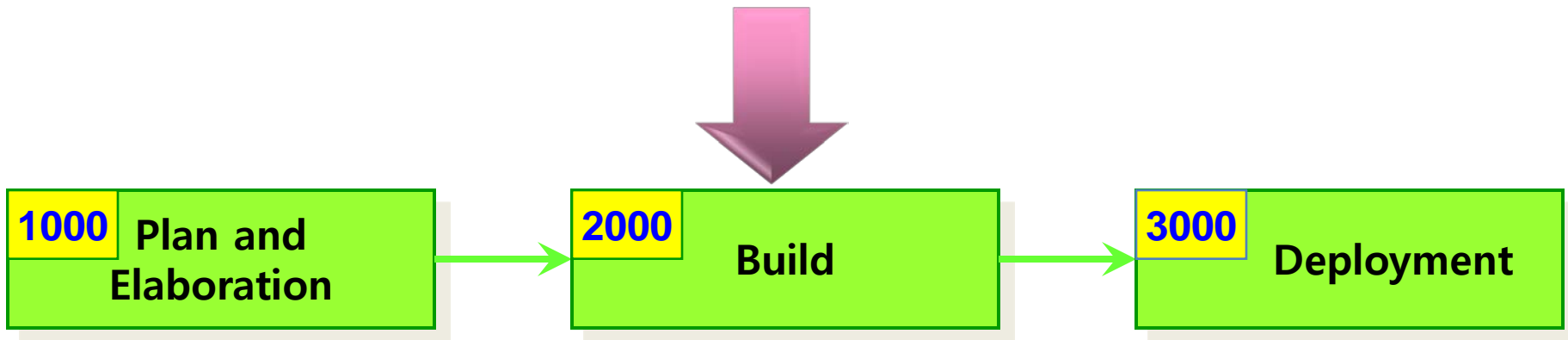


Case Study : Design

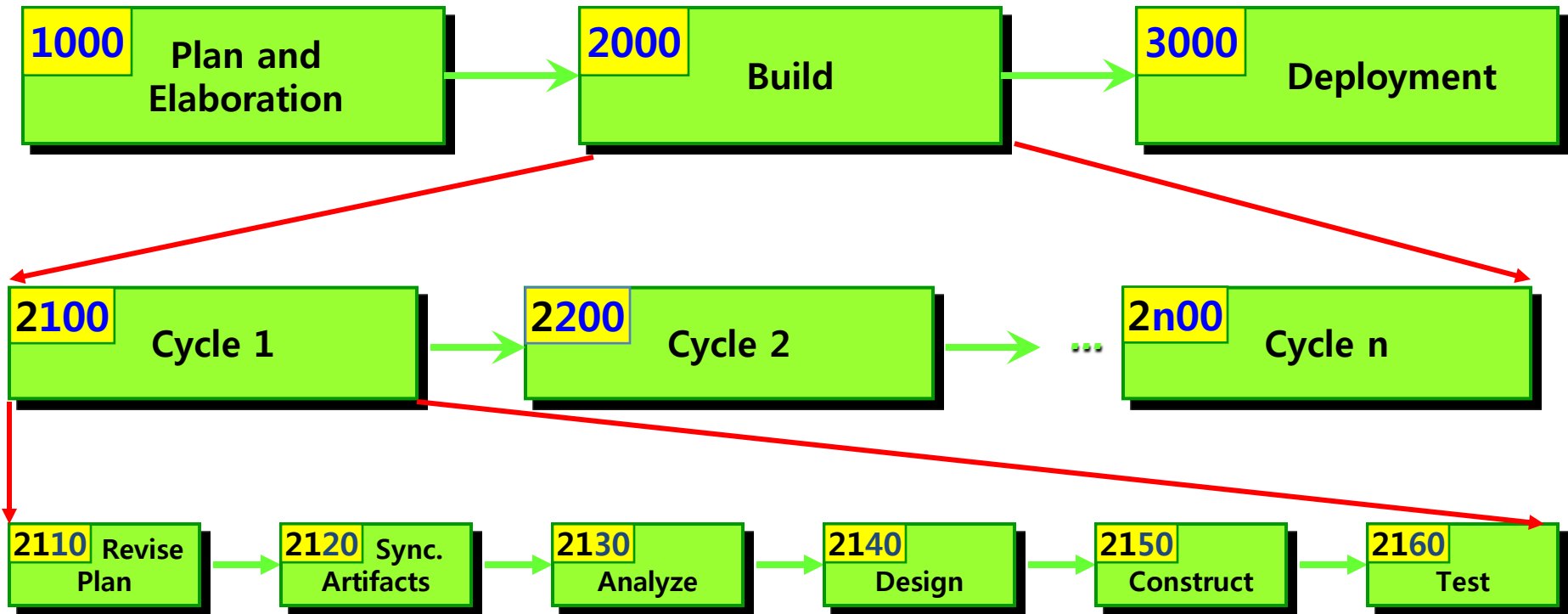
- Library Management System

Lecturer: JUNBEOM YOO
jbyoo@konkuk.ac.kr

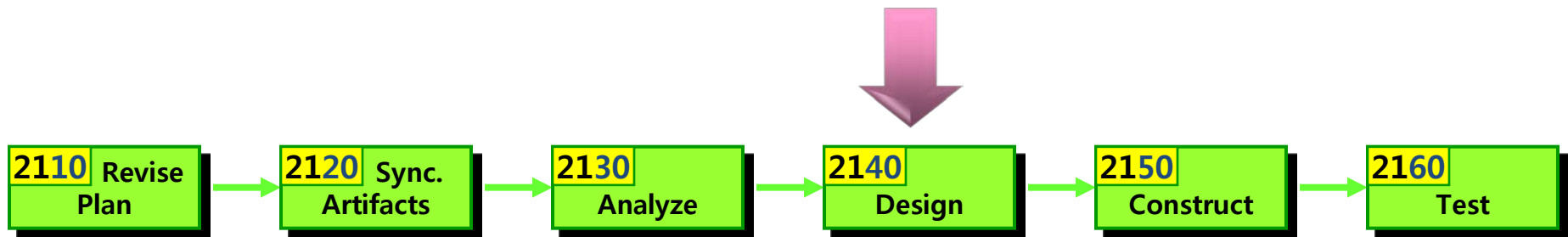
Stage 2000. Build



6 Phases of 'Build' Stage



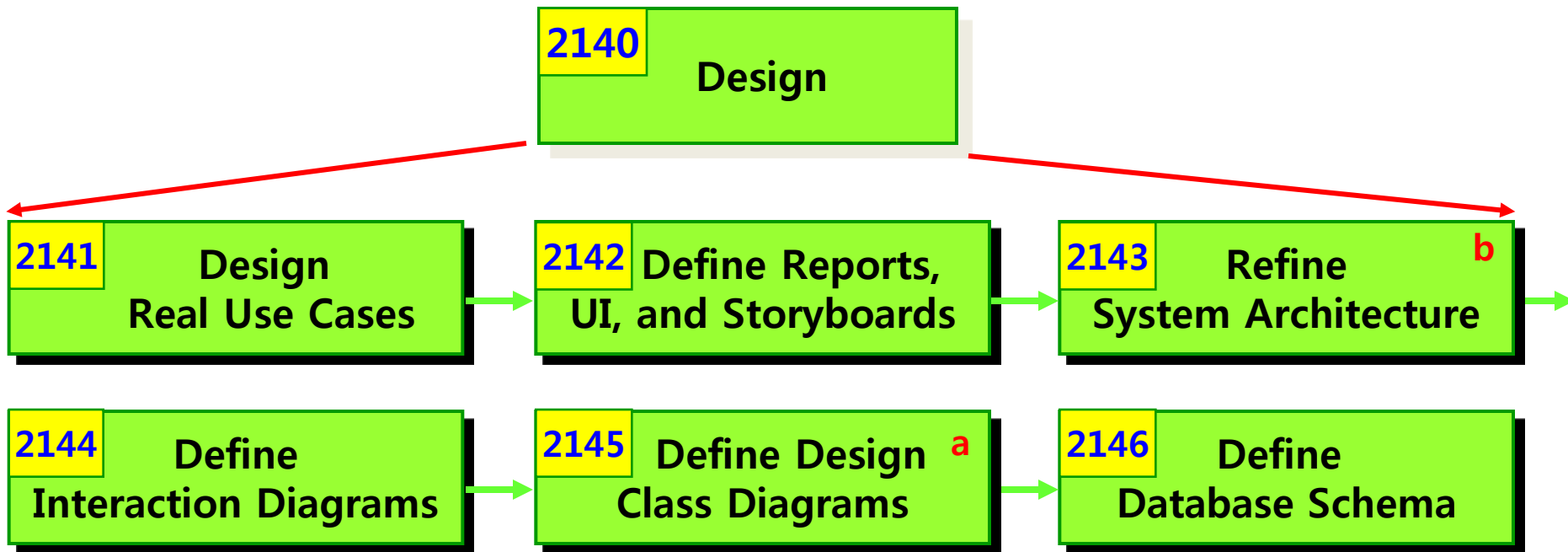
Phase 2040. Design



Phase 2040. Design

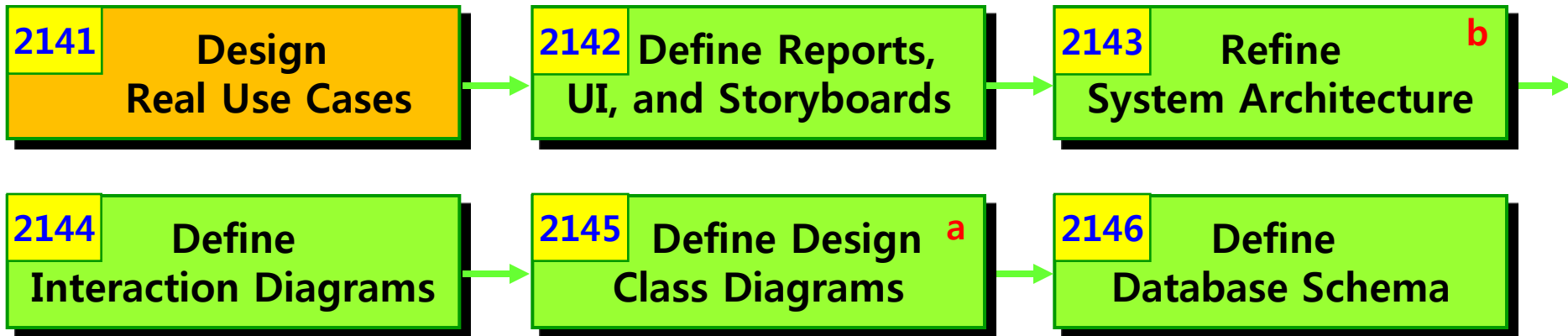
- Phase 2040 Activities

a. In parallel with interaction diagrams
b. Varied order



Activity 2041. Design Real Use Cases

- a. In parallel with interaction diagrams
- b. Varied order



Activity 2041. Design Real Use Cases

1. Make Reservation

| | |
|--------------------------------------|--|
| Use Case | 1. Make Reservation |
| Actor | Librarian |
| Purpose | Create a new reservation |
| Overview | (As in the business use case) |
| Type | Primary and Real |
| Cross Reference | System Functions: R1.1, R3.1 Use Case: "Add Borrower" |
| Pre-Requisites | A borrower should be registered. |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (A) A librarian inputs an <i>isbn</i> and <i>ssn</i> of the title 2. (S) Find a corresponding title 3. (S) Find a corresponding borrower 4. (S) Create a new reservation 5. (S) Store the new reservation 6. (S) Increase <i>reservationCount</i> in the borrower 7. (S) Increase <i>reservationCount</i> in the title |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | Line 2: If the title does not exist, display an error message. Line 3: If the borrower does not exist, display an error message. |

Activity 2041. Design Real Use Cases

2. Remove Reservation

| | |
|--------------------------------------|---|
| Use Case | 2. Remove Reservation |
| Actor | Librarian |
| Purpose | Remove a reservation information |
| Overview | (As in the business use case) |
| Type | Primary and Real |
| Cross Reference | System Functions: R1.2, R1.3 Use Case: "Lend Item" |
| Pre-Requisites | A borrower should be registered. A title should have been reserved. |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (A) A librarian inputs an <i>isbn</i> of the title 2. (S) Find a corresponding reservation 3. (S) Remove the reservation 4. (S) Decrease <i>reservationCount</i> of the borrower 5. (S) Decrease <i>reservationCount</i> of the title |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | Line 2: If the reservation does not exist, display an error message. |

Activity 2041. Design Real Use Cases

3. Lend Item

| | |
|--------------------------------------|---|
| Use Case | 3. Lent Item |
| Actor | Librarian |
| Purpose | Lend items to a borrower |
| Overview | (As in the business use case) |
| Type | Primary and Real |
| Cross Reference | System Functions: R1.3, R1.2, R3.1 Use Cases: "Remove Reservation", "Add Borrower" |
| Pre-Requisites | An item should exist. |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (A) A librarian inputs an item's <i>ID</i> and <i>ssn</i> of the borrower 2. (S) Find a corresponding borrower 3. (S) Find a corresponding item 4. (S) Create a new loan 5. (S) Store the new loan 6. (S) Set <i>validLoan</i> to true 7. (S) Increase <i>loanCount</i> of borrower 8. (S) Set <i>available</i> to false 9. (S) Decrease <i>AvailableCount</i> of the title |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | Line 2: If the borrower does not exist, invoke "Add Borrower" use case. |

Activity 2041. Design Real Use Cases

4. Return Item

| | | |
|--------------------------------------|--|---|
| Use Case | 4. Return Item | |
| Actor | Librarian | |
| Purpose | Return items loaned | |
| Overview | (As in the business use case) | |
| Type | Primary and Real | |
| Cross Reference | System Functions: R1.4.1, R1.4.2, R1.6 Use Cases: "Calculate Late-Return-Fee", "Notify Availability" | |
| Pre-Requisites | An item should have been loaned. | |
| Typical Courses of Events | <p>(A) : Actor, (S) : System</p> <ol style="list-style-type: none"> 1. (A) A librarian inputs an item's <i>ID</i> 2. (S) Find a corresponding loan 3. (S) Get item information from the loan 4. (S) Get title information from the item 5. (S) Get <i>loanPeriod</i> from the title 6. (S) Compute <i>calculateLateReturnFee</i> 7. (S) Check <i>reservationCount</i> of the title. | <ol style="list-style-type: none"> 8. (S) If the title is reserved, find the corresponding reservation 9. (S) Decrease <i>loanCount</i> of the loan. 10. (S) Decrease <i>loanCount</i> of the Borrower. 11. (S) Set <i>validLoan</i> of the borrower to false. 12. (S) Set <i>available</i> of the item to true. 13. (S) Increase <i>AvailbaleCount</i> of the title. |
| Alternative Courses of Events | N/A | |
| Exceptional Courses of Events | Line 2: If the loan does not exist, display an error message. | |

