

CALCULATING FAMILY EXPENSSES USING SERVICENOW

TEAM ID: NM2025TMID17921

TEAM MEMBERS: 4

,

TEAM LEADER: MADEEHA ANJUM.H

TEAM MEMBER: MAGESHWARAN.S

TEAM MEMBER 2: SHYAM.A

TEAM MEMBER 3: LOGESWARI.V

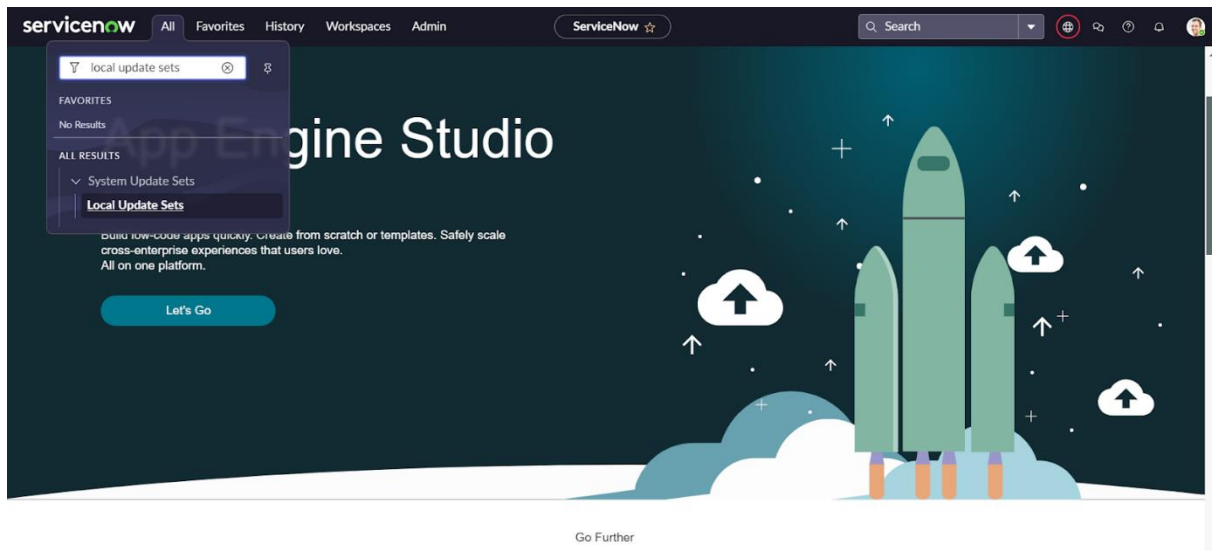
TASK INITATION

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 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: ☒

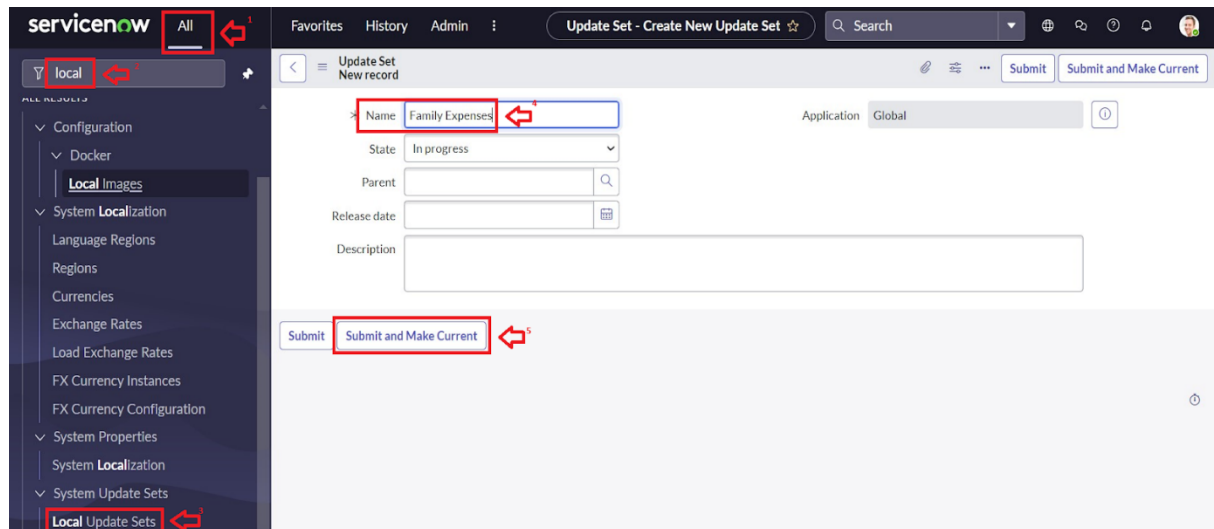
Create module: ☒

Add module to menu: -- Create new --

New menu name: Family Expenditure

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

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

Columns						
Table Columns						
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

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

Making Number Field an Auto-Number

- Double click on the Number Field/Column.
- Go down and double click on Advanced view
- In Default Value:

Use dynamic default : check the box

Dynamic default value : Get Next Padded Number

- 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**

-
- Go to All >> In the filter search for Number Maintenance >> select Number Maintenance
- Click on New.
- Enter the below Details:

Table : Family Expenses

Prefix : MFE

Number MFE

* 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 ↩ 1

* Name ↩ 2

Extends table

Application ⓘ

Create module ☒

Create mobile module ☒

Application Menu

Add module to menu ↩ 3

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 : 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.

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.

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

The screenshot shows a 'New record' form for a table named 'Daily Expenses'. The form includes the following fields and annotations:

- Table:** 'Daily Expenses' (Annotation 1: red box and arrow pointing to the field name)
- Prefix:** 'DFF' (Annotation 2: red box and arrow pointing to the field name)
- Number:** A numeric field with a value of '1,000'.
- Application:** 'Global'.
- Number of digits:** A field with a value of '7'.
- Submit:** A blue button at the bottom (Annotation 3: red box and arrow pointing to the button)

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.

