

# Middle School Programming

---

## *Course Syllabus*



**Developed by:**

Kelly Strick

[strickkr@gmail.com](mailto:strickkr@gmail.com)

Summer 2016, Updated 2017

## TABLE OF CONTENTS

<b>Table of Contents</b> .....	<b>2</b>
<b>1. Introduction</b> .....	<b>3</b>
1.1. Materials.....	4
<b>2. Introduction to STEM</b> .....	<b>5</b>
<b>3. STEM 1</b> .....	<b>11</b>
<b>4. STEM 2</b> .....	<b>18</b>

## 1. INTRODUCTION

The intent of this syllabus is to provide a logical progression educators can follow for middle school STEM classes with programming as a part of the regular curriculum. This syllabus is a revision after the courses were taught during the 2016-2017 school year. It is still a work in progress and will be updated again as the courses are taught this school year.

This material was created by a teacher new to programming and can easily be taught by the same. It is not necessary to have any programming experience, but would be helpful to work through the lessons prior to teaching the material. Any resources mentioned in this syllabus are free or have a free version. Purchasing licenses is not necessary to teach the material.

This course material was originally designed for a three-class progression: Introduction to STEM, STEM 1, and STEM 2. These are semester-long courses, with one quarter devoted to, or at least including, programming. The Intro class provides an introduction to programming using Tynker™ and Scratch™ material. While Tynker™ is not a free resource; it does provide six free-play lessons. This then progresses into lessons on Scratch™ programming, similar to building with Lego blocks because it uses “blocks” of code. STEM 1, the second course in the sequence, first uses Lego Mindstorms NXT kits, providing an opportunity for students to build and program robots to move in particular ways. This progresses into introductory Python programming using Codesters™ online classes. Codesters™ is free for anyone to join, but also has a Learning Management System (LMS) for a fee that allows students to be enrolled in classes and their progress tracked by the instructor. Finally, STEM 2, the last course, continues lessons in programming with Python and experience programming with Arduino for electronics and microcontrollers. If desired, the three units could be grouped together for use as a programming class rather than part of a STEM class curriculum. The daily plans in this syllabus are projected completion times; some may need adjusted based on students. These daily plans and completion times will be updated again after the courses are taught this school year.

Differentiation will be highly necessary in these courses as students will enter with varying degrees of exposure and experience with programming. Assignments based on each student’s level of skills, is recommended. It is also recommended that a pre-assessment be given to evaluate the students’ relative abilities.

Throughout the syllabus, some instruction will be individualized, as needed, which means that instruction should be geared to the individual students’ needs. Due to varying degrees of computer experience, some students may need more instruction than others. Beginning with a pretest and utilizing the formative assessments should give the instructor an idea of who will need more guidance. Individualized instruction may include one-on-one or small group assistance.

Formative assessments can include student submissions of sample programs as well as interviews between student and teacher at certain check points for projects. Summative assessments may include quizzes, tests, and projects. Quizzes are short, in-class activities which demonstrate knowledge of coding vocabulary, functions, etc. Students may be asked to write code, interpret code, or debug code. Sample assessments are provided with these materials. It would be particularly helpful to print assessment materials in color as students read code and errors. Sample solutions may not include all possible correct solutions. Students may represent their solutions in various ways. Final projects may be completed over several weeks and presented to the class upon completion. Logbooks should be used by students as a personal reference for the projects they are completing, strategies they are learning, and reflections over the process. Logbooks, ideally, will be routinely monitored by instructors. Providing feedback to students about their progress, problem-solving ideas, and providing additional resources is important.

Assessments and a Quick Reference Guide are available to print from the Middle School Programming Teacher Resource document. These documents are also works in progress. They have been revised once, but more teacher resources will be added as the course is taught during the 2017-2018 school year. These documents are available for unlimited teacher use and may be altered as needed.

Logbook entries should contain the following:

- Date for each entry
- Response to daily question, if necessary
- Log of personal activity, completed tasks, communications, team activity
- Research and analysis
- Sketches, pictures, screen shots, class notes, calculations, design process, rationale for decisions, links to helpful resources, videos, and /or tutorials
- Everything should be included whether or not tasks are successful.

While writing programs, students will often generate errors. A good practice would be for students to copy/paste the code and error message into their log book for future reference. They should also include a description of how they resolved the error. Educators might also save their error messages to use as examples for other students or to include as assessment items.

