Code Fellows

Instructor Handbook

Crafting an Excellent Educational Environment

Introduction

Welcome to Code Fellows

Welcome to our team of world-class instructors! We're glad to have you on board. We've captured many of our best practices in this handbook and linked pages. This document will help you get oriented, and will serve as your course material during training, as well as be your guide before, during, and after you teach your class. **Read it now, refer to it often, and please make suggestions for how we can improve it.**

When using this document, we recommend taking advantage of the Document Outline feature **(View > Show document outline)**. This feature will help you navigate the handbook as you're scrolling its pages.

Also, view the pages in Print layout (**View > Print layout**) so that all page numbers, headers, and footers are visible.

The Code Fellows Way

We Change Lives.

Our curriculum and class objectives are 100% geared toward creating great professional software developers. Graduates grasp the most relevant practical skills, have product sensibilities, and can work in technical teams. Theory is introduced to teach students to think or solve problems, but not for the sake of discussing an "interesting theory" alone. We have placed hundreds of graduates into jobs where they get to be creative and valued, in many cases doubling their annual salary.

We Are Empathetic.

At Code Fellows, empathy is the most important skill we practice. It leads to greater success personally and professionally, and as a bonus, it increases happiness the more it is practiced. Read more on this topic now, in this article, Your Most Important Skill: Empathy, by esteemed developer Chad Fowler.

We Empower Adaptability.

We are considered successful when we teach students how to approach new problems and quickly solve them, as well as to "learn how to learn" new skill sets and technologies. All of the classes are based on a compressed, practical, hands-on curriculum.

We Cultivate Engagement.

The style of education is learn-do-learn-do, in rapid succession. The role of an instructor is to guide the students through what they need to learn, give direct feedback on how well they are doing and how to do better, and support them through the technical and emotional journey that an intense environment and large change inspires. Our instruction is very interactive. Students invest a lot of time and money to take our classes, and this is a high-touch business. Instructors and their team are the first and primary contacts for this experience.

Based on educational theory (read a short summary of andragogy: *see also* Pedagogy vs Andragogy), industry standards, professional best practices, and **lots** of iteration, we have identified some very effective instructional methods.

> *We cannot teach people anything; we can only help them discover it within themselves.*
>
> \- Galileo Galilei

Your Training and Onboarding

Just as we are invested in the success of our students starting new careers as software developers, we are equally committed to ensuring every instructor gains the skills necessary to teach effective courses.

Instructors who complete our training will gain the knowledge and skills to effectively deliver a Code Fellows course based on our proven curriculum, achieving a Net Promoter Score (described below) of 85 or higher. You will obtain a Certificate of Instruction for each course you are trained to teach, by completing the course-specific training and passing a course-specific certification exam.

This training consists of working through the Basic Instructor Training Canvas course, in which you will enroll as a student. In that course, you will:

1. Demonstrate that you've read this handbook in its entirety.
2. Learn our basic instructional process.
3. Work through course-specific training for each course you are scheduled to teach.

Your timeline for completing these courses will vary based on what and when you are scheduled to teach. Your manager will work with the VP of Education to coordinate a personalized training plan. This plan will include training on general teaching, instructor best practices, and specific course materials.

Basic Instructor Training

Instructors are the key architects of the educational environment that supports rapid skill acquisition. In the classroom, at the lab, and through individual interactions, you get to act as coach, mentor, teacher, and guide to students who are exploring a strange new world. The cultural norms you enforce create the space and provide the direction for students to do their best work. The Basic Instructor Training course presents the effective tools and techniques we have learned over the years.

Enroll now in the Basic Instructor Training Canvas course.

This course is a set of assignments with the best guidance we can give you for instructional success. These assignments are a combination of independent study and presentation practice to prepare you to lead classes the Code Fellows Way.

We highly recommend two books for additional key insights.

Book 1: Growth Mindset

In *Mindset*, Carol Dweck presents her research on how the perspective of the learner can have a massive impact on the effectiveness of the learning. As an instructor, you'll want to absorb this material both as a learner yourself, and as an educator.

Book 2: What Great Trainers Do

The second book is *What Great Trainers Do* by Robert & Dorothy Bolton. Written for providing repeatable successes in the world of corporate training, this is a collection of detailed best practices that can serve as a default guide when you aren't sure what to do. If you've had some teaching experience, you'll likely find it very helpful. If you haven't spent much time in front of the classroom yet, don't worry: just know this guide is here to help you when you need it.

Basic Instructor Training also covers the other "pillars" of our learning environment: empathy, growth mindset, psychological safety, radical candor, diversity and inclusion.

Course-Specific Training

Each course we offer has distinct goals and learning objectives. Instructors must understand the unique qualities of each course, and be prepared to guide students through the challenges they will face. Your training continues with Course-Specific Training in the modules relevant to your teaching schedule.

Train on specific courses as you prepare to teach each one. You'll complete key milestone assignments using the same approach as your students. Working with students in the course gives you direct experience with the skill level you'll be teaching. Connecting with TAs reveals the typical challenges of the course. Further presentation practice helps you dial in the targeted terminology and most relevant techniques.

Course-specific training is most effective when you are able to join a currently-running course of the same level. Coordinate with your manager to get enrolled to train in each course as it runs.

Evaluating Your Progress

Feedback is a gift. Code Fellows encourages and expects all instructors to strive for mastery in their educational and technical skills, even beyond initial training. To this end, we provide ample opportunity for you to receive and respond to feedback on your teaching. Being open to Radical Candor is essential here.

You'll get to be observed by fellow instructors and your supervisors. Observers will benefit from watching what you do too, and how you do it. Observers will write up comments for you based on the concepts in this handbook, as well as the skills outlined in the Instructor Skills Mind Map. You'll also have the chance to observe other instructors in action, and provide feedback for them using the Instructor Observation Form.

Follow this process, when doing an observation (or being observed):

1. Arrange an agreeable time to do an instructor observation. If remote, or if the timing doesn't work out to observe in person, have the observed instructor share a video link.
2. Arrive promptly, to make the most of the scheduled time.
3. The observer will record detailed, nit-picky notes in the observation form linked above.
4. Submit the form; it will email the observer a copy of the notes.
5. After the lecture, the observer will review notes verbally with the instructor observed.
6. Finally, the observer will forward the email with notes to the instructor observed for their reference.

Capturing Feedback as an Observer

Progressively take notes on your *in the moment* thoughts as the instructor in training makes their way through the lecture material. For each note bullet point, preface your comment with a "+" or a "-". Here is a breakdown of what each of those symbols means:

- **+** good lecture/presentation moment - meets or exceeds expectations of the audience
- **-** area of growth within the lecture/presentation moment

When sharing your notes, remain candid and kind. Share best practices. Allow time for questions and additional discussion on lecture strategies when applicable.

Responding to Feedback

When reviewing the comments of someone who has observed you, you have the opportunity to get better at your craft. Remain in a growth mindset. Take written notes to refer to later. Talk through what you might have done differently. Show appreciation for the observer taking the time to watch you present, and sharing their thoughts.

Standard*: Every month, you observe another instructor, and are observed by another instructor, to continually improve your teaching craft.*

Education 101

Introduction to Teaching

Getting to Know Code Fellows

Take some time to read About Code Fellows in the Operational Handbook. The mission, vision, and values capture who we are and what we are boldly attempting to achieve. It’s up to you to live these out, as it’s the foundation of all we do.

Professionalism

Code Fellows is committed to an educational environment where all individuals are treated with respect and dignity. As a mentor and instructor, your behavior needs to establish a professional atmosphere. Please remember not to physically touch students without permission.

