updated	Date/Time	(empty)	40		false
×	Date	Date	(empty)	40		false
×	Amount	Integer	(empty)	40		false
	Created by	String	(empty)	40		false
×	Number	String	(empty)	40	javascript:getNextObjNumberPadded();	false
	Created	Date/Time	(empty)	40		false
	Updates	Integer	(empty)	40		false



Dictionary Entries

	Column label	Type	Reference	Max length	Default value	Display
×	Expense Details	String	(empty)	800		false
	Updated by	String	(empty)	40		false
	Sys ID	Sys ID (GUID)	(empty)	32		false
	Updated	Date/Time	(empty)	40		false
×	Date	Date	(empty)	40		false
×	Amount	Integer	(empty)	40		false
	Created by	String	(empty)	40		false
×	Number	String	(empty)	40	javascript:getNextObjNumberPadded();	false
	Created	Date/Time	(empty)	40		false
	Updates	Integer	(empty)	40		false
	Insert a new row...					



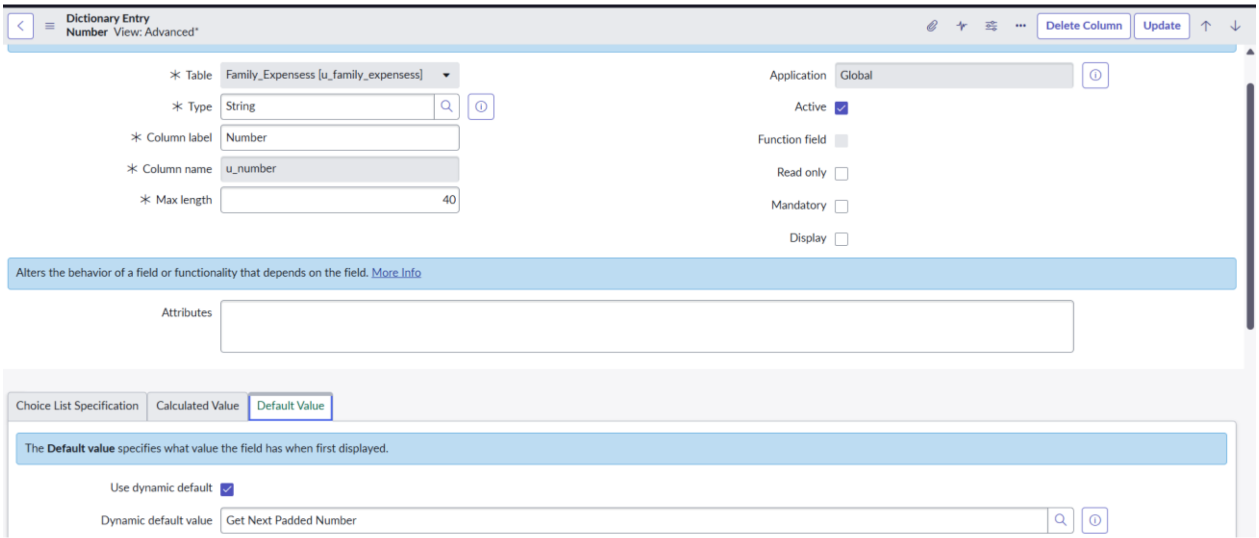
Access Controls

Name	Decision Type ▲	Operation	Type	Active	Updated by	Updated
u_family_expensess	Allow If	delete	record	true	admin	2025-10-31 17:39:42
u_family_expensess	Allow If	write	record	true	admin	2025-10-31 17:39:42
u_family_expensess	Allow If	create	record	true	admin	2025-10-31 17:39:41
u_family_expensess	Allow If	read	record	true	admin	2025-10-31 17:39:42

Right-click the header and select **Save**.

4. Making Number Field an Auto-Number

Open the Number field → Advanced view.
Check Use dynamic default and select Get Next Padded Number, then click Update.



✱ Table	Family_Expensess [u_family_expensess]	Application	Global
✱ Type	String	Active	☑
✱ Column label	Number	Function field	□
✱ Column name	u_number	Read only	□
✱ Max length	40	Mandatory	□
		Display	□

Alters the behavior of a field or functionality that depends on the field. More Info

