LA101R.MBR

Path: NXCLOUD/rpgsrc/LA101R.MBR **Generated**: 2026-01-08 13:32:28 **Processing Time**: 12989ms

Business Logic for LA101R

This document outlines the business rules that govern the maintenance of status codes within the system, based on an analysis of the RPG program LA101R. The primary focus is on how status codes are retrieved, displayed, and updated in the inventory management context.
The core logic for maintaining status codes is contained within the A1BLD subroutine in LA101R. The program interacts with two files, lstsl1 and lstslu, to read and update status code information based on user input.

Order Status and Header Rules

LA101R: lstsl1, lstslu

1. Retrieve Status Code Information

•**Logic:** The program retrieves information from the lstsl1 file based on a key. If the record is found, it populates various fields with the corresponding values from the file.
•**File:** lstsl1 (Status code master file)
•**Field:** lstsl1_key
•**Condition:** The process will not select a record if the key does not match an existing entry in lstsl1.

2. Display Status Code Information

•**Logic:** After retrieving the status code information, the program displays the data to the user through a formatted screen.
•**File:** a1bld (Display format for status code)
•**Field:** Various fields populated from lstsl1
•**Condition:** The display is executed only if the status code record is found.

Configuration and Authorization Rules

1. Update Status Code Information

•**Logic:** The program updates the lstslu file with the new values entered by the user. If the record exists, it updates the fields; if not, it writes a new record.
•**Files:**
•lstslu (Status code update file)
•**Fields:**
•laabrk (Status code)
•laabin (Inventory code)
•**Condition:** The update occurs if the record is found; otherwise, a new record is created.

2. Set User Information on Update

•**Logic:** When updating or writing a new status code, the program captures the user information and timestamps for auditing purposes.
•**File:** lstslu (Status code update file)

•**Field:** laaeus (User who made the update)
•**Condition:** This logic is executed during the update process.

Financial and Transactional Rules

1. Timestamp on Update

•**Logic:** The program records the timestamp of the last update to the status code for tracking changes.
•**File:** lstslu (Status code update file)
•**Fields:**
•laaeda (Last edit date)
•laaeti (Last edit time)
•**Condition:** The timestamp is updated only if the record is found and updated.

2. Default Values on Not Found

•**Logic:** If no record is found in lstsl1, the program initializes all fields to blank or zero before displaying the screen.
•**File:** lstsl1 (Status code master file)
•**Condition:** This logic is executed when the initial chain operation does not find a matching record.

Special Conditions (Program-Specific)

1. End Program Logic

•**Logic:** The program checks for the F3 key (exit) and sets the program's end indicator before returning control to the caller.
•**File:** N/A
•**Field:** N/A
•**Condition:** This logic is executed when the user presses the F3 key.

2. Initialization Subroutine

•**Logic:** The program initializes keys for reading and updating the status code, including setting the firm identifier.
•**File:** N/A
•**Fields:** w_firm (Firm identifier)
•**Condition:** This logic is executed at the beginning of the program.

Subprogram Calls Affecting Logic

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

1. A1BLD (Display Subroutine)

•**Trigger:** Called after retrieving status code information.
•**Logic:** Displays the status code information to the user.
•**Impact:** This call acts as a **major logical gateway** for user interaction with the status codes.

2. lstsl1 (Status Code Master File)

•**Trigger:** Accessed for reading status code information.

•**Logic:** Provides the necessary details for the status codes.
•**Impact:** This is a **critical data source** for maintaining status codes.

3. lstslu (Status Code Update File)

•**Trigger:** Accessed for updating or writing new status codes.
•**Logic:** Handles the persistence of changes made by the user.
•**Impact:** Represents the **handoff to the next major business function** of maintaining accurate status codes in the system.