Take care to address individual students using their identified pronouns (see the daily Instructor Course Enrollment Report). Update your vocabulary to stop using gendered group pronouns: prefer something like "Hi devs" or "GM Team!" to “Hey guys!”. And never refer to any student or coworker by a pet name (“honey,” “sweetie,” etc).

Don’t show favoritism, but do celebrate success with your students. You are welcome to use language creatively, but keep in mind some students will be offended by profanity, you don't know who else is in the room listening to the lecture or the playback of the recording.

Each individual has the right to study and work in a professional atmosphere that promotes equal learning opportunities and prohibits unlawful discriminatory practices, including harassment. Therefore, Code Fellows expects that all relationships among persons in the school will be business-like and free of bias, prejudice, and harassment. If you witness or experience anything to the contrary, please immediately contact hr@codefellows.com.

Further examples and definitions are in our Code of Conduct. You are expected to be fully familiar with and to abide by it. Read it now if you haven't already, or review it now as needed. If anything is unclear, or unacceptable, please raise your concerns immediately.

Students are held to a high standard of professionalism as defined by our Professional Competencies, distilled from a number of internal hiring and promoting guidelines from our partner companies. Since you will be evaluating students on how well they behave to this standard, you should exhibit exemplary behavior on all points.

Tooling

In your first few days at Code Fellows, you'll be paired with a trainer, and you'll see email from our HR system which will walk you through some onboarding tasks. These will include getting you set up with our standard toolset, including:

- **Google Apps**: We use this for email, calendar scheduling, lecture slides, and documentation. You'll get important meetings scheduled during this step, including weekly sync meetings with your manager, all-team meetings, and a weekly all-instructor meeting. If you can't see these, please alert your manager.
- **Slack**: We use Slack for chat-based messaging, communication that's nearer to real-time than email.
- **GitHub**: We create course-specific repos within the Code Fellows organization, and use GitHub for assignments and their review.
- **Canvas**: This is the learning management system we use to manage specific course instances and cohorts, and it complements the assignment reviews we do in Github. Many specific aspects of Canvas require their own introduction (e.g. arranging class times and assignments in the calendar, inclusion of pre-work, weighting of assignment groups, using the Gradebook, etc).

- **Asana**: This is our project and task management tool. You won't likely get set up with Asana, because instructors generally track tasks through GitHub. But if you end up using something else, it will be Asana.
- **PassPack**: We use this to manage Code Fellows credentials; if you create an account needed for your work at Code Fellows anywhere, be sure to add the credentials to this tool.
- **Invision Freehand:** Use for all Whiteboard illustrations during class. Allows collaboration with students during lecture times, and easy drawing management and sharing. Always use the text tool for text, shape tool for nodes, and the pen tool for arrows. Learn the shortcut keys for each tool.
- **Audio/Video recording tools**: We capture course lectures using cameras installed in the classrooms, as well as screen-captured audio/video using best of breed OS-specific tools (please review the SOP). The videos are shared online in a playlist created for each course instance.

Conducting Class Remotely with Zoom

Zoom is conferencing software which will allow you to broadcast audio, video, and your screen to multiple participants. It also allows for some interactivity.

It is free for basic usage, which is pretty full featured. The main restriction of the free plan is that calls with MORE than 2 people can only be up to 40 minutes long.

Therefore you will be provided with a Zoom license at your company email address for class instruction to allow for larger groups and longer sessions

Software Installation

Install the Zoom Desktop Application on your laptop.

Zoom Configuration

Follow the instructions in the Screen Recording SOP to ensure your Zoom account is correctly configured.

Class Operations: Lecture

Cloud Recordings

Properly configured, Zoom will record your lecture to the cloud, from when you start recording until you end the class. As soon as the recording is available for playback in the cloud, you will get an email with the video link, and a passcode.

Copy the access info, and paste it into the section at the top of the syllabus of your course in Canvas, linked from the class number.

Whiteboard

Zoom has a whiteboard mode. But it has some limitations. We instead use InVision Freehand for whiteboard, and share the screen over Zoom.

In a classroom, we recommend keeping an agenda on the whiteboard. In a virtual classroom, you can write out the agenda for the day, save the image, and use it as a virtual backdrop for your video feed.

Operating a great class remotely

1. Use multiple screens if you can.
    a. Share your working desktop screen with the students
        i. VSCode
        ii. Browser
        iii. Terminal
    b. In your second (un-shared) screen, open the other support apps.
        i. Helper code
        ii. Notes
        iii. Slack
            1. Like always, we don’t want students seeing our Slack.
        iv. Zoom menu bar
            1. This can get in the way as you’re operating your windows and workflow.
        v. Zoom participants list

            1. Good for a reference, but takes real estate.
        vi. Zoom chat window
            1. Good to have this going, but private messages from students show up here, too, so you don't want this in front of the class.

2. Use the participants list to "Call On" students.
    a. Alternately, create list of students before each class and post list to class Slack channel.
3. Require students to turn video on so that everyone can see each other. This lets you gauge understanding and keeps them more accountable.
    a. If students don't have a webcam, recommend one to them to buy.
4. Allow use of the "Annotations" feature (in the Zoom toolbar)
    a. It's a fun way for the students to raise their hands.
    b. It can also allow them to circle things on screen, point things out, etc.
        i. These are things we can't do in the regular classroom, so take advantage of the opportunity!

Methods of getting student engagement

Zoom offers a number of ways that students can potentially engage and interact with the presentation while it's happening.

1. "Raise Hand" button notifies the presenter that a student has raised a hand.
    a. This allows students to answer questions along the lines of "Who thinks it will happen this way?"
    b. Other non-verbal feedback options in Zoom include:
        i. Go slower or faster.
        ii. Time for a coffee break.
        iii. And more!
2. Annotations allows students to draw arrows, pen lines, or stamps (checks, "?", etc.)
    a. Students can interact live with the screen.
    b. These are things we can't do in the regular classroom, so take advantage of the opportunity!
    c. Students can use stamps to "star" or "?" lines of code that they do or don't get.

    d. It can also allow them to circle things on screen, point things out, etc.

3. Presenters can give keyboard and mouse control to have students take a stab at typing a line of code.

It's up to you to leverage these aspects of the tool.

There should be an expectation that students will talk. Call on them, in order, going down your Participant List. If they happen to be in a place where they can't use audio, they can respond in chat (in which case you may want to give them a yes/no question or a fill-in-the-blank).

Use the Breakout Rooms feature to create small groups where a few people can discuss a topic at a time. This is useful for taking 2 minutes to "tell your neighbor" how something works (in groups of 2-3), or for doing small-group code reviews (in groups of 3-4). These rooms are configurable on the fly, and can have students randomly distributed.

Utilize Polls as a way of getting anonymized feedback as you work through lecture. You'll need to prep the polls ahead of time, but once "voting" has closed, you are able to show the general results to the class. It's best to keep some generic polls handy, like for requesting a "Fist-to-Five" score on how people are feeling about a certain topic, or a simple "Option 1, Option 2, Option 3" poll, where you are verbalizing what the options are (eg: predictions for what will happen when the code runs?) while students are voting.

Class Operations: Lab

Without being able to gather in person, lab time presents some interesting challenges. Fortunately, the tools available go a long way toward mitigating these concerns.

Generally, even in person, TAs can only help 1 person at a time, or a few folks in a small group. This can be replicated in Remo, an event management platform. Instructors and TAs should "make the rounds" to check in on students while they are working.

If there is a backlog of students needing help, requests can be queued by posting questions to the Help Queue sidebar, with details added about their question and where they can be found. TAs can then address issues in order that they are posted. TAs will prioritize students who have lab time during the TAs shift, e.g. helping nights-and-weekends students during the evening TA shift.

