

Phase 4: Process Automation (Admin) – Detailed Steps

1. Validation Rules

Validation rules enforce data integrity by preventing incorrect records from being saved.

Step 1: Ensure Return Date \geq Issue Date

1. Go to **Setup** → **Object Manager** → **Book Transaction** → **Validation Rules**.
2. Click **New**.
3. Rule Name: Return_Date_Check.
4. Formula:
5. Return_Date__c < Issue_Date__c
6. Error Message: "Return Date cannot be before Issue Date."
7. Error Location: Field Return_Date__c.
8. Save and activate.

Result: Users cannot save a return date earlier than the issue date.

Step 2: Prevent Issuing if Available Quantity = 0

1. Go to **Book Object** → **Validation Rules** → **New**.
2. Rule Name: No_Book_Available.
3. Formula:
4. Available_Quantity__c <= 0
5. Error Message: "Cannot issue book. No copies available."
6. Error Location: Field Available_Quantity__c.
7. Save and activate.

Result: Books cannot be issued if none are available.

2.Flows (Lightning Flow / Process Builder)

Flows automate multi-step processes. You can choose **Record-Triggered Flows** (Lightning Flow) instead of Process Builder, as Salesforce recommends Flows now.

Step 1: Book Issue Flow

Goal: Automatically update inventory and set due date when a book is issued.

Steps:

1. **Go to Setup → Flows → New Flow → Record-Triggered Flow.**
2. **Object:** Book Transaction.
3. **Trigger:** When a record is **created**.
4. **Condition:** Transaction Type = "Issue".
5. **Elements inside Flow:**
 - **Get Records:** Get the related Book record.
 - **Update Records:** Decrement Available_Quantity__c by 1.
 - **Update Transaction:** Set Due_Date__c = Issue_Date__c + Loan_Period__c.
6. **Activate** the flow.

Result: Issue date sets due date and reduces available quantity automatically.

Step 2: Book Return Flow

Goal: Update transaction status, increment book quantity, calculate fine if overdue.

Steps:

1. **New Record-Triggered Flow → Object:** Book Transaction.
2. **Trigger:** When a record is **updated**.
3. **Condition:** Status = "Returned".
4. **Flow Elements:**
 - **Get Records:** Fetch related Book record.
 - **Update Records:** Increment Available_Quantity__c by 1.
 - **Decision Element:** Check if Return_Date__c > Due_Date__c.

- **Assignment:** If overdue, calculate fine:
- $\text{Fine_c} = (\text{Return_Date_c} - \text{Due_Date_c}) * \text{Fine_Per_Day_c}$
- **Update Transaction Record:** Set Fine and Status.

5. **Activate** the flow.

Result: Book return automatically updates inventory and calculates fine if overdue.

3.Approval Process

Approval Processes automate managerial or librarian approval for specific actions.

Step 1: Book Purchase Request Approval

1. Go to **Setup → Approval Processes → New Approval Process → Use Standard Wizard.**
2. **Object:** Book Purchase Request.
3. Steps:
 - Define entry criteria (e.g., Status = "Pending Approval").
 - Select approver (e.g., Librarian/Head Librarian).
 - Set **Initial Submission Actions:** Notify approver.
 - **Approval Actions:** Update status to "Approved" and trigger a flow to add book to inventory.
 - **Rejection Actions:** Update status to "Rejected".
4. Activate the approval process.

Result: All purchase requests are automatically routed for approval.

Step 2: Special Book Reservation Approval

- Same process as above but for **reserved books**, ensuring high-demand books are managed properly.

4.Email Alerts

Email alerts automatically notify members about important events.

Step 1: Due Date Reminder

1. Go to **Setup** → **Email Alerts** → **New Email Alert**.
2. **Object:** Book Transaction.
3. Email Template: "Book Due Reminder".
4. **Recipients:** Member (lookup field).
5. Trigger: **Time-Based Workflow/Flow:** 2 days before Due_Date__c.

Result: Members are reminded before books are due.

Step 2: Reservation Availability Notification

1. Email Template: "Your Reserved Book is Available".
2. Trigger: When reserved book becomes available (Available_Quantity__c > 0).
3. Send to the member who reserved.

Result: Members get automatic notification when their reserved books are ready.