RC100R.MBR

Path: NXCLOUD/rpgsrc/RC100R.MBR **Generated**: 2026-01-08 12:58:47 **Processing Time**: 12772ms

Business Logic for RC101R

This document outlines the business rules that govern the maintenance of fixed data for debt collection (inkasso) within the program RC101R. The primary focus is on the logic surrounding the management of debt collection records, including creation, updating, deletion, and display of records.

The core logic for managing debt collection records is contained within various subroutines in RC101R. The program utilizes subfiles to handle multiple records and provides functionality for user interactions through function keys.

Order Status and Header Rules

RC101R: RCA0PF

1. Record Creation

- **Logic:** A new record can be created if the input fields are valid and do not already exist in the database.
- **File:** rca0lr (Debt Collection Records)
- **Field:** c10fir
- **Condition:** The process will not create a record if c10fir is already found in the rca0lr file.

2. Record Update

- **Logic:** Existing records can be updated with new values from the input fields.
- **File:** rca0lur (Update Debt Collection Records)
- **Field:** rc0txt
- **Condition:** The update will occur only if the record with the specified key exists in the rca0lur file.

Configuration and Authorization Rules

1. User Authorization Check

- **Logic:** The program verifies if the user has permission to perform actions on debt collection records.
- **Files:**
- rca0l1 (Debt Collection Records)
- rca0lr (Read Debt Collection Records)
- **Fields:**
- l_firm (User Firm)
- rc0fir (Record Firm)
- **Condition:** The user must be associated with the firm specified in the records to perform any updates or deletions.

2. Input Validation

- **Logic:** The program checks if the input fields are filled correctly before processing.
- **File:** rca0lr (Debt Collection Records)

•**Field:** c20txt

•**Condition:** The process will not proceed if c20txt is blank.

Financial and Transactional Rules

1. Record Deletion

•**Logic:** Records can be deleted from the database upon user confirmation.

•**File:** rca0lur (Delete Debt Collection Records)

•**Fields:**

•d10fir (Record ID for deletion)

•rca0lu_0fir (Key for the record to delete)

•**Condition:** The record will be deleted only if it exists in the rca0lur file.

2. Record Copying

•**Logic:** Users can create a copy of an existing record.

•**File:** rca0lur (Copy Debt Collection Records)

•**Condition:** The copying process will only proceed if the original record exists.

Special Conditions (Program-Specific)

1. Inkasso Agency Validation (RC101R)

•**Logic:** The program checks if the specified inkasso agency is valid before proceeding with updates.

•**File:** rca0lr (Debt Collection Records)

•**Field:** c2inkf

•**Condition:** If c2inkf is not one of the recognized agencies, an error message is displayed.

2. Subfile Management (RC101R)

•**Logic:** The program manages the display and navigation of records in a subfile format.

•**File:** b1sfl (Subfile for Debt Collection Records)

•**Fields:** w_srrn, w_sfrn (Subfile pointers)

•**Condition:** The subfile is populated based on the current firm and user input.

Subprogram Calls Affecting Logic

Beyond direct file checks, several external subprograms are called that play a significant role in the workflow.

1. AA005R (Error Message Display)

•**Trigger:** Called when an invalid inkasso agency is entered.

•**Logic:** Displays an error message to the user.

•**Impact:** This call ensures that users are informed of invalid input, preventing further processing.

2. xc2bld (Record Maintenance Subroutine)

•**Trigger:** Called during the creation and updating of records.

•**Logic:** Handles the logic for maintaining records, including validation and database updates.

•**Impact:** This subroutine is crucial for ensuring data integrity and proper record management.

3. xc1msg (Duplicate Record Notification)

•**Trigger:** Called when attempting to create a record that already exists.

- **Logic:** Displays a message indicating that the record already exists.
- **Impact:** This ensures that users are aware of duplicates and can take appropriate action.