## 1.1. MATERIALS

- Computers with internet access
- Lego Simple & Powered Machines Base set #9686

- Lego Mindstorms Education kits (NXT or EV3 versions)
- Scratch Hot Sheets (created by SparkFun)
  - <https://learn.sparkfun.com/resources/78>
- Scratch Do Now Worksheets (created by Kelly Vaughan)
  - <http://scratched.gse.harvard.edu/resources/do-nows-beginning-scratch-instructions>
- Arduino kits (such as SparkFun Tinker Kit product #13930)
- *Invent Your Own Computer Games with Python* 3<sup>rd</sup> Edition by Al Sweigart
  - <https://inventwithpython.com/>
- *Think Python* 2<sup>nd</sup> Edition by Allen Downey
  - <http://greenteapress.com/thinkpython2/thinkpython2.pdf>
- *Python-for-Kids* by Jason R. Briggs
  - [http://raspbian.serveblog.net/wp-content/uploads/2016/02/python\\_for\\_kids.pdf](http://raspbian.serveblog.net/wp-content/uploads/2016/02/python_for_kids.pdf)

## 2. INTRODUCTION TO STEM

Day 1 **Class Activity** – Introduction to the Art of Computer Science (code.org); Binary Strips activity

**Instruction** – Expectations and learning targets; logbook usage; What is computer science? Vocabulary; Binary

**Resource** – [Intro to the Art of Computer Science](#)

Day 2 **Class Activity** – Count the Dots Activity: Binary Numbers Follow Up

**Instruction** – Review binary numbers

**Resource** – csunplugged.org Activity 1: [Binary Numbers](#)

Day 3 **Class Activity** – Treasure Hunt Activity – Finite State Automata

**Instruction** – Finite State Automata (simplified)

**Resource** – csunplugged.org Activity 11: [Treasure Hunt](#)

Day 4 **Class Activity** – Color By Numbers: Image Representation

**Instruction** – Image representation (pixels; compression)

**Resource** – csunplugged.org Activity 2: [Color by Numbers](#); info-study.net Unplugged Flash Activity: [Unplugged Flash Activity](#)

Day 5 **Class Activity** – Graphics: Bits and Points

**Instruction** – Vocabulary, scale, task scheduling

**Resource** – TryEngineering.org Graphics: Bits and Points Lesson: [Bits and Points](#)

Day 6 **Class Activity** – Bits and Points continued

**Instruction** – Individualized, as needed

Day 7 **Class Activity** – Introduction to Programming

**Instruction** – Programming vocabulary

**Resource** – Videos: [What Most Schools Don't Teach](#) and [What is Programming?](#) or [Programming](#)

Day 8 **Class Activity** – Marching Orders Activity

**Instruction** – Programming languages

**Resource** – csunplugged.org Activity 12: [Marching Orders](#)

Day 9 **Class Activity** – BrainPop movie

**Instruction** – Vocabulary, graphic organizer

**Resource** – Movie: [BrainPop Computer Programming Movie](#), vocabulary: [BrainPop Vocabulary](#)

Day 10 **Class Activity** – Welcome to Tynker

**Instruction** – Introduce Tynker and show demonstrations (Candy Quest and Code Monster)

**Resource** – [Tynker](#) and student cards ([www.tynker.com](http://www.tynker.com))

Day 11 **Class Activity** – Explore games: Lost in Space and Puppy Adventure

**Instruction** – Demo Lost In Space game; Demo Puppy Adventure – reinforce correct vocabulary; Compare/contrast two games

**Resource** – Tynker Lost in Space game: [Lost in Space](#); Tynker Puppy Adventure game: [Puppy Adventure](#)

Day 12 **Class Activity** – Sketch Racer

**Assessment** – Computer Programming BrainPop quiz

**Instruction** – Discussion of 3 games and their strategies

**Resource** – Tynker Sketch Racer game: [Sketch Racer](#)

Day 13 **Class Activity** – Tynker Candy Quest Lesson

**Instruction** – Individualized as needed

**Resource** – Tynker Hour of Code: Candy Quest: [Tynker](#)

Day 14 **Class Activity** – Tynker Dragon Dash Lesson

**Assessment** – Formative assessment

**Instruction** – Individualized as needed

**Resource** – Tynker Hour of Code: Dragon Dash: [Tynker](#)

Day 15 **Class Activity** – Tynker Comic Creator Lesson

**Instruction** – Individualized as needed

**Resource** – Tynker Comic Creator Lesson: [Tynker](#)

Day 16 **Class Activity** – Tynker continued

**Assessment** – Summative assessment

**Instruction** – Individualized as needed

Day 17 **Class Activity** – CodeMonkey game

**Instruction** – Individualized, as needed

**Resource** – playcodemonkey.com Code Monkey game: [CodeMonkey](#)

Day 18 **Class Activity** – CodeMonkey continued

Day 19 **Class Activity** – CodeMonkey continued

Day 20 **Class Activity** – Scratch Overview and exploration

**Instruction** – Scratch introduction and sample projects

**Resource** – Video: [Scratch Overview YouTube](#) or [Scratch Overview Video](#)

Day 21 **Class Activity** – Log: “What are 3 aspects of yourself that you could represent through images or sound?”; Scratch Do Now #1; About Me project