Conducting Presentations Remotely with Zoom

The most important thing for presentations is to work with your teams to practice ahead of time. Require teams to go through the full sequence of handoffs. If they get it wrong, or it isn't smooth, have them do it again.

The instructor of the presenting class, or their delegate, will host the Zoom call.

Use Zoom to schedule the call ahead of time, with the following settings:

- Name the meeting after your course code, i.e.: “Final Presentations: seattle-java-401d7”
- Uncheck require meeting password for publicly-accessible presentations.
- Turn video for hosts and guests on.
- Allow telephone and computer audio.
- Under advanced options check “mute participants on entry”.
- Do not set “record meeting automatically”; rather, be sure to begin recording when everything is situated, and the actual event is starting.
- Gather presenting students 15 minutes prior to do microphone, camera, and screen sharing checks.

Students are required to be on video. When a team begins to present, have students introduce themselves before their team driver shares their screen. This lets each student be on the main screen for everyone to see, which leads to better pitches. Have them start the screen share after all the personal introductions are finished.

Students in groups not currently presenting should have microphones and cameras off.

These are the important core tools you’ll need to be well-versed in, so feel free to ask lots of questions and spend time learning them. Get to know them well! Doing so will complete the first step in your training, and you'll be ready to move on to the next step: learning the foundational principles of teaching at Code Fellows.

Education 201

Foundations of Instruction

In this stage of your training, you'll learn how we instruct at Code Fellows at a more granular level, and understand what it means to master instruction at Code Fellows.

Prior to Course Start

Before teaching your first course at Code Fellows, you must understand the differences between pedagogy and andragogy, progressing from instructor-led to student-led learning (read how the Girl Scouts approach this), multiple intelligences, flexible strategies for teaching diverse learners, classroom management, and other topics in teaching and learning that impact a student's ability to learn. You'll also understand the principles of assessment, both formative and summative types. You'll know how to administer each type, and know how to use assessment feedback of both kinds to modify your teaching and counsel students to improve their performance.

Learn the curriculum by familiarizing yourself with the materials available. Everything needed for a given course can be found in a "Guide Document" Google Doc found at the course's folder in Google Drive.

On the first day of class, introduce your students to their most important course assets: peers, TAs, you, and the Code Fellows team *(as per the Orientation slides)*. Invite any TAs in attendance to introduce themselves, including a sentence or two regarding their background and their motivations for choosing to serve as a TA. Then, provide a high-level roadmap for the course using our standard slide materials.

Familiarize yourself with the tools you'll use as a Code Fellows instructor, so when you are in front of students, the focus is on the content.

Double-check the schedule. Are there any conflicts with your personal schedule? The first step in requesting time off is proper planning. As our courses require an instructor to be present for lecture, it's vital that you manage coverage amongst your fellow instructional team. If an Assistant Instructor is co-teaching the course with you, they are your first line of defense. If you do not have an Assistant Instructor, reach out to your Principal Instructor to get assistance in determining the best path forward. To request a day off, fill out the time off request form in BambooHR. This will send the request to a manager, where they can review and approve or deny from there. To avoid a denied request, be sure to fully note the plan for coverage as part of your request. If you are looking to take a partial day off, this process is not required; however, you'll still need to manage the time you are away with your manager and fellow instructors. Whenever you are out of the office, be sure to denote it on the "Out of Office" shared calendar, and block your personal calendar so the team knows you are not available for meetings.

During Your Course

Give a roadmap progress update to the class on a daily basis, so students get a sense for where they are in the course timeline. Also provide a why/what/how outline for each high-level topic, so that students have a context for what they're learning.

Great instruction at Code Fellows relies on the following tenets:

- We favor interaction and practice over lecture and theory.
- We're at the service of our students to keep them challenged—not to make it easier.
- Be yourself; you're part of the team because you're a culture fit here.

Students will expect you to know all about the inner workings of Code Fellows. Don't worry if you can't answer a question, but do help the student find the answer, and learn it yourself, so you can answer next time. Don't pretend to know more than you do.

During lecture, spend some time coding in real time, projecting your screen so that students can follow along. Pause and check for understanding. Ask a question and wait, for at least ten seconds at a minimum, but waiting 30 seconds or even a whole minute is acceptable. This is an uncomfortable amount of time to wait, but it takes students time to formulate questions and work up the confidence to ask them. Or, for instance, you can tell the students to try to formulate questions while you go refill your water bottle, or say you are going to stretch for a few moments (so that you're not just sitting there looking at them while they are thinking). Call on your students, and have them walk through how they came to the answer.

Lead by serving, without serving up the answers. Regularly ask yourself what you'd expect as a student, and gauge your performance accordingly. Stay on top of the weekly student survey feedback for your course. Reviewing the Retro/Learning Journal assignments for each class is also a great way to spot trends and issues as soon as possible.

It's up to you to keep the instruction team up to date with grading. Fast and regular feedback is a turbo boost to the learning process. The ideal class is a dialogue on both subject matter and process.

Stay healthy! Get your rest, and don't wear yourself out. Should you get sick during a course, your students will understand. Be sure to immediately inform your Principal Instructor at the first sign of illness, even if you think you might still make it into the classroom. If you know you can't come in, work with your Principal Instructor to get the course covered.

After Your Course

Rest up! Take some time off, if your schedule allows. Take a deep breath, and give yourself the mental space to relax, before diving back into your next course.

Health and safety are key professional competencies. Self-care is a major component of this. If you are feeling overloaded, or approaching burnout, be sure to immediately communicate that to your manager.

Education 301

Intermediate Instruction: How to Succeed

In this stage of your training, you'll learn what sets a good instructor apart from a great instructor at Code Fellows. Being prepared is key.

Prior to Course Start

As a Code Fellows instructor, your high-level goal is to help aspiring students prepare for a new coding job by guiding them along a very challenging path. There's a high bar for 401 course enrollment, which is handled by our Admissions team. Once accepted, not all students will pass the course, and for those who do, not all will get jobs in the industry. But when it's complete, and students are asked "would you recommend this course and your instructor to a friend with similar interests," *every one of them* should respond "yes."

Net Promoter Scores

Your efforts will result in high graduation rates, high job placement rates, high instructor ratings, and high Net Promoter Scores overall for the course. If you're not familiar with NPS, get familiar now. It's based on asking a single question with standardized language used across industries: "How likely are you to recommend [Company] to a friend or colleague?" Answer options range from "0 - Not at all likely" to "10 - Extremely likely." Promoters are those who answer with 9 or

10, and Detractors are those who answer with 6 and below. The “net” promoter number is calculated by subtracting the percentage of detractors from the percentage of promoters.

We feel NPS is one of the best ways to gauge the value we’re adding to the world, and we take it quite seriously. Combined with our standard survey questions, it gives us a proven and reliable metric of quality that helps us identify issues as they arise, rather than letting problems fester and explode.

Course assignments include student surveys, which contain NPS questions. It’s not easy to hit the **goal of 75** for a given course. For example, in a cohort of 20 students, the only way to reach that goal of 75 would be to have:

- greater than, or equal to, 15 promoters and no detractors,
- 16 promoters and 1 detractor, or
- 17-18 promoters and 2 detractors.

So, plan on doing what you can to optimize for the best results, ahead of time. Plan on emphasizing the importance of the weekly surveys repeatedly, without saying too much. For example, **“We love 9’s and 10’s, and if not, I’d love to know what I could be doing differently”** is fine, while an in-depth explanation of NPS isn’t.

Students also rate your performance on an NPS scale on their final survey. Similarly, a good score is 75, and our top performers regularly hit above 90.

