A brief summary of your project:

We are making an application for the **co-op program**, which will provide storage and retrieval of data related to various aspects of a co-op program. Some key aspects of the domain that the database will support includes: student profiles, job applications, coordinator information, and application history. We are planning to use Java Swing for the GUI, and we will be working on this project end-to-end.

Tasks:

- Ensure SQL Script works as intended
- Create GUI
    - Things to learn
        - How to create a button and a table
        - How to place the button and table in the desired configuration
        - How to place data inside the table
        - How to trigger the data refresh once a button is clicked
        - Have Text Boxes for users to input data
    - Design Ideas
        - Have a Job Board table of all Job Postings
        - Job Board should be separated into columns (these columns come from the Job Posting entity's attributes: ID, Position, Duration, Deadline, Time Submitted, Date Posted)
        - The University Name (University entity's attribute: Name) and Department name (Coop Program entity's attribute: Department) will be displayed at the top of the Job Board
        - Upon clicking a Job Posting, user should be able to see information about the company that posted it (Company entity's attributes: Name, Address, Website)
        - Create a user profile page that will consist of:
            - Student entity attributes (Student #, name, email, year, # of applications)
            - All the applications that the user has submitted (Application entity's attributes: ID, DateSubmitted, TimeSubmitted)
            - The number of interviews the user has coming up
            - User's coop advisor and their contact information (Coop Advisor entity's attributes: Name, Email)
        - For every query option, there will be a button that can be pressed that will bring up textboxes that users provide inputs into to customize their query
        - Ensure relational algebra correctly outputs the correct results

Timeline:

*Dates are subject to change depending on which parts take longer or shorter than expected
There is no specific task distribution, we have agreed upon all of us working on every task

Week 0-1 (Learning basics) (10/29-11/04):

- How to create a button and a table
- How to place the button and table in the desired configuration
- How to place data inside the table
- How to trigger the data refresh once a button is clicked
- Have Text Boxes for users to input data

Week 1-2 (Frontend) (11/05-11/11):

- Have a Job Board table of all Job Postings
- Job Board should be separated into columns (these columns come from the Job Posting entity's attributes: ID, Position, Duration, Deadline, Time Submitted, Date Posted)
- The University Name (University entity's attribute: Name) and Department name (Coop Program entity's attribute: Department) will be displayed at the top of the Job Board

Week 2-2.5 (Frontend) (11/12-11/15):

- Upon clicking a Job Posting, user should be able to see information about the company that posted it (Company entity's attributes: Name, Address, Website)
- Create a user profile page that will consist of:
    - Student entity attributes (Student #, name, email, year, # of applications)
    - All the applications that the user has submitted (Application entity's attributes: ID, DateSubmitted, TimeSubmitted)
    - The number of interviews the user has coming up
    - User's coop advisor and their contact information (Coop Advisor entity's attributes: Name, Email)

Week 2.5 - 4 (Backend) (11/16 - 12/01):

- For every query option, there will be a button that can be pressed that will bring up textboxes that users provide inputs into to customize their query
- Ensure relational algebra correctly outputs the correct results