The Coding Space

Course Brochure 2019

What makes us Different

Our classes are self-paced, allowing students to progress through our curriculum at their own speed.

Project Based Learning

In our classrooms, students learn entirely through creating projects. Projects allow students to work on their critical thinking and problem solving skills while simultaneously picking up the hard skills of coding.

Students create games in Scratch, progress to websites in HTML, CSS, and Javascript, as well as work in many other tools and languages depending on what keeps them both engaged and challenged. They create projects not by using tutorials, but rather by learning to think critically and create their own solutions to problems.

In addition to students being able to construct their own knowledge, project based curriculum also allows students to work at their own pace. It is the job of our teachers to make sure that students find projects that are both engaging and challenging at the same time.

Challenge Projects

Challenge Projects are at the core of the The Coding Space education. Projects are fun and engaging games, websites, or apps that students are challenged to figure out how to build.

Unlike virtually all other coding curriculums, The Coding Space doesn't have follow-along tutorials. In fact, Challenge Projects are "un-tutorials." They are similar to tutorials in that they guide students in building something, except at The Coding Space we don't include any of the actual instructions on how to do anything.

Challenge Projects list the high-level tasks students need to complete the project, but leave out how to do them. Students figure out the rest via tinkering, Googling, asking a friend, and iterating towards a solution, all while developing deep insights and intuition of high-level concepts.

Expert Mentorship

Fostering non-cognitive skills, like grit and intellectual self-confidence is a delicate balance of targeted encouragement and tough love.

Our teachers are highly trained mentors that are skilled in knowing exactly how much help to give a student, when, and how often. Rather than simply giving hints or answers, The Coding Space instructors draw ideas from students through leading questions and modeling strategies on how they could find answers for themselves. Would Google be helpful in this scenario? What keywords would you use?

IF YOU CAN DREAM IT, YOU CAN CODE IT.

After-School Classes

Fall Semester: September 16 - February 1
Winter Semester: February 3- June 5

Beginner
Coding in Scratch (Ages 8-14)
Skills covered:
Loops, Variables, Events

Advanced
Coding in WoofJS, HTML, CSS, or Javascript. (Ages 8-14)
Skills covered: Syntax, Arrays, Functions, HTML Tags, AJAX Requests, DOM

High School
Coding in WoofJS, HTML, CSS, or Javascript. (Ages 14-17)
Skills covered: Syntax, Arrays, Functions, HTML Tags, AJAX Requests, DOM

Beginner GirlCode
Coding in Scratch (8-14)
Skills covered:
Loops, Variables, Events

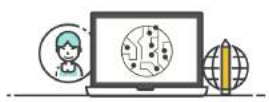
Advanced GirlCode
Coding in WoofJS, HTML, CSS, or Javascript. (Ages 8-14)
Skills covered: Syntax, Arrays, Functions, HTML Tags, AJAX Requests, DOM

Young Beginner
Coding in Code.org (Ages 6-7)
Skills covered:
Loops, Use of Mouse and Keyboard

Camps

Summer: June 15- August 28

Co-ed Camps
Upper West Side, Long Island, Park Slope

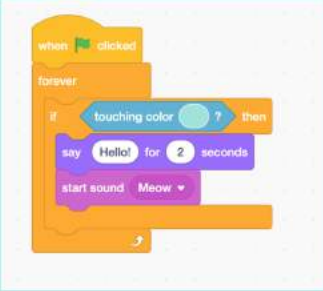


GirlCode Camps
Upper East Side

High School Camps
Upper East Side, Upper West Side, Park Slope

Beginner Classes

Coding with Scratch



Loops

A sequence of instructions that is continually repeated until a certain condition is reached

Variables

Variables are used to store, retrieve, and manipulate values that appear in your code

Events

The flow of the program is determined by events such as user actions (mouse clicks, key presses)

What Will My Child Learn First?

The cornerstone of our curriculum begins with block-based programming in MIT Scratch. The advantage of block-based programming is that students don't have to worry about small syntactic details on Day 1.

Parents often worry that because block-based programming is quicker to start coding, it is less educational than text-based programming. It's actually the opposite! Learning the syntax of a programming language is a tedious and rote task that doesn't engage students' critical thinking. Block-based programming skips directly to the difficult part of coding: formulating one's ideas into a logical sequence of steps.

Starting with a text-based programming language is like teaching a child to write before they can speak or to spell words in a foreign language before they know what they mean! We believe that meaning comes first, and spelling second.

Advanced Classes

Coding with WoofJS, HTML, CSS, or Javascript

What Happens After Scratch?

Our team at The Coding Space has spent thousands of hours creating the best possible solution to this answer. We developed WoofJS as an intermediate step where kids could go after Scratch. The idea was to allow students to leverage their existing Scratch knowledge but while using a web programming language. For every block in Scratch, we created an equivalent text-based command in JavaScript. WoofJS allows students to leverage their Scratch knowledge to learn JavaScript syntax.

WoofJS makes the transition as seamless as possible. There are no new concepts to learn. Students simply take what they know from Scratch and learn how to type it. This is the correct ordering of things: learning to speak before learning to spell. WoofJS is focused on the spelling.

What is WoofJS?

WoofJS is a JavaScript framework. In programming, a “framework” is a suite of functionality that can be added to a language. For example, Rails is a popular web framework for the Ruby programming language.

Programming in WoofJS is programming in JavaScript with some special words thrown in. It’s similar to how when you speak about baseball, it’s ultimately English just with some additional words, such as “home run”, “batting average”, and “strike out”. In the case of WoofJS, the additional words come directly from Scratch. For example, in Scratch there’s a block called “forever” which creates a loop that runs 30 times per second (which is about how fast the human eye perceives motion), so WoofJS adds a command to JavaScript called “forever” with the same functionality. Without WoofJS, this would still be possible, but it has a different name: “setInterval”. One of the beauties of WoofJS is that it keeps as much about Scratch constant so students can focus on learning JavaScript syntax, including common difficulties such as matching parentheses and brackets, indenting their code, and creating functions and variables.