The Admissions Process

Familiarize yourself with our admissions process and the steps required to become a Code Fellows student, including when and how students can test into a given course. Study our website and examine our course offerings in detail. Familiarize yourself with our Career Transition Services, made up of our Career Coaching curriculum and the Career Accelerator program. Read our School Catalog and what it references. Read our TA Handbook and know it well.

***Standard**: For students testing into Code 301 or Code 401, if they don’t pass on the first attempt, they can study and retake the admittance test one more time. If there is any concern* *with the test results or project submitted, it is ultimately the instructor's discretion as to whether the student is accepted to the course or not.*

During Your Course

Working With Feedback

The path toward mastery contains ongoing training loops and feedback loops; it's all about continuous improvement. You'll have a weekly sync meeting with the Principal Instructor, where you'll review course feedback from students. Prepare to be objective, and take constructive criticism in stride.

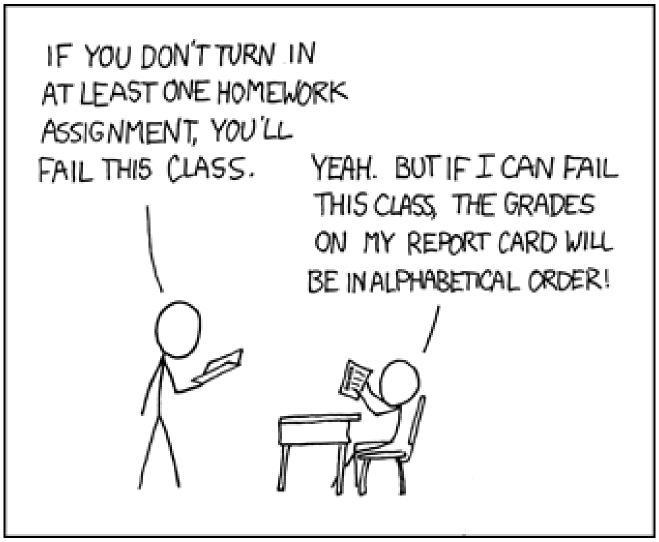
At the start of every week, take the opportunity to put some of these comments on a slide during your opening lecture and address them directly. Students appreciate being heard and having their concerns addressed. Sharing with the class what others are feeling also validates the personal experience of those who might be struggling, or are having a hard time putting words to the challenges they encounter.

Students may also occasionally raise concerns in person, or in the feedback surveys, that aren't related to course content. If it is anything related to the Code of Conduct, it should be immediately forwarded to conduct@codefellows.com. Concerns related to facilities, campus life, or anything else can be passed along to the Campus Director. The Campus Director will escalate issues further if appropriate, or work with you to propose a resolution to the student.

Giving Students Feedback

Feedback goes the other way as well, of course. Students always know their status in a class when grading is kept up to date. We strive for a 24-hour turnaround on grading. This helps you keep tabs on how the class is doing as a whole, and which individuals are struggling the most.

When students are below the passing threshold and in danger of not being able to recover, they should be informed of their options as early as the end of the first week. Some students just need the warning, and can push harder to get above the line.


IF YOU DON'T TURN IN AT LEAST ONE HOMEWORK ASSIGNMENT, YOU'LL FAIL THIS CLASS.
YEAH. BUT IF I CAN FAIL THIS CLASS, THE GRADES ON MY REPORT CARD WILL BE IN ALPHABETICAL ORDER!

When students continue to struggle, or miss class repeatedly or unexpectedly, a further intervention is required. Escalate the case to the Campus Director, who can make direct contact with the student. The Campus Director will meet with the student (and potentially you), decide on a course of action, send a followup email with the details to all parties, and log the email in SalesForce.

Some students will need to be "released" to focus on some self-study. Since our primary goal is to encourage their learning and help them move into a new career, we are happy to offer them a prorated refund (but please leave all discussion of finances to the Campus Director and Admissions team). If the team agrees a release is necessary, you will be expected to provide a personalized learning plan that gives the student some next steps to take (as self-study), based on their current progress.

Our students generally know when they are behind. We want them to finish the class, but we need to set their expectations at regular milestones. Prior to your weekly meetings with your manager, you'll fill out the Weekly Course Status Survey report, which includes input fields for identifying students who are in jeopardy of not passing the course. As needed, the results should be sent to the Campus Director (via an incident report or tagging in the form results doc), who will present students of concern with options, including a partial refund (as allowed by their contract), helping them make a plan to buckle down, or reviewing their plan progress. It is crucial to also record feedback on how your TAs are performing on a weekly basis.

A Personal Learning Plan may be requested by the Campus Director for a student being released, which you'll then provide. This plan should include:

1. What course material should be reviewed.
2. Which assignments should be redone.
3. Any project meetups they should attend.
4. Any online classes they should take.
5. Further personalized recommendations.

Common Challenges

One of the most significant challenges in teaching is addressing the range of skills that students bring to the classroom. You will have folks who have been in the tech industry for years sitting next to newbies who are creating their first apps. This can present a few different challenges for you as the instructor. Generally, the course material should be flexible to give challenge to all, often with stretch goals in place for those who hit the minimum requirements easily. When grading top-tier students, be sure to give recommendations for what they can do to move their code to a more professional level. For those who are struggling, more gentle comments are appropriate; congratulating them for getting working code shipped is also good for these students, even if the work just meets the minimum requirements.

If students start class without having completed the pre-work assignments in Canvas, explain to them that they will now need to do so alongside other classwork.

Career Coaching Days

Students in Code 401 classes will have six Career Coaching workshops. These half-day sessions will be scheduled on Fridays in daytime classes, and on weekend days in a nights-and-weekends class. The workshop will mostly be run by the Campus Director, so you don't need to have any curriculum planned or prepped for these days. Topics include personal pitches, resume writing, interview practice, and job search tools. The exception is workshop 5 which is about the technical interview process. This will be led by you or another instructor.

After Your Course

At the end of a course, you'll do a Course Retrospective Meeting to review what went well, what can be improved, NPS data, final surveys, and next steps.

Any students who completed the class, but ended with a failing grade may retake the course with an automatic 50% discount applied, as a benefit of being a student with us. Share this info with the student and encourage them to continue their learning journey. Leave all further discussions of financing the course to the Admissions team.

Education 401

Advanced Instruction & Mastery

In this stage of your training, you'll learn how we instruct at Code Fellows at a more granular level, and understand what it means to master instruction at Code Fellows.

Prior to Course Start

Gather a Team

Assuming your cohort has met the minimum enrollment of 10 students, as a Lead Instructor you should have at least one Lead Teaching Assistant or one Assistant Instructor. TAs are typically students who have taken your course recently and have the aptitude to help other students. Assistant Instructors are instructors in training to be Lead Instructors, and you can expect to share course preparation, lecture time, grading, and student mentorship with them. We keep job postings open for TAs and Instructors, so if you have candidates you'd like to queue up, direct them to our website to apply.

> ***Standard****: In the weeks leading up to course start, you should coordinate with the Code Fellows support staff to ensure that there will be one person on your team for every six students.*

A day or two prior to course start, or on day one at the latest, meet with your team to sync up on roles and responsibilities, as defined in their job contracts. Stress that assignments are to be graded within 24 hours of submission. This level of feedback is exactly why students come to take our classes.

