

AF010R.MBR

Path: NXCLOUD/rpgsrc/AF010R.MBR Generated: 2026-01-09 10:13:44 Processing Time: 13648ms

Business Logic for Post Code Register Maintenance

This document outlines the business rules that govern the maintenance of the post code register, based on an analysis of the RPG program AF010R. The primary focus is on how post codes are retrieved, displayed, and modified within the system.

The core logic for handling post code records is contained within the subfile subroutine in AF010R. The program processes records from the post code files, allowing users to view, create, update, and delete post code entries.

Order Status and Header Rules

Post_Code_Maintenance: aposl1, aposl2, aposlr, aposlu, af010d

1. Retrieve Post Code

- Logic:** The program retrieves post code records based on user input and displays them in a subfile format.
- File:** aposl1 (Post Code Table)
- Field:** aposl1_ponr
- Condition:** The process will not select a record if b1valg is not equal to 1, indicating that the user is not requesting a post code retrieval.

2. Update Post Code

- Logic:** Allows the user to update an existing post code record in the database.
- File:** aposlr (Post Code Record)
- Field:** aposlr_ponr
- Condition:** The update will occur if b1valg is equal to 2, indicating that the user has chosen to modify a post code.

Configuration and Authorization Rules

1. Subfile Initialization

- Logic:** Initializes the subfile for displaying post codes and prepares the user interface.
- Files:**
 - af010d (Workstation Display File)
 - aposl1 (Post Code Table)
- Fields:**
 - w_fcrn (First Record Number)
 - w_srrn (Relative Record Number)
- Condition:** The subfile is initialized when the program starts, and records are loaded based on the current state of the database.

2. Delete Post Code

- Logic:** Deletes a selected post code record from the database.

- File:** aposlu (Post Code Update Table)
 - Field:** aposlu_ponr
 - Condition:** The deletion will occur if b1valg is equal to 4, indicating the user has chosen to delete a post code.
-

Financial and Transactional Rules

1. Create New Post Code

- Logic:** Allows the user to create a new post code entry in the database.
- File:** aposlu (Post Code Update Table)
- Fields:**
 - apsted (Post Code Description)
 - apfylk (County)
- Condition:** The creation will occur if b1valg is equal to 1, indicating the user has chosen to add a new post code.

2. Display Post Code Details

- Logic:** Displays detailed information about a selected post code.
 - File:** aposlr (Post Code Record)
 - Condition:** The display will occur if b1valg is equal to 5, indicating that the user wants to view details of a post code.
-

Special Conditions (Program-Specific)

1. Subfile Processing (AF010R)

- Logic:** Handles the reading and processing of records in the subfile for display and interaction.
- File:** aposl1 (Post Code Table)
- Field:** aposl1_ponr
- Condition:** The subfile is processed during user interactions, and it will refresh based on the actions taken by the user. Note: The subfile management logic is crucial for maintaining the integrity of the displayed data.

2. Error Handling for Post Code Creation (AF010R)

- Logic:** Displays an error message if a post code already exists when attempting to create a new entry.
 - File:** aposlr (Post Code Record)
 - Fields:** aposlr_ponr (Post Code Number), c1ponr (Input Post Code)
 - Condition:** If the post code already exists in the database, an error message is displayed, preventing duplicate entries.
-

Subprogram Calls Affecting Logic

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

1. dsp_subfile (Display Subfile)

- Trigger:** Called during the initialization and processing of the subfile.
- Logic:** This subroutine handles the display of post code records in a user-friendly format.

- Impact:** This call acts as a **major logical gateway** for user interactions with the post code records.

2. clr_subfile (Clear Subfile)

- Trigger:** Called to clear the current contents of the subfile before loading new data.
- Logic:** This subroutine resets the subfile state, ensuring that old data does not persist.
- Impact:** This is a **destructive filtering step** that prepares the interface for new data.

3. crt_subfile (Create Subfile)

- Trigger:** Called to populate the subfile with records from the database.
- Logic:** This subroutine reads records from the post code tables and fills the subfile for display.
- Impact:** This represents the handoff to the next major business function, allowing users to interact with the post code data effectively.