Delivery Slot Scheduler

Problem Overview

A delivery company needs to allocate delivery time slots to customers. Each customer has a requested time window indicating when they are available. Your task is to determine the minimum number of delivery vehicles required and
to produce a possible schedule that assigns customers to vehicles without overlapping time windows.

Requirements

1. Each customer has a time window: StartTime to EndTime.

2. A vehicle can serve multiple customers as long as their time windows do not overlap.

3. You must:
a. Determine the minimum number of vehicles needed.
b. Produce a sample schedule (vehicle → list of customers).
c. Clearly explain your approach and assumptions.

Example Input

```
5
C1 9:00 10:00
C2 9:30 10:30
C3 11:00 12:00
C4 10:30 11:30
C5 12:00 13:00
```

Possible Output

```
Minimum vehicles needed: 2
Vehicle 1 → C1, C3, C5
Vehicle 2 → C2, C4
```

Meeting Room Booking Validator

Problem Overview

You are building a component for an office meeting room booking system. Employees submit booking requests by
specifying: RoomID, StartTime, EndTime, RequestedBy.
Your task is to validate bookings, identify conflicts, and suggest alternate rooms when possible.

Requirements

Your program must:

1. Detect conflicts — overlapping bookings for the same room.

2. Produce:
a. a list of valid bookings
b. a list of conflicting bookings

3. Suggest alternate rooms for conflicting bookings (if possible).

Additional Details

- There are M rooms.

- If a meeting conflicts, you must suggest another room whose schedule allows that meeting without collisions.

- If no room is free, mark it as unschedulable.

Example Input

Rooms: R1, R2
Bookings:
B1 R1 10:00 11:00 Alice
B2 R1 10:30 11:15 Bob
B3 R2 10:00 11:00 Charlie
B4 R1 11:00 12:00 David

Output Example

```
Valid:
B1 (R1)
B3 (R2)
B4 (R1)

Conflicts:
B2 conflicts in R1 → Suggested room: R2
```