Activity 2041. Design Real Use Cases

5. Calculate Late-Return-Fee

| | |
|--------------------------------------|---|
| Use Case | 5. Calculate Late-Return-Fee |
| Actor | None |
| Purpose | Compute late-return fee for an item returned late |
| Overview | (As in the business use case) |
| Type | Primary and Real |
| Cross Reference | System Functions: R1.4.1, R1.4.2 Use Cases: "Return Item" |
| Pre-Requisites | Lending time of an item should have expired |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (S) Calculate Late-Return-Fee of the item 2. (S) Display the Late-Return-Fee |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | N/A |

Activity 2041. Design Real Use Cases

6. Get Replacement-Fee

| | |
|--------------------------------------|---|
| Use Case | 6. Get Replacement-Fee |
| Actor | Librarian |
| Purpose | Compute replacement-fee for a lost title |
| Overview | (As in the business use case) |
| Type | Primary and Real |
| Cross Reference | System Functions: R1.5 Use Cases: - |
| Pre-Requisites | A title should be lost. |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (A) A librarian inputs an item's <i>ID</i> 1. (S) Find a corresponding loan 2. (S) Get an item from the loan 3. (S) Get a title from the item 4. (S) Get price of the title 5. (S) Compute <i>replacementFee</i> 6. (S) Set <i>validLoan</i> to false 7. (S) Update the loan 8. (S) Decrease <i>loanCount</i> of the borrower. 9. (S) Set the lost of the item to true. (?) 10. (S) Decrease <i>numOfItem</i> of the title. |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | Line 2: If the loan does not exist, display an error message. |

Activity 2041. Design Real Use Cases

7. Notify Availability

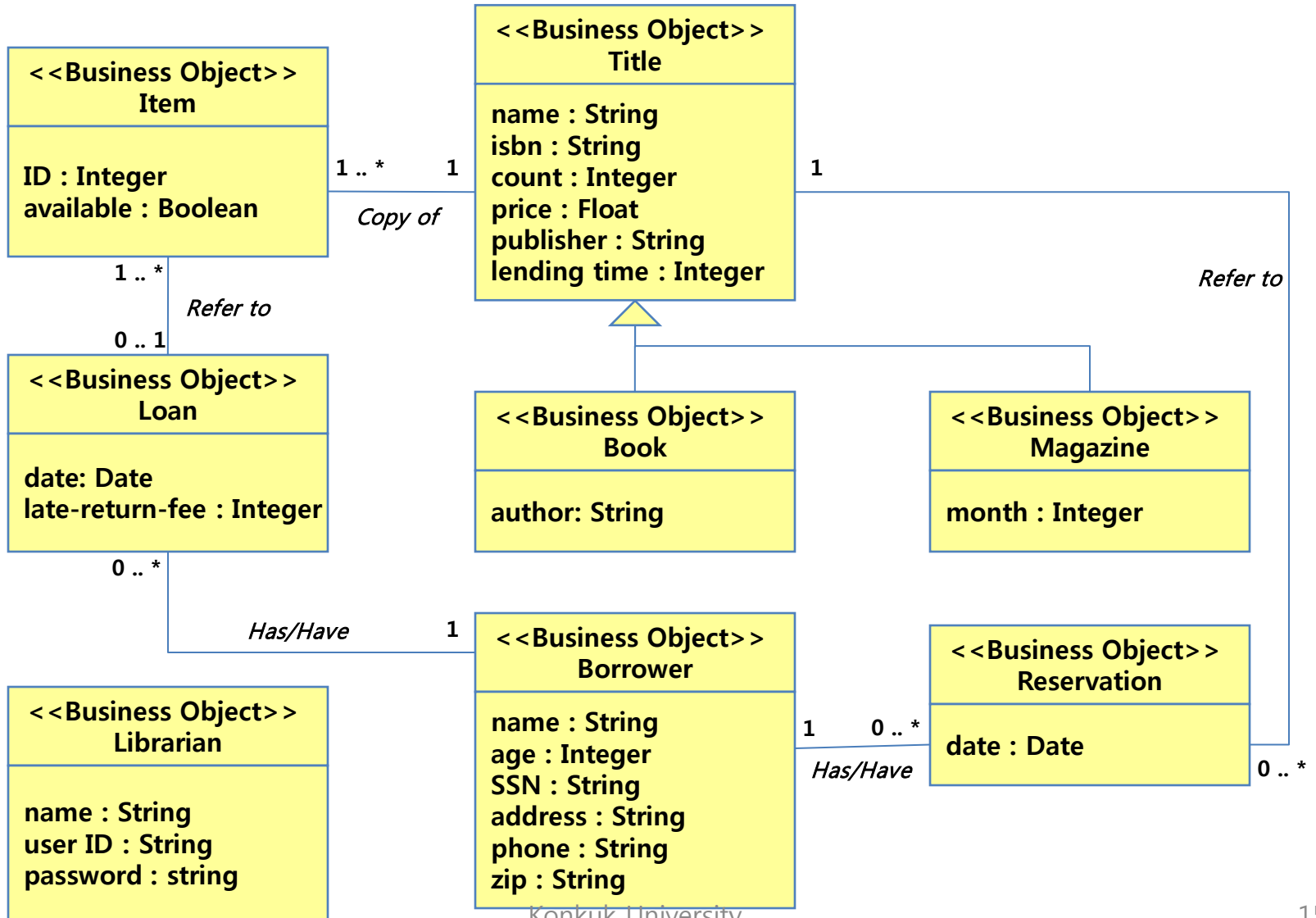
| | |
|--------------------------------------|--|
| Use Case | 7. Notify Availability |
| Actor | None |
| Purpose | Notify availability of a reserved item |
| Overview | (As in the business use case) |
| Type | Primary and Real |
| Cross Reference | System Functions: R1.4.1, R1.6, R2.4 Use Cases: "Return Item", "Add Item" |
| Pre-Requisites | An item should have been returned or a new item should have been added. |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (S) Print a post-card |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | N/A |

Activity 2041. Design Real Use Cases

8. Add Title

| | |
|--------------------------------------|---|
| Use Case | 8. Add Title |
| Actor | Librarian |
| Purpose | Register a new title |
| Overview | (As in the business use case) |
| Type | Primary and Real |
| Cross Reference | System Functions: R2.1, R2.4 Use Case: "Add Item" |
| Pre-Requisites | N/A |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (A) A librarian inputs title information such as name, isbn, price, publisher, loanPeriod (Book: author, Magazine:month, publishCycle) 2. (S) Find a corresponding title 3. (S) Create a new title 4. (S) Store the new title 5. (S) Invoke "Add Item" |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | Line 1: If the title already exists, display an error message. |

Activity 2033. Define Domain Model



Activity 2041. Design Real Use Cases

9. Remove Title

| | |
|--------------------------------------|---|
| Use Case | 9. Remove Title |
| Actor | Librarian |
| Purpose | Delete information of a title |
| Overview | (As in the business use case) |
| Type | Primary and Real |
| Cross Reference | System Functions: R2.2 Use Case: - |
| Pre-Requisites | N/A |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (A) A librarian inputs a title's <i>isbn</i> to remove 2. (S) Find a corresponding title 3. (S) Check if the corresponding title is reserved. 4. (S) If the title is reserved, Remove the reservation 5. (S) Check the item of the title is loaned. (and then what about items?) 6. (S) Remove the title |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | Line 2: If the title does not exist, display an error message. |

Activity 2041. Design Real Use Cases

10. Update Title

| | |
|--------------------------------------|--|
| Use Case | 10. Update Title |
| Actor | Librarian |
| Purpose | Update information of a title |
| Overview | (As in the business use case) |
| Type | Primary and Real |
| Cross Reference | System Functions: R2.3 Use Case: - |
| Pre-Requisites | N/A |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (A) A librarian inputs a title's <i>isbn</i> and information of the title to change 2. (S) Find a corresponding title 3. (S) Update the title (How?) |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | Line 2: If the item does not exist, display "Not Existing Title". Error message. Line 3: If the <i>isbn</i> is changed, then update items too, (How?) |

Activity 2041. Design Real Use Cases

11. Add Item

| | |
|--------------------------------------|---|
| Use Case | 11. Add Item |
| Actor | Librarian |
| Purpose | Add a new item |
| Overview | (As in the business use case) |
| Type | Primary and Real |
| Cross Reference | System Functions: R2.4 Use Cases: "Add Title" |
| Pre-Requisites | N/A |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (A) A librarian inputs an item's <i>id</i> 2. (S) Find a corresponding title 3. (S) Get an item's <i>ID</i> from the title 4. (S) Create a new item 5. (S) Store the new item 6. (S) Increase <i>numOfItem</i> of the title (?) 7. (S) Increase <i>availablecount</i> of the item |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | Line 2: Line 2: If the title does not exist, display an error message. |

Activity 2041. Design Real Use Cases

12. Remove Item

| | |
|--------------------------------------|--|
| Use Case | 12. Remove Item |
| Actor | Librarian |
| Purpose | Remove information of an item |
| Overview | (As in the business use case) |
| Type | Primary and Real |
| Cross Reference | System Functions: R2.1, R2.5 Use Case: "Remove Title" |
| Pre-Requisites | N/A |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (A) A librarian inputs an item's <i>ID</i> 2. (S) Find a corresponding item 3. (S) Check if the item is borrowed 4. (S) If the item is borrowed, decrease <i>numOfItem</i> of the title 5. (S) Decrease <i>availableCount</i> of the title 6. (S) Remove the item |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | Line 2: If the item does not exist, display an error message. Line 3: If the item was already borrowed, display an error message |

Activity 2041. Design Real Use Cases

13. Update Item

| | |
|--------------------------------------|--|
| Use Case | 13. Update Item |
| Actor | Librarian |
| Purpose | Update information of an item |
| Overview | (As in the business use case) |
| Type | Primary and Real |
| Cross Reference | System Functions: R2.6 Use Case: - |
| Pre-Requisites | N/A |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (A) A librarian inputs the item's <i>id</i> and information to change 2. (S) Find A corresponding item 3. (S) Update the item 4. (S) If a lost of the item is true, decrease <i>numOfItem</i> of the title. 5. (S) Decrease the availableCount of the title. 6. (S) If a lost of the item is false, increase <i>numOfItem</i> of the title. (What? Only for these cases "Update Item" are used?) 7. (S) Increase <i>availableCount</i> of the title |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | Line 2: If the item does not exist, display an error message. |

Activity 2041. Design Real Use Cases

14. Add Borrower

| | |
|--------------------------------------|--|
| Use Case | 14. Add Borrower |
| Actor | Librarian |
| Purpose | Register a new borrower |
| Overview | (As in the business use case) |
| Type | Primary and Real |
| Cross Reference | System Functions: R1.1, R1.3, R3.1 Use Cases: "Make Reservation", "Lend Item" |
| Pre-Requisites | N/A |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (A) A librarian inputs a borrower's name, ssn, and address. 2. (S) Find a corresponding borrower 3. (S) Create a new borrower 4. (S) Store the new borrower |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | Line 2: If the borrower exists already, display an error message. |

Activity 2041. Design Real Use Cases

15. Remove Borrower

| | |
|--------------------------------------|--|
| Use Case | 15. Remove Borrower |
| Actor | Librarian |
| Purpose | Remove information of a borrower |
| Overview | (As in the business use case) |
| Type | Primary and Real |
| Cross Reference | System Functions: R3.2 Use Case: - |
| Pre-Requisites | N/A |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (A) A librarian inputs the borrower's <i>ssn</i> 2. (S) Find a corresponding borrower 3. (S) Find a loan of the borrower 4. (S) If the loan is invalid, find a reservation 5. (S) Get the title of the reservation 6. (S) Decrease <i>reservationCount</i> of the title 7. (S) Remove borrower |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | Line 2: If the borrower does not exist, display an error message. Line 3: If the loan is still valid, display an error message. |

Activity 2041. Design Real Use Cases

16. Update Borrower

| | |
|--------------------------------------|--|
| Use Case | 16. Update Borrower |
| Actor | Librarian |
| Purpose | Update information of a borrower |
| Overview | (As in the business use case) |
| Type | Primary and Real |
| Cross Reference | System Functions: R3.3 Use Case: - |
| Pre-Requisites | N/A |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (A) A librarian inputs a borrower's <i>ssn</i> and information to change 2. (S) Find a corresponding borrower 3. (S) Update the borrower |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | Line 2: If the borrower does not exist, display an error message. |

Activity 2041. Design Real Use Cases

17. Log-In

| | |
|--------------------------------------|--|
| Use Case | 17. Log-In |
| Actor | Librarian |
| Purpose | Check access authority of a librarian |
| Overview | (As in the business use case) |
| Type | Secondary and Real |
| Cross Reference | System Functions: R4.1 Use Case: - |
| Pre-Requisites | A librarian should have user name and password. |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (A) A librarian inputs an <i>userID</i> and <i>password</i> 2. (S) Check if the <i>userID</i> and <i>password</i> are correct |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | Line 2: If the <i>userID</i> and <i>password</i> are not correct, display an error message. |

Activity 2041. Design Real Use Cases

18. Log-Out

| | |
|--------------------------------------|---|
| Use Case | 18. Log-Out |
| Actor | Librarian |
| Purpose | Exit the library management system |
| Overview | (As in the business use case) |
| Type | Secondary and Essential |
| Cross Reference | System Functions: R4.1 Use Case: - |
| Pre-Requisites | A librarian should have user name and password. |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (A) A librarian selects "LogOut" 2. (S) Check if the <i>userID</i> is correct and then exit the system |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | Line 2: If the <i>userID</i> is incorrect, display an error message. |

Activity 2041. Design Real Use Cases

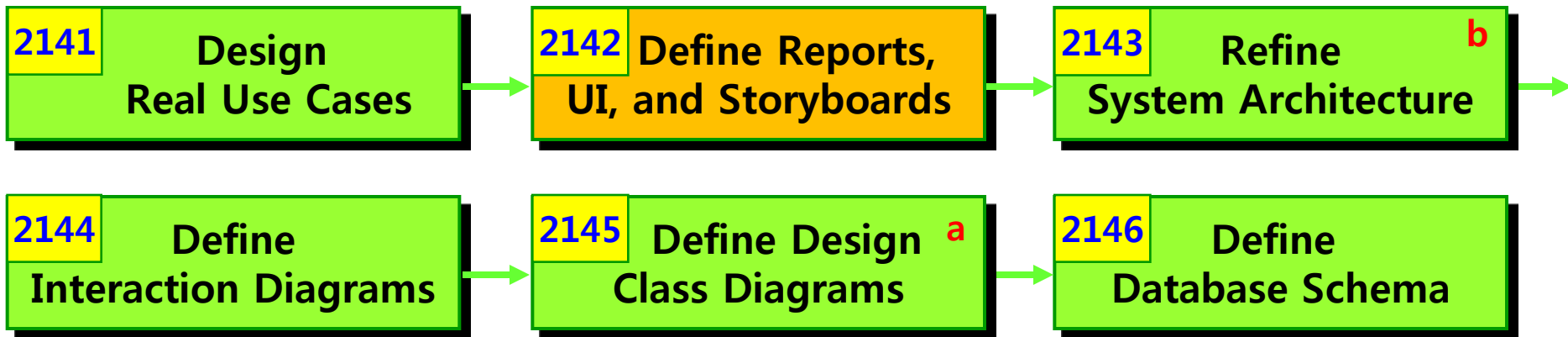
19. Count Loans

| | |
|--------------------------------------|--|
| Use Case | 19. Count Loans |
| Actor | Librarian |
| Purpose | Compute total count of the titles checked out |
| Overview | (As in the business use case) |
| Type | Secondary and Essential |
| Cross Reference | System Functions: R5.1 Use Case: - |
| Pre-Requisites | A librarian should have user name and password. |
| Typical Courses of Events | (A) : Actor, (S) : System 1. (A) A librarian requests loan count 2. (S) Get <i>numOfLoan</i> of the loan |
| Alternative Courses of Events | N/A |
| Exceptional Courses of Events | N/A (Really?) |

Activity 2042.

