

Lesson Plan: Intro to Programming Afternoon Workshop

For questions on this lesson plan, email compsci-feedback@khanacademy.org

This is based off a Saturday [CoderDojo](#) workshop, where the audience was 40 middle school girls. For the most part, the students had not programmed before, or if they had, it was with HTML or block-based programming. The workshop introduces students to programming with JavaScript/ProcessingJS on Khan Academy, mostly learning how to call functions with different arguments, and it gives them an opportunity to see the social side of programming with the pair projects. More specifically, the lesson covers three topics: [Intro to Programming](#), [Drawing](#) and [Coloring](#).

Before the workshop:

Assemble your team:

- Enlist volunteers to help students with projects and programming questions. Ask volunteers to do a few challenges and projects from the [Drawing](#) & [Coloring](#) tutorials to get familiar with the curriculum.
- Get students to sign up (if you are not conducting the workshop as part of an existing class). Ask students to bring a laptop (preferably with Chrome or Firefox installed), charger, and headphones. [See example signup](#).

Prepare your materials:

- Order [disposable headphones from Amazon](#), if you suspect some students may forget headphones.
- Print **handouts**

- [Graph paper handout](#) (at least 1 per person, but extras are useful too)
- [Documentation cheat sheet](#) (optional)
- [Robot game commands](#) (download this file too)
- [Pair programming worksheet](#) (1 per pair, optional)
- Cut the robot game commands and paperclip them into bunches. Each bunch should have 3 copies of each command.
- Get **pencils** and **pens** for students to use with the robot game and the graph handout.
- Create two **documents** to project during the workshop
 - [Example instructions document](#)
 - [Example projects document](#)

Day of workshop:

- Check that the curriculum loads and audio works on the projector.
- Make sure there are sufficient outlets/power strips.
- Display [a document](#) on the projector with welcome, instructions, and conversation-starter questions.
- Place 1 graph handout and writing utensil on each seat.
- Place 1 bunch of robot game commands at each pair of seats.

Suggested agenda:

- **12:20pm** - Volunteers start arriving.

- **12:30pm** - Students start arriving. They check in, set up their laptops, make sure their WiFi works, install a modern browser, and eat lunch.
- **12:40pm** - Volunteers meet to introduce themselves and review the agenda.
- **1:00pm** - Present ["What is programming?" slides](#), stopping at the fifth slide.
- **1:15pm** - Introduce robot programming game, display a [document with rules](#) on the projector.
 - Presenter does demonstration with a volunteer.
 - Students get in pairs.
 - Mentors give them the stack of commands and writing utensil.
 - Students take turns being the robot and the programmer.
- **1:30pm** - Show [say command on OSX](#) and different voices. Relate it to the SAY command from the game. (If presenter is not using a Mac, skip.)
- **1:35pm** - Present ["What is programming?" slides](#), the rest of the deck.
 - Have students identify the logos on the slides, for relevant slides.
 - Ask students for examples of their favorite games, apps, movies, on those slides.
- **1:45pm** - Scroll through the [curriculum on Khan Academy](#). Mention that there are talk-throughs, challenges, and a final project for each section.
- **1:50pm** - Introduce the idea of "pairing" for the projects:
 - Presenter does a demonstration with a volunteer.
 - Pair starts with deciding which laptop to use (pick the bigger one!). Close other laptop or open documentation on it.
 - Pair picks a team name, makes up a team chant and secret handshake.
 - Pair begins coding project.
 - Once done, pair puts URL of project in [doc](#).

- **1:55pm** - Watch ["Intro to Drawing"](#) together. Pause and point out interactivity of code, transcript.
- **2:00pm** - Do ["Challenge: H for Hopper"](#) together.
 - Get students to draw the first side on their graph paper.
 - Ask what would be a good number for the second side, try it.
 - Ask what would be a good number for the middle, try it.
 - Encourage them to just try things - it's okay if they're not right at first.
- **2:20pm** - Students begin working on self-paced curriculum independently. Display [a document with links and instructions](#) on the projector.
- **~3:00pm** (varies by student) - When students get to "Project: What's for Dinner?", they raise their hands.
 - Volunteer helps them find someone else who is also at that point. If there's nobody even close, then a volunteer could pair with them. Most of the time in a big enough group (like 35), the stars align. If you printed our [pair programming handouts](#), give them one. Otherwise, remind them to start by picking a single laptop to work on and coming up with their team details.
 - Volunteers should be there to answer questions about pair projects, to make sure that the pairs are both actually working on it (and switching off), and to inspire them to try new functions (like image, arc, etc.).
- **4:00pm** - Begin Show & Tell!
 - Going in the order shown in the doc, ask each team to come up. [See example doc.](#)
 - Ask for their team chant and secret handshake.
 - Show their project. Ask them to point out the shapes they made.

- Find something positive and unique to say about each project. For example, compliment creative food choices (so gourmet!), compliment clean code (yay comments!), color choice, point out some interesting command or optional argument they used, etc.
 - High-five them, give them some swag, applaud.
- **4:30pm** - Conclude the workshop. Show that there's more to learn in the curriculum, and emphasize that if they keep going, they'll be more powerful and able to do even more creative projects.