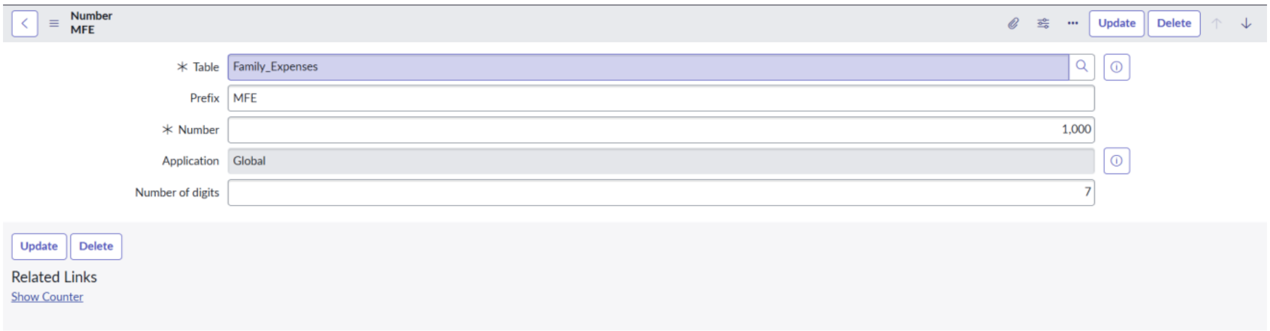
Attributes

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

Use dynamic default ☑

Dynamic default value: Get Next Padded Number

Search Number Maintenance → click New.
Set Table: Family Expenses, Prefix: MFE, and click Submit.



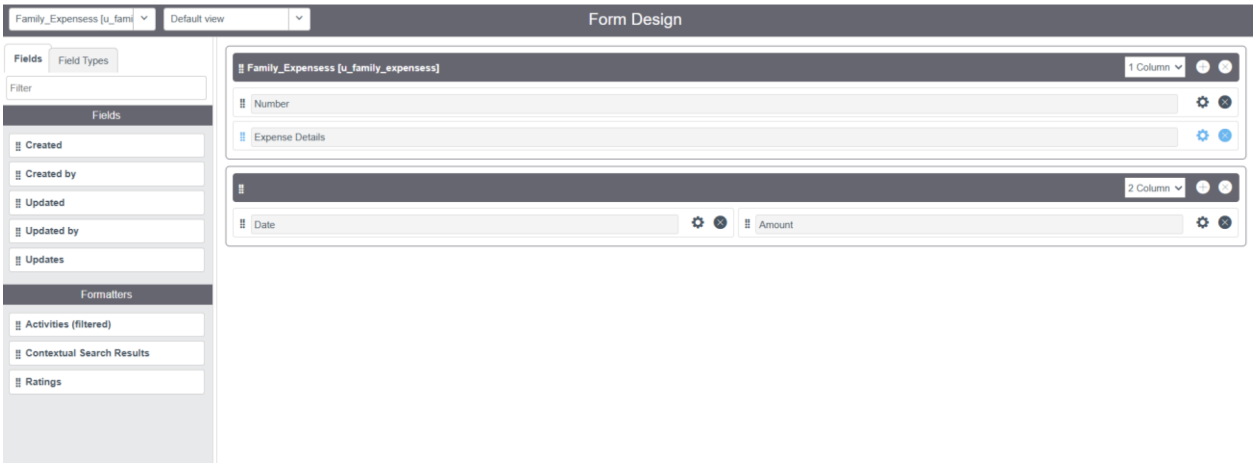
Number
MFE

* Table	Family_Expenses
Prefix	MFE
* Number	1,000
Application	Global
Number of digits	7

Related Links
Show Counter

5. Configure the Form

Search Family Expenses and open it.
Click New, then right-click the header → Configure → Form Design.
Arrange fields as needed.
Set Number as *Read-only* and make Date and Amount *Mandatory*.
Click Save.