Define Reports, UI, and Storyboards

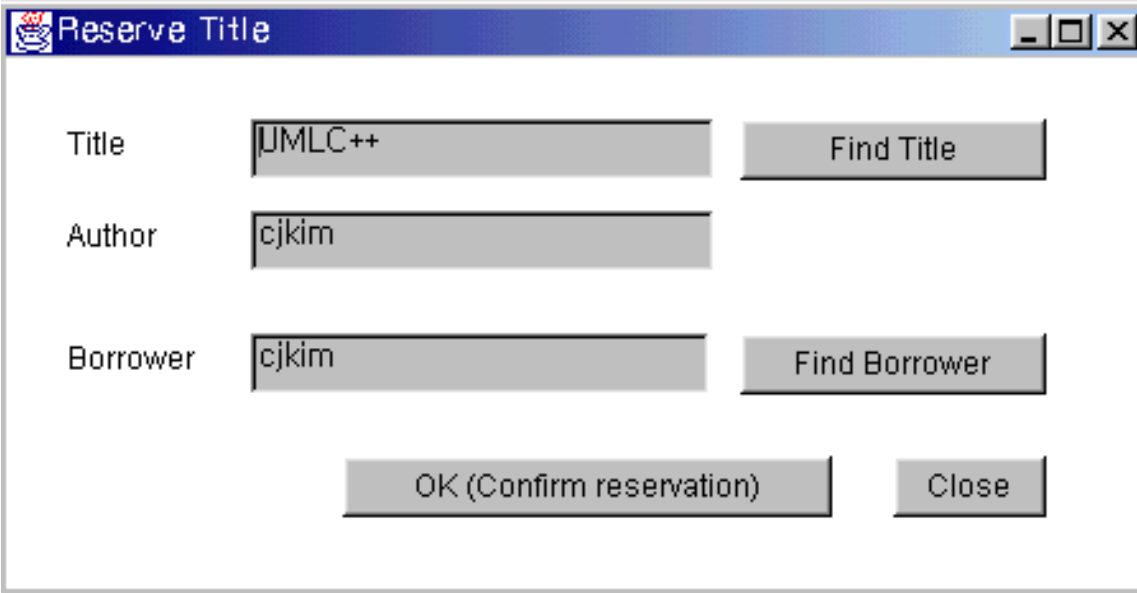
- a. In parallel with interaction diagrams
- b. Varied order



Activity 2042.

Define Reports, UI, and Storyboards

- Make Reservation



The image shows a screenshot of a software dialog box titled "Reserve Title". The dialog box has a blue title bar with a small icon on the left and standard window control buttons (minimize, maximize, close) on the right. The main content area is white and contains three rows of input fields and buttons:

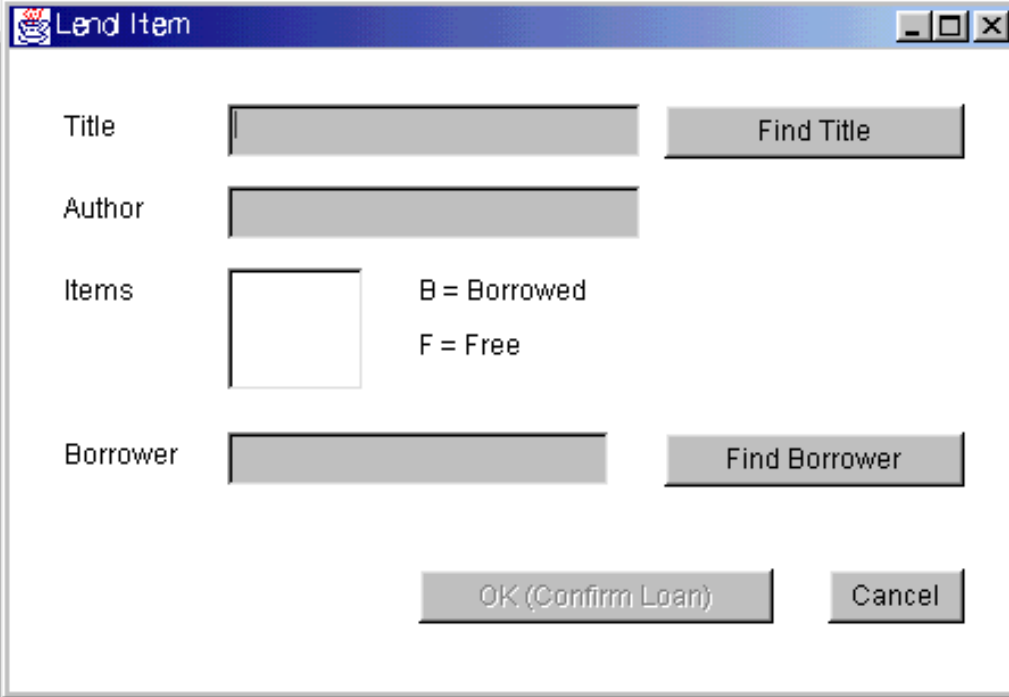
- The first row has a label "Title" on the left, a text input field containing "UMLC++", and a button labeled "Find Title" to the right of the field.
- The second row has a label "Author" on the left, a text input field containing "ckim", and no button to its right.
- The third row has a label "Borrower" on the left, a text input field containing "ckim", and a button labeled "Find Borrower" to the right of the field.

At the bottom of the dialog box, there are two buttons: "OK (Confirm reservation)" on the left and "Close" on the right.

Activity 2042.

Define Reports, UI, and Storyboards

- Lent Item



The image shows a screenshot of a Windows-style dialog box titled "Lend Item". The dialog box contains the following elements:

- Title:** A text input field followed by a "Find Title" button.
- Author:** A text input field.
- Items:** A small square input field, with the text "B = Borrowed" and "F = Free" displayed to its right.
- Borrower:** A text input field followed by a "Find Borrower" button.
- Bottom:** Two buttons: "OK (Confirm Loan)" and "Cancel".

Activity 2042.

Define Reports, UI, and Storyboards

- Count Loans

The screenshot shows a 'Title Information' dialog box with the following fields and buttons:

- Title Name: UMLC++ (Find button)
- Author: cjkim (OK button)
- ISBN / Nr: 700630-1031410
- Type: Book

Below the search fields, there are two sections:

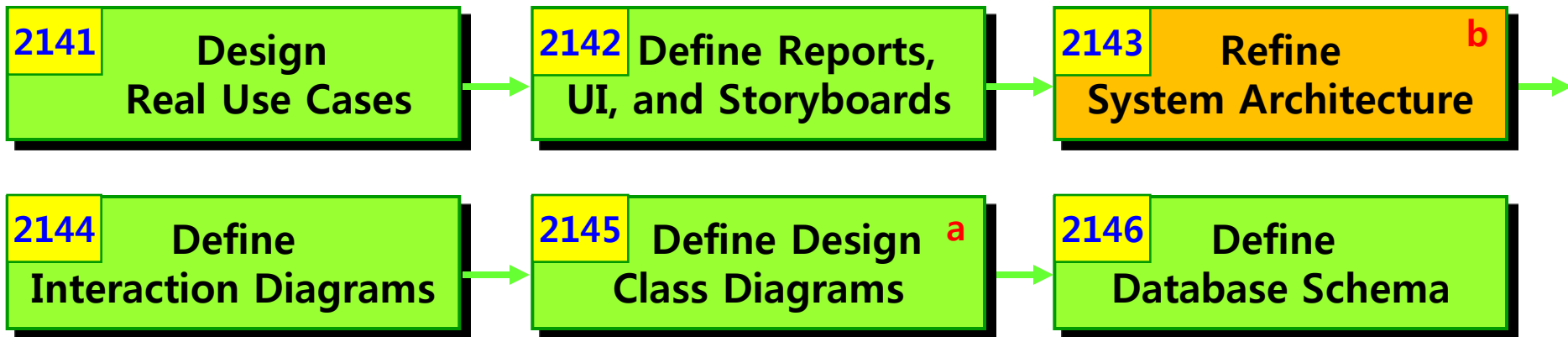
- Items Available:** (Item:1): Borrowed, (Item:2): Borrowed, (Item:3): Free
- Reservations:** 1111

At the bottom, there are two summary lines:

- Total : 3, Borrowed : 2, Free : 1
- Total : 1

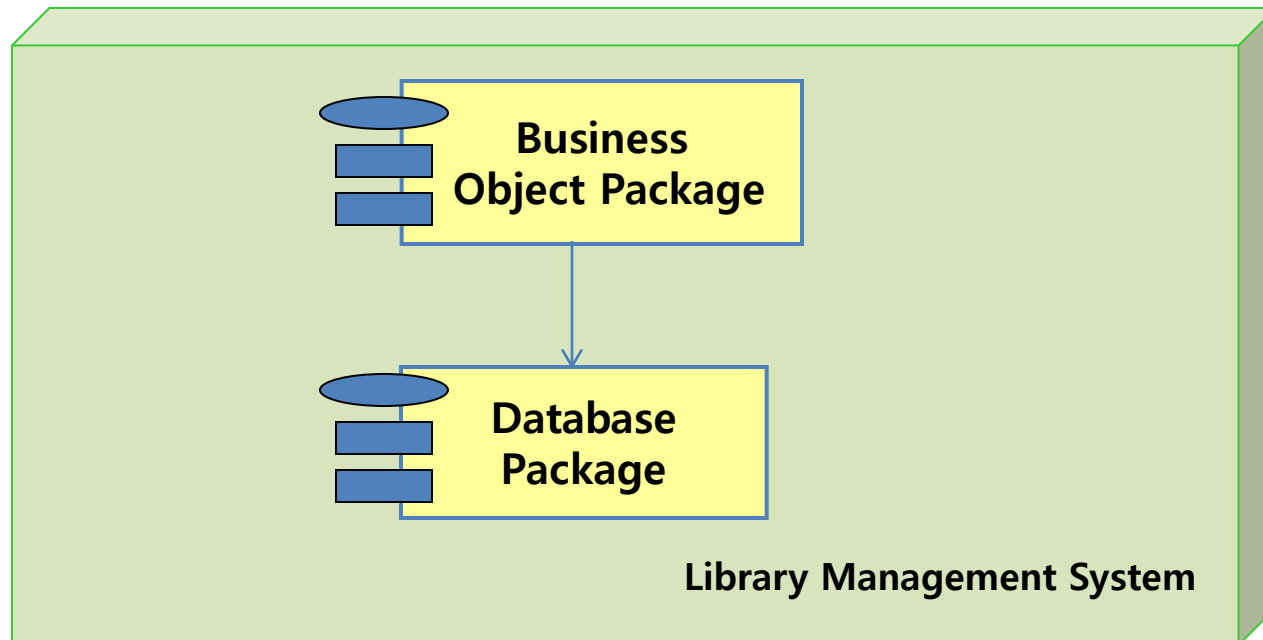
Activity 2043. Refine System Architecture

- a. In parallel with interaction diagrams
- b. Varied order



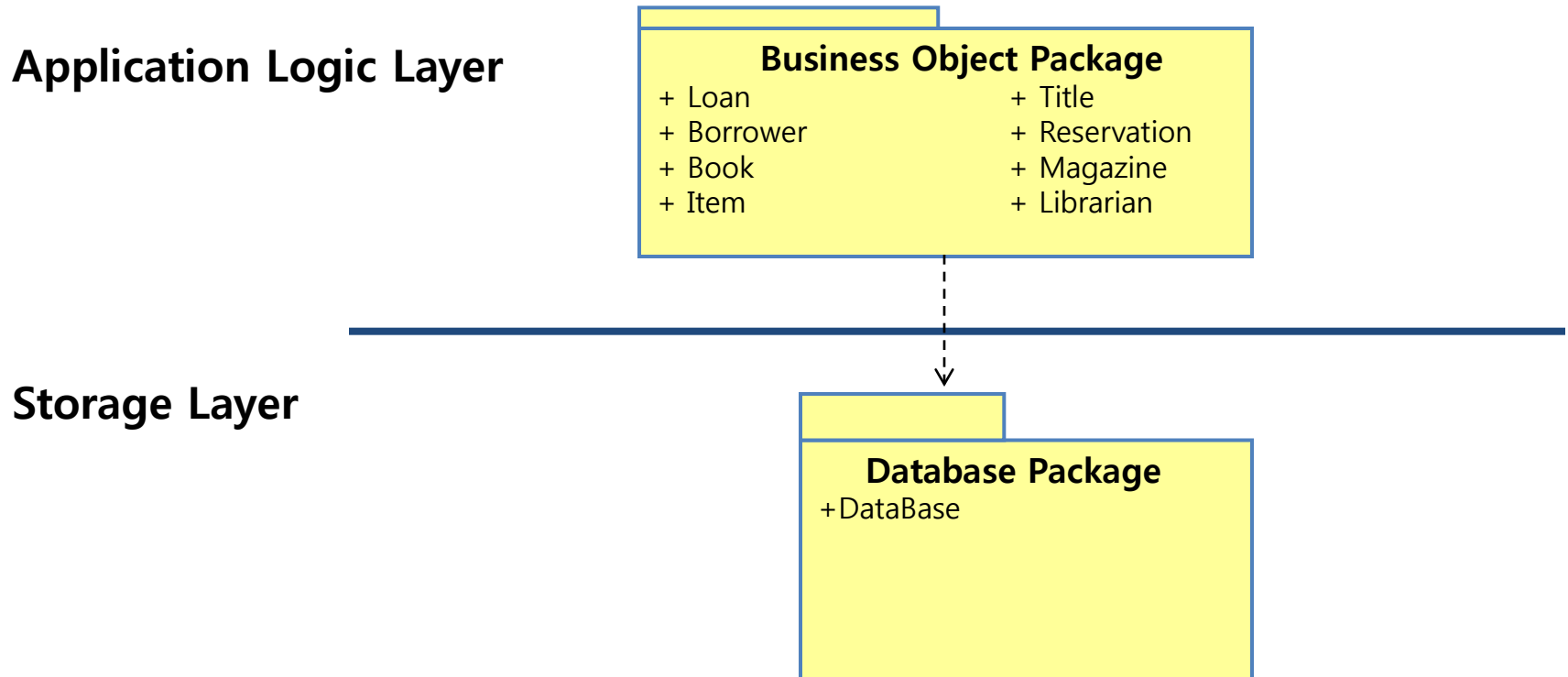
Activity 2043. Refine System Architecture

