

This is a guide to various aspects of the course, including how to attend lecture, visit TA hours, hand in assignments and quizzes, and use GitHub. If you're looking for support on Golang or SQL, please check out the respective guides, accessible via the home page. If you have any questions please feel free to contact the TA's at [cs1270headtas@lists.brown.edu](mailto:cs1270headtas@lists.brown.edu) or ask your question on Campuswire.

---

## 1 Lectures

The following is a step-by-step guide on how to join lectures.

1. Find the Zoom link on the course website or on the course calendar. Click on it to join the class Zoom call. If prompted for a password, find it on Campuswire.
  2. If you have a question during class, either use the “raise hand” feature or type your question into the chat. Your question will either be answered by the Professor or through chat.
  3. While attendance at lectures is not mandatory, we **highly** encourage you to come to lectures!
- 

## 2 TA Hours

The following is a step-by-step guide on how to get help at TA Hours.

1. Navigate and sign in to [SignMeUp](#) using your Brown login.
  2. Find the TA Hours queue for CSCI 1270 and click on it.
  3. Click “Join Queue” and type in a **detailed** explanation of why you are at hours. Be reminded to follow our Hours policy, available on the home page of the course website.
  4. Join the Zoom waiting room or head to the CIT room indicated on the queue and wait to be helped by the TA. If you are not available by the time you are called, the TA may move on to help other students.
  5. Get help!
- 

## 3 Quizzes

The following is a step-by-step guide on how to submit quizzes.

1. Quizzes will be timed and administered through Gradescope. You will be able to start the quiz any time on the day of the quiz; however, once you begin, you will have to submit the quiz within 90 minutes. When you are ready to take the quiz, navigate to the corresponding assignment in Gradescope and click into it.
  2. Your time will begin when you click “Start Assignment”.
  3. Complete the questions in the quiz.
  4. When completed, click “View your Submission” and “Submit”.
  5. You will be allowed to resubmit until your time is completely up.
-

## 4 Handin

Written assignments will be handed in through Gradescope. Written assignments don't have to be typeset in LaTeX, but they do have to be handed in as PDFs. Find the corresponding assignment on Gradescope and click "Submit Assignment". Upload your PDF handin and select the pages that correspond to each question. We highly recommend/appreciate that you separate questions into separate pages, so no page contains more than one question - this makes the grading process much smoother for us. When you are done, you should receive an email confirmation.

Coding assignments will also be handed in through Gradescope. Find the corresponding assignment and click "Submit Assignment". Choose the GitHub repository that corresponds to your assignment and choose the branch you'll be handing in. Click "Submit". After handing in, the autograder will verify that your project is formatted and structured correctly and run our test suite. After confirming that your project was handed in and autograded correctly, you may close the window.

---

## 5 Grading

All assignments will be autograded on Gradescope. If you have a question about your grade or are certain that a test case that failed should have passed, submit a regrade request detailing your experimentation process. The grade proportions between each type of test will vary depending on the assignment; consult the Gradescope assignment for the full, up-to-date rubric.

---

## 6 LaTeX

We recommend typesetting your homeworks in LaTeX where reasonable. We strongly recommend using [Overleaf](#) to write LaTeX in. Here is a [cheat sheet](#) and a [tutorial](#) to help get you started with LaTeX. We also have an [in-house cheatsheet](#) and [homework template](#) for you.

---

## 7 Git and GitHub

GitHub is a version control system system built atop Git. GitHub enables you to store your code in an online repository and share it with others, which is great for collaboration and, in the case of this course, handing in assignments. We'll be using GitHub Classroom to distribute stencil code for assignments, but most of your work in this course will be done on a single repository, which you will submit multiple times throughout the semester as you build out more of your database. You can create a Github account [here](#). Let's learn some basic Git commands.

- `git clone <url>` creates a local copy of an online repository.
- `git remote add origin <url>` links a local repository to an online repository.

- `git add <file>` adds a particular file to your repository, and `git add -A` adds all files in the repository folder to the repository.
- `git commit -m <message>` commits what you have added to your repository to the repository history.
- `git push origin <branch>` pushes your commits to an online repository.
- `git pull origin <branch>` pulls commits from an online repository. **Always run this before you start developing and before you commit!**
- `git stash` saves your local changes and reverses the state of your folder to the last commit. `git stash pop` or `git stash apply` reapplies the saved changes. This is useful for dealing with merge conflicts.
- `git checkout <branch>` switches to a separate branch of your repository. To create a branch, run `git checkout -b <new_branch>`.

There are many more commands to learn, but they will likely not be particularly useful in this course, seeing as we don't do any pair or group assignments. If you prefer working with a GUI, we recommend [GitHub Desktop](#).

Be sure to commit your code often and write detailed commit messages. This helps you look back on your code to find bits and bobs that you may have implemented in the past, especially once you have moved on from this course. Read [here](#) to learn how to write a good commit message.

---

## Feedback

As this is a new course, we appreciate any feedback you have for us! If you enjoyed this assignment, hated this assignment, or have other thoughts to share, please do so [here](#)!