**Case Study: Hotel Reservation System**

**Context:**

You are required to develop a console-based Hotel Reservation System using Core Java. The application will allow users to book rooms, manage reservations, and perform various business operations related to hotel management. The application will utilize Java Collections for storing room information and reservation details, implement Object-Oriented Programming principles, and include exception handling to manage potential errors.

**Requirements:**

**Features:**

1. **View Room Availability:**
   * Allow users to view the availability of rooms for a specific date range.
   * Display the available rooms with details such as room number, type, and price.
2. **Make Reservation:**
   * Allow users to make a reservation for a room.
   * Display the reservation details including room number, check-in and check-out dates, and total price.
3. **Cancel Reservation:**
   * Allow users to cancel a reservation.
   * Update room availability accordingly.
4. **Check-in and Check-out:**
   * Allow users to check-in to a reserved room.
   * Allow users to check-out from a room.
   * Update room status and availability.

**Classes and Objects:**

1. **Room Class:**
   * Attributes: number, type (single, double, suite), price, available (boolean indicating if the room is available).
   * Methods: Constructors, getters and setters, toString method.
2. **Reservation Class:**
   * Attributes: id, roomId, guestName, checkInDate, checkOutDate, totalPrice.
   * Methods: Constructors, getters and setters, toString method.
3. **HotelManager Class:**
   * Attributes: ArrayList<Room> rooms, ArrayList<Reservation> reservations.
   * Methods:
     + checkRoomAvailability(Date checkIn, Date checkOut): Checks room availability for a given date range.
     + makeReservation(int roomId, String guestName, Date checkIn, Date checkOut): Makes a reservation for a room.
     + cancelReservation(int reservationId): Cancels a reservation.
     + checkIn(int reservationId): Checks-in to a reserved room.
     + checkOut(int reservationId): Checks-out from a room.
     + Helper methods for input validation and exception handling.

**Deliverables:**

1. Complete source code for the Hotel Reservation System.
2. Documentation including:
   * How to run the application.
   * Instructions for each feature.
   * Explanation of the exception handling implemented.
3. A brief report on the application design and how Object-Oriented principles were applied.