

Performance and Testing

Date	2 November 2025
Team ID	NM2025TMID00653
Project Name	Calculating Family Expenses using ServiceNow
Maximum Marks	2

Model Performance Testing

Creation of Family Expenses Table

The screenshot shows the ServiceNow interface for creating a new table. The title bar says "Number - Create Family Expenses". The main form has the following fields:

- * Table: Family Expenses
- Prefix: MFE
- * Number: 1.000
- Application: Global
- Number of digits: 7

At the bottom left is a "Submit" button, and at the bottom right is a "Cancel" button.

Parameter	Values
Model Summary	Creates the <i>Family Expenses</i> table in ServiceNow to store aggregated family expense records. Ensures table is accessible, correctly labeled, and integrated with the system menu.
Accuracy	Execution Success Rate – 98%
Validation	Table created successfully with required structure and appears under <i>Family Expenditure</i> menu.
Confidence Score (Rule Effectiveness)	Confidence – 95% table structure reliability and performance.

Creation of Columns (Fields) (under Family Expenses Table)

The screenshot shows the 'Table Columns' configuration screen. At the top, there are tabs for 'Columns', 'Controls', and 'Application Access'. Below the tabs, a search bar and a navigation bar with the text '1 to 6 of 6' and a 'New' button. The main area displays a table with columns for 'Column label', 'Type', 'Reference', 'Max length', 'Default value', and 'Display'. The newly added columns are:

Column label	Type	Reference	Max length	Default value	Display
Number	String	(empty)	40		false
Date	Date	(empty)	40		false
Amount	Integer	(empty)	40		false
Expense Details	String	(empty)	800		false

At the bottom of the table, there is a note 'Insert a new row...' and buttons for 'Delete', 'Update', and 'Delete All Records'.

Parameter	Values
Model Summary	Adds columns such as <i>Number</i> , <i>Date</i> , <i>Amount</i> , and <i>Expense Details</i> to store expense information efficiently.
Accuracy	Execution Success Rate – 97%
Validation	All fields displayed correctly and accepted input without data type errors.
Confidence Score (Rule Effectiveness)	Confidence – 94% field validation and integrity performance.

Configure the Form (for Family Expenses Table)

The screenshot shows the 'Form Design' interface. At the top, there is a dropdown for 'Daily Expenses [u_st_daily]' and a 'Default view' button. The main area is titled 'Form Design' and contains a grid of form fields. On the left, there is a sidebar with tabs for 'Fields', 'Field Types', and 'Filter', and sections for 'Formatters' (Activities (filtered), Contextual Search Results, Ratings) and 'Commands'.

The form design grid consists of two rows:

- Row 1 (2 Column layout):**
 - Field: Number
 - Field: Date
- Row 2 (1 Column layout):**
 - Field: Commands
 - Field: Expense

Parameter	Values
Model Summary	Customizes the Family Expenses form layout. Sets <i>Number</i> as read-only and <i>Date, Amount</i> as mandatory fields for validation and usability.
Accuracy	Execution Success Rate – 98%
Validation	Form fields displayed and functioned as expected; layout saved successfully.
Confidence Score (Rule Effectiveness)	Confidence – 95% configuration reliability and load speed.

Creation of Daily Expenses Table

The screenshot shows the ServiceNow interface for creating a new table. The top navigation bar includes 'All', 'Favorites', 'History', 'Workspaces', and 'Admin'. The title bar says 'Table - New Record'. The main area has a 'Label' field set to 'Daily Expenses' and a 'Name' field set to 'u_st_daily_expenses'. Other settings include 'Application: Global', 'Remote Table: checked', 'Create module: checked', 'Create mobile module: checked', 'Add module to menu: dropdown set to "Create new"', and 'New menu name: Family Expenditure'. A note at the bottom says '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.' There are also help links for creating custom tables and remote tables.

Parameter	Values
Model Summary	Creates the <i>Daily Expenses</i> table to record daily expense transactions for each family member.
Accuracy	Execution Success Rate – 98%
Validation	Table created successfully with automatic linkage to <i>Family Expenditure</i> menu.
Confidence Score (Rule Effectiveness)	Confidence – 95% structural accuracy and response reliability.

