Question 1 [Databases, 30 marks]

NB: All tables can have hundreds of data entries. This is denoted by the "…"

a) Imagine you are working as a database developer, and you are given the following table in Excell format. The company you work for has asked you to add it to their relational database management system. Write an SQL query that is used to create a table called

FurnitureID	ItemName	Material	Price	StockQuantity
1	Eames Chair	Leather/Wood	450.00	12
2	Billy Bookcase	Particle Board	89.99	45
3	Marble Dining Table	Stone	1200.00	3
4	Task Lamp	Steel	35.50	28
…	…	…	…	..

"Furniture" with the same structure. Include the specification of primary key(s). Do not insert any data in the table.

[5 marks]

b) Consider the following scenario:

A local "Pet Grooming & Boarding" shop uses a single spreadsheet to track appointments. The owner is complaining that when a customer changes their phone number, they must update it in ten different places, and if they delete an appointment, they sometimes lose the owner's contact info entirely.

Table 1: Pet Grooming Appointments. ApptID stands for appointment ID

ApptID	OwnerName	OwnerPhone	PetName	PetType	Service	ServicePrice
501	Sarah Chen	555-1234	Barnaby	Dog	Full Groom	$60
502	Mike Ross	555-9876	Luna	Cat	Nail Trim	$20
503	Sarah Chen	555-1234	Felix	Cat	Full Groom	$45
504	Sarah Chen	555-1234	Barnaby	Dog	Teeth Brushing	$15
…	…	…	…	…	…	…

i) What anomalies can be identified based on this? Explain you answer

[5 marks]

ii) If you normalise the Table 1 into 3 normal form (3NF), how many tables will you end up having? What will be their columns? What will be the primary and foreign keys?

[10 marks]

Your answer should be in the form:
[Table1_Name]: [Column1_Name] [Column2_Name]…

[Table2_Name]: [Column1_Name] [Column2_Name]…
[Table3_Name]: [Column1_Name] [Column2_Name]…
….

If a column is selected as a primary or foreign key, use:
[Column1_Name] **(PK)** or [Column1_Name] **(FK)**

c) Consider a cinema database comprising of two tables:

Table 2: Members

MemberID (PK)	**FullName**	**JoinDate**
10	Sarah Jenkins	2026-01-01
20	Mark Sloan	2026-02-15
30	Elena Rodriguez	2026-03-01
…	…	…

Table 3: Tickets

TicketID (PK)	**CustomerID (FK)**	**MovieTitle**	**Price**
501	10	Inception	$12
502	10	The Matrix	$12
503	40	Interstellar	$15
…	…	…	…

i) Write a SQL query that joins the Members and Tickets tables. Your query should display the FullName, MovieTitle, and Price.

[6 marks]

Constraint: You must link the tables using the correct Primary Key and Foreign Key.

ii) Write a SQL query to display the **MovieTitle** and **Price** for all tickets where the price is **greater than 10**. The results must be sorted by Price in Descending order (highest price first).

[4 marks]

Question 2 [Web Development Principles - 30 marks]

1. **Request Parameters & Paths** A fitness app uses two different ways to find data. Evaluate the following:
    a. GET /api/v1/workouts/45
    b. GET /api/v1/workouts?type=running

Explain the difference between a **Path Variable** and a **Request Parameter** using these two examples. In which scenario would you use one over the other?

[10 marks]

2. **The Persistence Layer**
    a. Inside a Spring Boot Repository, you see the method:
    long countByStatus(String status);
    Explain what this method returns and how Spring Data JPA "writes" the code for you based on that name.
    b. Which specific annotation is used to link a Java Class to a specific table name in the database?

[10 marks]

3. **Authentication vs. Authorization** In the context of a fitness app where users have "Private" and "Public" profiles:
    a. Define **Authentication** and **Authorization**.
    b. If a user tries to view a "Private" profile that doesn't belong to them, which check has failed? Provide a logical example of how a Spring Boot Service might perform this check.

[10 marks]

Question 3 [Web Development Practice - 40 marks]

1. **JDL Modeling: Activities** You are modeling a **Workout Log**.
   a. Define a JDL for an Activity entity with: type (String, required), durationMinutes (Integer), caloriesBurned (Integer), and timestamp (Instant).
   b. Ensure type is limited to a maximum of 30 characters.
   c. Create an appropriate relationship between Activity and the built-in User entity.

   [15 marks]

2. **CURL & Methods**
   a. Write a curl command to **PATCH** an existing Activity (ID: 700) to change the durationMinutes to 45. Assume the application is running on localhost:8080.
   b. If the developer accidentally uses the **POST** method on /api/activities/700, what **4xx** error code is most likely to be returned? Explain why.

   [10 marks]

3. **Versioning & Deprecation** The app is moving to a new version to support "GPS Coordinates."
   a. How would you structure the @RequestMapping for a new ActivityResource to distinguish it from the old endpoint?
   b. If you keep the old endpoint active for 6 months while people migrate, what is this period of support called? Explain one strategy to notify developers using the old endpoint that they need to switch.

   [15 marks]