Write your first computer program

LESSON OVERVIEW

In this lesson, learners of all ages get an introductory experience with coding and computer science in a safe, supportive environment. This lesson has been designed for young learners, ages 4-10, but can be adapted for older learners using the differentiation suggestions provided.

LESSON OBJECTIVES

Students will:

- Define "coding" and "computer science"
- Identify key computer science vocabulary
- Identify places to go to continue learning computer science and coding

TEACHING SUMMARY

MATERIALS AND PREP

One Week Before Your Hour of Code
One Day Before Your Hour of Code

VOCABULARY

GETTING STARTED (5 MINUTES)

Setting the Stage

ACTIVITY (30-45 MINUTES)

WRAP UP (5 MINUTES)

Debrief
Celebrate
Next Steps

ASSESSMENT (2 MINUTES)

EXTENDED LEARNING

Beyond an Hour of Code

DIFFERENTIATION SUGGESTIONS

Adjustments for K-2 Teachers

TEACHING GUIDE

MATERIALS AND PREP

One Week Before Your Hour of Code

- Review the Hour of Code Educator Guide and Best Practices from Successful Educators in order to begin to plan your Hour of Code event.
- Register your Hour of Code event if you'd like to receive swag or classroom support.
- Review and complete the online tutorial yourself: Write your first computer program
- Be sure to test it first before asking your students to complete it. Check your technology and decide if you need to troubleshoot anything in advance of your Hour of Code.

One Day Before Your Hour of Code

- Print one or more of the Exit Ticket examples at the end of this lesson plan, or create your own.
- Each student who completes the activity should receive a certificate. Print one for everyone in advance to make this easier at the end of your Hour of Code.

VOCABULARY

- **code** - (v) to write code, or to write instructions for a computer.
- **Debugging** - Finding and fixing problems in your algorithm or program.
- **Program** - An algorithm that has been coded into something that can be run by a machine.

GETTING STARTED (5 MINUTES)

Setting the Stage

Welcome students to class and very briefly introduce the day's activity.

Say: "Today we're going to spend one hour learning to code. Has anyone here heard the term "code" before? What does it mean?"

Students might mention that a "code" is like a secret message, or that it's related to computers in some way.

Teaching Tip

One way to introduce the Hour of Code if you are not very familiar with coding yourself is to show one of our inspirational videos. Choose one you think your students will find inspiring, and share it now. For young learners, we suggest "The Hour of Code is Here."

Explain that in computer science, "code" means a set of instructions that a computer can understand. Let students know that today, they are going to practice "coding," "programming" and "debugging".

Define:

- **Coding** means to write code, or to write instructions for a computer.
- **Programming**, similarly, means to write code or instructions. Today, you will program with blocks on the computer (if you're using an online tutorial) or with pen and paper (if you're using an unplugged activity).
- **Debugging** means to check code for mistakes and try to fix errors.

Teaching Tip

You'll want to spend very little time front-loading or introducing your Hour of Code. With young learners, it's best to jump in as quickly as possible. Too much explanation or lecture at the beginning tends to spoil the fun, and fun is the whole point!

ACTIVITY (30-45 MINUTES)

Challenge your students to complete the Write your first computer program tutorial.

Depending on the age and ability of your students, you might consider:

- For younger students, we suggest you break your class into pairs or very small groups (three to four students each) and ask each group to work together to complete the tutorial using pair programming.
- For learners in the middle grades, we find that working independently on tutorials works well. Sometimes it helps to allow students to choose their own tutorial. If students aren't interested in Angry Birds, they can get a similar experience with the Code with Anna and Elsa or Make a Flappy Game tutorials.
- For older or adult learners, the Write your first computer program tutorial works extremely well either as an independent challenge or a pair programming activity.

Teaching Tip

Be sure to play through your chosen tutorials yourself, before asking your students to attempt them. That way you'll know what to expect and can make decisions about whether to let students choose their own tutorial, or if you want to assign tutorials based on student needs.

If a group or individual finishes early, they can attempt another tutorial by visiting code.org/learn.

WRAP UP (5 MINUTES)

Debrief

Ask students to reflect on the day's activities. What did they learn about coding? Programming? Debugging? How do they feel about computer science and code after spending one hour exploring?

Celebrate

Explain that you are spending one hour coding today, because this week is CS Education Week, and millions of other students across the globe have also been learning one Hour of Code this week. Congratulate students on being part of this world wide movement.

Give each student a certificate with his or her name on it.

Next Steps

Let students know that if they enjoyed today's activity, they have many options for continuing to code. Encourage students to visit code.org/learn for a list of options, or, if you're planning any of the extension activities that follow, tell students what's coming next in your classroom.

ASSESSMENT (2 MINUTES)

Give each student an "Exit Ticket" and ask them to summarize what they learned today, how they felt, or what they experienced. Students can draw, write or express themselves in any way they feel comfortable. Collect an "Exit Ticket" from each student on their way out of the room. If you like, you can post the "Exit Tickets" on a bulletin board or some place prominent in the school as a reminder of your Hour of Code.

EXTENDED LEARNING

Beyond an Hour of Code

After your Hour of Code ends, there are many ways to continue teaching computer science in your K-5 classroom. Here are some ideas:

- **Teach the Code Studio Computer Science Fundamentals courses.** These four courses are designed for young learners. Students work their way through a series of puzzles that teach them to code, and educators have access to engaging lesson plans that help make the learning coming alive. Code.org offers free professional development for these courses, online or in-person.
- **Invite a computer science expert to talk to your class about his or her work** Don't know any local computer scientists? Try signing up for a virtual classroom visit through Code.org's and Skype's Guest Speakers in Computer Science program.

DIFFERENTIATION SUGGESTIONS

Adjustments for K-2 Teachers

- Have the first level of the activity already pulled up on students' computer screens when they students start the activity.
- If possible, have students sit away from the computers while explaining the directions to the activity.
- Explain the 3 commands that the bird can do: move forward, turn right, and turn left.
- Do a live example of the first 3 levels. Try putting tape on the floor to make it look like the bird's maze. Model the commands yourself and then pick 1 or 2 student

volunteers to model for the class.

- Have students pair program by sitting 2 students at the same computer. Have the person controlling the mouse and keyboard first be a “1” and the other student who makes suggestions, points out errors, and asks questions be a “2.” Every 5 minutes, have the students switch roles.
- Practice clicking and dragging blocks before attempting to solve any of the puzzles. Also, practice dragging blocks to the trash can.

STANDARDS ALIGNMENT

▶ **Common Core English Language Arts Standards**

▶ **Common Core Math Standards**

▶ **CSTA K-12 Computer Science Standards**

▶ **ISTE Standards for Students**