JM100R.MBR

Path: NXCLOUD/rpgsrc/JM100R.MBR **Generated**: 2026-01-08 12:34:46 **Processing Time**: 10900ms

Business Logic for Campaign Header Maintenance

This document outlines the business rules that govern the maintenance of campaign headers, based on an analysis of the RPG program JM100R. The primary focus is on the logic for managing campaign data, including creation, updating, and deletion of campaign records.

The core logic for campaign header maintenance is contained within the main subroutine in JM100R. The program processes records related to campaigns, including their associated pricing and responsible parties, through various subroutines and file operations.

Campaign Header and Subfile Rules

Campaign Header Maintenance: fjkahl1, fjkahl2, fjkahl3, fjkahl4, fjkahlr, fjkahlu, fjkavlr, fjkavlu

1. Record Creation

•**Logic:** New campaign records can be created based on user input from the subfile. If the input fields are valid, the program writes a new record to the database.

•**File:** fjkahl1 (Campaign header file)

•**Field:** jkahl1_kamp

•**Condition:** A record will be created if the input fields are not blank and pass validation checks.

2. Record Update

•**Logic:** Existing campaign records can be updated. The program checks if the record exists and updates the fields accordingly.

•**File:** fjkahl1 (Campaign header file)

•**Field:** jkahl1_kamp

•**Condition:** The record will be updated if it is found in the database.

3. Record Deletion

•**Logic:** Campaign records can be deleted. The program will prompt the user for confirmation before deletion.

•**File:** fjkahl1 (Campaign header file)

•**Field:** jkahl1_kamp

•**Condition:** The record will be deleted if the user confirms the action.

Subfile Management Rules

1. Subfile Population

•**Logic:** The subfile is populated with campaign records from the database. The program reads records based on the current position and displays them in the subfile.

•**Files:**

•fjkahl1 (Campaign header file)

•**Fields:**

•jkahl1_kamp (Campaign identifier)

•**Condition:** The subfile will be populated as long as there are records to read from the database.

2. Subfile Navigation

•**Logic:** Users can navigate through the subfile using function keys. The program handles roll-up and roll-down actions to move through the records.

•**File:** fjkahl1 (Campaign header file)

•**Condition:** Navigation actions will be processed based on user input from the function keys.

Validation and Error Handling Rules

1. Input Validation

•**Logic:** The program validates user input for campaign fields. If any required fields are blank or invalid, the program will display an error message.

•**File:** fjkahl1 (Campaign header file)

•**Fields:**

•jkahl1_kamp (Campaign identifier)

•**Condition:** Validation checks will trigger error messages if fields do not meet criteria.

2. Error Messaging

•**Logic:** The program provides feedback to the user through error messages if an action cannot be completed due to validation failures.

•**File:** fjkahl1 (Campaign header file)

•**Condition:** Error messages will be displayed based on specific validation failures.

Special Conditions (Program-Specific)

1. Campaign Price Maintenance (JM110R)

•**Logic:** When maintaining campaign prices, the program calls another program JM110R to handle the pricing logic.

•**File:** fjkavlr (Campaign price file)

•**Field:** jkavlr_vkam

•**Condition:** This call is made when the user selects to maintain prices for a specific campaign.

2. Duplicate Record Check

•**Logic:** Before creating or updating a campaign, the program checks if a record with the same identifier already exists.

•**File:** fjkahl1 (Campaign header file)

•**Field:** jkahl1_kamp

•**Condition:** If a duplicate is found, the program will prompt the user with a warning message.

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 during input validation for price groups.

•**Logic:** Validates the price group against existing records.

•**Impact:** Ensures that only valid price groups are accepted, preventing errors in campaign pricing.

2. VL711R (Price Group Retrieval)

•**Trigger:** Called during program initialization to retrieve the current price group.

•**Logic:** Fetches the price group associated with the current firm.

•**Impact:** Ensures that the correct price group is displayed and used for campaign maintenance.

3. JM110R (Campaign Price Maintenance)

•**Trigger:** Called when the user opts to maintain campaign prices.

•**Logic:** Handles the logic for updating or creating campaign prices.

•**Impact:** Represents a critical step in ensuring that campaign pricing is correctly managed and updated in the system.