- Steps 1~3 : Drawing Deployment Diagram



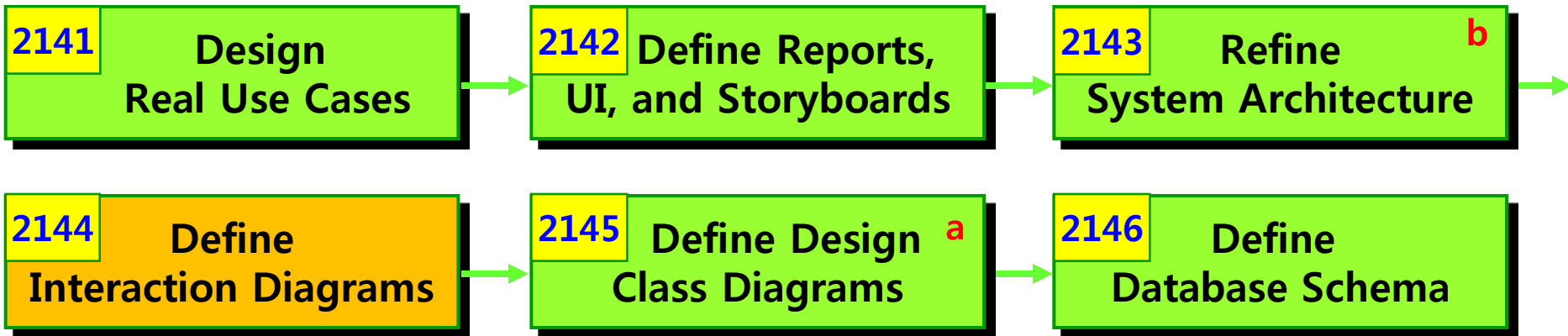
Activity 2043. Refine System Architecture

- Steps 4~7 : Drawing Package Diagram

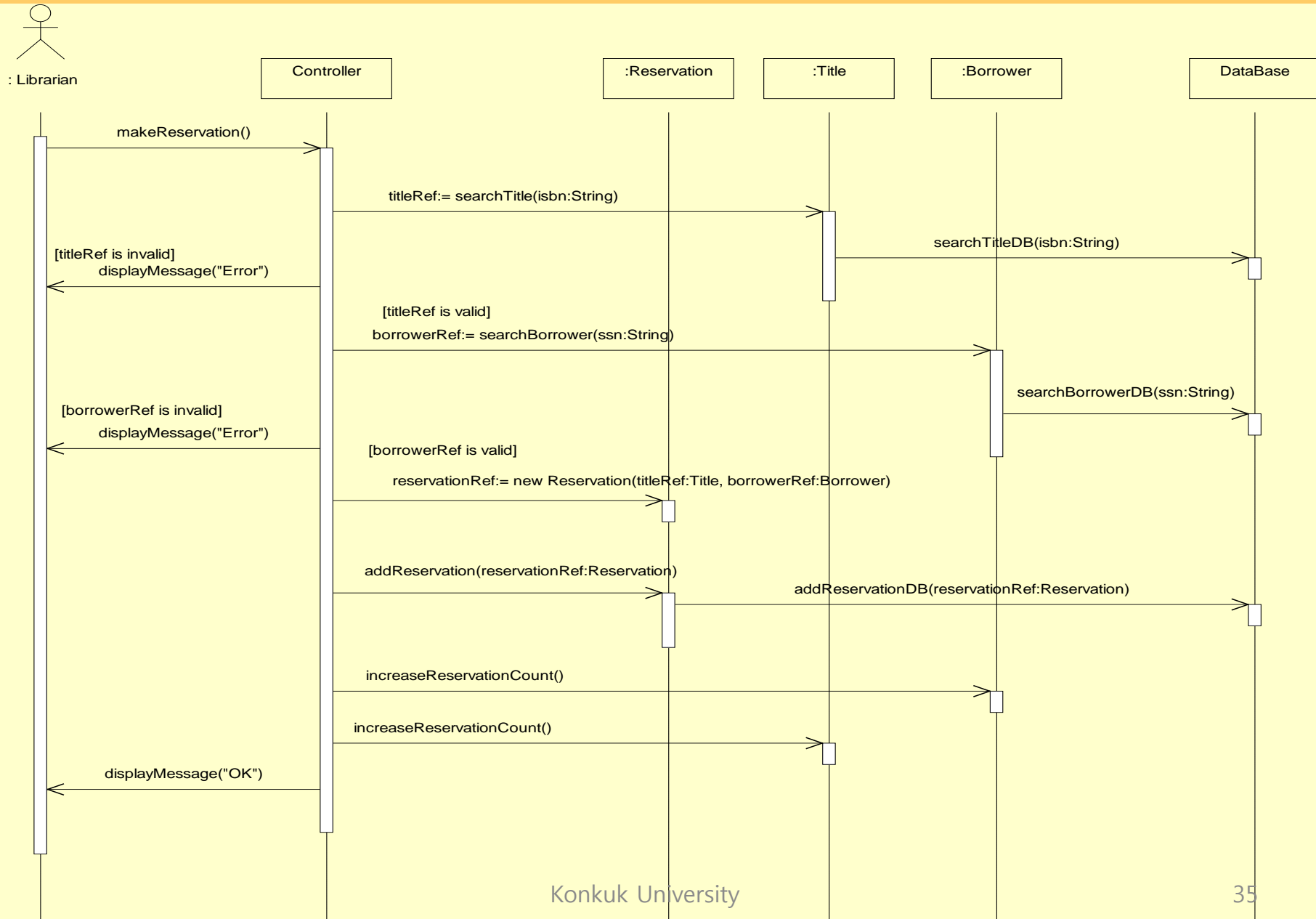


Activity 2044. Define Interaction Diagrams

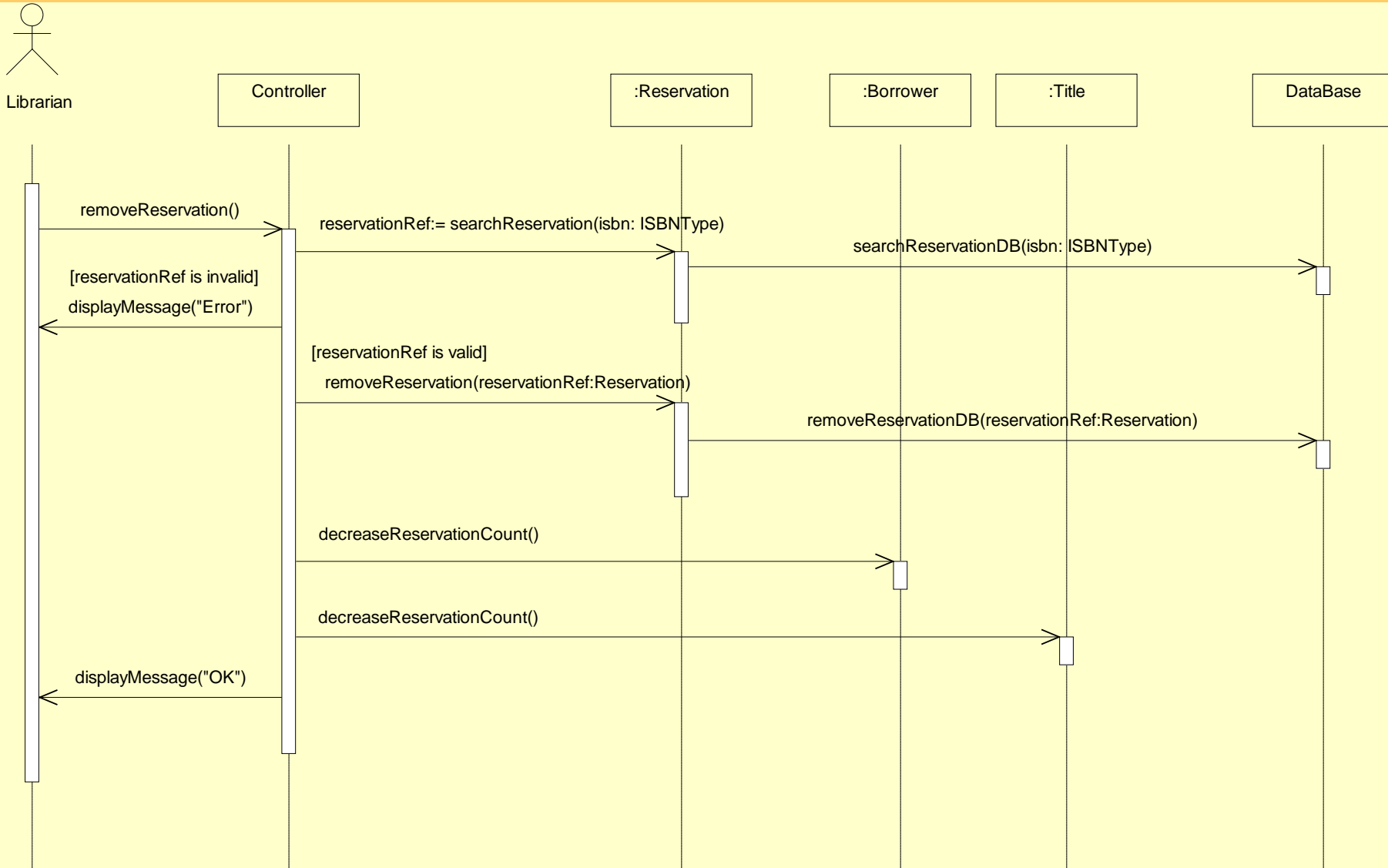
a. In parallel with interaction diagrams
b. Varied order



