



MANGAYARKARASI COLLEGE OF ENGINEERING

Department Of Computer Science Engineering

Completed a project on

CALCULATING FAMILY EXPENSES USING SERVICENOW

Submitted By

Team ID : NM2025TMID04304

1. **Team Leader** : Vansika R K
2. **Team Member** : Srimadhumitha S
3. **Team Member** : Subbulakshimi N
4. **Team Member** : Dharshini M

Calculating Family Expenses Using ServiceNow

Problem Statement

In a project of managing family expenses is often challenging due to the lack of a centralization system for tracking income, expenses, and savings. Families typically rely on manual methods such as spreadsheet, paper records, or multiple apps, which can lead to errors, duplication, and difficulty in analyzing financial patterns.

Objective

Simplify Expense Tracking- provide a centralization system for recording and categorizing all family expenses

Improve Financial Awareness-offer real-time dashboards and reports to give clear insights into spending patterns

Enhance Budget Planning-support better financial planning by identify high-spending categories and opportunities to save

Skills: Servicenow Development, Workflow&Automation, Scripting in Servicenow, Reporting&Dashboards, Integration Skills

TASK INITIATION

Update

Activity 1:

1.Open Service now

2.Go to All>>search for Local Update Set>Click on New



3.Enter the Details as:

Name:Family Expenses

4.And then click on Submit and make current

The screenshot shows the ServiceNow interface for creating a new update set. The left sidebar contains a navigation menu with the following items: 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, and Local Update Sets. The 'Local Update Sets' item is highlighted. The main form is titled 'Update Set - Create New Update Set' and includes a search bar and a 'Submit and Make Current' button. The form fields are: Name (Family Expenses), State (In progress), Parent, Release date, and Description. The Application is set to Global. At the bottom, there are two buttons: 'Submit' and 'Submit and Make Current', with the latter being highlighted.

Table

Activity 2:

- 1.Open Service now
- 2.Go to all>search for Tables>Click on New
- 3.Enter the Details:

Label:Family Expenses

Name:Auto-Populated

New menu name:Family Expenditure

The screenshot shows a configuration window for a new menu item. On the left, there is a list of fields with their labels and names. The 'Label' field is set to 'Family Expenses' and the 'Name' field is set to 'u_st_family_expenses'. Both fields are highlighted with red boxes. On the right, there are several options: 'Application' is set to 'Global', 'Remote Table' is checked, 'Create module' is checked, and 'Create mobile module' is checked. There is also a 'Create menu' button at the bottom right.

Field	Value
Label	Family Expenses
Name	u_st_family_expenses

Application: Global

Remote Table: ☒

Create module: ☒

Create mobile module: ☒

Create menu

4.Click on save

Activity 2:

1.Open Service now

2.Near Columns Double Click near insert a new row

Give the details as:

Column label:Number

Type:String

3.Double click on insert a new row again

Give the details as:

Column label:Date

Type:Date

4.Double click on insert a new row again

Give the details as:

Column label:Amount

Type:Integer

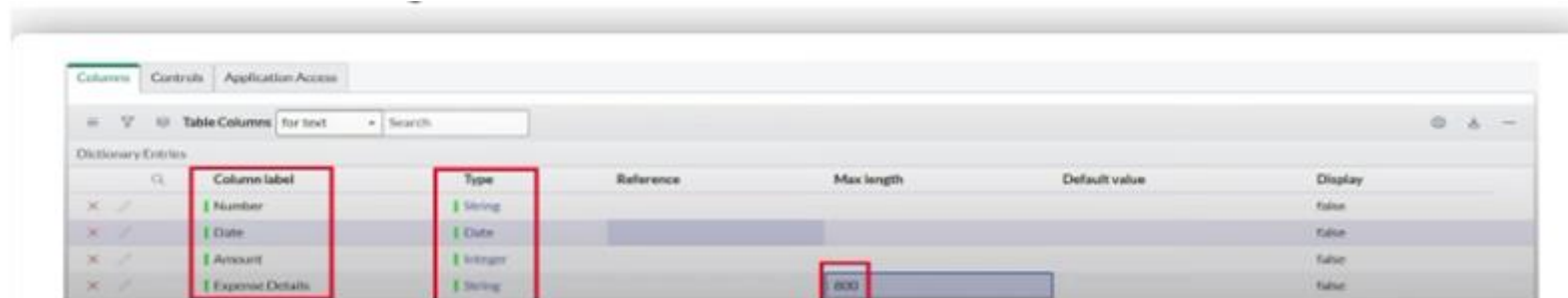
5 .Double click on insert a new row again

