10 Presentations

Description

Students should come prepared to present based on the assignment introduced in a previous lesson.

Presentations are celebrations! Try to use this time to build a culture of support and collaboration among your students. Students should be able to present even if they don't consider their games "finished." Non-working games can lead to interesting class discussions so students should not feel embarrassed about non-functioning code.

Objectives

- I can clearly communicate my design process through an oral presentation and visual aids.
- I can describe and defend technical decisions I made while coding my game.

Standards

- *7-8.CT.10 // 9-12.CT.10: Algorithms and Programming* (Document the iterative design processes of developing a computational artifact that incorporates user feedback and preferences. // Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users.)

Brain-Starter (5 min)

SAY and display on a slide: "Standing up in front of a group of people and showing your work can be really hard!"
"At your table, come up with a list of "Norms" that you think we should follow as a class."
"An example norm, 'We should listen to the speaker and not talk during presentations."
"What other norms can you think of?"

Presentations (50 min)

Determine an order for student presentations either randomly or have students volunteer for a spot.

Allow each student to present their work, then allowing time for questions.

Make sure students show and describe the purpose of a part of their code.

Closing (5 min)

End on a high note! Have students celebrate each other's work through shout outs! Shout outs can be linked to game, presentation, or how they collaborated with peers.

Sentence stem:
“I want to shoutout [name] for [compliment]!”