**Scenario based spring boot questions**

**1. Subscriber Management System**

**Scenario:**  
Build a Spring Boot application to manage **mobile subscribers**.

**Entity: Subscriber**

Long id

String name

String mobileNumber

String planType // Prepaid, Postpaid

String region

**Required Endpoints:**

* POST /subscribers – Add a new subscriber
* GET /subscribers – View all subscribers
* GET /subscribers/{id} – Get subscriber by ID
* PUT /subscribers/{id} – Update subscriber details
* DELETE /subscribers/{id} – Delete subscriber

**Postman Testing Tips:**

* Use POST with raw JSON to create subscribers.
* Use GET to verify data.
* Try updating planType from Prepaid to Postpaid using PUT.

**✅ 2. Call History Tracker**

**Scenario:**  
Create a Spring Boot API to **record and view call history** for users.

**Entity: CallRecord**

Long id

String callerNumber

String receiverNumber

int durationSeconds

LocalDateTime callTime

String callType // Incoming, Outgoing

**Required Endpoints:**

* POST /calls – Log a new call
* GET /calls – Get all call records
* GET /calls/caller/{callerNumber} – Get all calls made by a number
* DELETE /calls/{id} – Delete a call record

**Postman Testing Tips:**

* Use GET /calls/caller/{callerNumber} to filter calls.
* Test DELETE to simulate call log deletion.

**✅ 3. Data Usage Monitoring**

**Scenario:**  
Build an API to monitor **data usage per device** on the network.

**Entity: DataUsage**

Long id

String userId

String deviceType // Smartphone, Dongle, Tablet

double dataUsedInMB

LocalDate usageDate

String networkType // 3G, 4G, 5G

**Required Endpoints:**

* POST /usage – Add a new data usage entry
* GET /usage – View all data usage records
* GET /usage/user/{userId} – View usage by user
* GET /usage/device/{deviceType} – View usage by device type
* PUT /usage/{id} – Update a record
* DELETE /usage/{id} – Delete a record

**Postman Testing Tips:**

* Check how data changes when you PUT a new value for dataUsedInMB.
* Try querying usage by both userId and deviceType.