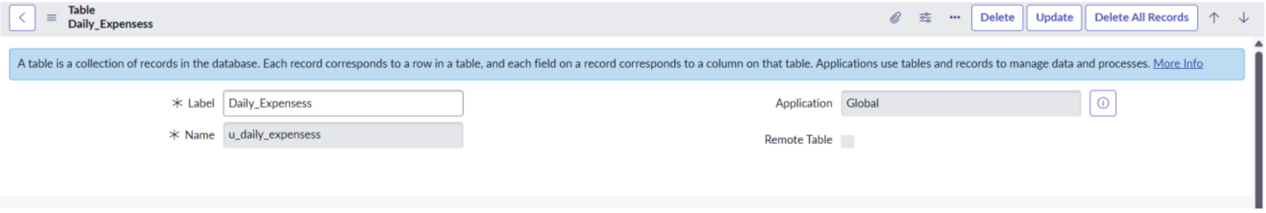
Form Design

Family_Expensess [u_family_expensess] (1 Column): Number, Expense Details

(2 Column): Date, Amount

6.Creation of Daily Expenses Table

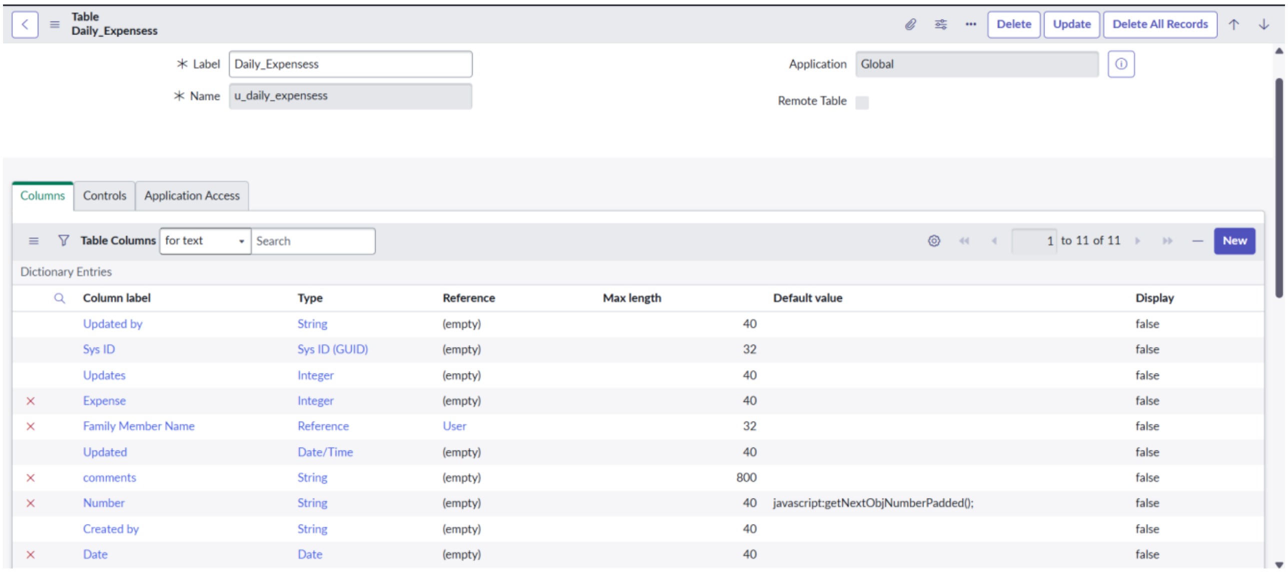
Search Tables → click New.



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	Daily_Expensess	Application	Global
* Name	u_daily_expensess	Remote Table	□

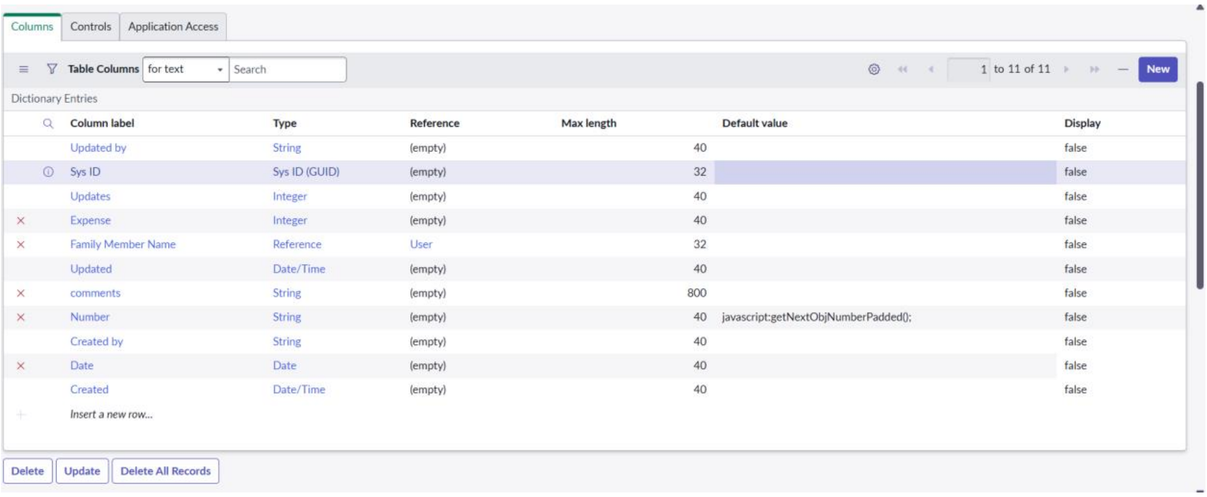
Enter Label: Daily Expenses, add to Menu: Family Expenditure, then right-click the header and select Save.



