

Unit 1: Coding, Games & Animation
7th Grade Digital Art
17 Classes

Rev. May 2025

Essential Questions

- How does coding enable creativity and individual expression?
- What practices and strategies will help me as I write code?
- How do software developers manage complexity and scale?

Enduring Understandings with Unit Goals

EU 1: How can the skills used in game design transfer to many other high-demand fields?

- Skills in problem-solving, teamwork, marketing, and interface design are essential components when developing a video game.

EU 2: Games consist of multiple stages with interactable objects.

- Design multiple stages to play through with variance in challenge.
- Create objectives or puzzles that allow for an immersive gameplay experience.

Standards

ISTE Standards

Standard 1: Empowered Learner. Students leverage technology to take an active role in choosing, achieving, and demonstrating competency in their learning goals, informed by the learning sciences.

Standard 2: Digital Citizens. Students recognize the rights, responsibilities and opportunities of living, learning and working in an interconnected digital world, and they act and model in ways that are safe, legal and ethical.

Standard 3: Knowledge Constructor. Students critically curate a variety of resources using digital tools to construct knowledge, produce creative artifacts and make meaningful learning experiences for themselves and others.

Standard 6: Creative Communicator. Students communicate clearly and express themselves creatively for a variety of purposes using the platforms, tools, styles, formats and digital media appropriate to their goal.

ISAAC Vision of the Graduate Competencies

Competency 1: Write effectively for a variety of purposes.

Competency 2: Speak to diverse audiences in an accountable manner.

Competency 3: Develop the behaviors needed to interact and contribute with others on a team.

Competency 4: Analyze and solve problems independently and collaboratively.

Competency 5: Be responsible, creative, and empathetic members of the community.

Unit Content Overview

1. Coding

- Create animations, computer games, and interactive projects using Scratch, a graphical programming language developed at MIT.
- Utilize interface controls to work in the environment to create interactive programs.
- Apply code to create, correct, and modify their own Scratch programs.

2. Computer Concepts

- Apply computer science concepts to control, compute, and accurately respond to user commands within original programs.
- Evaluate and modify the code created by others to create a functional program.

Daily Learning Objectives with *TWPS*

Students will be able to...

- Use visual programming language, Scratch, to gain an understanding of input, output and the logic of Block code work.**
 - *When you press a key or click a sprite in a game, how does the game know what to do next?*
- Apply graphic effects to create visual interest, create their own sprites, backgrounds, and costumes.**
 - *What makes a game or animation visually fun or unique—and how can you create that look yourself?*
- Write blocks of code that changes the costumes of sprites, add animations to sprites and repeat blocks using loops. Make a sprite do simple tasks, and create code to better control how a sprite moves. Use a mouse to change instructions and create effects.**

- *Have you ever seen a character that blinks, dances, or changes shape? How do you think that kind of animation works in Scratch?*
- Use the sensing blocks to detect what a user is doing; get information from the user, and track the location of a sprite. **
 - *How can you make your Scratch project “notice” what the player is doing and respond to it?*
- Create variables to explain how data changes in variables and how to show and hide blocks. Insert backdrops, change backdrops and use sensing commands. **
 - *What do you think happens when a game needs to keep score or switch levels—how does the computer remember what’s going on?*
- Use broadcasts as a trigger to start scripts and change backdrops; how to trigger a script based on video or audio input. **
 - *How do you think one sprite can “talk” to another sprite in a game to make something happen, like a new level starting or music playing?*
- Use variables in your scripts to store information such as user information, or track information such as the score in of a game**
 - *If you were building a game and wanted it to remember your high score, how would you make that happen in Scratch?*
- Write blocks of code that create music in Scratch. Add sound to a sprite; have a sprite play music. **
 - *Have you ever seen a character that blinks, dances, or changes shape? How do you think that kind of animation works in Scratch?*
- Create simple drawings. Write blocks of code that insert sound in Scratch; Change the costumes of sprites, add animations to sprites and repeat blocks using loops.
 - *What happens when you combine drawing, sound, and animation in one project—what could you create with all three?*

Instructional Strategies/Differentiated Instruction

- Daily Warm Up Activities
- Lecture slides with guided note-taking
- Flexible grouping
- Exit slips
- Graphic Organizers
- Creating authentic connections for students
- Rephrasing and restatement of information and concepts
- Student use of headphones
- Independent reading
- Outlining of text

Assessments

FORMATIVE ASSESSMENTS:

- Quick Write responses
- Exit Slips
- Accountable Talk Discussions
- Do Now
- Oral questioning
- Graphic organizers
- Thumbs up/Thumbs down

SUMMATIVE ASSESSMENTS:

- Unit Task scoring guide

Unit Task

Unit Task Name: Scratch Game,

Description: Students create a 45: radio advertisement promoting a new bubble gum product. (EU1) Students create a simple ‘Guessing Game’ and a Pong Game in Scratch which makes use of user interaction. (EU2) Students create a simple story in Scratch where they develop skills in using suitable backgrounds and sounds. (EU1)

Unit Resources

Scratch.Org