Give the details as:

Column label:Expense Details

Type:String

Max length:800



6.Click on save

Activity 2:

1. Open Service now
2. Double click on the Number field/Column
3. Go down and double click on Advanced view
4. In Default Value:

Use dynamic default: check the box

Dynamic default value: Get Next Padded Number

5. click on update

6. Go to all>>search for Number Maintenance>>select number maintenance

7. click on new

8. Enter the details:

Table: Family Expenses

Prefix: MFE

Number MFE

Table Family Expenses

Prefix MFE

Number 1,000

Application Global

Number of digits 7

Update Delete

8. Click on Submit

Activity 2:

1. Open Service now
2. Go to all >> search for Family Expenses >> Open family Expenses
3. Click on new
4. Go to header and click there >> click on configure >> Select Form Design
5. Drag Drop the form as requirement

Family Expenses [u_family_expenses] 2 Column

Number Date

Amount Expense Details

6. Make Number Read-Only Field by Clicking
7. Make Date, Amount Mandatory field by Clicking
8. Click on save

Table(Daily Expenses)

Activity 3:

1. Open Service now
2. Go to all>search for Tables>Click on New
3. Enter the Details:

Label: Daily Expenses

Name: Auto-Populated

Add Module to menu : Family Expenditure

The screenshot shows the 'New Table' form in ServiceNow. On the left, the 'Label' field is set to 'Daily Expenses' (marked with a red box and a red arrow labeled '1') and the 'Name' field is set to 'u_daily_expenses' (marked with a red box and a red arrow labeled '2'). The 'Extends table' field is empty. On the right, the 'Application' is set to 'Global'. The 'Create module' and 'Create mobile module' checkboxes are both checked. Below these, there is a section for 'Application Menu' with a button labeled 'Add module to menu'. This button is highlighted with a red box and a red arrow labeled '3', and it is set to 'Family Expenditure'.

4. click on save

Activity 3:

1.Open Service now

2.Near Columns Double Click near insert a new row

Give the details as:

Column label:Number

Type:String

3.Double click on insert a new row again

Give the details as:

Column label:Date

Type:Date

4. Double click on insert a new row again

Column label:Amount

Type:Integer

5 .Double click on insert a new row again

Column label:Expense Details

Type:String

Max length:800

6.Double click on insert a new row again

Give the details as:

Column label:Comments

Type:String

Max length:800

6.Click on save

Activity 3:

1.Open Service now

2.Double click on the Number field/Column

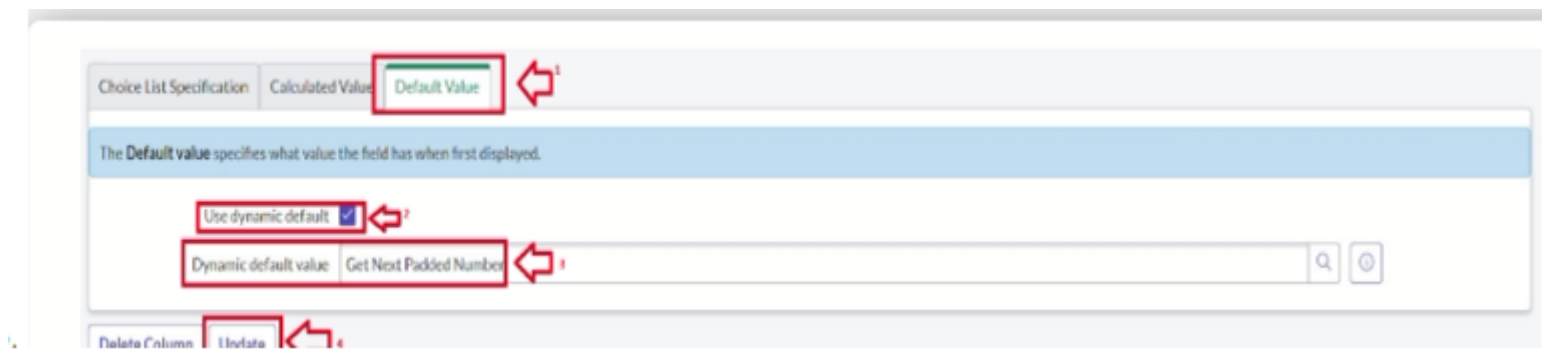
3.Go down and double click on Advanced view

4.In Default Value:

Use dynamic default:check the box

Dynamic default value:Get Next Padded Number

5.click on update



6.Go to all>>search for Number Maintenance>>select number maintenance

7.click on new

8.Enter the details:

Table:Daily Expenses

Prefix:MFE

The screenshot shows a 'Number New record' form. At the top, there is a header bar with a back arrow, a search icon, a help icon, a menu icon, and a 'Submit' button. Below the header, the form contains several fields: a 'Table' field with the value 'Daily Expenses' (annotated with a red box and arrow 1), a 'Prefix' field with the value 'MFE' (annotated with a red box and arrow 2), a 'Number' field with the value '1,000', an 'Application' field with the value 'Global', and a 'Number of digits' field with the value '7'. At the bottom left, there is a 'Submit' button (annotated with a red box and arrow 3) and a red arrow pointing left.