***Standard**: Assignments are graded and feedback is returned within 24 hours of submission; it's your responsibility to ensure that your team meets this goal.*

Communicate with Enrolled Students and Staff

There are three primary written communication channels you'll use with students:

1. **Slack**: Use Slack for ad-hoc communication with staff, students, and alumni. There are two teams we use: *Code Fellows Staff* for the internal team (messages persist, and are searchable), and *Code Fellows* for students, alumni, and TAs (the 10k most recent messages are viewable). Take some time to learn Slack's features— it's a powerful tool. In particular, know how to use *@channel* to send a time-sensitive message to a group of people, and how to use `backticks` to escape code examples you share. You and your cohort will use the following channels and private groups for communication:
    a. **Course staff groups**: you'll be added to private groups specific to the course level (and language if applicable), along with other staff, for broader cross-campus communication specific to your course.
    b. **Class cohort channel**: most communication will happen in a private class channel, named by your course code, e.g.: **seattle-201d1.**
    c. **Cohort teaching team group**: a private group for your team of TAs will be created, so you can check in with how grading is going, and they can discuss student struggles and successes with you.
    d. **Topic channels**: stay apprised of what's happening with alumni and your tech stack by joining any other channels that look interesting to you.
2. **Canvas**: Use Canvas to create long-term discussions, or to create anything you want to persist. Useful for setting up carpools, organizing neighborhood study groups, sharing group project ideas, or discussing any other topic that you don't want lost in the flow of Slack.
3. **Email**: Your students can be contacted directly via email. All emails to students should CC your Campus Director, and any other relevant instructional or administrative staff.

Prepare Course Kickoff Materials

It's important to set the foundation for a new cohort properly, right out of the gate. On the first day of class, there are some critical points you and the Campus Director should communicate to students. Verify with the Campus Director that these are included in your day-one course assets (e.g. 201, 301, or 401 slides), and are addressed in this order:

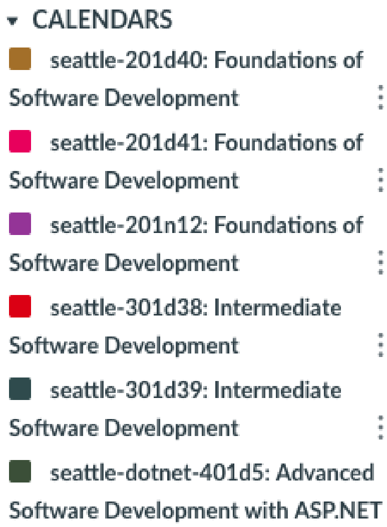
Set Up Canvas

Our Education team will do the initial setup of Canvas for you during the Course Roadmap Meeting, and you'll be invited to the course as a Teacher. You will publish the entire course two weeks ahead of the start date so that students can see and submit pre-work assignments for grading prior to the course start.

Start by configuring your own account settings, including crucial timezone settings and notification settings (pick how you'd like to be alerted to new submissions, etc). Complete your profile, including your picture and a brief bio.

Learn how to use Canvas as a teaching tool. Review what you learned in the TA Training course, and also ensure you understand Modules, Assignments, Quizzes, and the Calendar.

Your course material will be imported from the corresponding course template. Canvas will attempt to adjust dates automatically, but this invariably fails to line things up perfectly. Align all assignments around holidays as needed, and verify their availability dates, due dates, and due times are correct. For morning lectures, assignments are due the night after the lecture is given, at 11:59pm. For afternoon lectures that start at 2pm, assignments are usually due before the start of the next lecture.



▾ CALENDARS

🟫 seattle-201d40: Foundations of Software Development ⋮

🟥 seattle-201d41: Foundations of Software Development ⋮

🟪 seattle-201n12: Foundations of Software Development ⋮

🟥 seattle-301d38: Intermediate Software Development ⋮

⬛ seattle-301d39: Intermediate Software Development ⋮

🟩 seattle-dotnet-401d5: Advanced Software Development with ASP.NET

Nights and weekends instructors will need to ensure assignments are scheduled according to the recommended time table.

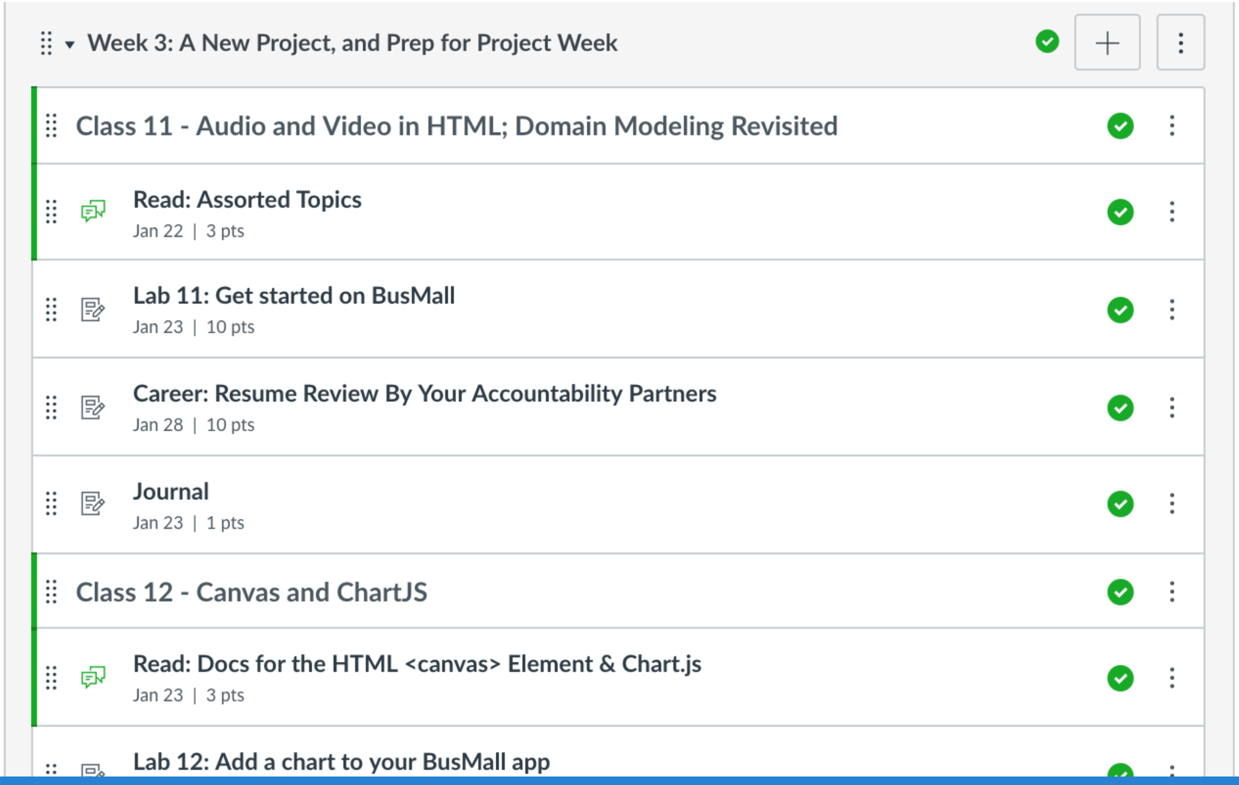
The easiest way to make these adjustments is from the Calendar view. Select your course from the "Calendars" sidebar. It may take a few seconds for data to load. You can drag and drop assignments to update their due dates. You can click an assignment on a calendar day, then click “edit” to update the due time as needed.

Once you have assignments on the proper days, review them from the Canvas Modules page. Each group of five classes forms a module, with “Text Headers” added to the module for each class number. Verify assignments are under the proper headers.

While you are arranging assignments, be sure to **read each one** and make any needed edits (via GitHub pull requests) to the wording of each for clarity. Use your best writing and text formatting skills to make each assignment comprehensible to students. Verify that links are going to the correct places.

Next, proofread and update the syllabus page. Dates, if any, are likely inaccurate; adjust as needed. Ensure teaching and support staff are correctly identified, with accurate contact information.

