UNIVERSITI KUALA LUMPUR

CITY CAMPUS

MALAYSIAN INSTITUTE OF INFORMATION TECHNOLOGY

Name of Course	**OBJECT ORIENTED PROGRAMMING**
Course Code	**ISB16003**
Lecturer	**Dr Zulkhairi Mohd Yusof**
Student Name & ID	
Semester / Year	**Oct 2025**
Distribution Date	**8th December 2025**
Presentation Date	**5th January 2025**
Report Due Date	**11th January 2025**
Assessment	**Project – 3 or 4 students per group**

Project Overview

This is a 4-week group project for students with Java programming experience and understanding of **Object-Oriented Programming (OOP)** principles. The assignment is designed for groups of 4 students to collaborate on designing, implementing, and testing a Java application. The project should challenge students to apply OOP concepts (encapsulation, inheritance, polymorphism, abstraction), handle data structures, and implement user interaction.

Students will design and implement a **Smart Booking and Reservation System (SBRS)** using **Java Swing** and **OOP principles**. The core structure and required OOP elements are identical for all groups, ensuring fair grading. However, each group will select one of the eight themes below, customizing the system's user interface (UI), specific data fields, and behavior to match their chosen domain.

Duration: 4 Weeks
Team Size: 3–4 students per group
Tools: Java, Swing (not AWT), no external libraries
Submission: ZIP file with source code, UML diagram, user guide, and presentation

---

Core Requirements (All Groups)
All groups must implement the following **OOP and GUI features**:

1. OOP Design (Topics 1.0–3.7)

- Abstract Class: abstract class Reservation with abstract method String getConfirmationDetails().
- Concrete Subclasses: Implement the three specific subclasses for your chosen theme (e.g., BookLoan, RoomBooking, EquipmentRental for the Library theme). Each must have unique fields.
- Final Class: final class BookingID (immutable, validates format, e.g., "LIB-2025-001").
- Static Utility Class: static utility class ReservationManager with methods like generateReport() and getTotalReservations().
- Interface: Interface Payable with void processPayment() and default void applyDiscount().
- Nested Class: Reservation.ClientDetails (static) with fields: name, contact.
- Inner Class: Reservation.Reminder (non-static) that alerts when a reservation's due date is near.

2. Exception Handling (Topic 4.0)

- **Custom Exception**: InvalidReservationException (e.g., invalid date, negative quantity, over-capacity)
- Use try-catch when:
  - Parsing dates
  - Reading/writing files
  - Validating user input
- Use a finally block to close file resources

3. Text-Based Applications (Topic 5.0)

- Save reservations to reservations.txt (formatted)
- Load events from file on startup
- Append new reservations to file
- Handle FileNotFoundException, IOException

4. Building OO GUI with Swing (Topics 6.0–8.0)

- Use **Swing components**: JFrame, JPanel, JButton, JTextField, JComboBox, JList, JTextArea
- Use **layout managers**: BorderLayout, GridLayout, FlowLayout
- Implement **menu bar** with:
  - File → New, Save, Load, Exit
  - View → List All, Calendar View
  - Help → About
- Change **font and color** of key components (e.g., overdue items in red)
- Use **event handling**:
  - Button clicks (ActionListener)
  - List selection (ListSelectionListener)
  - Menu actions

Themed Project Themes (One per Group)
Each group selects **one theme** and customizes the UI, concrete subclasses, fields, and behavior accordingly.

Theme #	Theme Name	System Concept	Concrete Subclasses	Custom Fields & Behavior
1	Library Resource Management	Smart Library System (SLS)	BookLoan, RoomBooking, EquipmentRental	Add: dueDate, fineRate, patronID. UI: book covers, overdue alerts, search bar.
2	Vehicle Fleet Management	Smart Fleet System (SFS)	CarRental, TruckLease, BikeShare	Add: mileage, fuelLevel, serviceDate. UI: map view, maintenance log, vehicle status icons.
3	Hospital Appointment System	Smart Health System (SHS)	DoctorVisit, LabTest, SurgerySlot	Add: patientID, doctorName, insuranceCode. UI: calendar view, urgent flag, patient history summary.
4	Restaurant Table Booking	Smart Dining System (SDS)	DinnerReservation, PrivateEvent, TakeoutOrder	Add: tableNumber, partySize, specialRequest. UI: floor plan, order status tracker, server assignment.
5	Hotel Room Management	Smart Hospitality System (SHS)	StandardRoom, SuiteBooking, ConferenceHall	Add: checkInDate, nights, roomRate. UI: room status board, housekeeping notes, guest services menu.
6	Pet Adoption & Shelter System	Smart Animal System (SAS)	Adoption, FosterCare, VetAppointment	Add: animalID, breed, vaccinationStatus. UI: animal photos, medical history log, adoption application form.
7	Film & Photography Studio Booking	Smart Studio System (SSS)	PhotoShoot, VideoRecording, EquipmentRental	Add: studioRoom, lightingSetup, rentalDuration. UI: studio layout diagram, equipment inventory list, time slot picker.
8	University Course Enrollment	Smart Academic System (SACS)	LectureEnrollment, LabRegistration, TutorialBooking	Add: courseCode, creditHours, prerequisites. UI: course catalog search, conflict checker, student transcript view.

Note: All groups use the same core class structure (Reservation, BookingID, ReservationManager, etc.), but the UI and specific data fields within the concrete subclasses must be tailored to the chosen theme.

4-Week Development Plan

Week	Task
1	Design UML diagram. Implement core classes: Reservation, concrete subclasses, ReservationManager, BookingID. Write custom exception.
2	Implement file I/O: save/load reservations. Build console menu for testing. Add reservation validation and exception handling.
3	Build Swing GUI: main window, input panel, reservation list. Add menu bar, layout, fonts, colors. Connect to backend.
4	Implement event handling (buttons, list selection). Test all features. Prepare user guide and presentation.

Deliverables

Each group submits a **ZIP file** named: SBRS_GroupX_ThemeY.zip containing:

1. **Java Source Files.rar** (zip all .java files in correct package)
2. **Answers.pdf** include:
   - This assignment cover page (with your names)
   - Part A Group members, members contribution, theme, special features
   - Part B UML Class Diagram and UML Table
   - Part C Java codes (for ALL java files. Put the java name file on top of each program)
   - Part D User Guide (including screen shot)
   - Part E 10-slide presentation (design consideration, features, demo)

Java Codes Marking Rubric

Criteria	Notes
OOP Design (1.0–3.7)	Abstract, final, static, interface, nested classes, encapsulation
Exception Handling (4.0)	Custom exception, try-catch-finally, input validation
File I/O (5.0)	Save/load from text file, handle exceptions
Swing GUI (6.0–8.0)	Components, containers, layout, menu bar
Event Handling (7.0)	ActionListener, ListSelectionListener, proper event flow
Customization & Creativity	Theme integration, UI design, unique features
Code Quality	Comments, naming, structure, no duplication