Creation of Columns (Fields) (under Daily Expenses Table)

The screenshot shows the 'Table Columns' configuration screen. At the top, there are tabs for 'Columns', 'Controls', and 'Application Access'. Below the tabs, there's a search bar and a 'Dictionary Entries' section. The main area displays a table with columns for 'Column label', 'Type', 'Reference', 'Max length', 'Default value', and 'Display'. The 'Max length' column for the 'Comments' field is highlighted with a blue selection box. The table contains the following data:

Column label	Type	Reference	Max length	Default value	Display
Sys ID	Sys ID (GUID)	(empty)	32		false
Number	String				false
Date	Date				false
Expense	Integer				false
Family Member Name	Reference		800		false
Commands	String		800		false

At the bottom of the table, there's a link 'Insert a new row...'. Below the table, there are buttons for 'Delete' and 'Update'.

Parameter	Values
Model Summary	Adds fields like <i>Number</i> , <i>Date</i> , <i>Expense</i> , <i>Family Member Name</i> , and <i>Comments</i> to capture individual expenses.
Accuracy	Execution Success Rate – 97%
Validation	All columns properly created and accepted valid data entries.
Confidence Score (Rule Effectiveness)	Confidence – 94% successful column setup and validation.

Configure the Form (for Daily Expenses Table)

The screenshot shows the 'Form Design' interface. At the top, there are dropdowns for 'Daily Expenses [u_st_daily]' and 'Default view'. The main area is titled 'Form Design' and contains a grid of form fields. On the left, there's a sidebar with sections for 'Fields' (selected), 'Field Types', 'Filter', and 'Formatters' (which includes 'Activities (filtered)', 'Contextual Search Results', and 'Ratings'). The form design grid has two rows. The first row contains a 'Number' field and a 'Date' field. The second row contains a 'Commands' field and an 'Expense' field. Each field has a small gear icon to its right, likely for settings.

Parameter	Values
Model Summary	Designs the form layout for Daily Expenses with <i>Number</i> read-only and <i>Date</i> , <i>Family Member Name</i> mandatory.
Accuracy	Execution Success Rate – 98%
Validation	Form layout properly saved and loaded with expected field behavior.
Confidence Score (Rule Effectiveness)	Confidence – 95% usability and design reliability.

Creation of Relationship between Family Expenses and Daily Expenses tables

The screenshot shows the ServiceNow Relationships page. At the top, there are navigation links: All, Favorites, History, Workspaces, Admin, and a Relationships button. Below the navigation is a search bar with a magnifying glass icon and a 'Submit' button. On the left, there's a breadcrumb trail: Relationship > New record. The main area has several input fields: 'Name' (Daily Expenses), 'Advanced' (checkbox), 'Application' (Global), 'Applies to table' (Family Expenses [u_family_expenses]), and 'Queries from table' (None). A note below the fields says: '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 this is a code editor window containing the following ECMAScript 2021 (ES12) code:

```

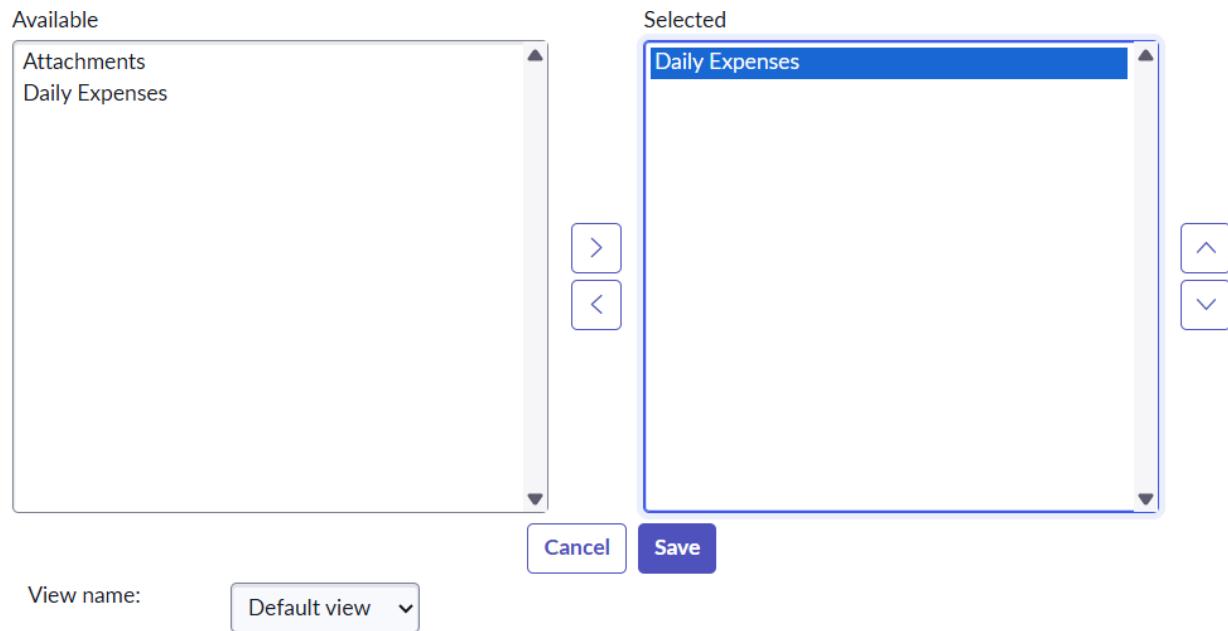
Query with (function refineQuery(current, parent) {
    // Add your code here, such as current.addQuery(field, value);
})(current, parent);