Unpublish everything except pre-work and reading assignments for the first week of class, and then publish the course. Then review and publish as much of the rest of the course as possible. At a minimum, publish each course week by the Thursday of the preceding calendar week at the very latest, to ensure students have a solid mental roadmap for where the course is taking them next.



Week 3: A New Project, and Prep for Project Week

Class 11 - Audio and Video in HTML; Domain Modeling Revisited

Read: Assorted Topics
Jan 22 | 3 pts

Lab 11: Get started on BusMall
Jan 23 | 10 pts

Career: Resume Review By Your Accountability Partners
Jan 28 | 10 pts

Journal
Jan 23 | 1 pts

Class 12 - Canvas and ChartJS

Read: Docs for the HTML <canvas> Element & Chart.js
Jan 23 | 3 pts

Lab 12: Add a chart to your BusMall app

Quizzes

Quizzes should be used to reinforce the key concepts from lecture. Questions should directly map to learning objectives. We use quizzes as a formative assessment, giving students a chance to practice recall of key ideas and reinforcing vocabulary. We use multiple-choice questions whenever possible so that students can immediately see the results. Quizzes are configured to allow unlimited retakes, so students can repeat the quiz as needed to get full points, and fully drill on those key concepts. Canvas is configured to keep the highest score. These quizzes are not about trick questions or making the students do further research. They are about assisting the learning process. Similar to the “spaced repetition” style of language learning, a few questions in every quiz should cover material from 2, 5, or 7 days prior, to aid students in recall of key concepts.

Reading Assignments

Reading assignments provide a preview of the day’s lecture topics. They have a due date of one minute before the class starts (e.g., 8:59 a.m. or 1:59 p.m.). All reading assignments should be created as Graded Discussion assignments. This allows students to see responses to the reading from fellow classmates. In Canvas, configure the assignments with “Allow threaded replies” and “Users must post before seeing replies,” so students will share their own thoughts, and then be able to comment on classmates’ submissions. The submission instructions will guide students about what to write in response to the reading (generally, “explain <topic> in your own words”), and also mention a recommendation to review classmate responses.

Attendance

Passing a class requires 90% attendance. The instructor has the responsibility of ensuring that attendance is taken on a daily basis. This can be handled by a Lead TA, if one is assigned to the class, but it is lead instructor's ultimate responsibiliy. Students should notify the instructor if they will miss a class. Exceptions to the 90% rule are handled case-by-case, at the instructor’s discretion. Your Canvas instance should have Roll Call attendance tracking enabled, which connects attendance records with an attendance assignment, and is given a certain number of points. Using the “Seating Chart” mode is recommended, as students tend to sit in the same seat each day.

When a student is absent without having notified an instructor prior, the instructor should message the student to let them know they were missed. If there is no response, or if the student misses class again, then the instructor should email the student (cc'ing the Campus Director and Admissions) with a friendly check-in, to see what's up. If the student has passed 50% of allowable absences, then the email should include a reminder of the attendance policy, specifying how many more absences will be allowed before the student is released from the class.

Once the student exceeds the allowed number of absences, the Campus Director can handle discussing release from the class with the student.

Validate Curriculum Assumptions

Our curriculum is reviewed regularly and adapted in step with the industry's ever-changing technical landscape. At this point, you should have reviewed the course syllabus. If some topics or tools seem out of date or questionable, take notes and include your recommendations and arguments for/against those topics or tools. If you want to validate any assumptions underlying your recommendations, find data from current job postings in your area. For further research, ask to be put in touch with our hiring partners, who will have the best perspective on what tools and technologies are needed most in the current market. Then bring your recommendations to the team via the Curriculum Change process. This process should be fairly quick. Please raise the alarm if that's not your experience—the last thing we want is to become a slow-moving bureaucracy, unable to change and adapt quickly! At the same time, with a distributed team, it's important to maintain consistency for our students, and have the ability to scale and run courses at campuses. Until you've received approval from the team at HQ, move forward with the existing curriculum.

***Standard**: Like many other things in the Code Fellows world, our curriculum is standardized for use amongst numerous instructors across numerous schools, and doesn't change without going through our review and approval process. This same general philosophy is applied to all shared instructional resources, whether they be GitHub repo path naming and layout standards, survey questions, Canvas setup, etc.*

Student Interviews

If you’re teaching a Code 301 or Code 401 course, you may be asked to help out with interviewing students who are testing directly into the course prior to course kickoff. This is done according to the Test-In SOP.

> ***Standard**: Student interviews are conducted using a three-step process to determine their likelihood of success, and advised on best next steps through empathetic mentorship. The output of this step is your notes on each student, captured in an email or shared directly with the Admissions team.*

During Your Course

Kickoff

Getting the course off to a good start on Day 1 is crucial for setting the tone and establishing good rapport with your students.

The opening will be handled by the Campus Director, to welcome students and give them some orientation to how our campus operates. You can then dive into brief **introductions** to help students get to know one another, you, and the TAs. Follow this with a course-specific **overview** to show students where you’ll be heading together. Finally, **prepare** students for their first lab assignment, with the course content for the day. Class 1 should contain meaningful learning, not just orientation.

The schedule looks like this for a daytime class that holds lecture in the morning:

1. 8:45 a.m.: Welcome party, optional for students, to greet new students on campus.
2. 9:00 a.m. sharp: Campus Director starts promptly with an Orientation Presentation
    a. Code 201 Kickoff Slides
    b. Code 301 Kickoff Slides
    c. Code 401 Kickoff Slides
3. 9:45 a.m.: 10 minute stretch break.

4. 9:55 a.m.: Return with introductions. Have each student answer these three questions in 30 seconds or less:
    a. "What is your name, and how can we remember it?"
    People often have funny connections they make to their names, which helps you (and classmates) remember their names better.
    b. "Where are you from?"
    Everyone has an area they identify with, and getting to share this is meaningful, especially when classmates find others from their same region.
    c. "In just 3 words, what's the why behind the why you are here?"
    People are here to get a job. But the reasons they came here for that are varied and fascinating. This question should be introduced with a brief explanation of the 5 whys style investigation.
    d. Help students learn each other's names by playing the Name Game. Have each student name all the students who preceded them in introductions. This is a great way to ensure every student knows their name was heard, and an excellent chance for you to practice learning names. Start with those who chose to sit in the front row to thank them for their eagerness.
5. 10:30 a.m.: Present a course overview, or "roadmap." Define any key terms needed for understanding the structure of the class (e.g., "front end vs back end" or "MVC"). Present students with a knowledge map of course topics, or a week-by-week or day-by-day topic guide. Go into detail on the teaching techniques you will use to set the expectation that it won't be eight hours (or even three hours) of lecture a day.
6. 11:00 a.m.: Take a ten-minute stretch break. Encourage everyone who stays in their seat to move around for at least a moment.
7. 11:10 a.m.: Resume exactly at the time you said you would, in order to make it clear that you know how to manage a schedule, and are serious when you tell students to come back at a certain time. Now is the time to dig into the course content for the day, and give students everything they need to complete the assignment(s) during lab time.
8. 11:59 a.m.: End promptly at the agreed time to break for lunch. As needed, give students direction on how to get to the coworking space, and tell them what they should do with their stuff if they want to leave the building.

Lecture

During Every Lecture

After the kickoff, you'll have a lot more freedom in what you do with the in-class component of your course. We call this "lecture" even though it isn't—and never should be—strictly lecturing. Plan to structure your day according to the recommended daily schedule.