9.click on submit

Activity 3:

1.Open Service now

2.Go to all>>search for Daily Expenses>>Open Daily Expenses

3.Click on new

4.Go to header and click there>>click on configure>>Select Form Design

5.Drag Drop the form as requirement

The image shows a form editor interface with two sections. The top section is titled "Daily Expenses [u_daily_expenses]" and has a "2 Column" layout. It contains four fields: "Number", "Date", "Family Member Name", and "Expense". Each field has a settings icon (gear) and a delete icon (X). The bottom section is titled with a single column layout and contains a "Comments" field, also with settings and delete icons.

6.Make Number Read-Only Field by Clicking

7.Make Date,Family Member Name Mandatory field by Clicking

8.Click on save

Relationship

Activity 4:

1.Open Service now

2.Go to all>>search for relationships>>open relationships

3.click on new

4.Enter the details:

Name:Daily Expenses

Applies to table:select Family Expenses

Daily Expenses:select Daily Expenses

5.click save

Configuring Related List

Activity 4:

1.open Service now

2. .Go to all>>search for Family Expenses>>Open FamilyExpenses

3.Click on new

4.Go to header and click there>>click on configure>>select Related Lists

5.Add Daily Expenses to selected area

6.click on save



Business rule

Activity 5:

1. Open Service Now
2. Go to all>>search for Business Rules
3. Under system definition select business rules then click new
4. Enter the Details:

Name: Family Expenses BR

Table: Select Daily Expenses

Check Advanced

The screenshot shows the 'Business Rule' configuration form in Service Now. The form has a header 'Business Rule' and a sub-header 'New record'. Below the header is a blue bar with a description: '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'. The form contains several fields: 'Name' with the value 'Family Expenses BR', 'Table' with the value 'Daily Expenses [u.daily_expenses]', 'Application' with the value 'Global', 'Active' with a checked checkbox, and 'Advanced' with a checked checkbox. Red boxes and arrows highlight the 'Name', 'Table', and 'Advanced' fields, with numbers 1, 2, and 1 respectively indicating the order of steps.

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 Family Expenses BR

Table Daily Expenses [u.daily_expenses]

Application Global

Active ☒

Advanced ☒

5. In when the run check insert and update

The screenshot displays a configuration window for a business rule. At the top, there are three tabs: 'When to run' (highlighted with a red box and a red arrow pointing to it), 'General', and 'Advanced'. Below the tabs, a blue instruction bar reads: 'Specify whether the business rule should run on Insert or Update. Use Filter Conditions to specify under which conditions.' The 'When' dropdown is set to 'before'. The 'Order' field contains the value '100'. To the right, a list of actions with checkboxes is shown: 'Insert' (checked), 'Update' (checked), 'Delete' (unchecked), and 'Query' (unchecked). A red arrow points to the 'Insert' and 'Update' checkboxes. Below this, the 'Filter Conditions' section includes two buttons: 'Add Filter Condition' and 'Add "OR" Clause'. Underneath these buttons are three input fields: '-- choose field --', '-- oper --', and '-- value --'. At the bottom, the 'Role conditions' section has a small icon.

