Task3 description

Your application allows you to manage `Task` and `Project` entities.

You have to add missing annotations and code over entity classes and converters to make it possible to store `Task` and `Project` and their relations in specified tables.

Requirements

1. `Project.updates` should be persisted in a table with the following structure:
    1. Table name: `PROJECT_EVENT_UPDATES`;
    2. Columns:
        1. `PROJECT_ID`: foreign key;
        2. `USER_NAME`: value of `ItemUpdate.user`;
        3. `EVENT_DATE`: value of `ItemUpdate.date`.
2. `Task.watchers` should be persisted in a table with the following structure:
    1. Table name: `TODO_WATCHERS`;
    2. Columns:
        1. `TODO_ID`: foreign key;
        2. `LOGIN`: value of `String` from the `Set`.
3. `Task.comments` should be persisted in a table with the following structure:
    1. Table name: `TODO_COMMENTS`;
    2. Columns:
        1. `TODO_ID`: foreign key;
        2. `AUTHOR`: value of `Comment.author`;
        3. `ADD_DATE`: value of `Comment.date`;
        4. `COMMENT`: value of `Comment.comment`.
4. `Task.updates` should be persisted in a table with the following structure:
    1. Table name: `TODO_UPDATES`;
    2. Columns:
        1. `TODO_ID`: foreign key;
        2. `AUTHOR`: value of `ItemUpdate.user`;
        3. `UPDATE_DATE`: value of `ItemUpdate.date`.
5. `Task.tags` should be persisted in a table with the following structure:
    1. Table name: `TODO_TAGS`;
    2. Columns:
        1. `TODO_ID`: foreign key;
        2. `NAME`: key from the `Map`;
        3. `DESCRIPTION`: value from the `Map`.

Tables

The following tables should exist after implementation:

1. `PROJECTS` with columns `PROJECT_ID`, `NAME`;
2. `PROJECT_EVENT_UPDATES` with columns `PROJECT_ID`, `USER_NAME`, `EVENT_DATE`;
3. `TODOS` with columns `TODO_ID`, `SUBJECT`, `DETAILS`;
4. `TODO_WATCHERS` with columns `TODO_ID`, `LOGIN`;
5. `TODO_COMMENTS` with columns `TODO_ID`, `AUTHOR`, `ADD_DATE`, `COMMENT`;
6. `TODO_UPDATES` with columns `TODO_ID`, `AUTHOR`, `UPDATE_DATE`;
7. `TODO_WATCHERS` with columns `TODO_ID`, `NAME`, `DESCRIPTION`.

Assumptions

- Class names and function signatures should not be changed.

Hints

In order to satisfy the requirements you should:

- add missing annotations from the `javax.persistence` package to the `Task` and `Project` classes.

Source Code

The following classes have already been implemented:

ItemUpdate

```
@Embeddable
@NoArgsConstructor(access = AccessLevel.PRIVATE)
@AllArgsConstructor
@Getter
public class ItemUpdate {
    private String user;
    private LocalDate date;
}
```

Comment

```
@Embeddable
@NoArgsConstructor(access = AccessLevel.PRIVATE)
@AllArgsConstructor
@Getter
public class Comment {
    @Column(name = "AUTHOR")
    private String author;
    @Column(name = "ADD_DATE")
    private LocalDate date;
    @Column(name = "COMMENT")
    private String comment;
}
```

In your solution, focus on correctness. Imports will not be needed and possible misspelled annotations will not be the focus of the assessment.