

# LO101R.MBR

**Path:** NXCLOUD/rpgsrc/LO101R.MBR **Generated:** 2026-01-08 12:58:35 **Processing Time:** 17783ms

## Business Logic for LO101R

This document outlines the business rules that govern the registration of orders/purchases and related information, based on an analysis of the RPG program LO101R. The primary focus is on the validation and processing of order header information.

The core logic for order processing is contained within the various subroutines in LO101R. The program retrieves and validates data from the order header file and other related files, ensuring that all necessary conditions are met before updating the order information.

### Order Status and Header Rules

**LO101R:** LOHEL1, LOHELU, SIHEL1, RLTRL7, VVARL1, VLTYL1, VMVAL1, LODTLU, RLEVL1, RP1PL1, SIHELA

#### 1. Check Existence of Order Header

• **Logic:** The program checks if the order header exists in the LOHEL1 file. If found, it retrieves various fields related to the order.

• **File:** LOHEL1 (Order Header Register)

• **Field:** LOHEL1\_KEY

• **Condition:** The process will not proceed if the order header is not found.

#### 2. Validate Department Code

• **Logic:** If the department code is provided, it checks if it exists in the department code register.

• **File:** KODE-REGISTER (Department Code Register)

• **Field:** C2AVDE

• **Condition:** The process will flag an error if the department does not exist.

#### 3. Validate Purchaser Code

• **Logic:** Similar to the department code, it checks if the purchaser code is valid.

• **File:** KODE-REGISTER (Purchaser Code Register)

• **Field:** C2INNK

• **Condition:** An error is flagged if the purchaser does not exist.

#### 4. Validate Warehouse Code

• **Logic:** The program checks if the warehouse code is valid.

• **File:** KODE-REGISTER (Warehouse Code Register)

• **Field:** C2LAGE

• **Condition:** An error is flagged if the warehouse does not exist.

#### 5. Validate Payment Terms

• **Logic:** Checks if the payment terms code exists in the payment terms register.

• **File:** KODE-REGISTER (Payment Terms Register)

• **Field:** C2BETB

• **Condition:** An error is flagged if the payment terms do not exist.

## **6. Validate Currency Code**

•**Logic:** Validates the currency code against the currency register.

•**File:** KODE-REGISTER (Currency Register)

•**Field:** C2VALK

•**Condition:** An error is flagged if the currency code does not exist.

## **7. Validate Shipping Method**

•**Logic:** Checks if the shipping method code exists in the shipping method register.

•**File:** KODE-REGISTER (Shipping Method Register)

•**Field:** C2LMAT

•**Condition:** An error is flagged if the shipping method does not exist.

# **Configuration and Authorization Rules**

## **1. Check for Valid Invoice Issuer**

•**Logic:** Validates that the invoice issuer exists in the supplier register.

•**File:** RLEVL1 (Supplier Register)

•**Field:** C3FLEV

•**Condition:** An error is flagged if the invoice issuer does not exist.

## **2. Check for Duplicate Invoice Number**

•**Logic:** Checks if the invoice number has been used previously for the supplier.

•**File:** SIHELA (Historical Invoice Register)

•**Field:** C3SFAK

•**Condition:** An error is flagged if the invoice number has been previously registered.

# **Financial and Transactional Rules**

## **1. Validate Invoice Date**

•**Logic:** Ensures that the invoice date is not greater than the due date.

•**File:** SIHELA (Historical Invoice Register)

•**Field:** C3FKDA, C3FFDA

•**Condition:** An error is flagged if the invoice date exceeds the due date.

## **2. Check Validity of KID Number**

•**Logic:** Validates the KID number format and checks its validity against the KID register.

•**File:** SIHELA (Historical Invoice Register)

•**Field:** C3BKID

•**Condition:** An error is flagged if the KID number is invalid.

# **Special Conditions (Program-Specific)**

## **1. Update Order Header**

•**Logic:** Updates the order header with the validated information.

•**File:** LOHELU (Order Header Update)

•**Field:** Various fields including LOMKOD, LOAVDE, LOINNK, etc.

•**Condition:** The update occurs only if the order header exists and all validations pass.

## **2. Handle Changes in VAT Code**

- **Logic:** Adjusts the VAT code based on changes in the order header.
- **File:** LODTTL (Order Line Table)
- **Field:** C2MKOD
- **Condition:** This check is triggered if the VAT code on the order header changes.

## Subprogram Calls Affecting Logic

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

### 1. CO402R (Check Warehouse)

- **Trigger:** Called to validate the warehouse against the user's default warehouse.
- **Logic:** Checks if the warehouse is newly created and differs from the default.
- **Impact:** This affects the department code assignment if the warehouse is new.

### 2. RA507R (Department Query)

- **Trigger:** Called when querying the department code.
- **Logic:** Retrieves department details based on the provided department code.
- **Impact:** Ensures that the department information is accurate before proceeding.

### 3. RS709R (Purchaser Query)

- **Trigger:** Called to retrieve purchaser information.
- **Logic:** Validates and fetches details for the purchaser code.
- **Impact:** Ensures that the purchaser is valid before processing the order.

### 4. RS720R (Warehouse Query)

- **Trigger:** Called to validate warehouse information.
- **Logic:** Retrieves details for the specified warehouse.
- **Impact:** Ensures that the warehouse is valid and operational.

### 5. AA007R (User Confirmation)

- **Trigger:** Called to prompt the user for confirmation on critical validations.
- **Logic:** Displays messages and waits for user input.
- **Impact:** This acts as a gatekeeper for critical decisions in the order processing.

Use clear, business-friendly language. Fill in all sections with details from the code. Output only the Markdown.