**Instruction** – Explanation of project requirements

**Resource** – Scratch Curriculum Guide: <https://drive.google.com/file/d/0B4qIZ8hFqev8STdXNW03Rmtabjg/view> ; Scratch Do Now Worksheets: <http://scratched.gse.harvard.edu/resources/do-nows-beginning-scratch-instructions> ; Scratch About Me example project: [About Me](#)

Day 22 **Class Activity** – “About Me” work

**Instruction** – Individualized, as needed



Day 23 **Class Activity** – Log: “What are 5 situations where you use instructions?, What are instructions good for?”; Programmed to Dance: “Bossy” vs. “Bossed”

**Resource** – Vimeo Dance videos: [Dance Moves 4](#), [Dance Moves 1](#), [Dance Moves 2](#), [Dance Moves 3](#)

Day 24 **Class Activity** – “Create” with Scratch; Scratch Do Now #3

**Instruction** – Getting Started with Scratch

**Resource** – Scratch.mit.edu: [Scratch](#) ; Scratch Do Now Worksheets: <http://scratched.gse.harvard.edu/resources/do-nows-beginning-scratch-instructions>

Day 25 **Class Activity** – Scratcher Hot Sheet

**Assessment** – Formative Assessment #1 (see Teacher Resource Documents)

**Instruction** – individualized, as needed

**Resource** – cdn.sparkfun.com: [Scratcher Hot Sheet](#)

Day 26 **Class Activity** – Log: “What are two strategies that you could use when you get stuck while designing?”; Dance Party; Scratch Do Now #2

**Instruction** – Difference between a sprite and a costume

**Resource** – Dance Party handout from Scratch Curriculum Guide: <https://drive.google.com/file/d/0B4qIZ8hFgev8STdXNW03Rmtabjg/view>; ; Scratch Do Now Worksheets: <http://scratched.gse.harvard.edu/resources/do-nows-beginning-scratch-instructions>; Scratch.mit.edu: [Dance Party](#)

Day 27 **Class Activity** – Scratch Do Now #4; Project Work

**Instruction** – individualized, as needed

**Resource** - Scratch Do Now Worksheets: <http://scratched.gse.harvard.edu/resources/do-nows-beginning-scratch-instructions>

Day 28 **Class Activity** – Debug it!

**Instruction** – What does “debug” mean?

**Resource** – Debug it! Handouts from Scratch Curriculum Guide:

<https://drive.google.com/file/d/0B4qIZ8hFgev8STdXNW03Rmtabjg/view>; [Debug it! 1](#),  
[Debug it! 2](#), [Debug it! 3](#), [Debug it! 4](#), [Debug it! 5](#)

Day 29 **Class Activity** – Log: “What is a game?”; Brainstorm for Maze creating

**Instruction** – How to brainstorm productively

**Resource** – Maze Handout from Scratch Curriculum Guide:

<https://drive.google.com/file/d/0B4qIZ8hFgev8STdXNW03Rmtabjg/view>; [Maze-creating](#)

Day 30 **Class Activity** – Log: “What did you like about the Maze project? How might you want to change it?”; Maze extension projects; Students teach

**Instruction** – Student led

**Resource** – Scratch Curriculum Guide [Maze Extensions](#), Score: [Score Extension](#), Timer: [Timer Extension](#), Enemies: [Enemies Extension](#), Levels: [Levels Extension](#), Rewards: [Rewards Extension](#)

Day 31 **Class Activity** – Buffalo Cart Hot Sheet Challenge

**Assessment** – Formative Assessment #2 (see Teacher Resource Document); logbook check

**Instruction** – Individualized, as needed

**Resource** – [cdn.sparkfun.com](http://cdn.sparkfun.com): [Buffalo Cart Hot Sheet](#)

Day 32 **Class Activity** – Log: “What is a variable? What is it good for?”; Game project