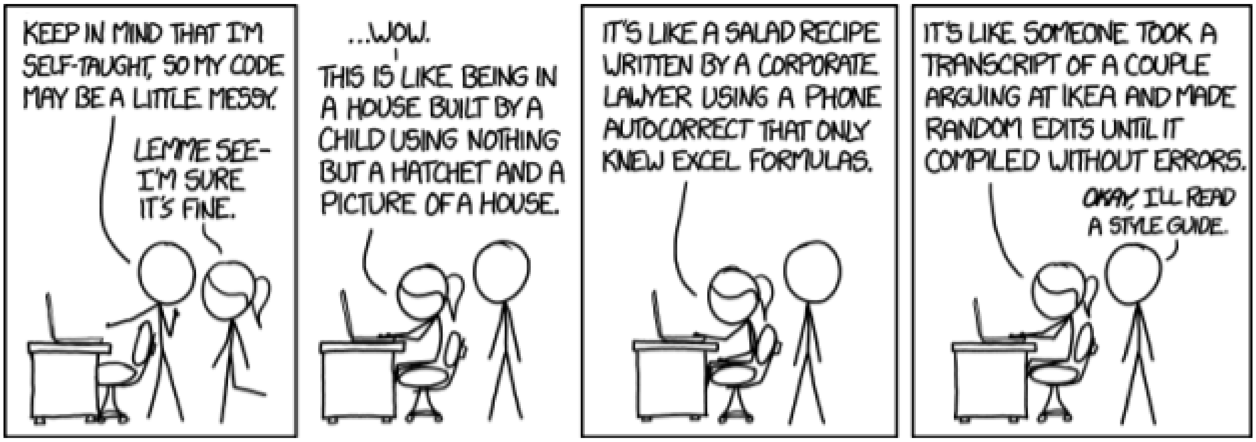
Before class starts, write a title and an outline for the day on the whiteboard. This should clearly connect to the overarching theme for the week, showing students: "Here's where we are today" in regards to the course roadmap. Draw a square next to each item on the board, and color it in as you complete that part of the lecture.

In-class lecture has two objectives on a daily basis:

1. Reviewing and discussing the lab assignments students have done previously.
2. Presenting new topics to prepare students for success in their following lab assignments.

Review Lab Assignments: 1 - 1.5 hours

You might want to start with warmups, which may be provided in the course Facilitator Guide repo. Alternatively, students can do quick stand-up meetings in small groups.


KEEP IN MIND THAT I'M SELF-TAUGHT, SO MY CODE MAY BE A LITTLE MESSY.
LEMME SEE– I'M SURE IT'S FINE.
...WOW.
THIS IS LIKE BEING IN A HOUSE BUILT BY A CHILD USING NOTHING BUT A HATCHET AND A PICTURE OF A HOUSE.
IT'S LIKE A SALAD RECIPE WRITTEN BY A CORPORATE LAWYER USING A PHONE AUTOCORRECT THAT ONLY KNEW EXCEL FORMULAS.
IT'S LIKE SOMEONE TOOK A TRANSCRIPT OF A COUPLE ARGUING AT IKEA AND MADE RANDOM EDITS UNTIL IT COMPILED WITHOUT ERRORS.
OKAY, I'LL READ A STYLE GUIDE.

Code review comments may be nit-picking, but they won't be personal. Please include some positive reinforcement in every review!

Reviewing previous work can be accomplished several ways. Whatever your preferred method, every day should include some code review of student code. Have a volunteer (or selected student) project their code onto the screen for all to see. If the student was stuck on something, you can troubleshoot as a class. Otherwise, you'll lead a careful code review, complimenting what you see that's done right, and for things that are done less well, offering suggestions for how to improve the codebase. Ensure no comments are directed at the author, but rather focus on improving syntax and iterating on the code style and organization, moving it toward the standard presented in lecture.

Code review is often cited by students in feedback as one of the most useful parts of the course. This is something no online tutorial can offer, and sharing your experience and eye for quality code is a key part of the value that students hope to get out of your course.

Take a 10-minute break at the end of the review time.

Prepare for Lab Assignments: 1 - 2 hours

The remainder of the lecture time is dedicated to presenting students with the new topic for the day. Start this by reminding students of where they have been, and where they are headed. Potentially, show a working demo of what they will build by the end of lab time.

Each day should have only one or two new topics, related to the weekly theme and building on the previous day's work.

This is where your teaching skills really come into play. Live coding demos are one of the best ways to illustrate to students how to implement new techniques, and tackle new topics. It's engaging for students, because of the risk of failure inherent in any "live performance." While you certainly want to have your demos prepared, don't be afraid if things go wrong; rather, use that as an opportunity to teach a professional debugging process.

Read more on this in the recommended book *What Great Trainers Do*, and in our guide to giving a great lecture.

Any slides presented during lecture should be uploaded (if they aren't already in the cloud) and shared in Slack. Similarly, the video recordings of the entire lecture time should be linked from the Canvas course.

Be warned: When you are using existing slides, they will only feel natural to you after much practice. As you are presenting, students should feel as if you developed the slide deck yourself. An instructor making comments like "Why did he put that on this page?" or "I don't know why that's here" are inappropriate and detract significantly from the learning experience. Add and update presenter notes as needed, and ensure you've gone through the deck sufficiently before presenting it. If you want to make changes to a standard deck, follow the Curriculum Change process as needed.

Lab

Lab Assignments

Every day, students will have an assignment or two to work on during lab time. These should be well-defined ahead of the course's start. Take the time to thoroughly familiarize yourself with the assignments and with the solutions. Students will expect you to know the ins and outs of what they will need to do to complete the work. You may also have TAs on your instructional team that aren't deeply familiar with the assignments. If that's the case, you will need to ramp them up as well, so that they can effectively assist students.

Most courses should encourage collaboration during lab time *and* have some assignments that are required to be done in pairs. We've found it effective to structure lab time around both a paired and an individual effort. This can be done by requiring the first feature to be completed in pairs, or you can set a time limit of two hours for paired work. Then, require the remainder of the assignment to be completed individually.

All details of lab assignments are written and stored in the GitHub repo for the course. Work is submitted as a pull request URL, via Canvas. Feedback from the grader should be given directly on the PR, and an appropriate grade should be entered in Canvas afterward.

Helping Students in Lab

When students come to you (or a TA) for help, or if you assist someone while circulating, consider it an ad-hoc pair programming session. You will act as the navigator, and the student will act as the driver. As you direct the student in what to type, they build "muscle memory" in their fingers and think through what you are saying in an active way. This improves learning and gives them more confidence that they can fix the problem or address the challenge themselves. While navigating, focus on guiding students to ask the right questions, rather than just supplying the right answers.

Some students are very sensitive to screen smudges, so be sure to keep fingers off the screen when pointing at code (if you must, use a capped pen or a tablet stylus). Simple rule: ***Don't touch student computers.*** (This can help reduce the spread of colds and other illnesses as well.)

WANNA SEE AN OPTICAL ILLUSION?
HOLD YOUR KEYBOARD UP IN FRONT OF YOU AND LOOK AT THE HOME ROW.
NOW CROSS YOUR EYES A LITTLE SO THE 'G' AND 'H' OVERLAP.
KEEPING FOCUSED, LIFT THE KEYBOARD OVER YOUR HEAD.
EWW!
HA HA

Evaluating Student Progress

At all times, students should know where they stand regarding their success in a class. To this end, we strive for 24-hour turnaround on grading assignments. You may have significant TA help on this, but you should always be involved in the grading process to ensure students are getting adequate and timely feedback on their work.

Assignments should be weighted according to the standard grading rubric.

At the end of every module, you may want to dedicate from 10 minutes to 45 minutes of class time to review all the concepts covered during the previous four lectures. For a quick and effective way to do this; divide a whiteboard into sections for each recent class, and have students call out terms, concepts, or tools they learned and used on that day. Looking over all the concepts, students should be able to see at a glance if there is anything in particular they need to focus on.

