

*Programming Language:*  
Scratch

*Supported Devices:*  
Mac  
Windows  
Chromebook

*Instructional Models:*  
Direct Instruction  
Instructional Scaffolding  
Use of Learning Objectives  
Relevant Vocabulary  
Bloom's Taxonomy of  
Questions  
Project-Based Instruction  
Independent Study

*Supported Learning Models:*  
Classroom  
Flipped Classroom  
Blended  
Hybrid  
Synchronous  
Asynchronous

*Aligned With National  
CS Standards*

*Reinforces:*  
Logical thinking  
Creativity  
Persistence  
Resilience  
Communication skills  
Structural thinking  
Problem-solving

### INTRODUCTION TO PROGRAMMING WITH SCRATCH

#### COURSE DESCRIPTION

Essentials Scratch Course features 96 lessons for each academic year in coding immersion. The course is designed with instructional scaffolding to promote academic equity and success across all learning levels. In this course, students will begin with learning how to use Scratch to create projects. Once students have a good understanding of how to use Scratch, they will use it to learn programming concepts.

The course is divided into two types of lessons: project and challenge, followed by quarterly cumulative capstone projects.

- **PROJECT LESSONS** guide students through the creation of fun, visually appealing, and engaging projects in Scratch. As students build projects, new concepts are introduced that further their understanding of coding logic.
- **CHALLENGE LESSONS** help develop students as programmers, allowing them to formulate solutions to problems independently.
- **CAPSTONE PROJECTS** at the end of each quarter act as a milestone for students to apply everything that they have learned in an in-depth project.

Through guided instruction, Essentials teaches students how to independently debug their code, a critical skill used by programmers. Developing this learning is made easier by the Essentials methodology of integrating these concepts gradually. By the end of the course, students will have the necessary skill sets to be comfortable coding in Scratch and creating projects.

### LEARNING OBJECTIVES

CodeWizardsHQ developed Essentials to support students in real-world, comprehensive learning. Students not only learn how to code but to think like programmers. Each lesson is designed to build computational thinking while learning how to code using Scratch.

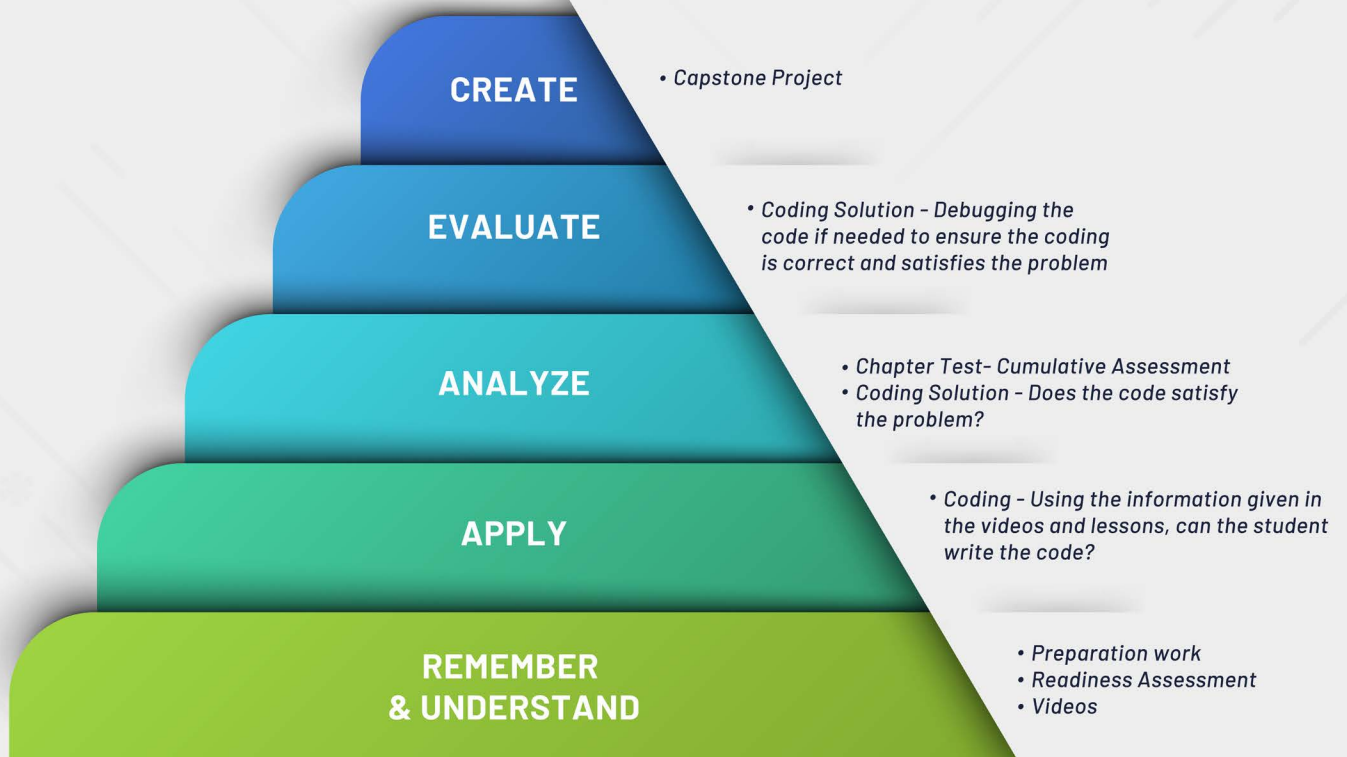
### BY THE END OF THE COURSE STUDENTS WILL BE ABLE TO:

- Customize elements in Scratch using Paint Editor
- Add sounds to their projects and understand color values
- Understand that a script is a set of instructions that run in order
- Incorporate user interactivity in projects
- Learn how to implement and manage sprites in the Scratch environment
- Employ and understand the uses of various blocks
- Understand how to interact with the coordinate plane
- Create and utilize clones
- Know what variables are and understand their behavior
- Identify basic data types and how to correctly use them
- Confidently use decisions, loops, and functions to create various projects
- Use various Scratch extensions
- Learn about Booleans, their behavior, and how to use them
- Understand how, when, and why computer programs make decisions

### RESOURCES INCLUDED:

- Learning Management System
- Cloud-based platform
- Administrative dashboard
- Learning targets and planning tasks
- Pre-lesson activities
- Teacher/student engagement actions
- Assessing and advancing questions
- Engaging concept videos
- Proprietary slide decks
- Comprehensive assessment tools
- Auto-graded quizzes and tests
- Common errors and their solutions
- Cumulative capstone projects
- Progress Tracking

## ESSENTIALS CURRICULUM IN APPLICATION OF BLOOM'S TAXONOMY



**BEGIN YOUR ESSENTIALS CODING JOURNEY TODAY!**