**Practice Lab: ERD and Normalization Skills Drill**

**Objective:**

Practice essential data modeling steps including requirement gathering, normalization, diagramming, naming, and version control using Git for an ERD.

**Topic 1: Requirements Interview Drill**

**Objective:** Simulate stakeholder interviews to extract entities and attributes.

**Tasks:**

* Write 5 interview questions to understand requirements for a hospital management system.
* Based on mock responses, list out at least 4 entities with their key attributes.
* Identify 3 relationships among those entities.

**Challenge:**  
Use your questions and answers to create a draft textual ER diagram with basic relationship mapping.

**Topic 2: ER Cardinality & Optionality**

**Objective:** Define how entities relate to one another.

**Tasks:**

* For a hotel booking system:
  + Define relationship cardinality between Customer and Booking.
  + Define optionality between Room and Booking.
* Mark one-to-one, one-to-many, or many-to-many with optional/mandatory roles.

**Challenge:**  
Draw a relationship matrix (textual or table form) for 4 entities (e.g., Customer, Room, Booking, Payment).

**Topic 3: Attribute Atomicity**

**Objective:** Normalize attributes to atomic form.

**Tasks:**

* Given the following attribute: FullAddress = "123 Main St, Springfield, IL, 62701", split it into atomic fields.
* Normalize ContactInfo = "John Doe | john@example.com | +91-99999-88888".

**Challenge:**  
Write 3 non-atomic attributes from a retail order system and propose atomic breakdowns.

**Topic 4: BCNF Walk-through**

**Objective:** Normalize a composite table into BCNF.

**Tasks:**

* Given this table:

| OrderID | CustomerName | ProductName | ProductCategory | Price |

* Identify functional dependencies.
* Convert it to 1NF, 2NF, 3NF, and then BCNF.

**Challenge:**  
Redesign the table structure in SQL CREATE TABLE statements after BCNF decomposition.

**Topic 5: Crow-foot ERD Tooling**

**Objective:** Use a diagramming tool to create a visual ERD.

**Tasks:**

* Use dbdiagram.io or any ERD tool.
* Define tables for a library system: Book, Member, IssueRecord, Librarian.
* Connect relationships using Crow’s Foot notation.

**Challenge:**  
Export the diagram as PNG and simulate how the IssueRecord table changes if one book can have multiple copies.

**Topic 6: Naming Standards**

**Objective:** Apply industry-standard naming conventions.

**Tasks:**

* Rename the following table and column names per best practice:
  + Table: EMPLOYEES\_LIST → ?
  + Column: Emp.Name → ?
* Apply:
  + snake\_case for columns
  + singular names for tables
  + avoid spaces or special characters

**Challenge:**  
Given a table with poorly named columns, rewrite the full table schema with naming best practices.

**Topic 7: ERD Versioned in Git**

**Objective:** Track ERD file versions using Git.

**Tasks:**

* Initialize a Git repo.
* Add an ERD file (e.g., .dbml or .png).
* Commit with message: "Initial conceptual ERD".

**Challenge:**  
Simulate a change (e.g., new entity or relationship), modify the ERD, commit again, and show Git history using git log.