FP100R.MBR

Path: NXCLOUD/rpgsrc/FP100R.MBR **Generated**: 2026-01-08 11:56:30 **Processing Time**: 12675ms

—

Business Logic for Special Pricing Maintenance

This document outlines the business rules that govern the special pricing maintenance process, based on an analysis of the RPG program FP100R. The primary focus is on the logic related to managing special pricing records, including the creation, updating, and deletion of these records. The core logic for special pricing maintenance is contained within the various subroutines in FP100R. The program processes records from multiple files, allowing users to manage special pricing based on various criteria such as customer categories, product types, and pricing groups.

—

Order Status and Header Rules

Special Pricing Maintenance: fprilr, fprilu, fpril1, fpril2, fpril3, fpril4, fpril5, fpril6, fpril7, fpril8, fpril9

1. Record Selection

•**Logic:** The program selects records based on various criteria such as pricing group, customer category, and product type.

•**File:** fprilr (Special Pricing Records)

•**Field:** fprilr_prgr

•**Condition:** The program will not select a record if fprilr_prgr is blank or if other key fields such as fprilr_kund or fprilr_vare do not match the input criteria.

2. Record Update

•**Logic:** When updating records, the program checks if the record exists and updates the relevant fields.

•**File:** fprilu (Updated Special Pricing Records)

•**Field:** fprilu_prgr

•**Condition:** The update will only occur if the record is found; otherwise, an error message is generated.

—

Configuration and Authorization Rules

1. Customer Category Validation

•**Logic:** The program validates customer categories against existing records to ensure they are valid.

•**Files:**

•rkunl1 (Customer Records)

•ra11pf (Customer Category Records)

•**Fields:**

•rkunl1_kund (Customer ID from rkunl1)

•ra11pf_kkod (Customer Category Code from ra11pf)

•**Condition:** The process will not proceed if the customer category does not exist in the ra11pf file.

2. Product Validation

•**Logic:** The program checks if the product exists before allowing any pricing updates.

•**File:** vvarl1 (Product Records)
•**Field:** vvarl1_vare
•**Condition:** The process will not allow updates if the product does not exist in the vvarl1 file.

Financial and Transactional Rules

1. Pricing Calculation
•**Logic:** The program calculates the selling price and cost price based on discounts applied to the base price.
•**File:** fprilr (Special Pricing Records)
•**Fields:**
•fprilr_sapr (Selling Price)
•fprilr_kopr (Cost Price)
•**Condition:** The calculation will only occur if both the selling price and cost price are defined; otherwise, the program will call another program to fetch these values.

2. Discount Application
•**Logic:** Discounts are applied to the selling price based on predefined discount rates.
•**File:** fprilr (Special Pricing Records)
•**Condition:** Discounts will be applied if the discount fields (c2rab1, c2rab2) are greater than zero.

Special Conditions (Program-Specific)

1. Subfile Handling (FP100R)
•**Logic:** The program manages subfile operations, including display, update, and delete functionalities.
•**File:** b1sfl (Subfile for Special Pricing)
•**Field:** b1valg (User action)
•**Condition:** The program will handle actions based on user input, such as creating, copying, or deleting records.

2. Error Handling (FP100R)
•**Logic:** The program includes error handling for various user inputs and system checks.
•**File:** dspfbk (Display Feedback)
•**Fields:** Various fields for error messages.
•**Condition:** Errors will be flagged if required fields are blank or if invalid data is entered.

Subprogram Calls Affecting Logic

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

1. RA530R (Price Group Validation)
•**Trigger:** Called when validating price groups.
•**Logic:** This program checks if the price group exists and retrieves its description.
•**Impact:** This call acts as a **validation step** for pricing information.

2. RK500R (Customer Information Retrieval)
•**Trigger:** Called to retrieve customer information based on customer ID.

•**Logic:** This program fetches customer details and updates related fields.

•**Impact:** This is a **critical information retrieval step** that ensures accurate customer data is used.

3. VP701R (Price Retrieval)

•**Trigger:** Called when calculating prices if they are not already set.

•**Logic:** This program retrieves the selling and cost prices for a given product.

•**Impact:** This represents a **key step in ensuring pricing accuracy** for special pricing records.