Guidelines:

- Time: 90mins
- Write modular and clean code.
- **A driver program/main class/test case is needed to test out the code by the evaluator with multiple test cases**. But do not spend too much time in the input parsing. Keep it as simple as possible.
- Evaluation criteria: Demoable & functionally correct code, Code readability, Proper Entity modelling, Modularity & Extensibility, Separation of concerns, Abstractions. Use design patterns wherever applicable
- You are not allowed to use any external databases like MySQL. Use only in memory data structures.
- No need to create any UX
- Please focus on the Bonus Feature only after ensuring the required features are complete and demoable.

Problem Statement

Create an application like **Trello** which is a project management application, where you can manage your project by tracking smaller tasks. For reference : https://trello.com/

Basic Features and workings :

- The application contains multiple boards to signify different projects
- Each board contains different lists to signify sub-project
- Each list contain different cards signifying smaller tasks
- Each card can be assigned to a user or may remain unassigned

Definitions :

- User: Each user should have a userId, name, email.
- Board: Each board should have a id, name, privacy (PUBLIC/PRIVATE), url, members, lists
- List: Each list should have a id, name and cards
- Card: Each card should have a id, name, description, assigned user, priority, created time, finished time, ETA, status

Requirements :

- We should be able to create/delete **boards**, add/remove people from the members list and modify attributes. Deleting a board should delete all lists inside it.
- We should be able to create/delete **lists** and modify attributes. Deleting a list should delete all cards inside it.
- We should be able to create/delete cards, assign/unassign a member to the **card** and modify attributes.

- We should also be able to move cards across lists in the same board
- Ability to show all boards, a single board, a single list and a single card
- Default privacy should be public
- Cards should be unassigned by default
- Ids should be auto-generated for board/list/card

Bonus for SDE1 :

- Bookmark (add to favorite) card/list/board. Show user's bookmarked cards/lists/boards

Additional Complexity for SDE2 :

- Search by filter / sort - assigned user / created date range / ETA / priority etc
- Watch (subscribe to changes on card/board) - user should get informed for the changes.

Bonus for SDE2 :

- Handling concurrency when multiple users edit the same card.