* Label	Daily_Expensess	Application	Global
* Name	u_daily_expensess	Remote Table	□

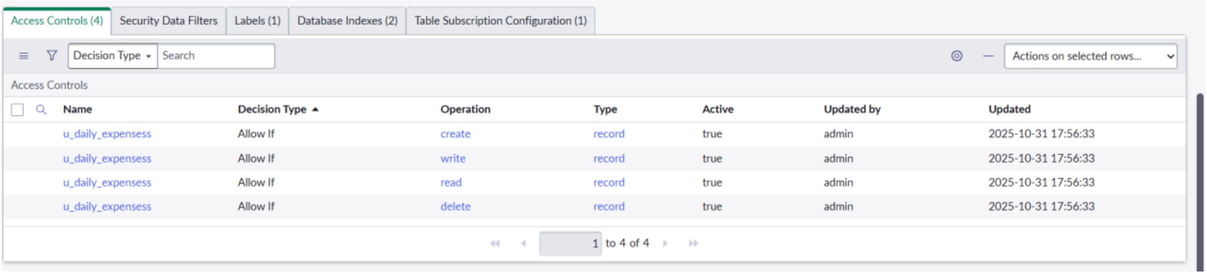
Dictionary Entries

	Column label	Type	Reference	Max length	Default value	Display
	Updated by	String	(empty)	40		false
	Sys ID	Sys ID (GUID)	(empty)	32		false
	Updates	Integer	(empty)	40		false
×	Expense	Integer	(empty)	40		false
×	Family Member Name	Reference	User	32		false
	Updated	Date/Time	(empty)	40		false
×	comments	String	(empty)	800		false
×	Number	String	(empty)	40	javascript:getNextObjNumberPadded();	false
	Created by	String	(empty)	40		false
×	Date	Date	(empty)	40		false



Dictionary Entries

	Column label	Type	Reference	Max length	Default value	Display
	Updated by	String	(empty)	40		false
ⓘ	Sys ID	Sys ID (GUID)	(empty)	32		false
	Updates	Integer	(empty)	40		false
×	Expense	Integer	(empty)	40		false
×	Family Member Name	Reference	User	32		false
	Updated	Date/Time	(empty)	40		false
×	comments	String	(empty)	800		false
×	Number	String	(empty)	40	javascript:getNextObjNumberPadded();	false
	Created by	String	(empty)	40		false
×	Date	Date	(empty)	40		false
	Created	Date/Time	(empty)	40		false
+	Insert a new row...					

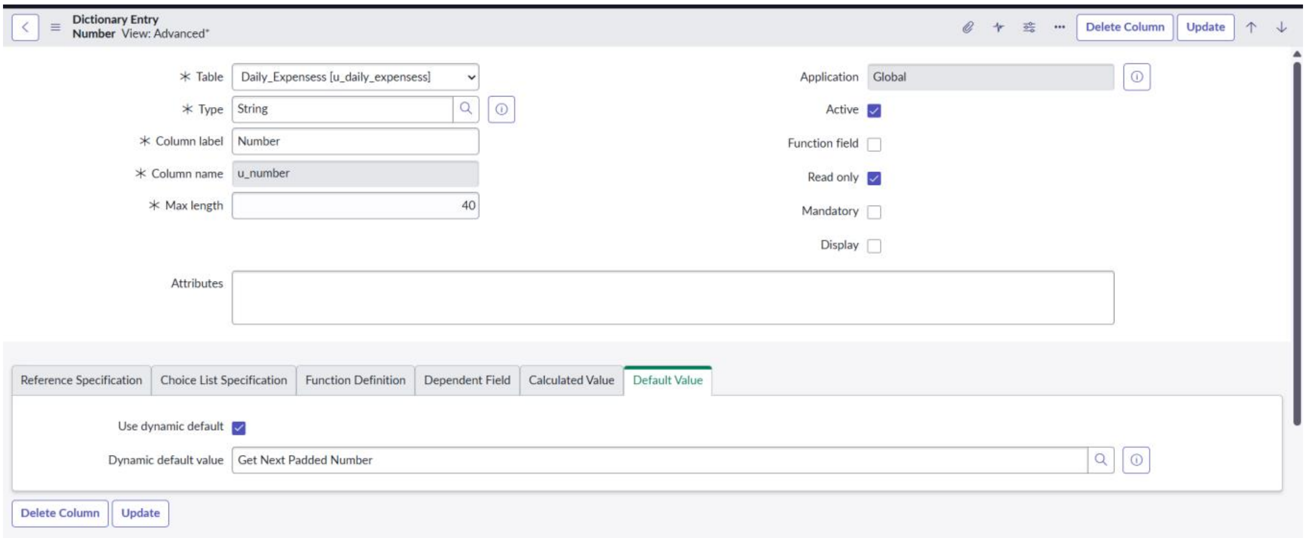


Name	Decision Type	Operation	Type	Active	Updated by	Updated
u_daily_expensess	Allow If	create	record	true	admin	2025-10-31 17:56:33
u_daily_expensess	Allow If	write	record	true	admin	2025-10-31 17:56:33
u_daily_expensess	Allow If	read	record	true	admin	2025-10-31 17:56:33
u_daily_expensess	Allow If	delete	record	true	admin	2025-10-31 17:56:33

