

RP001R.MBR

Path: NXCLOUD/rpgsrc/RP001R.MBR **Generated:** 2026-01-09 10:13:44 **Processing Time:** 10353ms

Business Logic for Koder, vedlikehold

This document outlines the business rules that govern the maintenance of codes within the ERP system, based on an analysis of the RPG program RP001R. The primary focus is on the handling of project categories and their associated descriptions.

The core logic for maintaining codes is contained within the *inzsr subroutine in RP001R. The program processes records related to project categories, allowing users to create, update, delete, and view these records through a subfile interface.

Order Status and Header Rules

Koder, vedlikehold: rpkol1, rpkol2, rpkolr, rpkolu

1. Record Creation

- Logic:** A new record can be created if the input fields are not blank.
- File:** rpkolr (Project category records)
- Field:** rpkolr_kkod
- Condition:** The process will not create a record if c1kode is blank.

2. Record Update

- Logic:** If a record exists, it can be updated with new values.
- File:** rpkolur (Updated project category records)
- Field:** rpktxt
- Condition:** The update will occur if the record is found in the database.

Configuration and Authorization Rules

1. User Authorization Check

- Logic:** The program checks the user's firm and type to ensure they have the necessary permissions.
- Files:**
 - rpkol1 (Project category key records)
 - rpkol2 (Project category description records)
- Fields:**
 - w_firm (User's firm)
 - w_type (User's type)
- Condition:** The process will not proceed if the user's firm and type do not match the records.

2. Input Validation

- Logic:** The program validates that required fields are filled before processing.
- File:** rpkolr (Project category records)
- Field:** c2text
- Condition:** The process will not proceed if c2text is blank.

Financial and Transactional Rules

1. Record Deletion

- Logic:** A record can be deleted if the user confirms the action.
- File:** rpkolur (Records to be deleted)
- Fields:**
 - d1kode (Code to delete)
 - rpkolu_kkod (Key for the record being deleted)
- Condition:** The deletion will occur if the user confirms the delete action.

2. Record Viewing

- Logic:** Users can view details of a selected record.
 - File:** rpkolr (Project category records)
 - Condition:** The view will be triggered if the user selects a record from the subfile.
-

Special Conditions (Program-Specific)

1. Subfile Handling (RP001R)

- Logic:** The program manages a subfile to display multiple records and handle user interactions.
- File:** b1sfl (Subfile for project categories)
- Field:** b1valg
- Condition:** The subfile will be populated based on user selections and actions.

2. Error Handling (RP001R)

- Logic:** The program provides feedback for errors encountered during operations.
 - File:** dspfbk (Display feedback structure)
 - Fields:** d_fcrn (Feedback code)
 - Condition:** The feedback will be displayed if any errors occur during processing.
-

Subprogram Calls Affecting Logic

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

1. dsp_subfile (Display Subfile)

- Trigger:** Called to display the subfile after processing records.
- Logic:** Manages the display of the subfile and handles user interactions.
- Impact:** This call acts as a **major logical gateway** for user interaction with the records.

2. clr_subfile (Clear Subfile)

- Trigger:** Called to clear the subfile before populating it with new data.
- Logic:** Resets the subfile state and prepares it for new entries.
- Impact:** This is a **destructive filtering step** that ensures the subfile only contains relevant data.

3. crt_subfile (Create Subfile)

- Trigger:** Called to populate the subfile with records from the database.
- Logic:** Reads records and fills the subfile for display.
- Impact:** Represents the handoff to the next major business function, allowing users to interact with the data.