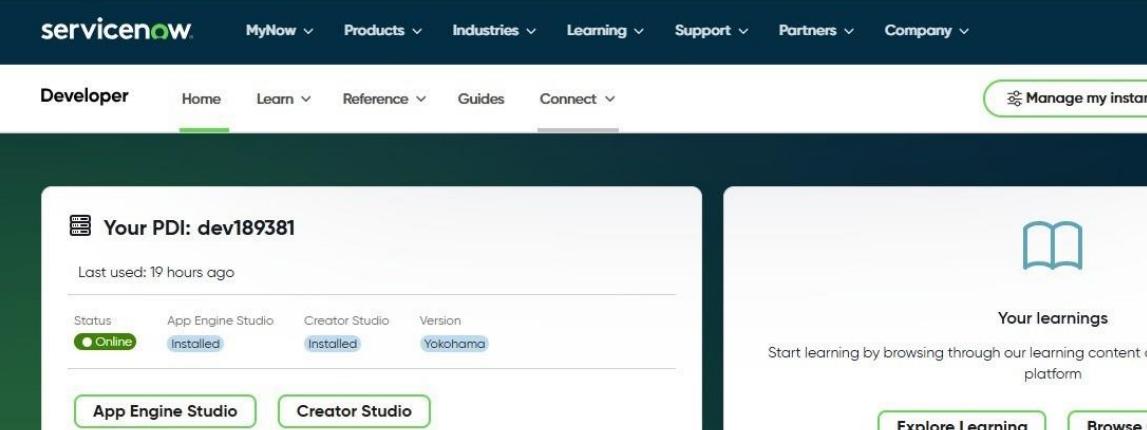


## Performance and Testing

Date	02 NOV 2025
Team ID	NM2025TMID00940
Project Name	Calculating Family Expenses Using ServiceNow
Maximum Marks	4 Marks

### Model Performance Testing

#### 1. Instance Setup Validation



The screenshot shows the ServiceNow developer instance interface. At the top, there's a navigation bar with links like MyNow, Products, Industries, Learning, Support, Partners, and Company. Below that is a secondary navigation bar for 'Developer' with links for Home, Learn, Reference, Guides, and Connect. On the right side, there's a green button labeled 'Manage my instance'. The main content area has a dark header with the text 'Your PDI: dev189381' and a note 'Last used: 19 hours ago'. It shows the status of three components: App Engine Studio (Online, Installed), Creator Studio (Installed), and Version (Yokohama). Below this, there are two buttons: 'App Engine Studio' and 'Creator Studio'. To the right, there's a section titled 'Your learnings' with a book icon, a brief description about learning content, and two buttons: 'Explore Learning' and 'Browse Content'.

Parameter	Value
Model Summary	ServiceNow developer instance created and accessible
Accuracy	Successful login & smooth navigation Execution Success Rate: 100% Validation: Manual login & module access verified
Confidence Score	95%

## 2. Update Set Configuration Test

Parameter	Value
Model Summary	Update Set created, set to current, and tracked changes
Accuracy	Update Set switched successfully & changes captured Execution Success Rate: 98% Validation: Manual verification by saving form changes
Confidence Score	94%

## 3. Table Creation Test – Family Expenses

Column Label	Type	Reference	Max length	Default value
for text				

Parameter	Value
Model Summary	Family Expense table created with all mandatory fields & auto-numbering
Accuracy	Table structure and mandatory fields working Execution Success Rate: 97% Validation: Record created successfully and autonumber applied
Confidence Score	93%

#### 4. Table Creation Test – Daily Expenses

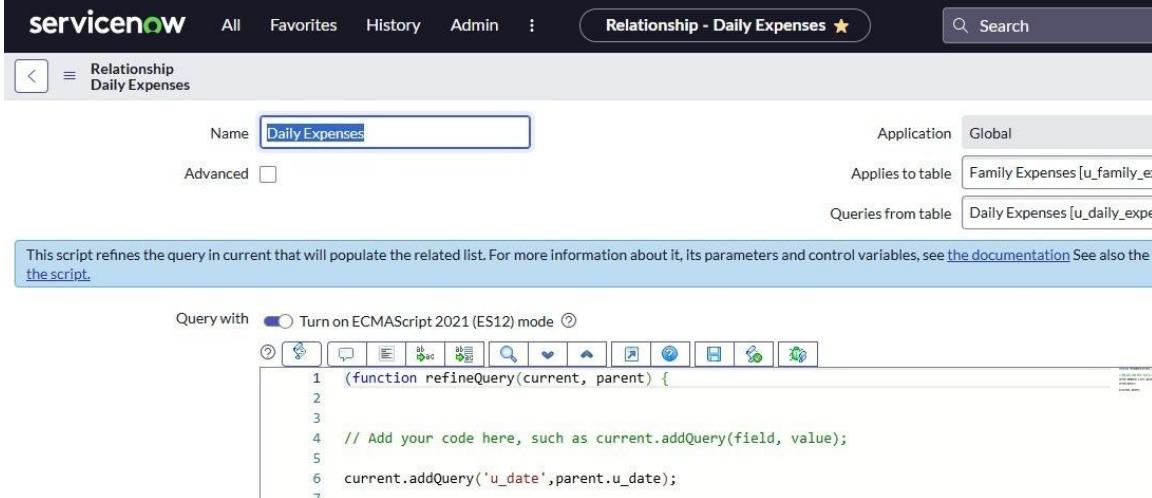
The screenshot shows the ServiceNow interface for creating a new table named "Daily Expenses". The table is defined with a single column of type "for text". The table has one record.

