JX100R.MBR

Path: NXCLOUD/rpgsrc/JX100R.MBR **Generated**: 2026-01-08 15:03:59 **Processing Time**: 15421ms

Business Logic for Price Maintenance Process

This document outlines the business rules that govern the price maintenance process for items in the ERP system, based on an analysis of the RPG program JX100R. The primary focus is on how item prices are created, updated, and validated within the system.

The core logic for price maintenance is contained within the various subroutines in JX100R. The program handles the retrieval, display, and modification of item pricing information, ensuring that all necessary validations are performed before any changes are committed to the database.

Order Status and Header Rules

Price Maintenance: fjvprl1, fjvprlr, fjvprlu, fjvarl1, fjlevl1, fjvlelr, favall1

1. Check for Valid Date

•**Logic:** The program checks if a valid "from date" is filled out when creating or updating a price record.

•**File:** fjvprlr (Price Records)

•**Field:** jvprlr_gdat

•**Condition:** "The process will not proceed if jvprlr_gdat is not valid."

2. Display Outdated Prices

•**Logic:** Outdated prices are displayed when the user first accesses the price maintenance screen.

•**File:** fjvprlr (Price Records)

•**Field:** jvprlr_rsts

•**Condition:** "If the price is marked as outdated, it will be shown in the maintenance screen."

Configuration and Authorization Rules

1. Field Protection Based on Company Settings

•**Logic:** Certain fields are protected based on company-specific validation rules.

•**Files:**

•favall1 (Field Validation Settings)

•**Fields:**

•avall1_apgm (Program Name)

•**Condition:** "Fields will be protected if the validation settings indicate that they should be."

2. Supplier Validation

•**Logic:** The program checks if the supplier exists in the database before allowing the price to be updated.

•**File:** fjlevl1 (Supplier Records)

•**Field:** jlevl1_ldor

•**Condition:** "The process will not allow updates if the supplier ID does not exist."

Financial and Transactional Rules

1. Price Update Logic

•**Logic:** When updating a price, the program checks if the price is valid and not outdated.

•**File:** fjvprlu (Price Update Records)

•**Fields:**

•jvprlu_gdat (Price Validity Date)

•jvprlu_rsts (Price Status)

•**Condition:** "The update will only proceed if jvprlu_gdat is valid and jvprlu_rsts indicates the price is active."

2. Deletion of Price Records

•**Logic:** The program allows deletion of price records if certain conditions are met.

•**File:** fjvprlr (Price Records)

•**Condition:** "The deletion will only occur if the price record is not currently in use."

–

Special Conditions (Program-Specific)

1. Display Message for Existing Records (JX100R)

•**Logic:** If a user attempts to create a price record that already exists, a warning message is displayed.

•**File:** fjvprlr (Price Records)

•**Field:** jvprlr_ldor

•**Condition:** "If a record with the same supplier and item already exists, the program will show a warning message."

2. Cross Docking Logic (JX100R)

•**Logic:** The program checks for cross-docking requirements when updating prices.

•**File:** fjvprlu (Price Update Records)

•**Fields:** jvprlu_ldor (Supplier ID), jvprlu_gdat (Price Validity Date)

•**Condition:** "If cross-docking is applicable, additional validations will be performed before the update."

–

Subprogram Calls Affecting Logic

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

1. JK101R (Logistics Update Program)

•**Trigger:** Called during the price update process.

•**Logic:** Updates logistics-related information for the item being modified.

•**Impact:** "This call ensures that all logistics data is synchronized with the updated pricing."

2. JV790R (Search Text Update Program)

•**Trigger:** Called after a successful price update.

•**Logic:** Updates search text for the item in the database.

•**Impact:** "This ensures that any changes to pricing are reflected in search functionalities."

3. JL500R (Supplier Validation Program)

•**Trigger:** Called when validating supplier information.

•**Logic:** Checks if the supplier exists and is valid.
•**Impact:** "This represents a critical validation step before any price updates are allowed."