When to run | General | Advanced

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

When: before

Order: 100

Insert ☒

Update ☒

Delete ☐

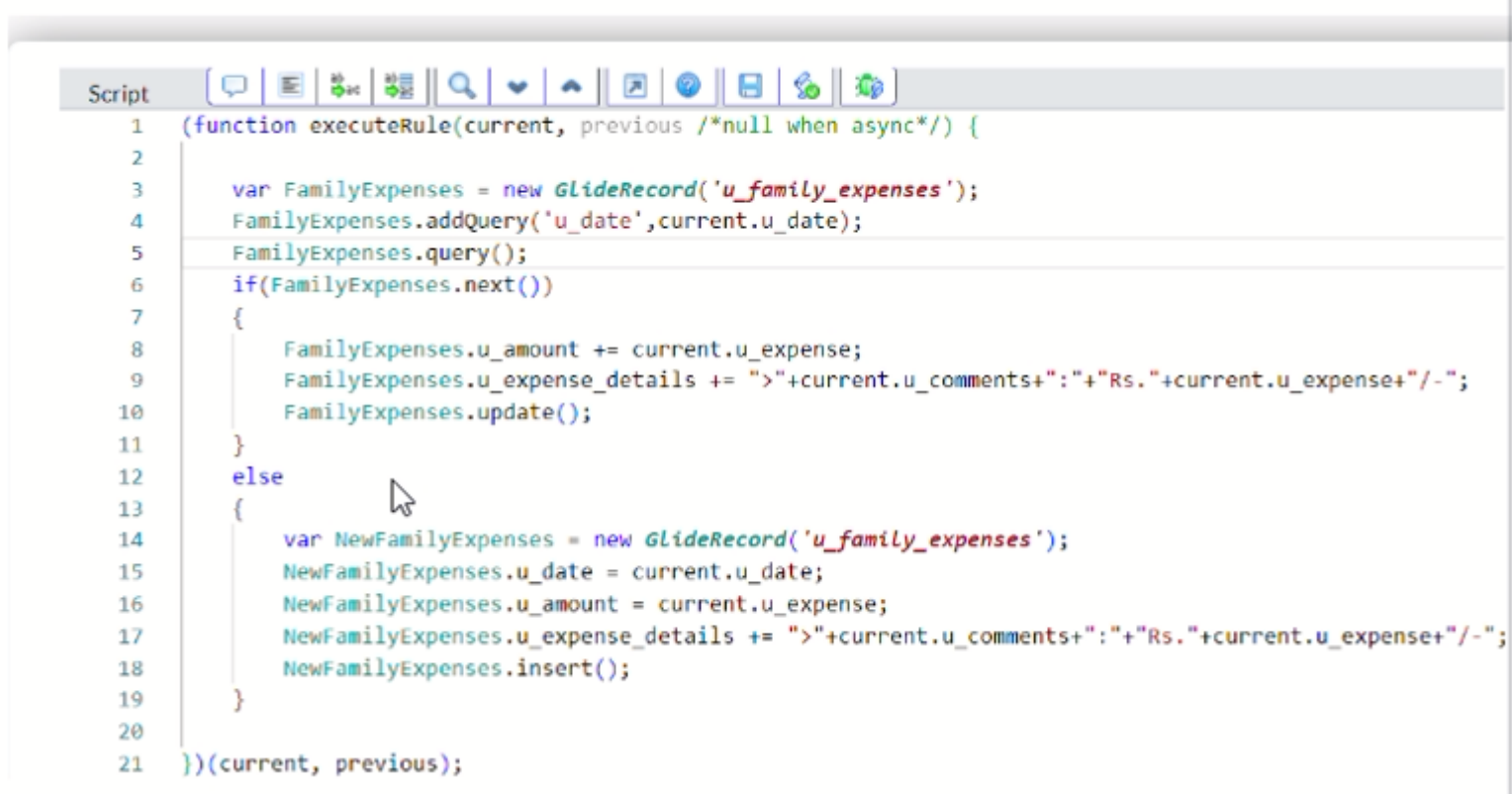
Query ☐

Filter Conditions: Add Filter Condition Add "OR" Clause

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

Role conditions

6.In Advance(we write the code)



```
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);
```

7.click on save

Configure The Relationship

Activity 6:

1.Open Service now

2.Go to all>>search for relationships>>open relationships

3.In open Daily Expenses relationship

For Applies to table:select Family Expenses

In Query with:write the below Query

4.click on update

The screenshot shows the 'Relationship Daily Expenses' configuration page in ServiceNow. The page has a header with a back arrow, a title 'Relationship Daily Expenses', and 'Update' and 'Delete' buttons. Below the header, there are fields for 'Name' (Daily Expenses), 'Application' (Global), and 'Advanced' (unchecked). The 'Applies to table' field is set to 'Family Expenses [u_family_expenses]' and is highlighted with a red box and a red arrow labeled '1'. The 'Queries from table' field is set to 'Daily Expenses [u_daily_expenses]'. Below these fields is a blue information bar that reads: '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](#).' Below the information bar is a 'Query with' section with a toolbar and a text area containing a JavaScript function:

```
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);
```

 The text area is highlighted with a red box and a red arrow labeled '2'. At the bottom of the page, there are 'Update' and 'Delete' buttons, with the 'Update' button highlighted by a red arrow labeled '3'.