

Queries to be run on database for use cases :

1. *Name:* Signup

Description:

1. Search for the entered email_id in the user table.
2. If email_id already exists the show error message that the user already registered.
3. Else INSERT new user in the user document/table.

2. *Name:* Login

Description:

1. Search for the entered email_id in the user document/table.
2. If email_id || user_id not present then show error message that user doesn't exist.
3. Else if entered credentials == database credentials =>start a new session and turn logged_in=true

3. *Name:* Like a post

Description:

- a. INSERT the *user_id* to the *likers* array of the post .
- b. *likes +=1*
- c. INSERT this in the activity array of the user.
- d. INSERT a new notification in the *notifications* array of the *user_id* associated with the post.?

4. *Name:* Unlike a post

Description:

- a. REMOVE the *user_id* to the *likers* array of the post .
- b. *likes -=1*

5. *Name:* Comment

Description:

- a. INSERT the *user_id* and the *comment_text* to the *comments* array of the post .
- b. *comments_count* +=1
- c. INSERT this in the activity array of the user.
- d. INSERT a new notification in the *notifications* array of the *user_id* associated with the post.

6. **Name:** add post

Description:

- a. INSERT new post into the post document/table and generate a new *post_id* .
- b. SEARCH for the hashtags included in this post into the hashtags , **if** the hashtag already exists => INSERT this *post_id* into an array of posts in the hashtag document ,else INSERT new object in the hashtag document .
- c. *no_of_posts* +=1 (for each # included in the post)
- d. INSERT notification into each *user_id*'s notification array which are tagged into the post.
- e. INSERT this in the activity array of the user.
- f. INSERT this post's *post_id* in the *posts* array of the user.

7. **Name:** edit post

Description:

- a. SEARCH for the *post_id* and UPDATE the attributes of the post.
- b. SAVE changes in the database.
- c. RETURN updated post.

8. **Name:** Connection request

Description:

- a. SEARCH for the *user_id* (of the user whom you want to connect)

- b. If *user_id* already present in *sent_invitations* array , then you cannot resend the invitation
 - c. Else INSERT sender's *user_id* to the *connection_requests* array of the receiver's user.
 - d. INSERT this request in the *notifications* array of the receiver's user.
9. **Name:** Respond to connection request

Description:

- a. If response == “accept” => INSERT *user_id* of sender into receivers connections array && INSERT *user_id* of receiver into senders connections array .
- b. INSERT this in the *activity* array of the user.
- c. INSERT this request in the *notifications* array of the receiver user and the sender user.
- d. Else if response == “decline” => DELETE *user_id* of senders from the *connection_requests* array of the receiver.

10. **Name:** Save post

Description:

- a. INSERT the *post_id* in the *saved_post* array of users.

11. **Name:** View Profile

Description:

- a. SEARCH for the *user_id* in the database
- b. RETURN the profile page of user with *user_id*

12. **Name:** Edit Profile Intro

Description:

- a. UPDATE attributes of *intro* in user profile.
- b. SAVE changes into profile changes
- c. Return the updated profile

13. **Name:** Edit Profile Photo

Description:

- a. UPDATE url of *profile_photo* in user profile .
- b. SAVE changes into profile in database.
- c. Return the updated profile

14. **Name:** Wall Photo

Description:

- a. UPDATE url of *wall_photo* in user profile .
- b. SAVE changes into profile in database.
- c. Return the updated profile

15.**Name:** Like a comment on a post

Description:

- a. INSERT this like in the *notifications* array of the user whose comment is liked.
- b. INSERT *user_id* of the user who liked the post into the likes attribute of the *comments* of the post.

16.**Name:** Edit post visibility

Description:

- a. UPDATE the visibility setting [private/connections/public]
- b. Save value in the database

17.**Name:** Edit open_to setting

Description:

- a. UPDATE open_to setting [hiring/looking for job/none]
- b. SAVE status in the database .

18.**Name:** Cancel connection request

Description:

- a. DELETE request from the recipient *user_id*'s *connection_requests* list.

19.**Name:** View my connections

Description:

- a. DISPLAY list of all the user profiles connected to you.

20.**Name:** Mute a user

Description:

- a. SEARCH for the *user_id* in the connections.
- b. UPDATE *mute_status* in the *connections* attribute = true

21.**Name:** Follow a page

Description:

- a. SEARCH for the *page_id* in the connections.
- b. INSERT *user_id* of the user into the *followers* list of the page with *page_id*
- c. INSERT *page_id* in the *pages_followed* by the user with *user_id*.
- d. INSERT this in the activity list of the user.

22.**Name:** Follow a hashtag

Description:

- a. SEARCH for the hashtag in the hashtags table.
- b. INSERT *hash_id* of that hashtag in the *hashtags* list of the user.
- c. INSERT this in the activity list of the user.

23.**Name:** Create a page

Description:

- a. a.Search for the entered *page_id* in the pages table.
- b. If *page_id* already exists the show error message that the user has already registered.
- c. Else INSERT new page in the user document/table.

24.**Name:** Edit page_name

Description:

- d. SEARCH for the new *page_name* in the page document
- e. If *page_name* already exists show error message
- f. Else UPDATE the *page_name* and SAVE the document.

