LI Databases and Web programming

Lab sheet Week 2 Solution

Scenario: To-Do Application

You are asked to design the database for a **To-Do Item Application**.

The application allows users to manage their **tasks** (to-do items), organize them by **projects**, and assign ownership. Each task also has a **priority**, a **status**, and can be linked to a **context**.

Here are the main details:

- A **To-Do Item** has:
  - a unique identifier,
  - a text description,
  - a due date, and
  - a completion date.
- Each **To-Do Item** belongs to one **Owner**.
  - An **Owner** has:
    - an ID,
    - a first name, and
    - a surname.
- Each **To-Do Item** is linked to a **Project**.
  - A **Project** has:
    - a project ID,
    - a project name, and
    - a budget.
- Each **To-Do Item** is associated with a **Context**.
  - A **Context** has:
    - a context ID, and
    - a description.
- Each **To-Do Item** also has a **Priority** (e.g., High, Medium, Low).
- Each **To-Do Item** also has a **Status** (e.g., Pending, In Progress, Completed).

Task 1: Extract Entities

From the scenario, list the **entities** and their **attributes**.
Write them in this format:

- Entity: Attributes

Task 3: Write the JDL File

After identifying the entities and relationships, write a **JDL model** for this system.

Task 4: Visualize the ER Diagram

- Open **JDL Studio** (https://start.jhipster.tech/jdl-studio/).
- Paste your JDL code.
- Generate and **draw the ER diagram**.

Solution

Task 1

Entity: ToDoItem

Attributes: id, description, dueDate, completionDate, priority, status

Entity: Owner

Attributes: id, firstName, surname

Entity: Project

Attributes: id, name, budget

Entity: Context

Attributes: id, description

Task 2-3:

```
entity Owner {
 firstName String required,
 surname String required
}

entity Project {
 name String required,
 budget BigDecimal
}

entity Context {
 description String required
}

entity ToDoItem {
 description String required,
 dueDate Instant,
 completionDate Instant,
 priority ToDoPriority required,
 status ToDoStatus required
}

enum ToDoPriority {
 High, Medium, Low
}

enum ToDoStatus {
 Pending, InProgress, Completed
```

}

relationship ManyToOne {

ToDoItem{owner} to Owner,

ToDoItem{project} to Project,

ToDoItem{context} to Context

}

Task 4: Visualize in JDL Studio

Copy the above JDL code.

Go to https://start.jhipster.tech/jdl-studio/

Paste the code and generate the ER diagram.

