Community Part of Table explain

**Community Chat Features:**

1. **Join a Group**:
   * Users can join a book club by clicking the "Join Group" button on the book reading page.
   * The relationship between users and groups is tracked through the group\_members table.
2. **Sending Messages**:
   * Once in a group, users can send messages.
   * Each message is stored in the chat\_messages table, including:
     + group\_id: ID of the group where the message was sent.
     + sender\_id: ID of the user sending the message.
     + message: The content of the message.
     + created\_at: Timestamp of when the message was sent.
3. **Leaving a Group**:
   * When a user leaves a group, their status is updated in the user\_group\_status table.
   * The status field can be set to 'left' to indicate the user is no longer part of the group.

**Database Tables and Their Functions:**

1. **chat\_messages**:
   * **Purpose**: Stores the messages sent by users in a group chat.
   * **Key Fields**:
     + message\_id: Unique ID for each message.
     + group\_id: The ID of the group where the message was sent.
     + sender\_id: The ID of the user who sent the message.
     + message: The content of the message.
     + created\_at: Timestamp of when the message was sent.
   * **Example**: Each time a user sends a message in a book club, a new entry will be added to this table.
2. **community\_groups**:
   * **Purpose**: Stores details about the groups (book clubs) that users can join.
   * **Key Fields**:
     + group\_id: Unique ID for each community group.
     + admin\_id: The user ID of the group admin (the person who created the group).
     + created\_at: Timestamp of when the group was created.
   * **Example**: When a user creates a new book club, a new entry will be added to this table.
3. **group\_members**:
   * **Purpose**: Tracks the members of each community group.
   * **Key Fields**:
     + membership\_id: Unique ID for the membership entry.
     + group\_id: The ID of the group the user is a part of.
     + user\_id: The ID of the user in the group.
     + joined\_at: Timestamp of when the user joined the group.
     + online\_status: Tracks if the user is online or offline.
     + last\_active: Timestamp of the last time the user was active.
   * **Example**: Every time a user joins a group, a new entry is added to this table to reflect the membership status.
4. **user\_group\_status**:
   * **Purpose**: Keeps track of the status of a user in a group (whether the user has joined or left).
   * **Key Fields**:
     + user\_id: The ID of the user.
     + group\_id: The ID of the group.
     + status: The status of the user in the group (either 'joined' or 'left').
   * **Example**: When a user leaves a group, this table will update their status to 'left'.

**Workflow:**

1. A user clicks the "Join Group" button, and the group\_members table is updated with their user\_id and group\_id.
2. The user can send messages, which are recorded in the chat\_messages table.
3. If the user leaves the group, their status is updated in the user\_group\_status table to 'left'.
4. The group is administered by the community\_groups table, with each group having an admin (admin\_id).

All the table that belong to the community part

CREATE TABLE `chat\_messages` (

`message\_id` int(11) NOT NULL,

`group\_id` int(11) NOT NULL,

`sender\_id` int(11) NOT NULL,

`message` text NOT NULL,

`created\_at` timestamp NOT NULL DEFAULT current\_timestamp()

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

CREATE TABLE `community\_groups` (

`group\_id` int(11) NOT NULL,

`admin\_id` int(11) NOT NULL COMMENT 'User ID of the group admin',

`created\_at` timestamp NOT NULL DEFAULT current\_timestamp() COMMENT 'Timestamp when group was created'

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

--

CREATE TABLE `group\_members` (

`membership\_id` int(11) NOT NULL,

`group\_id` int(11) NOT NULL COMMENT 'Group ID',

`user\_id` int(11) NOT NULL COMMENT 'User ID of the group member',

`joined\_at` timestamp NOT NULL DEFAULT current\_timestamp() COMMENT 'Timestamp when user joined',

`online\_status` enum('online','offline') DEFAULT 'offline',

`last\_active` timestamp NOT NULL DEFAULT current\_timestamp() ON UPDATE current\_timestamp()

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

CREATE TABLE `user\_group\_status` (

`user\_id` int(11) NOT NULL,

`group\_id` int(11) NOT NULL,

`status` enum('joined','left') DEFAULT 'joined'

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;