1 to 4 of 4

7.Making Number Field an Auto-Number

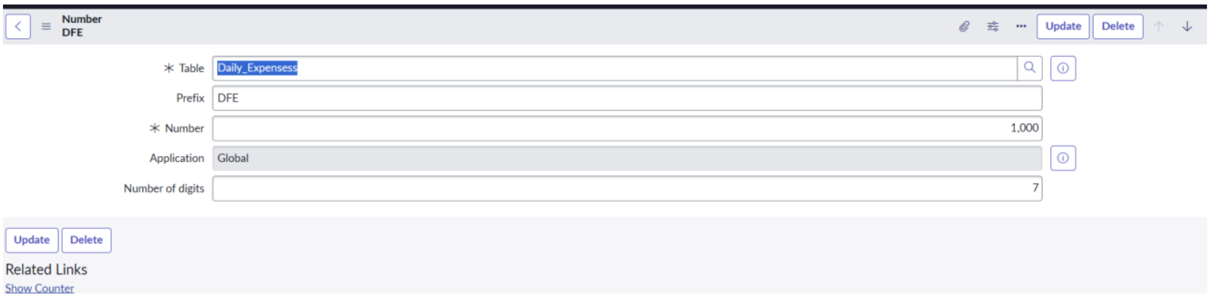
Open the Number field → Advanced view.
Enable Use dynamic default and choose Get Next Padded Number, then click Update



* Table	Daily_Expensess [u_daily_expensess]	Application	Global
* Type	String	Active	☑
* Column label	Number	Function field	□
* Column name	u_number	Read only	☑
* Max length	40	Mandatory	□
		Display	□
Attributes			

Use dynamic default	☑
Dynamic default value	Get Next Padded Number

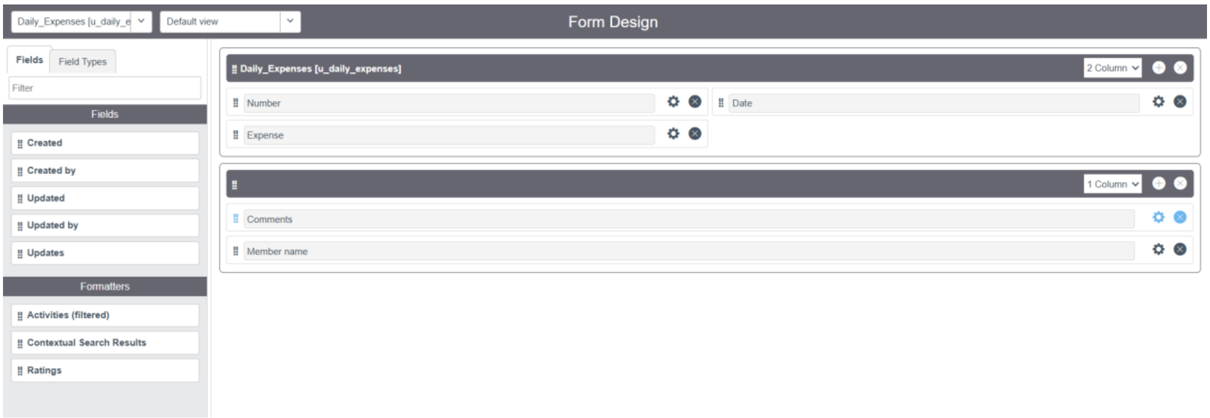
Search Number Maintenance → New.
Set Table: Daily Expenses, Prefix: DFE, and click Submit.



* Table	Daily_Expensess
Prefix	DFE
* Number	1,000
Application	Global
Number of digits	7

