

RY820R.MBR

Path: NXCLOUD/rpgsrc/RY820R.MBR **Generated:** 2026-01-08 15:04:27 **Processing Time:** 11673ms

Business Logic for RY820R

This document outlines the business rules that govern the process of updating the match register, based on an analysis of the RPG program RY820R. The primary focus is on how items are updated with information from the item register.

The core logic for updating the match register is contained within the main program logic of RY820R. The program reads through the match register and updates items with information from the item register based on specific conditions.

Match Register and Item Update Rules

RY820R: fjmatl1, fjvarl1, fjmatlu

1. Check for Blank Item Code

• **Logic:** The program skips the current item if the item code (jovar2) is blank.

• **File:** fjmatl1 (Match Register)

• **Field:** jovar2

• **Condition:** The process will not select a record if jovar2 is equal to *blank.

2. Update Item Information

• **Logic:** If the item exists in the item register and the old item code (joldo2) is different from the new item code (jvldor), the program updates the match register with the new item information.

• **File:** fjvarl1 (Item Register)

• **Field:** jvarl1_vare

• **Condition:** The item is only updated if it exists in the item register and the old code is not equal to the new code.

Key Initialization and Data Retrieval Rules

1. Initialize Keys for Match Register

• **Logic:** The program initializes keys for reading from the match register, item register, and for updating the match register.

• **Files:**

• fjmatl1 (Match Register)

• fjvarl1 (Item Register)

• **Fields:**

• w_firm (from LDA)

• jvarl1_vare (from fjvarl1)

• **Condition:** Keys are set up to ensure proper access to the relevant records.

2. Retrieve Firm Information

• **Logic:** The program retrieves the firm information from the local data area (LDA) to use in processing.

• **File:** N/A (Local Data Area)

•**Field:** l_firm

•**Condition:** This is a necessary step before processing the records.

Special Conditions (Program-Specific)

1. End Program Logic

•**Logic:** The program sets the last record indicator (*inlr) to on and returns control to the calling program.

•**File:** N/A

•**Condition:** This indicates the end of processing for the current program.

2. Subroutine for Program Initialization

•**Logic:** The program initializes keys for reading and updating records and retrieves the firm information from the local data area.

•**File:** N/A

•**Fields:** w_firm (Firm information)

•**Condition:** This is executed at the beginning of the program to prepare for processing.

Subprogram Calls Affecting Logic

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

1. N/A

•**Trigger:** N/A (No external subprogram calls identified in the provided code)

•**Logic:** N/A

•**Impact:** N/A

2. N/A

•**Trigger:** N/A

•**Logic:** N/A

•**Impact:** N/A

3. N/A

•**Trigger:** N/A

•**Logic:** N/A

•**Impact:** N/A

This document provides a comprehensive overview of the business logic contained within the RY820R program, focusing on the updating of the match register with item information.