Joins

For practice, I'd suggest please solve all questions from the links below. Practicing problems from just one db makes things a bit redundant, the more examples you can solve, the better.

SQL JOINS — Exercises, Practice, Solution
https://www.w3resource.com/sql-exercises/joins-hr/index.php
https://learnsql.com/blog/practice-sql-joins/

Joins is an extremely crucial topic, please do not overlook it. All job interviews and any technical interview will test your SQL skills. It's one of the top most used and tested skill and known to all developers, and joins is easily the favorite questions to test students during interviews.

Also sharing some past interview questions. They range from easy to difficult, so please have a look and see if you can solve all of them. If you cannot solve them, now is the time to prepare better, you have half a SEM ahead of you.

https://practice.geeksforgeeks.org/tag-page.php?tag=SQL
https://hackr.io/blog/top-sql-interview-questions
https://towardsdatascience.com/sql-questions-summary-df90bfe4c9c
https://www.edureka.co/blog/interview-questions/sql-interview-questions

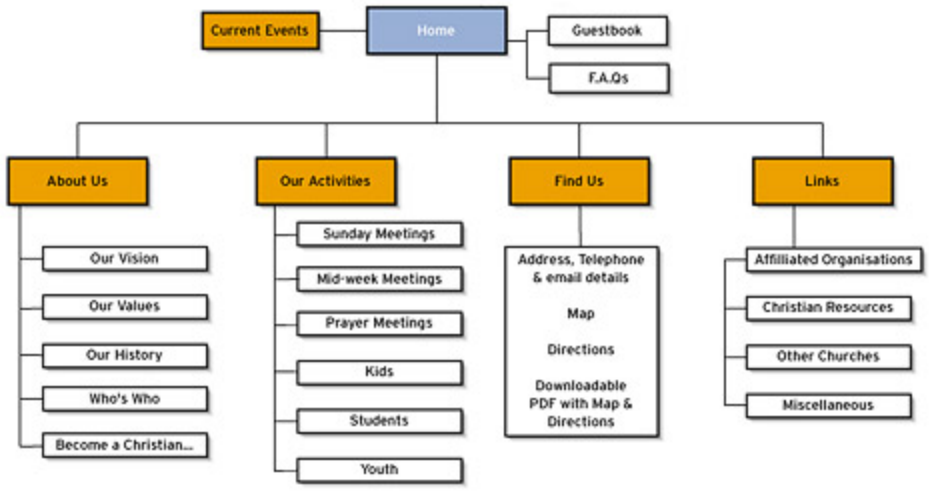
If there are topics you haven't done yet, it's okay, you will either learn it soon or if It's something very specific they might change the question for you because it's a vast subject.

For hardcore questions, lookup past IIT papers or Leetcode and such websites- they are very popular, hence not attaching them here.

Please check next page for more.

Project

1. Create a collab repo with all teammates and me
2. Create the basic file structure as asked
3. Add a simple readme.md for the entire repo (update it as you keep working on it)
4. Create a project board (start adding tasks if you want, the more you spread out your tasks over time are the more marks you are going to get for the overall development aspect)
5. In the documentation directory, add a PDF/MD file called Architecture.md/pdf (md will be easy to update)
6. This document must contain 4 major sections namely- Overview, Frontend Design, Backend Design, Database Design
7. Each section should have subsections/details as follows :
    a. Frontend
        i. Pages (dedicate one heading to each page) - describe the content and features of what's going to come for each page in your UI
        ii. Features — what features are your UI going to have other than the main idea, e.g. login/logout, register, settings page, etc. (remember these are very generic examples, you must provide ones specific to your project)
        iii. DB connectors if they are going to be used here (separate modules/pckgs only do not mix it up with ui code, import and use)
    b. Backend
        i. API endpoints(one heading for each) — describe the incoming parameters and the responses and its function
        ii. Features if any
        iii. DB connectors if they are going to be used here (separate modules,pckgs only do not mix it up with server code, import and use). Note : one separate class has to be made for each table or make db modular enough to accept db name and restrict functions. Since you are new to it I'd suggest making different classes.(will explain reason in class if needed)
    c. Database
        i. Tables and their names and purpose
        ii. Schema designs
    d. Overview
        i. Intro to the design of the project
        ii. A diagram representing front end back end and arrows showing how they connect (via HTTP API or via connectors of db)
        iii. Give a general flow of the application through a tree diagram (website navigation chart) e.g., :

1. 

DEADLINES

29th Oct 2021 — Version 1 of the architecture document (only the planning aspect, note this will also be graded)

The document (assignmet.md) need not contain all the info right away, but add as much. Before you come back from Diwali vacay need the first draft ready that is your first action plan, after that edits to the document will only be allowed if approved by me.

12th Nov 2021 — Version 2 (final version i.e. V3 will be before your presentation at the end of the SEM) submission for architecture document. (must contain all information with 90% final design and must have already started working on it, 10% changes will be allowed in v3, given adequate reasons)

You can make changes to the document without approval, but I will grade you on your older version only if you don't get the newer changes approved by me. Request for approval of change must make sense, if I see the older document was all rubbish and your actual stuff is only added later, I won't approve of it. Hence, do not fill nonsense on it right now, thinking you'll update it later.

Transition from v1 to v2 to v3 will also contribute to your grade. If it's not seen clearly and all work is only done in v2 or only in v3, you will lose marks.