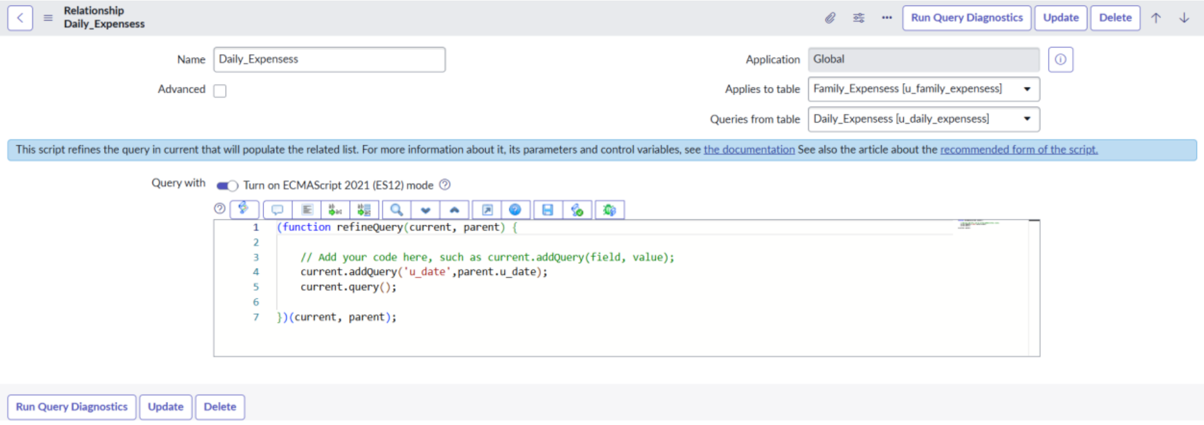
8.Configure the Form

Search Daily Expenses and open it.
Click New, then right-click the header → Configure → Form Design.
Arrange fields as needed.
Set Number as *Read-only* and make Date and Family Member Name *Mandatory*.
Click Save.



9.Creating Relationship Between Tables

Search Relationships → click New.
Set Name: Daily Expenses, Applies to Table: Family Expenses, Related Table: Daily Expenses, then click Save.



Name	Daily_Expensess	Application	Global
Advanced	□	Applies to table	Family_Expensess [u_family_expensess]
		Queries from table	Daily_Expensess [u_daily_expensess]

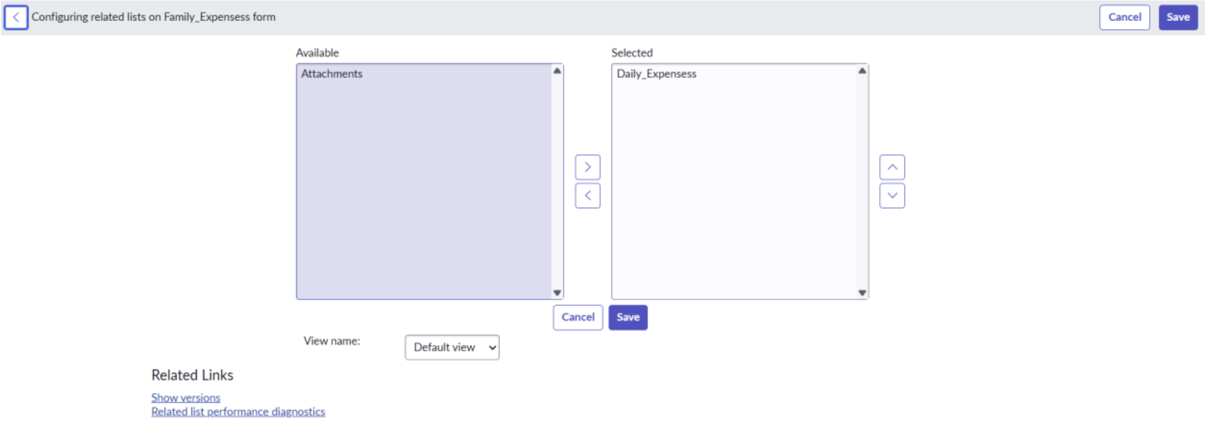
This script refines the query in current that will populate the related list. For more information about it, its parameters and control variables, see the documentation See also the article about the recommended form of the script.

Query with ● Turn on ECMAScript 2021 (ES12) mode

```
(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);
```

