

# NN700R.MBR

**Path:** NXCLOUD/rpgsrc/NN700R.MBR **Generated:** 2026-01-08 15:04:08 **Processing Time:** 9760ms

## Business Logic for NN700R

This document outlines the business rules that govern the export of customer information from the online store, based on an analysis of the RPG program NN700R. The primary focus is on the logic that processes customer data and prepares it for export.

The core logic for exporting customer information is contained within the main program section of NN700R. The program retrieves customer data from various files, constructs a structured message, and writes it to an output file.

## Customer Data and Export Rules

**NN700R:** rkunl1, fkprl1, apospf, anbl13, anbhl1, netkund, netkrappr

### 1. Customer Existence Check

- Logic:** The program checks if the online store and its owner exist. If not found, the process is terminated with an error status.
- File:** anbl13 (Online store information)
- Field:** anbl13\_fgr, anbl13\_fir
- Condition:** The process will not proceed if the online store record is not found.

### 2. Message Header Creation

- Logic:** The program constructs the message header with a unique reference number and type identifier.
- File:** N/A
- Field:** d\_unhrefno, d\_unhtypeid
- Condition:** This step is always executed after confirming the existence of the online store.

### 3. Customer Data Processing

- Logic:** The program iterates through customer records, checking for mandatory fields and writing valid records to the output.
- File:** rkunl1 (Customer information)
- Field:** rkftnr, rknavn, rksted
- Condition:** Records are excluded if they lack a business number or postal address.

### 4. Customer Project Data Processing

- Logic:** For each customer, the program checks for associated projects and writes them to the output if they meet the date criteria.
- File:** fkprl1 (Customer project information)
- Field:** fltdat
- Condition:** Projects are only included if the project date is greater than or equal to the current date.

## Validation and Error Handling Rules

### 1. Email Address Validation

- Logic:** The program checks if the customer email address is present. If missing, the record is excluded, and a warning is logged.
- File:** rkunl1
- Field:** rkemal
- Condition:** If rkemal is blank and rkebiz is 1, the customer is excluded from the export.

## 2. Mandatory Fields Check

- Logic:** The program verifies that all mandatory fields are filled. If any are missing, it logs a message and sets the status to 'R' (record excluded).
  - File:** rkunl1
  - Field:** rkftnr, rksted
  - Condition:** The process will log a warning and exclude the record if these fields are not populated.
- 

# Message Structure Rules

## 1. Message Footer Creation

- Logic:** After processing all customer records, the program constructs the message footer indicating the total number of records processed.
  - File:** N/A
  - Field:** d\_untnosegmen
  - Condition:** This step is executed after all customer records have been processed.
- 

# Subprogram Calls Affecting Logic

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

## 1. NN700R (Customer Export Program)

- Trigger:** The program is executed to export customer information.
- Logic:** It retrieves customer and project data, constructs messages, and writes them to an output file.
- Impact:** This call represents the core functionality of exporting customer data.

## 2. APOSPF (Postal Code Lookup)

- Trigger:** When processing customer addresses, the program checks for postal code information.
- Logic:** It retrieves postal information based on the provided postal number.
- Impact:** This ensures that customer addresses are complete and valid for export.

## 3. ANBHL1 (Online Store Owner Lookup)

- Trigger:** When the program checks for the existence of the online store.
- Logic:** It retrieves information about the store owner.
- Impact:** This is crucial for validating the store's existence before proceeding with customer data processing.