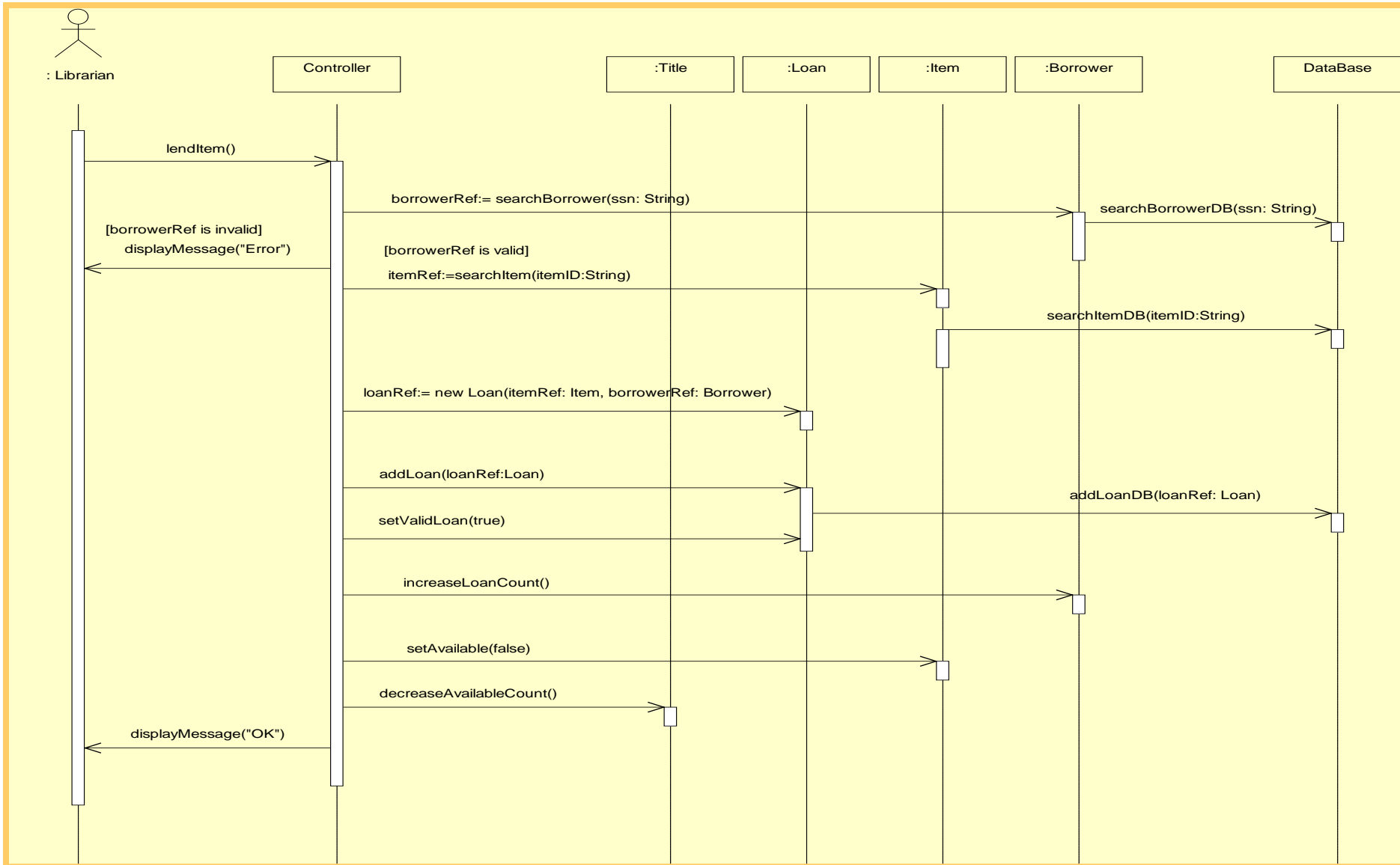
1. Make Reservation



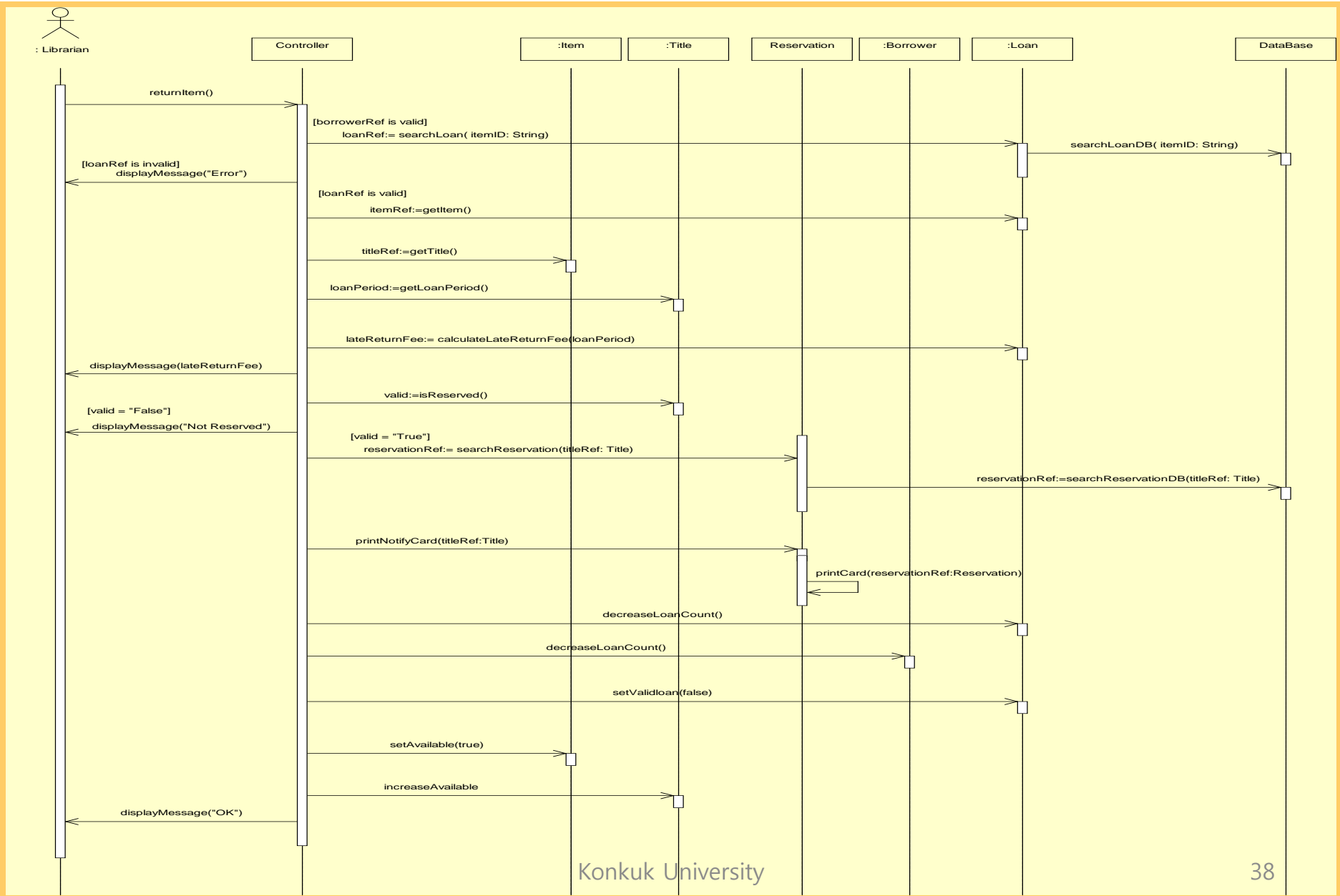
2. Remove Reservation



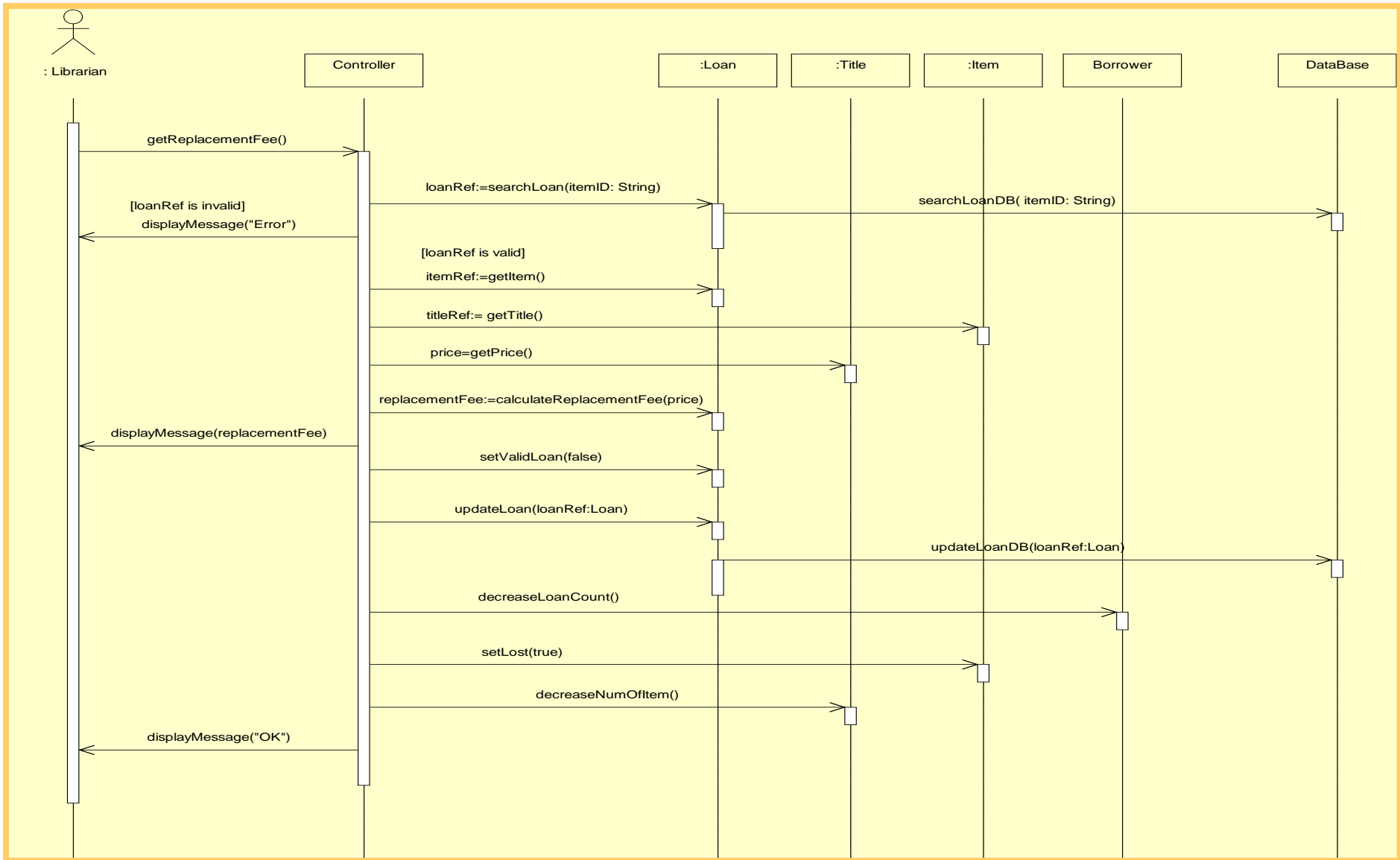
3. Lend Item



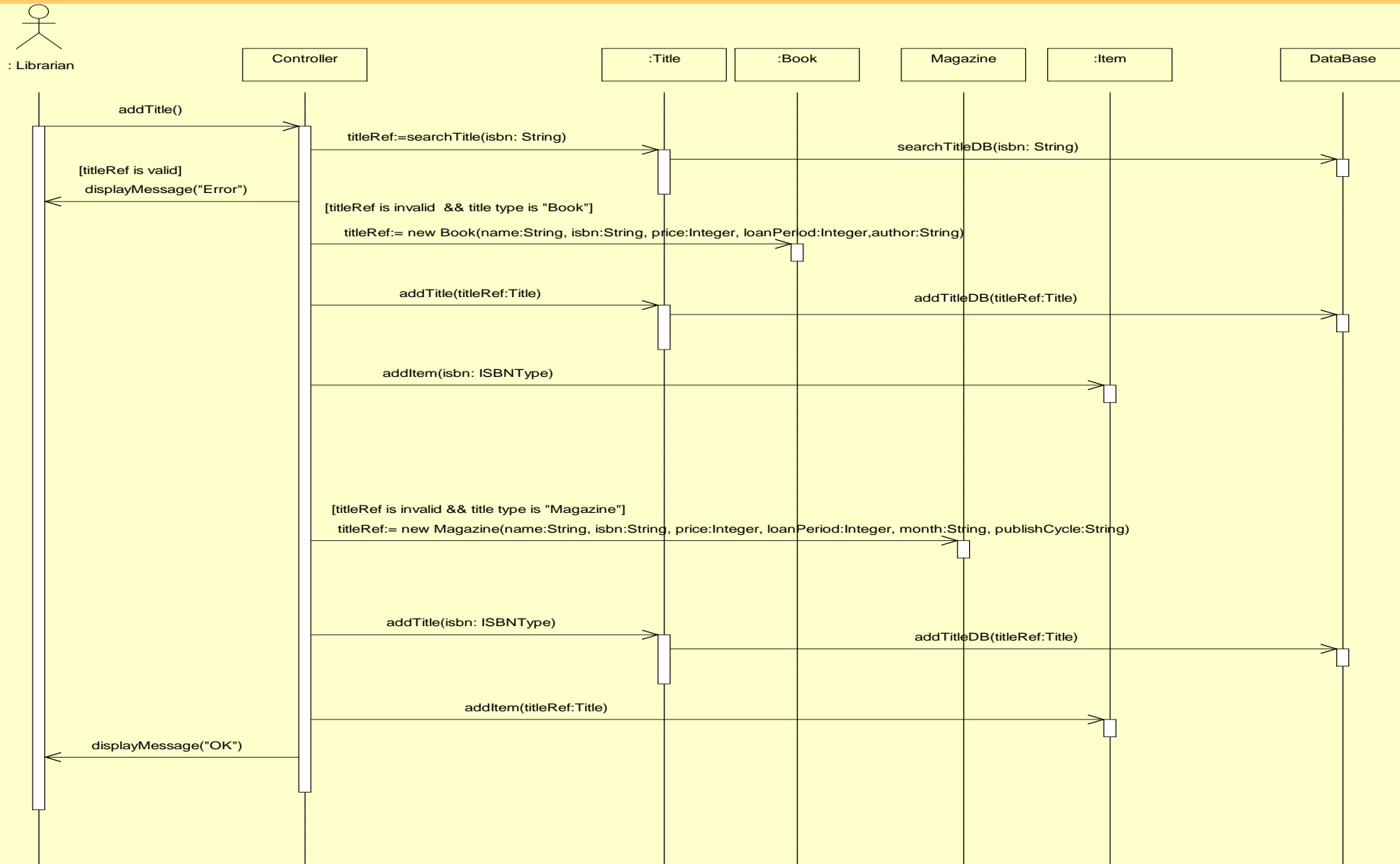
4. Return Item



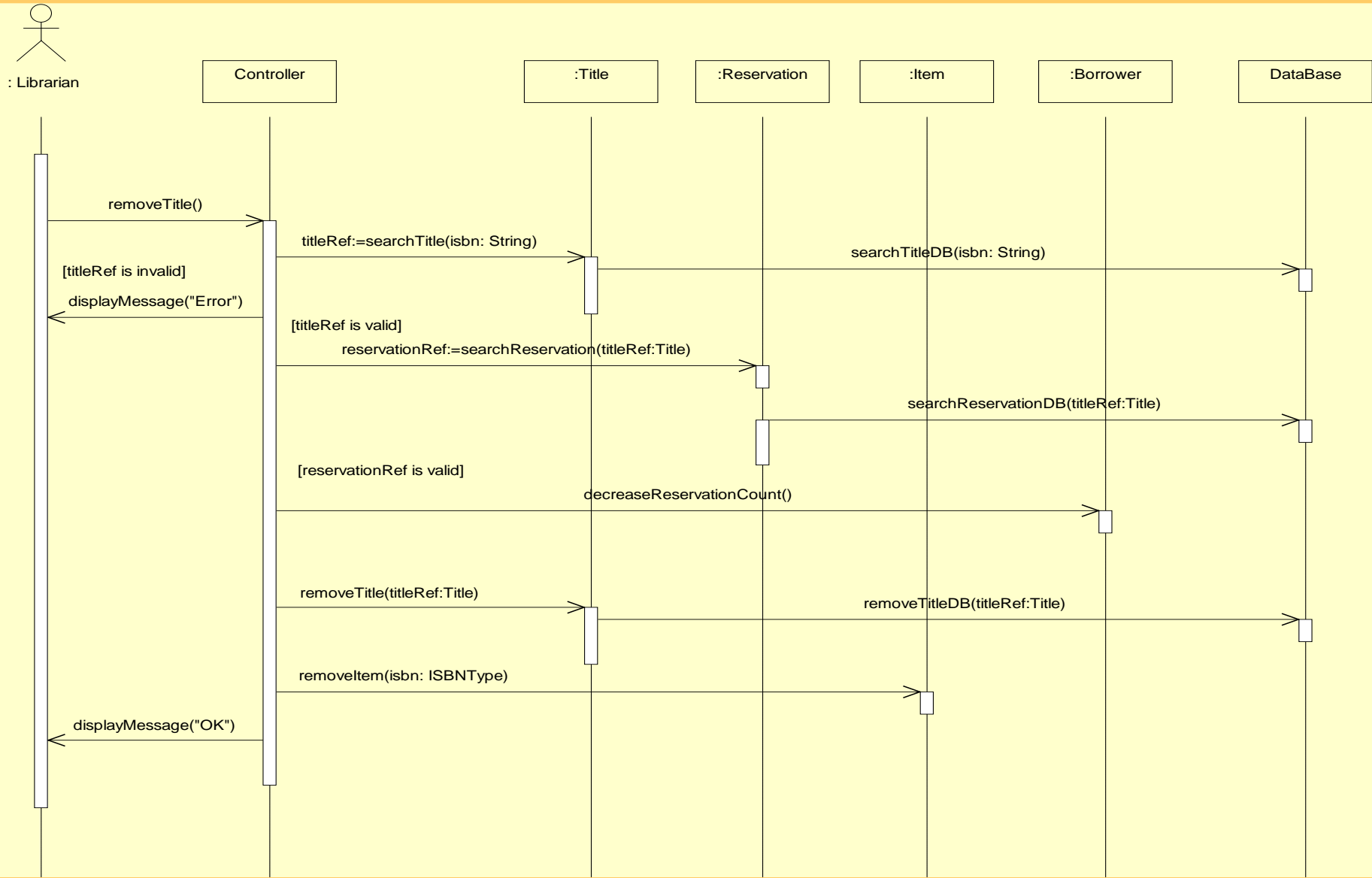
6. Get Replacement-Fee



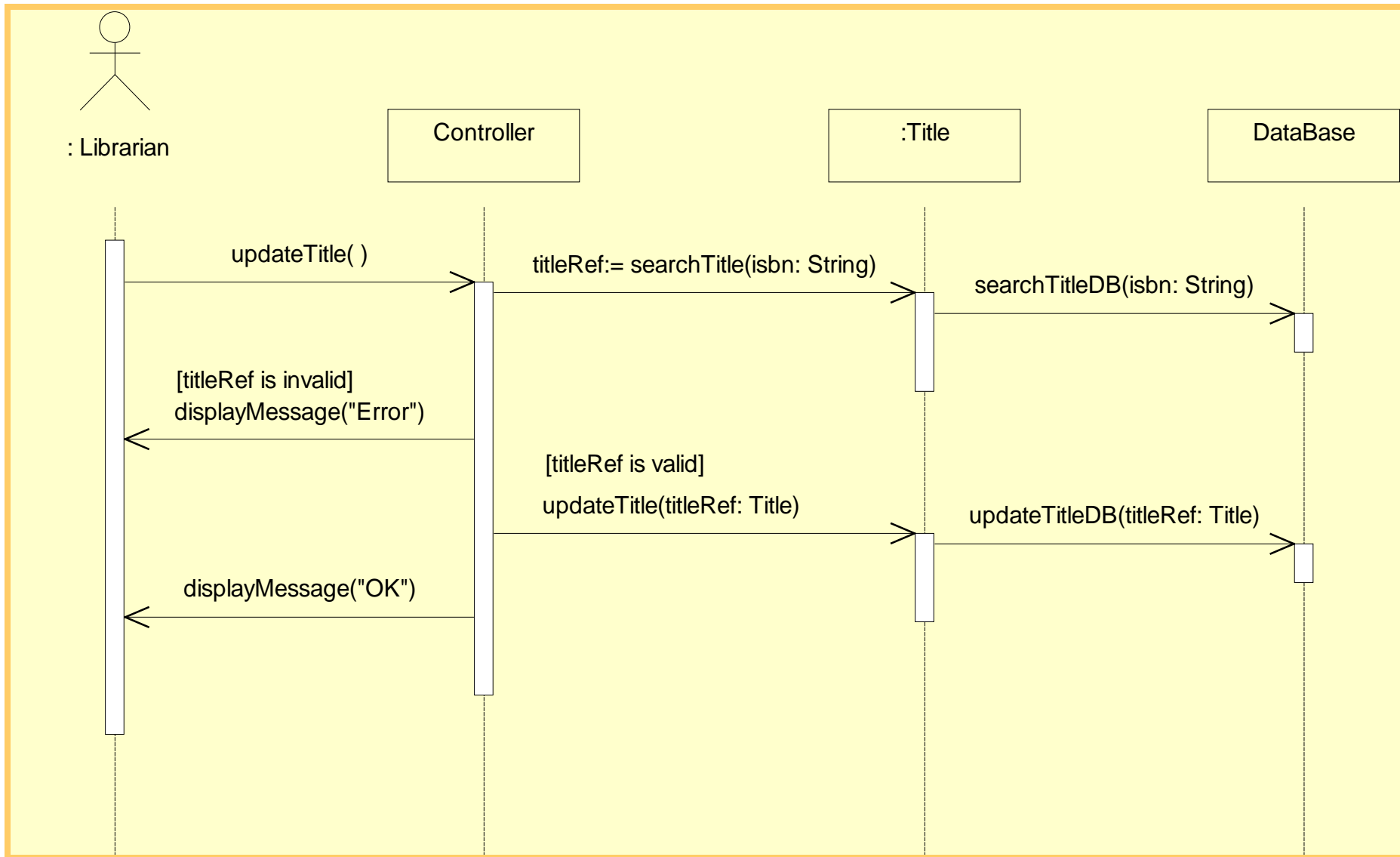
8. Add Title