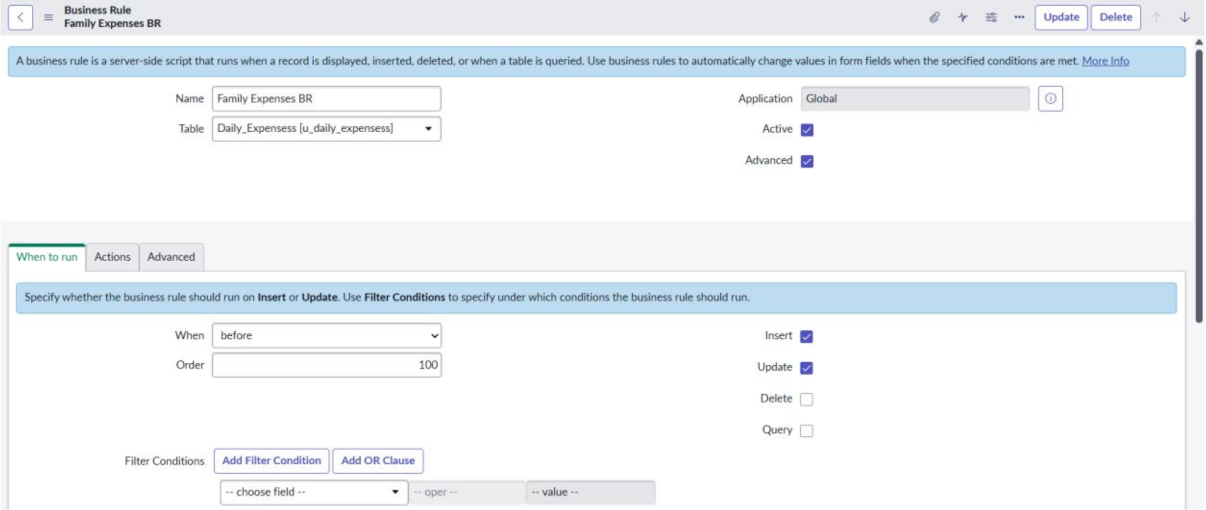
10.Configuring Related List

Open Family Expenses → New → Configure > Related Lists → add Daily Expenses → Save.



11.Business Rule Creation

Navigate to All → Business Rules → New.
Name: Family Expenses BR
Table: Daily Expenses
Add query: required



Business Rule
Family Expenses BR

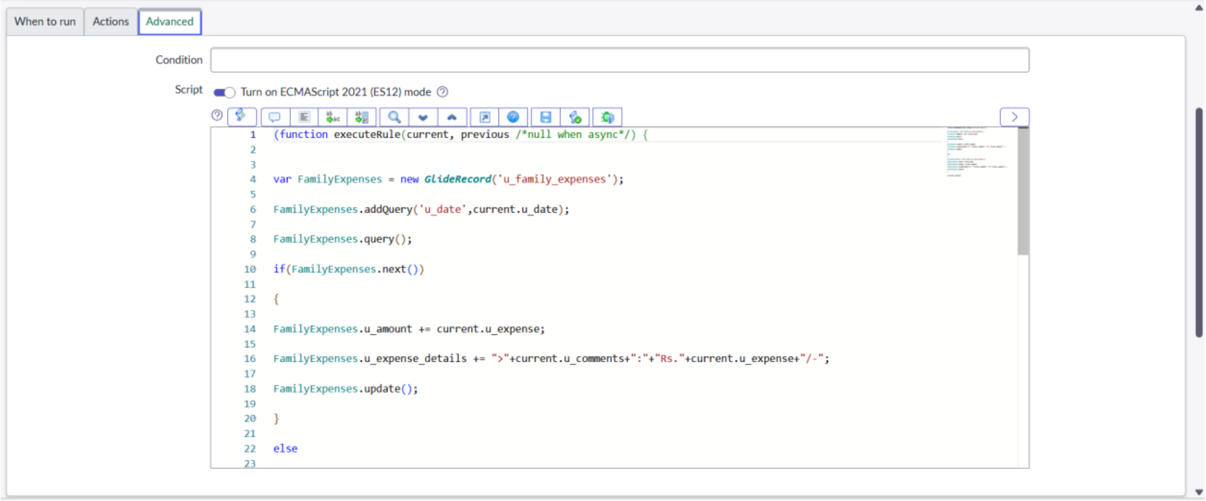
A business rule is a server-side script that runs when a record is displayed, inserted, deleted, or when a table is queried. Use business rules to automatically change values in form fields when the specified conditions are met. More Info

Name	Family Expenses BR	Application	Global
Table	Daily_Expensess [u_daily_expensess]	Active	☑
		Advanced	☑

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

When	before	Insert	☑
Order	100	Update	☑
		Delete	☐
		Query	☐

Filter Conditions: Add Filter Condition | Add OR Clause
-- choose field -- | -- oper -- | -- value --



Condition

Script: Turn on ECMAScript 2021 (ES12) mode

```
(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

```

12.Configure Relationship

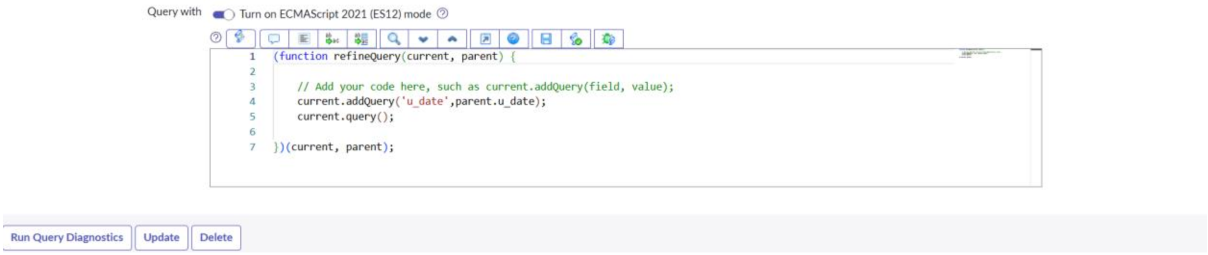
Go to All → Relationships and open Daily Expenses Relationship.
Set Applies to table: Family Expenses.

