

# Lesson 1: Welcome to CSA

45 minutes

## Overview

### What does software engineering mean to me?

Believing in one's own software engineering identity develops confidence and perseverance. In this lesson, students are introduced to key characteristics and skills of software engineers and explore common perceptions. Students realize that they have many of the same characteristics and skills as software engineers and discover their software engineering identity.

## Agenda

### Warm Up (5 minutes)

#### Changing the Face of CS

### Activity (30 minutes)

#### Design Thinking: Group Brainstorm

#### Design Thinking: Gallery Walk and Discussion

#### Skills of a Software Engineer

### Wrap Up (10 minutes)

#### Developing a Software Engineering Identity

#### Assessment (Optional)

## Objectives

Students will be able to:

- Identify characteristics and skills of a software engineer
- Relate characteristics and skills of a software engineer to personal skills and identity

## Preparation

- Gather sticky notes (at least 20 for each student)
- Print copies of the Software Engineering Skills handout (one half sheet for each student)
- Check the **Teacher's Lounge** for verified teachers on the CSA Forum to find additional strategies or resources shared by fellow teachers

## Links

**Heads Up!** Please make a copy of any documents you plan to share with students.

For the students

- **Change the Face of Computer Science** - Video
- **Software Engineering Characteristics** - Handout
- **Unit 1 Guide** - Handout

## Vocabulary

- **software engineer** - a person who designs, develops, and tests software for home, school, and business use

# Teaching Guide

## Warm Up (5 minutes)

### Changing the Face of CS

#### Remarks

Welcome to Computer Science A! We will learn a lot about computer science and programming this year. To get started, let's hear about how you can change the face of computer science.

 **Display:** Show the video - *Change the Face of Computer Science*.


## Activity (30 minutes)


### Design Thinking: Group Brainstorm (10 minutes)

#### Remarks

You won't just explore software engineering in this class - you will *be* software engineers.

We will use a common design thinking activity known as group brainstorming to gather some ideas about software engineering.

 **Do This:** Review the lesson objectives.

 **Distribute:** Give each student at least 20 sticky notes.

 **Do This:** Direct students to write a word or phrase on a sticky note in response to the prompt.

#### Teaching Tip


Encourage students to write down as many as they can think of. There are no wrong answers! As students write words or phrases, circulate the room to support their beliefs by:

- Agreeing with the statements they have written.
- Affirming the value of a particular idea.
- Making connections between statements on different students' desks to help them confirm that they are part of a community of people who are not so different from themselves.

#### Remarks

As you wrote down ideas, you were probably more confident about some than others. To help create a collective understanding of how we view software engineering right now, we will create groups of sticky notes. Your goal is to identify themes and similarities that you see when looking at all of your sticky notes together. It's up to you to decide how to group them!

**Group:** Place students in groups of three or four.

 **Do This:** Direct students to share the ideas they wrote, group sticky notes based on similar themes, and display their notes on designated wall space.

#### 💡 Teaching Tip

Alternatively, students can display their notes on a desk.

## Design Thinking: Gallery Walk and Discussion (10 minutes)

### 🎤 Remarks

Let's take a look at the ideas and themes your classmates identified through a Gallery Walk.

**📋 Do This:** Direct students to participate in a Gallery Walk to view other ideas and groupings. Students write what they notice and wonder on a new sticky note.

#### 💡 Teaching Tip

A Gallery Walk allows students to view multiple ideas, perspectives, and solutions. Students walk around the classroom to view each other's work, allowing each student to share ideas with the class and giving a voice to students who may be hesitant to share verbally.

**📋 Discuss:** Click through the animated slide to display the prompts.

- *What do you notice?*
- *What do you wonder?*

**Discussion Goal:** Students identify key ideas and common themes they noticed in other groups. Students may wonder why differences exist across groups and the cause or basis of these ideas.

**📋 Discuss:** Click through the animated slide to display the prompts.

- *Where do we get our perceptions from?*
- *Why?*
- *What message do we get from these?*

**Discussion Goal:** Students share how popular culture and media influence their perceptions about software engineers. Students share examples of how movies, TV shows, and social media portray software engineers.

#### 💡 Teaching Tip

Students might feel hesitant to share their thoughts. You could choose to modify this discussion by:

- Having small groups write down their thoughts on a student whiteboard.
- Having students write a response on a scratch sheet of paper, then throw it across the room. Students pick up a piece of paper close to them and read it to themselves. You can choose to call on students to share the response they picked up.

Collect the sticky notes. The ideas students shared will be revisited in Lesson 15. Choose common words and phrases to enter on the indicated slide in Lesson 15.


**📋 Do This:** Define the term - *software engineer*.


## Skills of a Software Engineer (10 minutes)

### 🎤 Remarks

It's helpful to hear from an expert about something we may not know much about. Let's hear from

It's helpful to hear from an expert about something we may not know much about. Let's hear from some software engineers about their work and the skills they use.

 **Display:** Show the video – *Software Engineering: Skills and Characteristics*.

 **Discuss:** *What characteristics and skills are important for software engineering?*

**Discussion Goal:** Students recall characteristics and skills discussed in the video, such as problem-solving, collaboration, creativity, and curiosity. Students share ideas about why these characteristics and skills are important for software engineering.

 **Discuss:** *When have you shown these characteristics inside or outside of school? How?*

**Discussion Goal:** Students realize they have already displayed many of the characteristics and skills of a software engineer in school and at home.


## Wrap Up (10 minutes)

### Developing a Software Engineering Identity

#### *Remarks*

These characteristics and skills are the most important parts of being a software engineer. Even though you may not realize it yet, you already have the mindset of a software engineer.

 **Distribute:** Give each student a copy of the Software Engineering Characteristics handout.

 **Do This:** Direct students to choose one software engineering characteristic or skill that they feel is their strongest and one that they feel they have the most room for growth.

#### Teaching Tip

Collect the Software Engineering Skills handouts. Students revisit these in Lesson 15 to reflect on their software engineering identity.

 **Do This:** Review the concepts covered in this lesson.

 **Display:** Key Vocabulary

#### Teaching Tip

The Unit Guide that is distributed in the next lesson has a space for vocabulary. Students can use this to write down definitions or fill in examples either throughout a lesson or at the end when the Key Vocabulary slide is displayed.

---

## Assessment (Optional)

**Software Engineering Characteristics:** The Software Engineering Characteristics handout can be used as a formative assessment to check that students can identify a software engineer's characteristics and skills and relate them to their skills and identity.



This curriculum is available under a  
Creative Commons License (CC BY-NC-SA 4.0).

If you are interested in licensing Code.org materials for commercial purposes **contact us**.