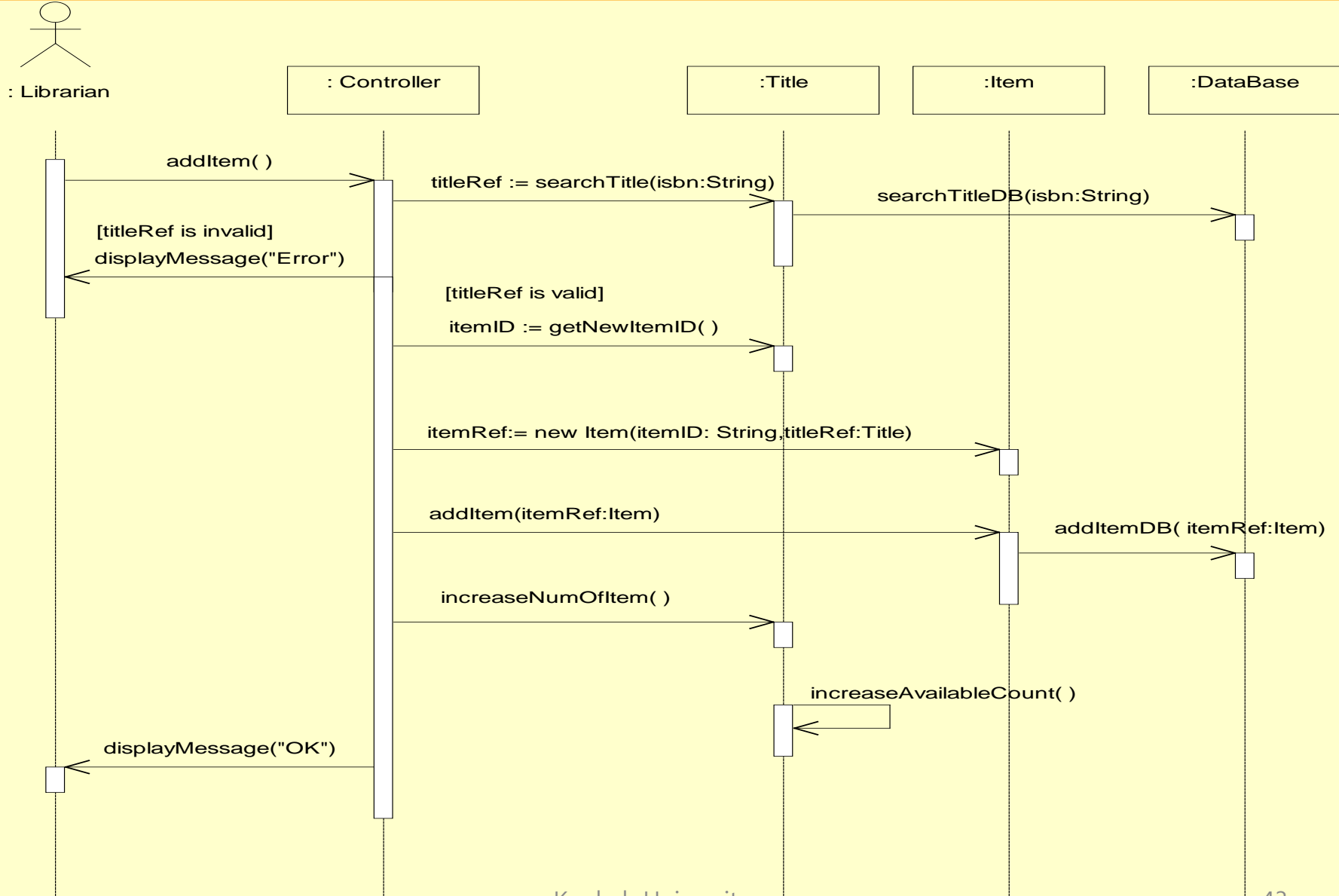
9. Remove Title



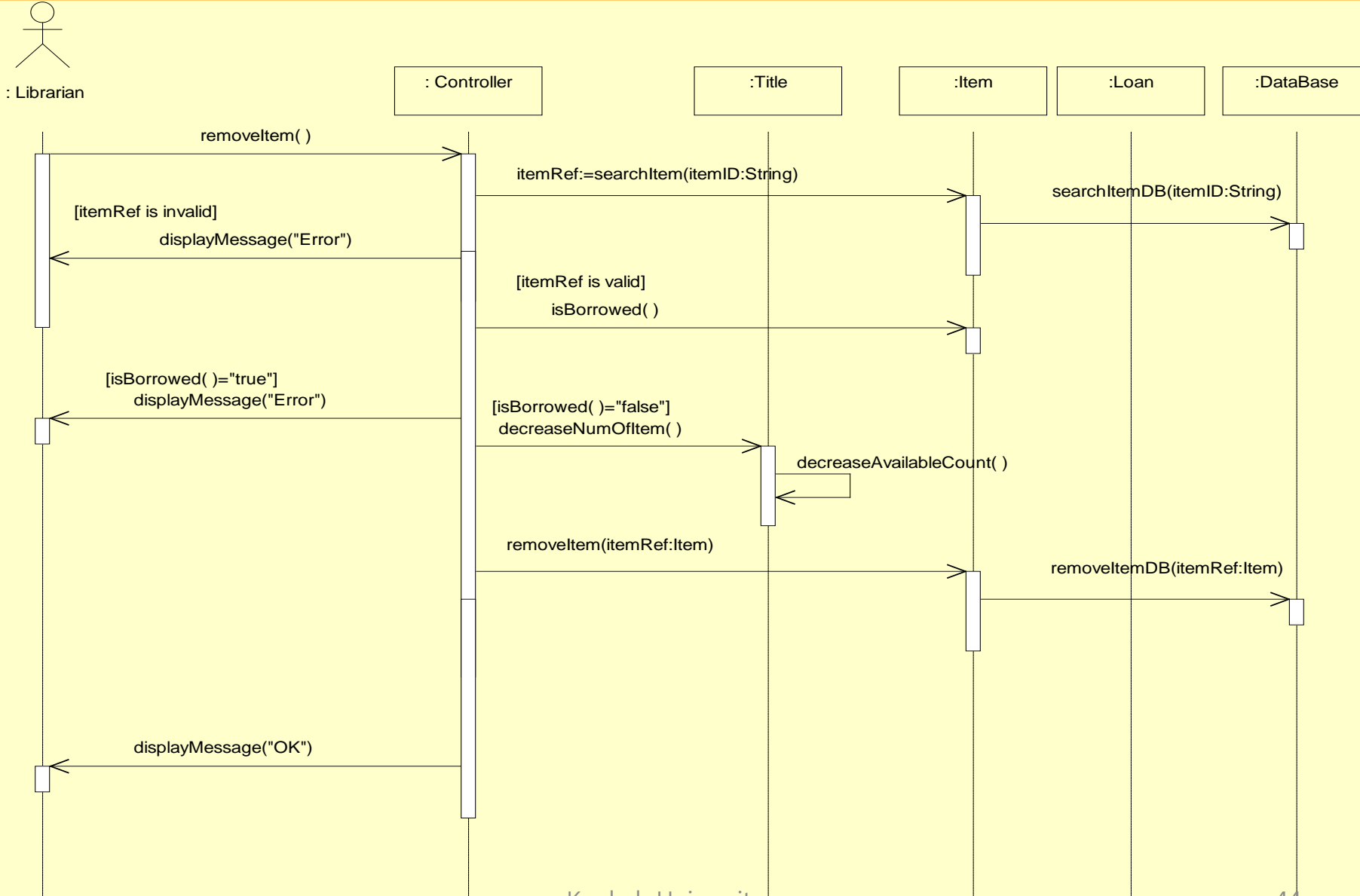
10. Update Title



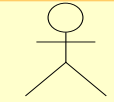
11. Add Item



12. Remove Item



12. Update Item



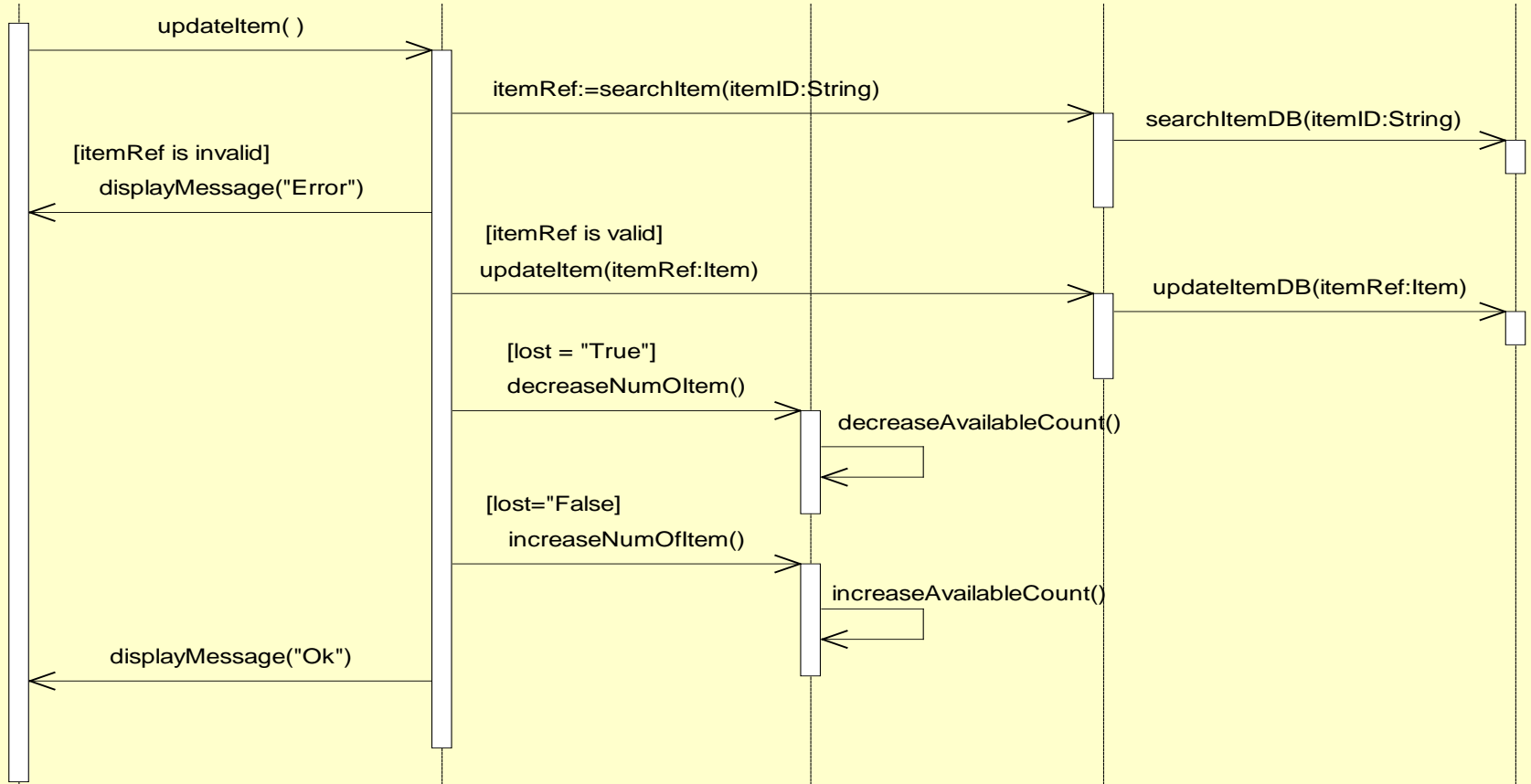
: Librarian

: Controller

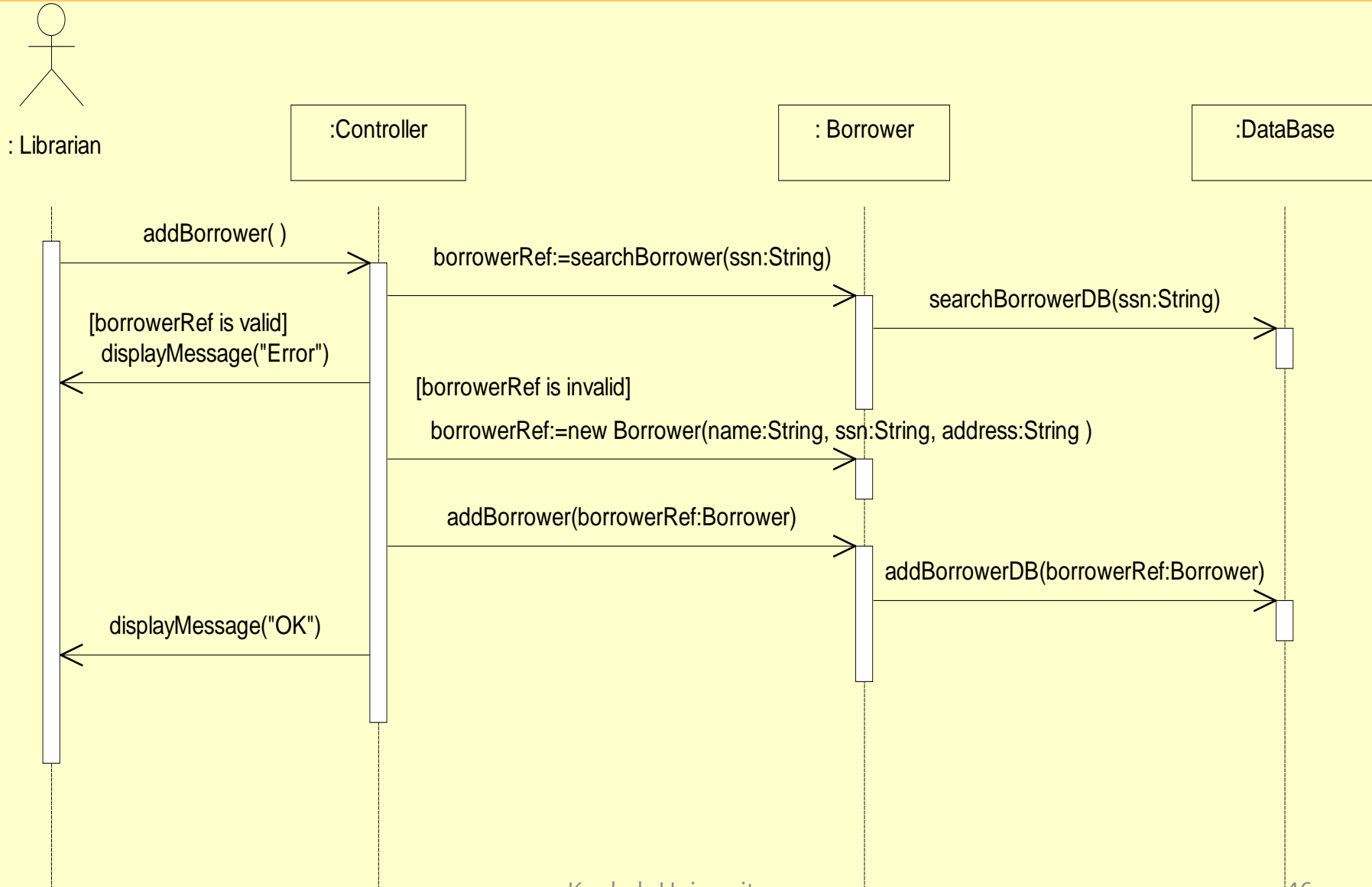
Title

Item

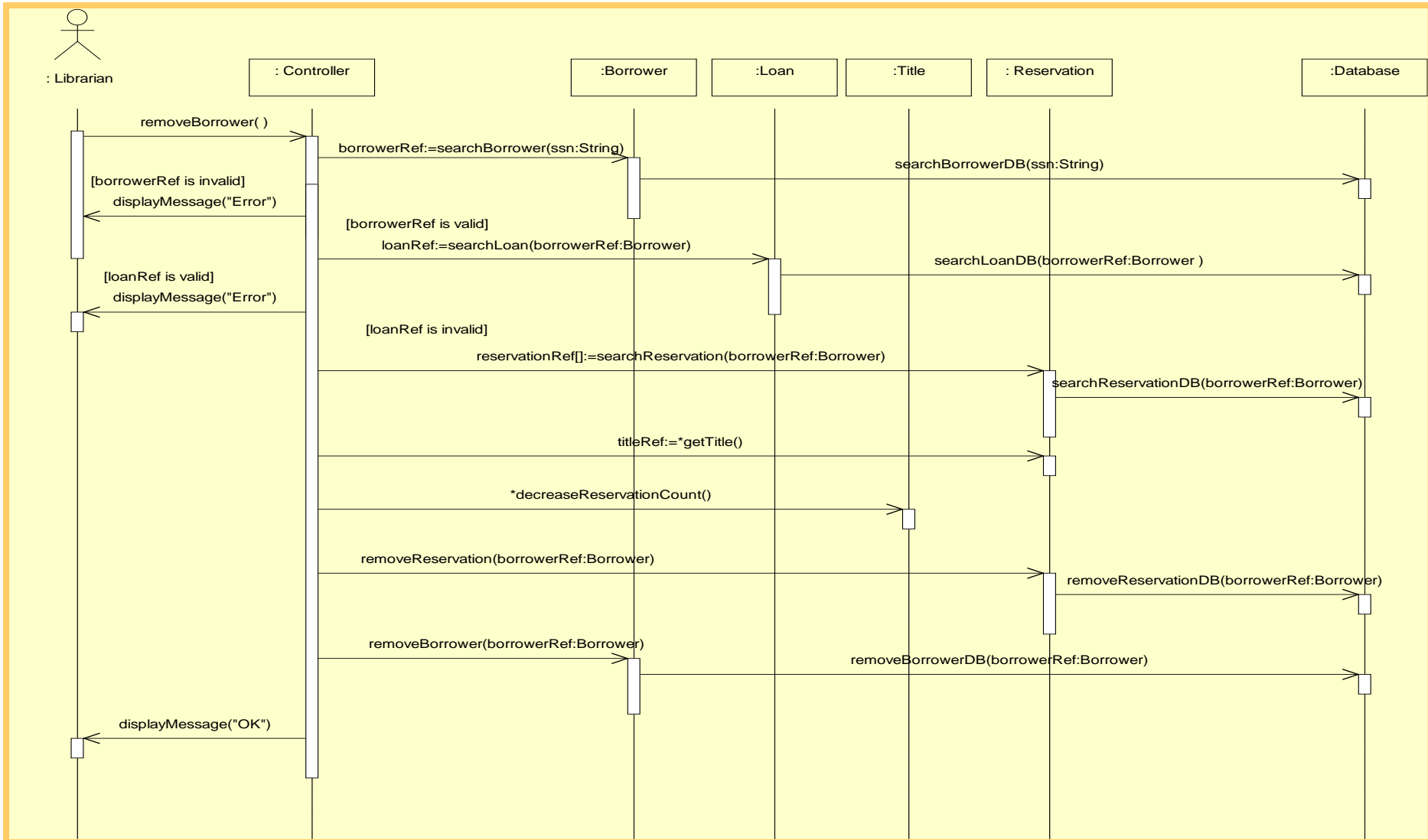