Add Query:

(function refine Query (current, parent) {

current. add Query('update', parent. u_date);

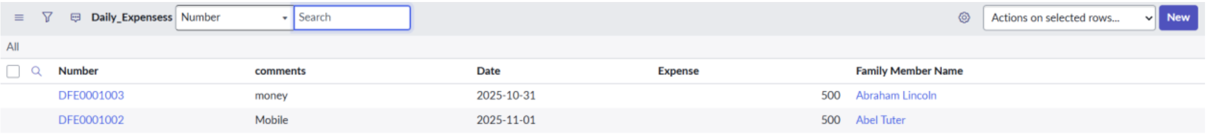
```
  current. query ();
}) (current, parent);
```



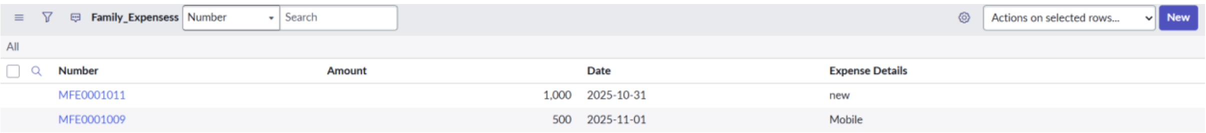
```
(function refineQuery(current, parent) {

    // Add your code here, such as current.addQuery(field, value);
    current.addQuery('u_date',parent.u_date);
    current.query();

})(current, parent);
```

Update the Relationships by clicking on update.

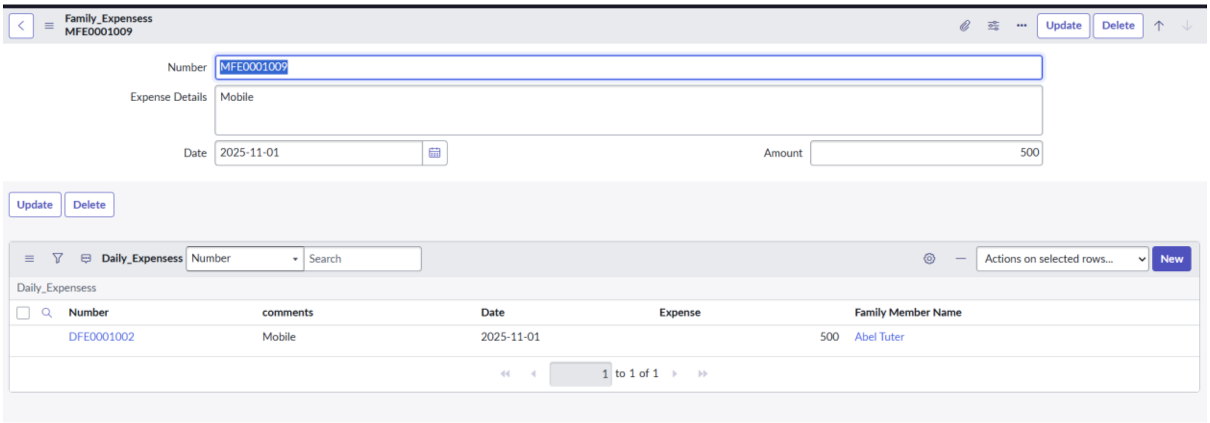
- Built in **ServiceNow** to manage family spending.
- **Daily Expenses** table records:
  - Date, Amount, Comments, Family Member
- **Family Expenses** table shows:
  - Total spent per date, with summary
- Tables are **linked**:
  - Daily entries update family totals automatically
- Helps track spending and keep financial records organized.



Number	comments	Date	Expense	Family Member Name
DFE0001003	money	2025-10-31	500	Abraham Lincoln
DFE0001002	Mobile	2025-11-01	500	Abel Tuter



Number	Amount	Date	Expense Details
MFE0001011	1,000	2025-10-31	new
MFE0001009	500	2025-11-01	Mobile



Number: MFE0001009

Expense Details: Mobile

Date: 2025-11-01

Amount: 500

Number	comments	Date	Expense	Family Member Name
DFE0001002	Mobile	2025-11-01	500	Abel Tuter

1 to 1 of 1

Conclusion:

In conclusion, the *Family Expenses Calculation System* built on ServiceNow provides an efficient and organized way to manage household finances. By leveraging ServiceNow's automation and data management capabilities, the system simplifies expense tracking, ensures accuracy, and offers real-time insights into family spending patterns. This project not only enhances financial transparency but also promotes better budgeting and informed decision-making—ultimately contributing to improved financial stability and well-being for families.

Done By,

Team ID: NM2025TMID06301

Team Leader: Ranish T

Team Member: Simiyon J

Team Member: Uthaya Kumar M

Thank You!