25.**Name:** Edit Page Photo

Description:

- a. UPDATE url of *page_photo* in pages document .
- b. SAVE changes into page in database.
- c. Return the updated page.

26.**Name:** Edit Page wall Photo

Description:

- a. UPDATE url of *page_wall_photo* in pages document .
- b. SAVE changes into page in database.

- c. Return the updated page.

27.**Name:** Edit page password

Description:

- a. Enter current password, if current password $\neq password$ show error message that invalid password.
- b. Else if entered password $= password \Rightarrow$ ask for new password.
- c. UPDATE *password*.

28.**Name:** Edit Page Description

Description:

- a. UPDATE *description* of the page and SAVE changes into the database
- b. RETURN updated page

29.**Name:** Edit Contact information.

Description:

- a. UPDATE *phone || email || website || address*.
- b. SAVE changes into page in database.
- c. Return the updated page.

30.**Name:** Mute a page

Description:

- a. SEARCH for the *page_id* in the connections.
- b. UPDATE *mute_status* in the *connections* attribute = true

31.**Name:** Change Privacy setting

Description:

- a. UPDATE *privacy_sett* in the *user* attribute,
- b. Choose *profile_visibility* = false or true.
- c. Choose *head_visibility* = false or true.
- d. Choose *article_visibility* = false or true.
- e. Choose *edu_visibility* = false or true.

- f. Choose `details_visibility = false` or `true`.
- g. SAVE changes into the user in the database.

32. Name: Add education

Description:

- a. INSERT new education object in user's *education* field .
- b. Create a *post* for a new education update for the user.
- c. INSERT this into *my_items* .
- d. INSERT this information into the *activity* array of the user.

33. Name: Add work experience

Description:

- a. INSERT new work-experience object in user's *work_experience* array.
- b. Create a *post* for a new *work_position* update for the user.
- c. INSERT this into *my_items* .
- d. INSERT this information into the *activity* array of the user.

34. Name: Add Licence/Certificate

Description:

- a. INSERT new object in user's *licence/certificates* array.
- b. Create a *post* for a new update for the user.
- c. INSERT this into *my_items* .
- d. INSERT this information into the *activity* array of the user.

35. Name: Add Volunteer Experience

Description:

- a. INSERT new object in user's *volunteer_experience* field .
- b. Create a *post* for a new update for the user.
- c. INSERT this into *my_items* .
- d. INSERT this information into the *activity* array of the user.

36. Name: Add Skill

Description:

- a. INSERT the new skill into the *skills* array of the user document.

37. Name: Add Course

Description:

- a. INSERT new object in user's *courses* array.
- b. Create a *post* for a new update for the user.

- c. INSERT this into *my_items* .
- d. INSERT this information into the *activity* array of the user.

38.Name: Add Publication

Description:

- a. INSERT new object in user's *publications* array.
- b. Create a *post* for a new update for the user.
- c. INSERT this into *my_items* .
- d. INSERT this information into the *activity* array of the user.

39.Name: Add Patent

Description:

- a. INSERT new object in user's *patent array*.
- b. Create a *post* for a new update for the user.
- c. INSERT this into *my_items* .
- d. INSERT this information into the *activity* array of the user.

40.Name: Add Project

Description:

- a. INSERT new object in user's *projects* array.
- b. Create a *post* for a new update for the user.
- c. INSERT this into *my_items* .
- d. INSERT this information into the *activity* array of the user.

41.Name: Add Awards

Description:

- a. INSERT new object in user's *awards* array.
- b. Create a *post* for a new update for the user.
- c. INSERT this into *my_items* .
- d. INSERT this information into the *activity* array of the user.

42.Name: Add Test_Scores

Description:

- a. INSERT new object in user's *test_scores* array.
- b. Create a *post* for a new update for the user.
- c. INSERT this into *my_items* .
- d. INSERT this information into the *activity* array of the user.

43.Name: Add Language

Description:

- a. INSERT new object in user's *language* array.

- b. Create a *post* for a new update for the user.
- c. INSERT this into *my_items* .
- d. INSERT this information into the *activity* array of the user.

44.Name: Save Post

Description:

- a. \$push ObjectId of the post into *saved_posts* array of the *user*

45.Name:Save Jobs

Description:

- a. \$push ObjectId of the job into the *saved_jobs* array of the *user*.

46.Name:View Notifications

Description:

- a. FIND all the notifications in the *notifications* array of the user.

47.Name:View my_items

Description:

- a. FIND all the objects in the *accomplishments* and *background* fields of the user.

48.Name:Change open to settings

Description:

- a. UPDATE *open_to* boolean settings of the user document.

49.Name:View activity

Description:

- a. FIND all the activities in the *activity* array of the user.

50.Name:Request Recommendation

Description:

- a. \$push ObjectID of user_1 in the *recommendation_requests* array of the user_2

51.Name:Respond to a Recommendation request

Description:

- a. If accepted \$push ObjectID of user_1 into user_2's recommendation array.
- b. \$pull the ObjectId from the *recommendation_requests* array of the user.

52.Name:Apply for jobs

Description:

- a. \$push ObjectId of the user into *applicants* array of the user
- b. \$push ObjectId of jo into *my_jobs* array of the user