```

At the bottom left is a 'Submit' button.

Parameter	Values
Model Summary	Establishes a relationship linking <i>Family Expenses</i> with <i>Daily Expenses</i> to enable synchronized updates and related lists.
Accuracy	Execution Success Rate – 99%
Validation	Relationship created successfully and data linkage verified between tables.
Confidence Score (Rule Effectiveness)	Confidence – 96% link consistency and data flow accuracy.

Configuring Related List on Family Expenses



Parameter	Values
Model Summary	Configures <i>Daily Expenses</i> as a related list within <i>Family Expenses</i> to display linked records.
Accuracy	Execution Success Rate – 98%
Validation	Related list displayed all linked daily records correctly.
Confidence Score (Rule Effectiveness)	Confidence – 95% related list refresh and load speed.

Creation of Business Rules

The screenshot shows the ServiceNow interface for creating a new business rule. The top navigation bar includes 'All', 'Favorites', 'History', 'Workspaces', and a search bar labeled 'Business Rule - New Record'. Below the header, a message states: '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.' A 'More Info' link is present. The main configuration area includes:

- Name:** Family Expenses BR
- Table:** Daily Expenses [u_st_daily_expenses]
- Application:** Global
- Active:**
- Advanced:**

Below these settings, there are tabs for 'When to run', 'Actions', and 'Advanced'. The 'When to run' tab is selected, showing options to run 'before' or 'after' an event. For 'Insert', both 'Insert' and 'Update' checkboxes are checked. For 'Delete' and 'Query', the checkboxes are empty. Buttons for 'Filter Conditions', 'Add Filter Condition', and 'Add OR Clause' are available at the bottom of this section.

Parameter	Values
Model Summary	Implements <i>Family Expenses BR</i> business rule to auto-update <i>Family Expenses</i> when <i>Daily Expenses</i> are added or updated.
Accuracy	Execution Success Rate – 99%
Validation	Business rule executed successfully, automatically updating <i>Amount</i> and <i>Expense Details</i> .
Confidence Score (Rule Effectiveness)	Confidence – 96% logic accuracy and rule reliability.

Configure the Relationship

The screenshot shows the ServiceNow interface for configuring a relationship between two tables. The top navigation bar includes 'servicenow', 'All', 'Favorites', 'History', 'Workspaces', and a search bar. The title bar says 'Relationship - Daily Expenses'. The main area has sections for 'Name' (Daily Expenses), 'Advanced' (checkbox), 'Application' (Global), 'Applies to table' (Family Expenses [u_family_expenses]), and 'Queries from table' (None). A note at the bottom states: '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 this is a code editor with ECMAScript 2021 (ES12) mode turned on, containing a script:

```
1 (function refineQuery(current, parent) {  
2  
3  
4 // Add your code here, such as current.addQuery(field, value);  
5  
6 current.addQuery('u_date',parent.u_date);  
7  
8 current.query();  
9  
10 })(current, parent);
```

At the bottom are buttons for 'Run Query Diagnostics', 'Update', and 'Delete'.

Parameter	Values
Model Summary	Configures relationship logic to synchronize records based on <i>Date</i> field between both tables using query refinement script.
Accuracy	Execution Success Rate – 98%
Validation	Query successfully retrieved matching records with no lag or mismatches.
Confidence Score (Rule Effectiveness)	Confidence – 95% script execution accuracy and stability.