: DataBase



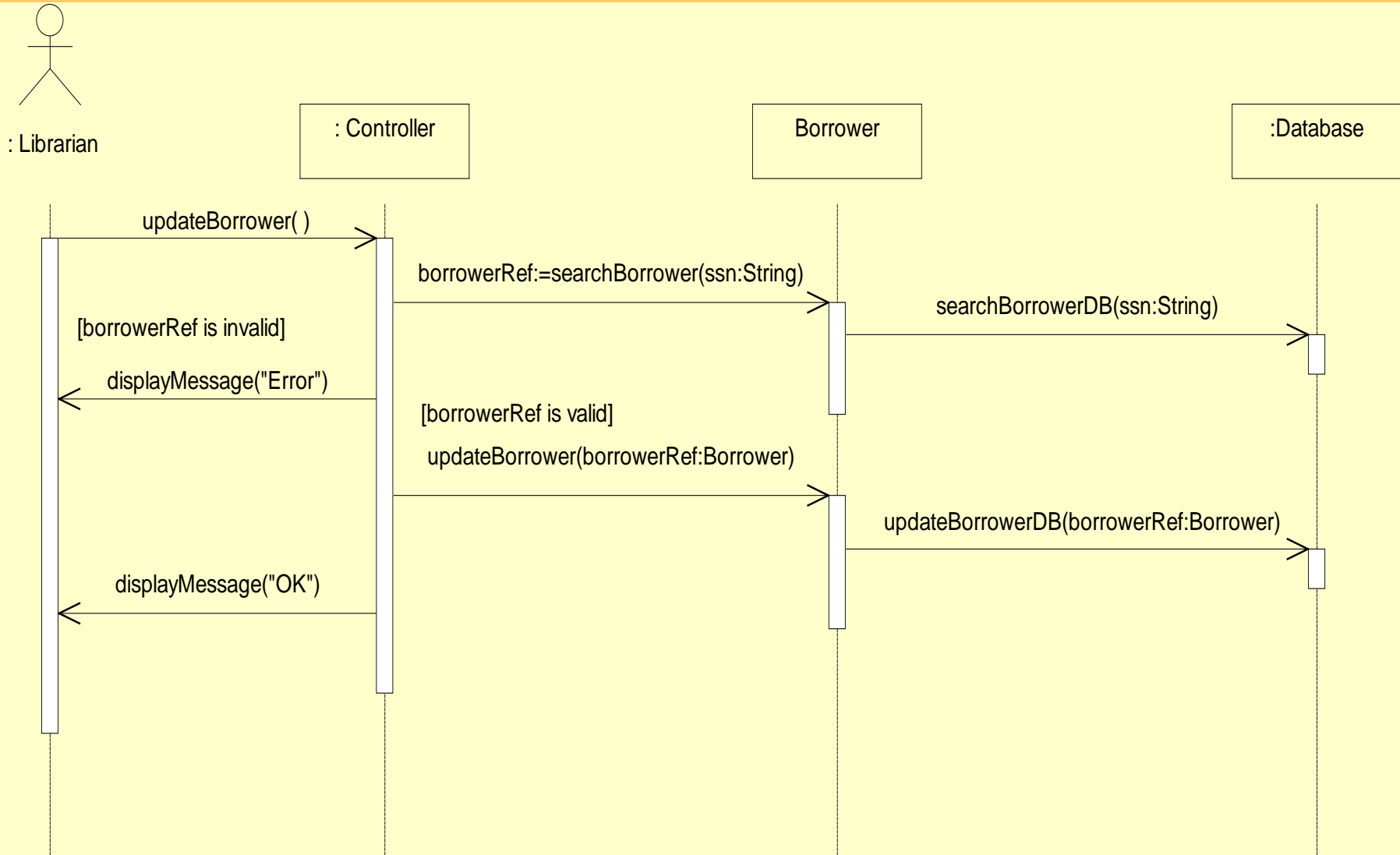
14. Add Borrower



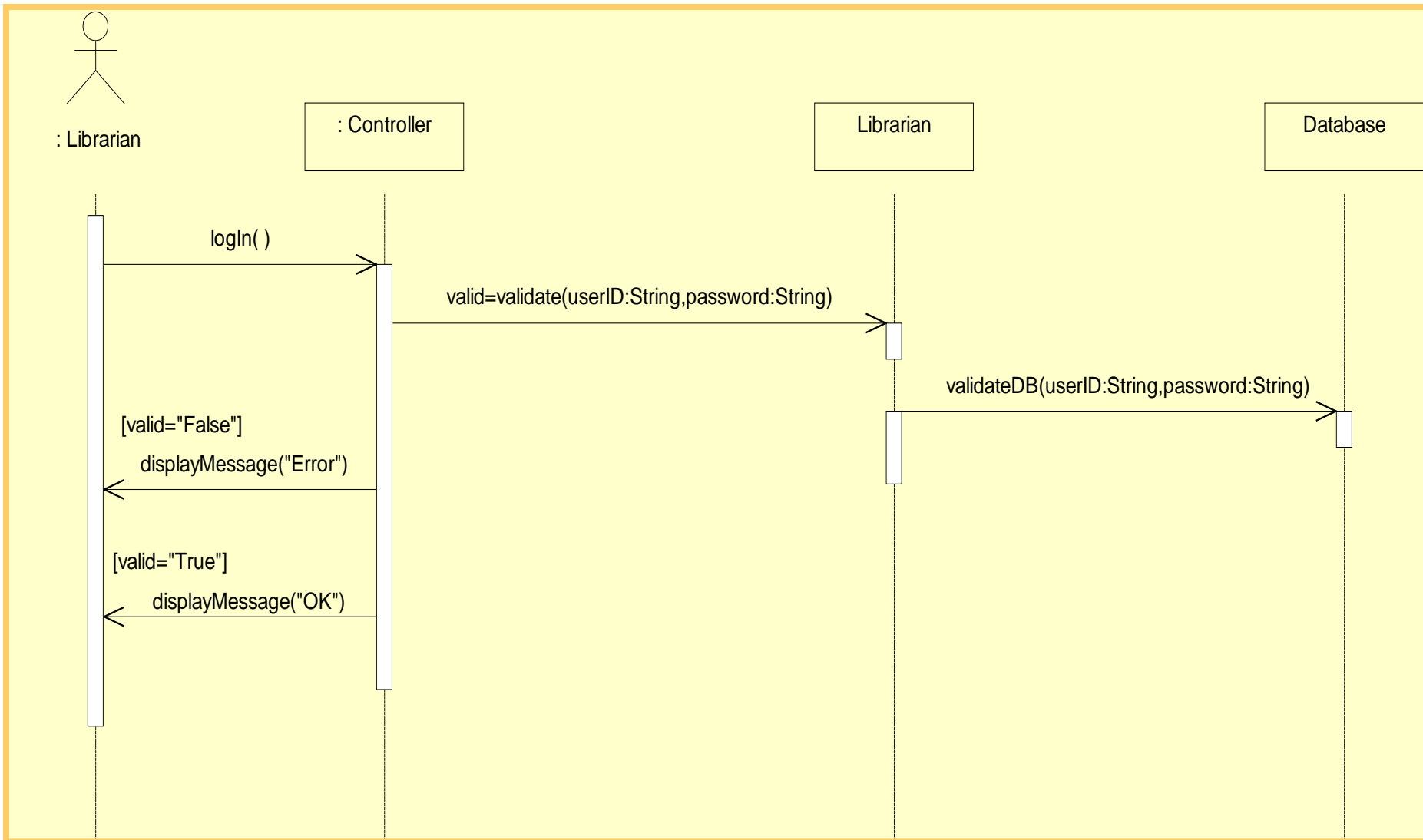
15. Remove Borrower



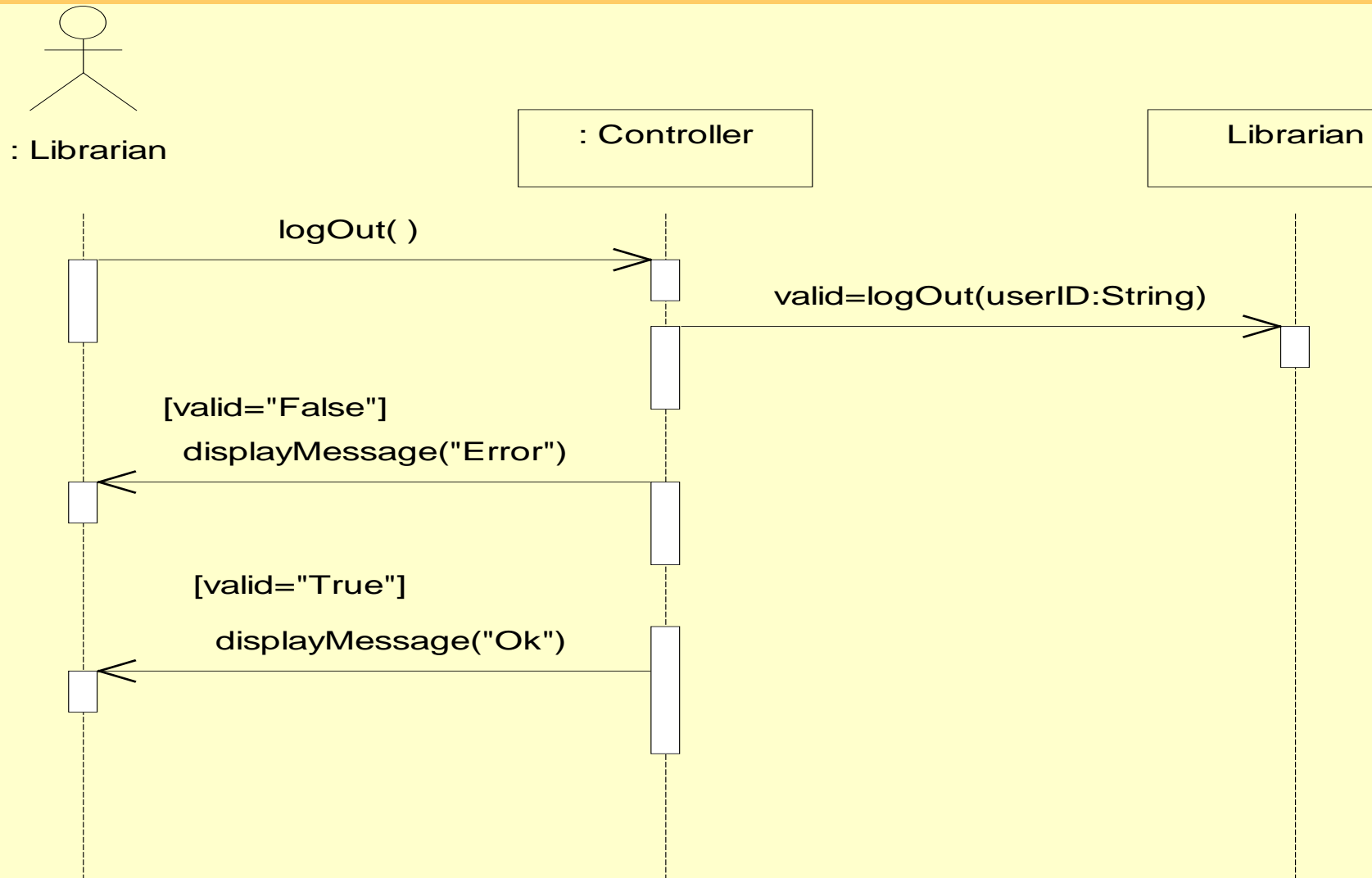
16. Update Borrower



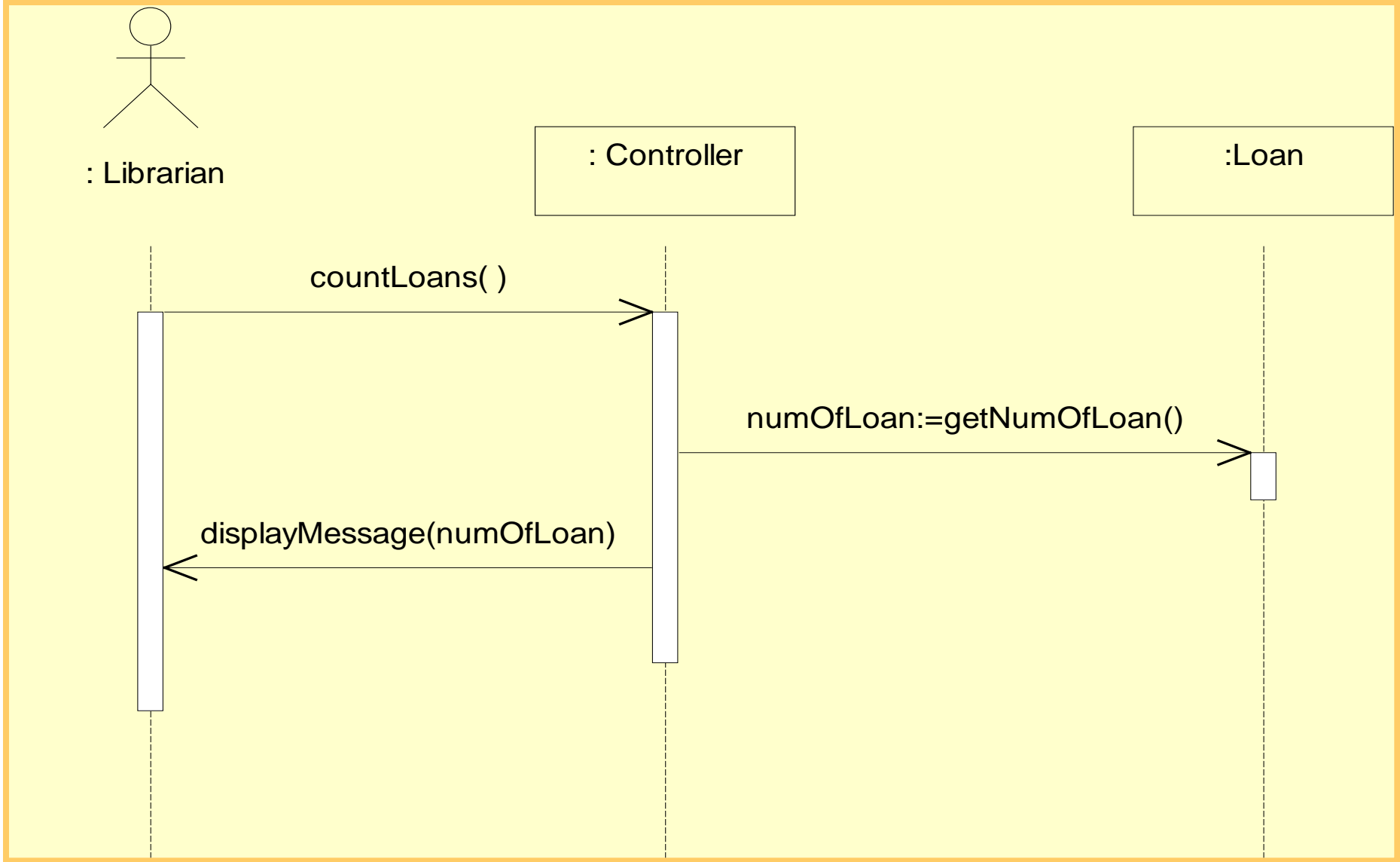
17. Log-In



18. Log-Out

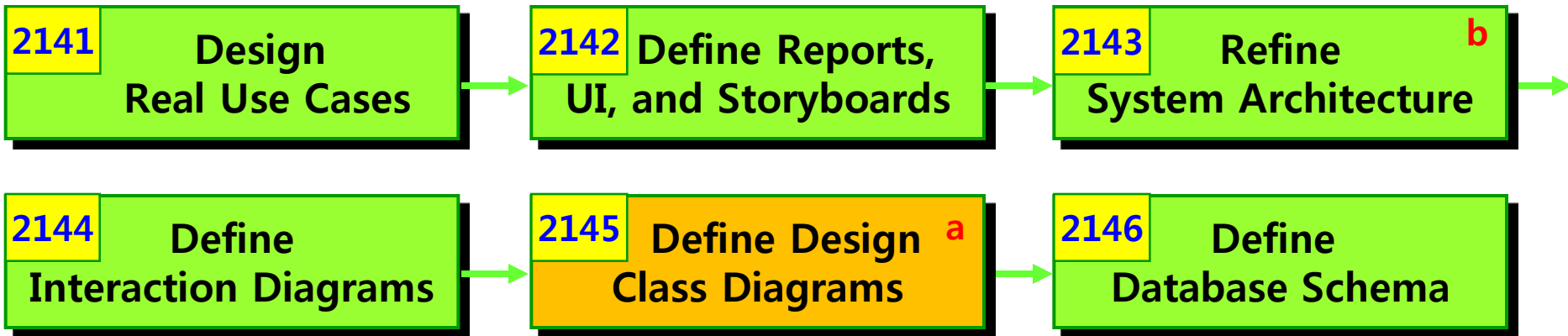


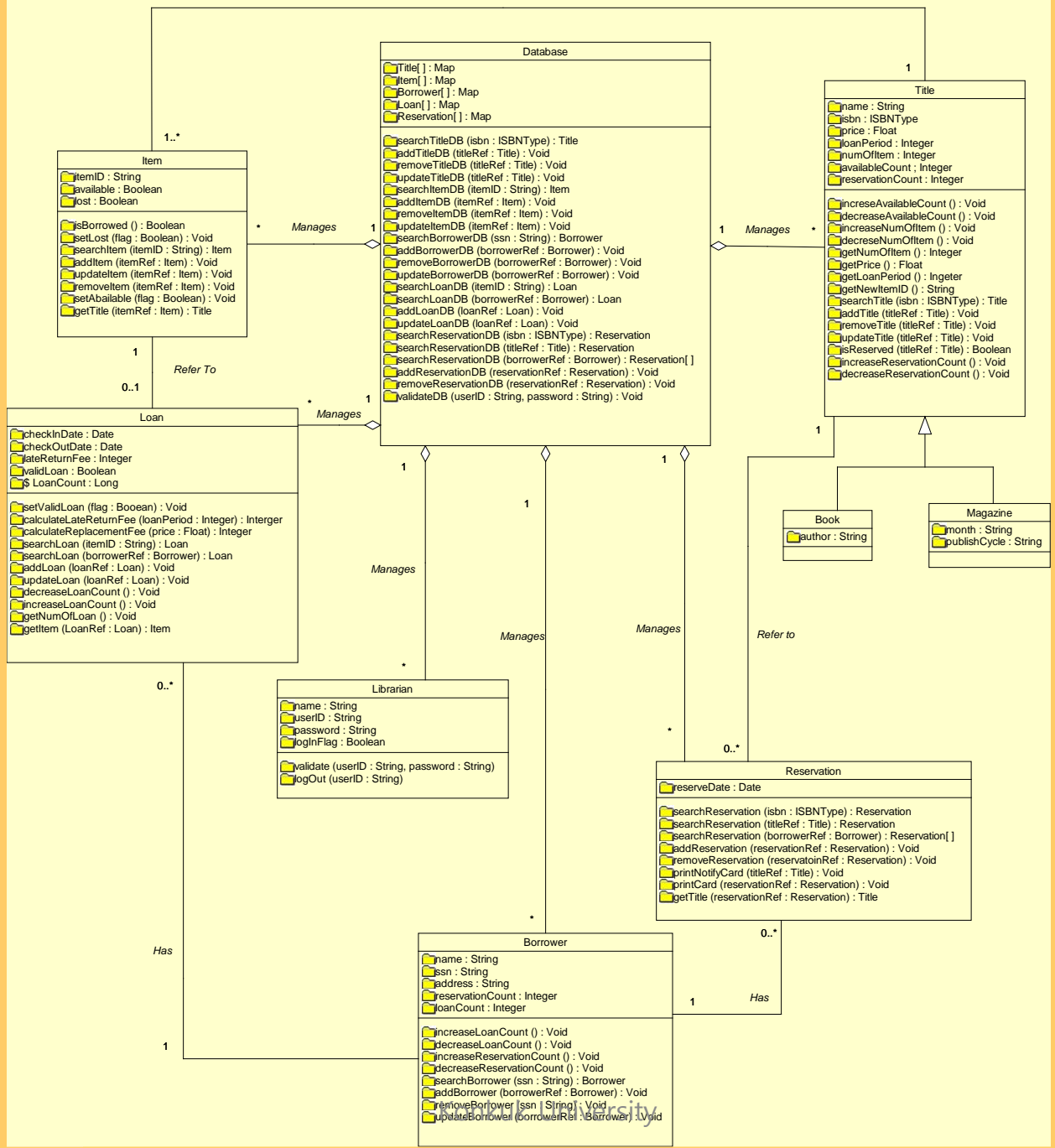
