

Instructional Model:

Teacher Led
Engaged Learning
Project Based Lessons

Course Category:

Programming

Minimum Grade Level:

3rd Grade

Prerequisites:

None

Programming Language:

Scratch

Software used in Course:

Scratch,
Google Chrome

Technology Options:

Mac
Windows
Chromebook

COURSE DESCRIPTION

Students understand foundational computer science principles such as loops, debugging, and conditional statements in this project-based course. Using Scratch, an open-source block-coding language created by the MIT Media Lab, students build games and solve puzzles to develop computational thinking and analytical skills. Educators introduce students to safe Internet behaviors, a variety of STEM careers, and real-world computer science vocabulary through participation in group discussion activities. After completion of this course, students will gain the basic skills needed to take higher-level, computer science classes.

STANDARDS COMPLIANCE

- 100% national and state computer science standards alignment - standards map provided
- Reinforces Math, ELA, and Social-Emotional Learning competencies

STUDENT OUTCOMES

Each lesson plan is designed to achieve specific student outcomes related to computer science competencies.

Sample outcomes for this course include:

- Evaluate and address errors in an algorithm by testing and debugging.
- Identify and explain how to deal with suspicious websites through discussions with classmates.
- Apply an understanding of loops, events, randomness, and conditionals through creating programs.
- Utilize loops by creating algorithms to navigate a maze.
- Apply software planning concepts.
- Understand the role and career of a web designer through discussions with their classmates.

101

PROGRAMMING 101

Introduction to computational thinking with Scratch

RESOURCES INCLUDED

- Teacher Training videos
- Summative Assessments
- Formative Assessments
- Syllabus
- Computer science standards compliance mapping
- Full year of step-by-step lesson plans
- Pacing guide
- Vocabulary words and definitions
- Coding activities
- Unplugged activities
- Digital citizenship activities
- Hardware activities (Optional)

PILLARS OF ENGAGEMENT



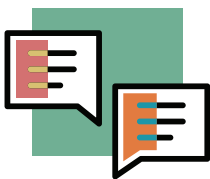
CODING *visual*

Codelicious engages visual learners with computer-based projects, vocabulary activities, as well as written and visual imagery, while building foundational and advanced computer science skills.



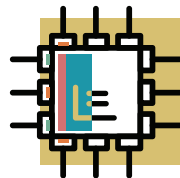
UNPLUGGED *kinesthetic*

Designed to be conducted off-line with creative activities or movement, Codelicious leverages unplugged activities to reinforce computer science concepts.



DIGITAL CITIZENSHIP *auditory*

With discussions, collaboration sessions, and student presentations, Codelicious provides computer science curriculum that enables the auditory learner to thrive.



HARDWARE *tactile*

Hands-on learning with Codelicious curriculum builds upon computer science principles through hardware projects, problem solving activities involving everyday materials, and real-world applications.