

Instructor guide

Target audience

- Everyone teaching online workshops about computational topics or interested in teaching.
- Previous and future instructors and helpers of CodeRefinery workshops.

Timing

Session 1

- 15 min: Intro
- 45 min: Lesson design and development
- 15 min: Break
- 45 min: Lesson template
- 15 min: Break
- 30 min: How we collect feedback and measure impact
- 10 min: Outro and feedback

Session 2

- 15 min: Intro
- 10 min: About the CodeRefinery project and CodeRefinery workshops in general
- 30 min: Collaborative notes and interaction
- 15 min: Break
- 35 min: Workshop overview, roles, onboarding/installation, helpers
- 20 min: Sound
- 15 min: Break
- 30 min: Screenshare
- 10 min: Outro and feedback

Session 3

- 15 min: Intro
- 30 min: Computational thinking
- 15 min: Break
- 30 min: Teaching philosophies
- 30 min: Co-teaching

- 15 min: Break
- 35 min: Sharing teaching tips, tricks, tools etc
- 10 min: Outro and feedback

Session 4

- 15 min: Intro
- 10 min: Why we stream
- 20 min: Behind the stream
- 15 min: Video editing (part 1)
- 15 min: Break
- 25 min: Video editing (part 2, exercise)
- 20 min: OBS introduction
- 15 min: Break
- 30 min: OBS setup
- 15 min: What's next?

Talking points for each sessions intro

- Brief CodeRefinery intro
- Instructors intros
- Notes intro
 - Check-in
 - Icebreaker
 - Participant intros in breakoutrooms (Random assign first, then same for whole session)
- Daily schedule, learning objectives
- Code of conduct
- Chat (please use collaborative notes)

Participant preparations for each session

Copied from e-mail communication with participants before each session.

Session 1: In general you will only need your brain for the exercises and discussions, but some will have the option to go deeper, which may require some accounts or tools to be set-up. We will inform about these things the week before each session.

For Session 1 we recommend to have a GitHub account (<https://coderefinery.github.io/installation/github/#github>). There will be an optional exercise where you can build a lesson locally. This requires a basic Python environment (<https://github.com/coderefinery/sphinx-lesson-template/blob/main/requirements.txt>), but do not worry if you cannot or do not want to set up these, there will be plenty of other exercises.

Session 2: In general you will only need your brain for the exercises and discussions. To get

the most out of this workshop day we recommend that you join the session with the same computer, display and microphone setup as you would usually use for teaching/ presenting. It may also help to get familiar with the different options your computer and Zoom setup provides for screenshare and audio adjustments. But it is not a must.

Session 3: In general you will only need your brain for the exercises and discussions. If you want to share some “cool gem” in the last episode of this workshop day, please update your registration (link to edit can be found in the registration confirmation e-mail sent by support@coderefinery.org or noreply@neic.org). This is very low barrier, you do not have to be the developer of a tool or technique to share with the group what makes it useful for you; and it does not even have to be cool, sometimes the small “normal” things are the best. You can check out some topics that have been proposed so far in this GitHub issue (#90), you can also comment there if you want to add or change your topic. These will be lightning talk length, so something between 2-5 min depending on how many are interested to share something, but we can collect them all in the collaborative notes for further exploration/ reading. But it is not a must.

Session 4: This upcoming session (session number 4) will cover the technical aspects of CodeRefinery-style online teaching. These topics have never before been covered in this much detail, so if it's interesting to you, don't miss it. (You might also want to pass this on to others who are interested in technical production of online events; registration is still open!).

First, we talk about why we stream (how we got here, advantages, disadvantages, how to be an instructor in a livestreamed workshop, and what it looks like from a broad level from the production side)

Then, we show how we are able to release videos so quickly. This is useful far beyond CodeRefinery, and there are also some hands-on exercises. Come prepared to set up a Python environment and the ffmpeg command line tool if you want to try these.

Finally, we see details about how to set up Zoom→Livestream environment for yourself. There isn't time to do this as a proper exercise, but it's the starting point and the instructor will offer help after the course for those interested. If you want to try this out during the session, install OBS Studio in advance.