The screenshot shows the 'Form Design' configuration interface for the 'Daily Expenses' table. The form is titled 'Daily Expenses [u_daily_expenses]' and is set to '2 Column' layout. The fields are arranged in two columns:

- Left Column:** 'Number' and 'Date'.
- Right Column:** 'Family Member Name' and 'Expense'.
- Bottom Section:** A single column containing the 'Comments' field.

Each field has a gear icon for configuration and a close icon.

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

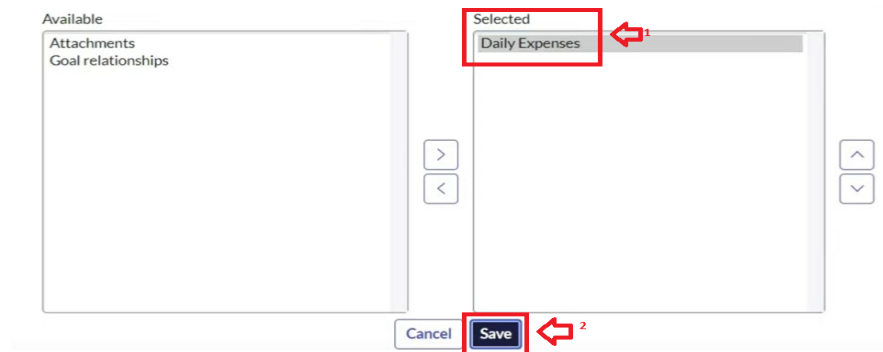
Applies to table : Select Family Expenses

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: Family Expenses BR ← 1

Table: Daily Expenses (u_daily_expenses) ← 2

Application: Global 1

Active ☒

Advanced ☒ ← 3

4. In when to run Check Insert and Update

When to run ← 1 Advanced

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

When: before

Order: 100

Filter Conditions: Add Filter Condition Add "OR" Clause

-- choose field -- -- oper -- -- value --

Role conditions: ✎

Insert ☒ ← 2

Update ☒ ← 3

Delete ☐

Query ☐

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

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

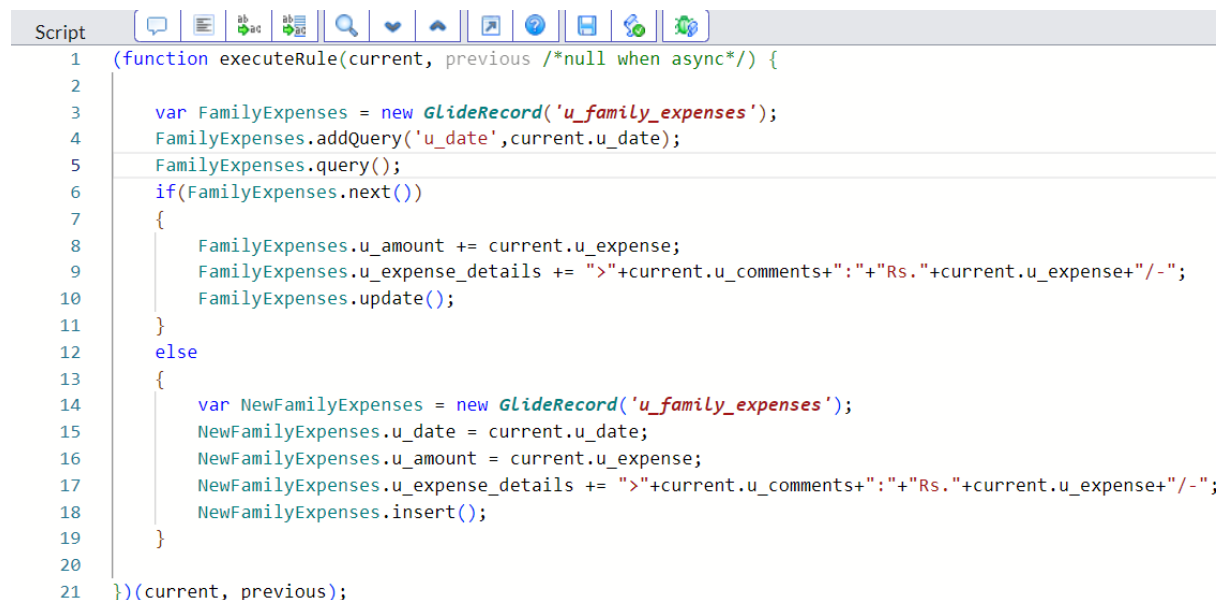
var FamilyExpenses = new GlideRecord('u_family_expenses');
FamilyExpenses.addQuery('u_date',current.u_date);
FamilyExpenses.query();
if(FamilyExpenses.next())
{
FamilyExpenses.u_amount += current.u_expense;
FamilyExpenses.u_expense_details +=
">" + current.u_comments + ":" + "Rs." + current.u_expense + "/-";
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

```

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 Family Expenses Calculator developed using ServiceNow provides a simple, efficient, and user-friendly platform to manage and analyze household spending. By utilizing ServiceNow's workflow automation and database capabilities, the project ensures that expense records are stored securely, categorized properly, and made accessible anytime. This not only helps families maintain financial discipline but also offers clear insights into monthly and yearly spending patterns.

The integration of expense tracking with ServiceNow demonstrates how cloud-based platforms can be extended beyond enterprise use to solve everyday challenges in personal finance management. Overall, this project highlights the importance of digital solutions in promoting financial awareness, encouraging savings, and supporting better decision-making for families.