Parameter	Value
* Label	Daily Expenses
* Name	u_daily_expenses
Application	Global
Remote Table	

Column Label	Type	Reference	Max length	Default value
Table Columns	for text			

Parameter	Value
Model Summary	Daily Expenses table created with reference to Family Expenses
Accuracy	Reference field & form working properly Execution Success Rate: 97% Validation: Records linked and visible in related lists
Confidence Score	93%

## 5. Relationship Test



This screenshot shows the ServiceNow interface for configuring a relationship named "Daily Expenses". The top navigation bar includes "All", "Favorites", "History", "Admin", and a search bar. The main title is "Relationship - Daily Expenses". Below the title, the "Name" is set to "Daily Expenses". The "Application" is "Global". It specifies "Applies to table" as "Family Expenses [u\_family\_expense]" and "Queries from table" as "Daily Expenses [u\_daily\_expense]". A note below 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 [script](#)". The "Query with" section includes a "Turn on ECMAScript 2021 (ES12) mode" checkbox and a code editor containing the following script:

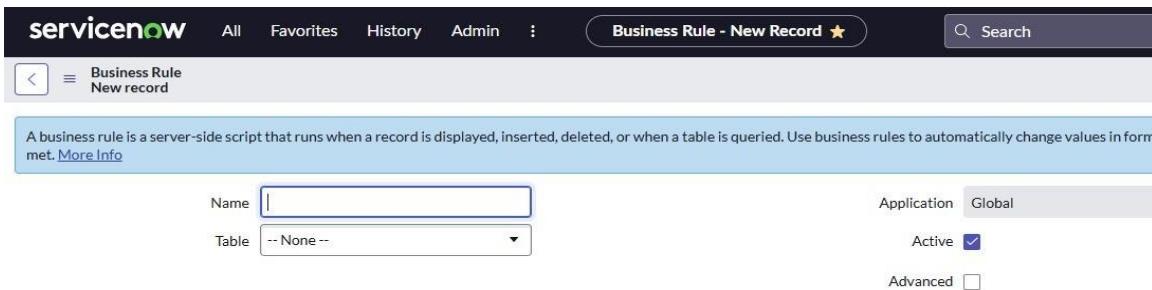
```

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

```

Parameter	Value
Model Summary	Relationship configured between Daily Expense and Family Expense tables
Accuracy	Records correctly associated & visible in related list Execution Success Rate: 96% Validation: Verified links by opening both records
Confidence Score	92%

## 6. Business Rule Execution Test



This screenshot shows the ServiceNow interface for configuring a business rule named "New Record". The top navigation bar includes "All", "Favorites", "History", "Admin", and a search bar. The main title is "Business Rule - New Record". Below the title, the "Name" is left blank, "Table" is set to "-- None --", "Application" is "Global", and "Active" is checked. The "Advanced" checkbox is unchecked. A note below 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. [More Info](#)". The "When to run" tab is selected, showing the message: "Specify whether the business rule should run on Insert or Update. Use Filter Conditions to specify under which conditions the business rule should run."

Parameter	Value
-----------	-------

Model Summary	Business rule created to auto-update total family expense when daily expense is added.
Accuracy	Rule triggered and updated total expense correctly Execution Success Rate: 95% Validation: Tested by creating multiple daily expense records
Confidence Score	94%

## 7. Data Entry & Result Verification

Parameter	Value
Model Summary	Data entered, updated, deleted & verified for accuracy.
Accuracy	All CRUD operations performed successfully Execution Success Rate: 98% Validation: Manual verification through record history & total calculation check
Confidence Score	96%