In Code 401 classes, students should be practicing whiteboard exams on a daily basis to develop and demonstrate their coding skills. Initially, they can work on problems in teams. As the course progresses, they are expected to have more independence in solving these code challenges. By the end of the course, they should be evaluating each other with the whiteboard grading rubric, and sharing comments with each other to improve their skills.

The final exam will be given after class 15 in 201 as a quiz in Canvas. Instructors can allow 1 retake, in coordination with Admissions, to be completed before class 19. Prior to allowing a retake, instructors should meet with the student to get a read on how the student is feeling. The instructor can ask what strategies they used, what they might try differently, and what they need to do before attempting the exam again.

A final assessment project for 301 is given on class 14 with retakes during project week as needed. It is a solo project with 4 hours allotted for completion. See 301 Final Exam Facilitator Guide for details.

The final technical exam should be given in week nine of 401, and retakes in week 10 as needed. Retakes follow the same process as above, except Admissions doesn't need to be looped in. A third attempt is generally allowed only at the instructor's discretion, and is administered by the instructor directly when they have a clear idea of how the student will approach it differently. Follow the process to schedule each student, and record the results of the exam. If other instructors are available for the technical exam, coordinate with the team to help schedule them as testers instead of you. The experience is a better simulation of an interview if students can be tested by an instructor other than their own.

Meeting with Students 1:1

Students benefit from direct interaction with instructors periodically (see suggested meeting intervals below). Plan for 15 to 30 minutes (depending on class size), and keep the topic open-ended: allow students to bring any concerns or topics they want to discuss.

A 1:1 meeting should be scheduled ASAP for any student who is struggling and needs a plan for getting their course grade up. Otherwise, 1:1 meetings should happen with all students in the second week of Code 201 or Code 301. In Code 401 classes, 1:1 meetings should happen at least once per half of the course.

These meetings create a key time to provide students with candid feedback on their progress. Share their course grade with them, so they know you know their level of delivery. Highlight their successes and caution them on areas where they need to seek improvement. Encourage those who are dealing with imposter syndrome, provide extra stretch goals for those who are excelling, and give a Personal Learning Plan to those who need a clear road to passing the course.

You may want to utilize the Professional Competencies as a framework for highlighting any non-technical issues where the student needs to improve.

Take notes on your meeting with the student via a Student Meeting Form, so your comments can be accessed across departments. Admissions will be automatically notified of your submission.

Evaluating Student Professionalism

Towards the end of every course (at the start of the project module), instructors provide feedback to each student on their professionalism. As professionalism is a requirement to pass the class, instructors give a grade on student performance in this area. Similar to exams, a score of 80 (out of 100) or higher is considered passing. Giving a score in the 90s indicates you enthusiastically endorse the student as they move to the next phase of their learning or career.

Record feedback in this form. Part of this process is verifying that their resumes are crafted professionally, and without error (see instructions in the form). Your answers and the overall professionalism scores are then emailed to you, so you can copy them into Canvas. Also include your notes to the students on where they are showing professional strength, and where they need to improve as a professional.

Guest Speakers

If you would like to bring in a guest lecturer, or have a TA present to the class, you (or another instructor) must be present in the classroom, as per state licensing requirements. TAs should be given focused topics, ample time to prepare, with presentations timeboxed to 15 minutes. TAs should not present more than two times per month, unless they are under consideration for an Assistant Instructor role. Share the Instructor Skills mind map with them ahead of time, and use the Observation Form while they are presenting to give them feedback on their technique.

Any exceptions to this should be approved by the Principal Instructor or VP of Education.

Project Work

The fourth module of 201 and 301 classes—and the fifth and tenth modules of 401—are Project Work sessions. Night/Weekend classes are stretched in duration, but devote a proportional amount of time to project work. Project teams should be selected and formed based on the Project Team formation guide.

Teams have one week to build out a project from scratch. Milestones for project progress are listed in the course repo, on a day-by-day basis. Students will complete a daily survey in

Canvas where they can report their individual progress and raise any concerns they have about their group. As the Lead Instructor, you should monitor these reports to see if any teamwork intervention meetings are required.

One of the biggest challenges students face is properly scoping their projects. Work with each team to pick a feature set that's achievable as an MVP by the midpoint of project week. Present them with a clear rubric at the start of the week that lays out how they incorporate the concepts from the class into their project.

Students present on the final day of class. It's a great time to celebrate all they learned, and highlight the growth that they have shown.

Project grades are divided between overall team results, and individual contributions.

Working With TAs

If one of your TAs leaves the course for any reason, check in with support staff to ensure they are properly off-boarded and compensated only for the portion of their contract they completed. The Support Team will work to find and train a replacement TA.

TAs will handle the bulk of the grading and daily lab interactions, which means they might have a better read on which students are succeeding and which are struggling. Keep asking for updates on this, so you know how your class is doing. Results of this should be reported regularly in the Weekly Course Status Survey, where you can identify top students and raise concerns about students who might be struggling. You can also log your impressions of how well your TAs are doing, so standout TA efforts (or problems) are officially noted.

Beyond Your Course

Concluding a course strongly goes a long way to "putting a bow" on this gift of new knowledge you have presented to your students.

If you and your grading team stay on top of scoring submissions in a timely manner, getting final grades completed is much easier. For students who are not moving on directly to another class, you may work with them to make a plan for re-submitting any incomplete or late assignments (no more than a week beyond course completion), and then communicate this plan to Admissions.

For a signed graduation certificate after a 401 class, students must be passing the course, with unsubmitted assignments scored as 0 points. If a student is not passing at the time of course conclusion, the instructor should communicate this with the student. If the student wants to get everything done, they should put a day-by-day plan into writing, listing out what they will accomplish. They will email this plan to you, and you can approve it if it looks reasonable. In your reply, clearly state the final cut-off date and time for submissions (either what is described by the plan, or 11:59 p.m. on the Friday immediately following the last day of the course). If they do not follow their own plan, they do not pass the course.

All grading should be completed by the end of the course. Grading will be locked at the time of the course retrospective meeting, which is typically about 10 days after the last day of the course. Certificates should be sent out to all those passing as per the instructions in the course retrospective meeting agenda.

Probably, most of your students will pass the course, and some may be very excited about their new skill set. Some may say to you, “This was awesome! Let me know if I can do anything to help you or Code Fellows!” This is a great time to suggest they leave a review of your course on Course Report, or perhaps you’ll want to trade recommendations on LinkedIn.

Top students who have good people skills often make good TA candidates. You can encourage them to apply for the job from the link in the footer of www.codefellows.org.

Student Job Searching

While job searching, students will remain connected to Code Fellows. They’ll fill out a monthly survey about their job search activities, and have an opportunity to ask for assistance. The Campus Director will provide ongoing support, like resume reviews and coaching sessions as needed. Students will likely have questions about this process. Familiarize yourself with the services offered by the Career Accelerator Program, and ensure your students know who to contact when they need help.

Floating Instructor

Occasionally, as an instructor you will have significant breaks between courses. Use this time to contribute to the Facilitator's Guide for your course(s).

Conclusion

Congrats! If you've read through this handbook thoroughly, you're well on your way to becoming a great Code Fellows instructor. You can now complete your quizzes in the Basic Instructor Training Canvas course (see Your Training and Onboarding section of this doc). As your final task in the process, please **share with your training coordinator at least one comment about this document to confirm you've read it**. Hopefully it's something constructive to help us improve it, but a simple "Read it!" will do. Stretch goal: suggest a good XKCD comic to insert into a relevant section of the handbook (we've received permission from the author to use the images this way). Thank you, and happy instructing!