19. Count Loans



Activity 2045. Define Design Class Diagram

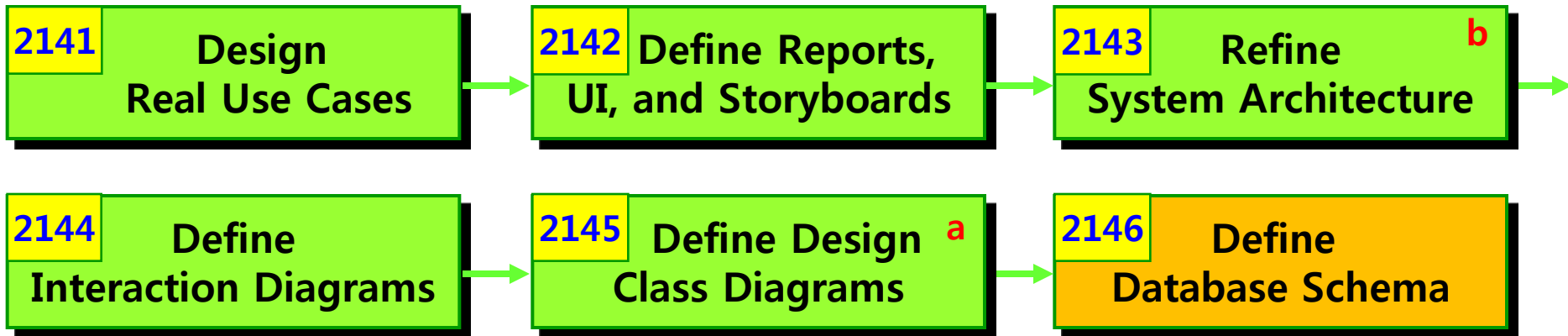
a. In parallel with interaction diagrams
b. Varied order





Activity 2046. Define Database Schema

- a. In parallel with interaction diagrams
- b. Varied order



Phase 2040. Design

- Phase 2040 Activities

a. In parallel with interaction diagrams
b. Varied order