**Instruction** – Individualized, as needed

**Resource** – Games starter projects handouts

Day 33 **Class Activity** – Crab Defender Hot Sheet Challenge

**Assessment** – Formative assessment

**Instruction** – Individualized, as needed

**Resource** – [cdn.sparkfun.com](http://cdn.sparkfun.com): [Crab Defender](#)

Day 34 **Class Activity** – Log: “What has been your favorite Scratch project to work on so far? What are 3 ideas for something you could work on next?”; Brainstorm for Final project; project planner

**Instruction** – Project planner

**Resource** – Planner handout

### 3. STEM 1

Day 1 **Class Activity** – Legos Mindstorms inventory

**Instruction** – Expectations for logbooks; Introduction of Mindstorms and NXT “brains”

**Resource** – Legos Mindstorms Education kits (NXT or EV3)

Day 2 **Class Activity** – Lego NXT

**Assessment** – What is a Program? Worksheet

**Instruction** – TeachEngineering.org PowerPoint

**Resource** – TeachEngineering.org worksheet: [What Is a Program?](#); TeachEngineering.org PowerPoint: [What is a Program?](#)

Day 3 **Class Activity** – Build robots; learn to program

**Instruction** – Programming of robots

**Resource** – [lego.com](http://lego.com): [Lego Mindstorms Learn To Program](#) and [Mindstorms User Guide](#)

Day 3 **Class Activity** – Investigating Distance

**Instruction** – Individualized, as needed

**Resource** – communityeducation.lego.com: [Investigating Distance](#)

Day 4 **Class Activity** – Moving Straight

**Instruction** – Individualized, as needed

**Resource** – communityeducation.lego.com: [Moving Straight](#)

Day 5 **Class Activity** – Work/”Catch-Up” Day

**Assessment** – Logbook checks

**Instruction** – Individualized, as needed

Day 6 **Class Activity** – Move Straight: A Dickey Situation

**Instruction** – Individualized, as needed

**Resource** – communityeducation.lego.com: [Move Straight: A Dickey Situation](#)

Day 7 **Class Activity** – Dancing Robots

**Assessment** – Informal and formative assessment: Can students design and build a robot that will dance to music for 20 seconds? The dance must include at least 3 different dance moves.

**Instruction** – Individualized, as needed

**Resource** – Dance music

Day 8 **Class Activity** – Move until Touch (using touch sensors)

**Instruction** – Sensors: what are they and how do they work?

**Resource** – [www.education.rec.ri.cmu.edu](http://www.education.rec.ri.cmu.edu): [Move Until Touch](#)

Day 9 **Class Activity** – Challenge: Move until Touch Vacuum Challenge

**Instruction** – Individualized, as needed

**Resource** – [www.education.rec.ri.cmu.edu](http://www.education.rec.ri.cmu.edu): [Vacuum Challenge](#)

Day 10 **Class Activity** – Catch Up Day

**Assessment** – Vacuum Challenge; logbook check

**Instruction** – Individualized, as needed

Day 11 **Class Activity** – Repeating Behaviors

**Instruction** – Loops

**Resource** – [www.education.rec.ri.cmu.edu](http://www.education.rec.ri.cmu.edu): [Repeating Behaviors](#)

Day 12 **Class Activity** – Buried Treasure Challenge

**Assessment** – Buried Treasure Challenge

**Instruction** – Individualized, as needed

**Resource** – [www.education.rec.ri.cmu.edu](http://www.education.rec.ri.cmu.edu): [Buried Treasure Challenge](#)

Day 13 **Class Activity** – Log: 1) Think of your favorite app, game, or website. These are all programs. Try to list all the things that it can do. 2) How do you think the app, game, or website you chose was made?

**Instruction** – Discussion; vocabulary; set up Codesters accounts; complete activity 1 as a class; model activities 2 and 3; complete activity 4 as a class; students complete activities 5 and 6

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 14 **Class Activity** – Review previous day’s activities; students complete activities 7-10

**Instruction** – Importance of syntax (quotes, parentheses, and correct spelling)

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 15 **Class Activity** – Students complete activities 11 – 13.

**Instruction** – Importance of order within a program

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 16 **Class Activity** – Students complete activities 14 – 17.

**Instruction** – Individualized, as needed

**Assessment** – Activity 15 and 16 Exit Ticket

**Extension** – Activity 17 (changing background and sprite; adding dialogue and actions)

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 17 **Class Activity** – Codesters Lesson #2: students complete activities 2 – 6.

**Instruction** – Vocabulary; coordinate plane and guiding of sprites around stage

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 18 **Class Activity** – Students complete activities 7 – 10.

**Instruction** – Class discussion of “debugging”

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 19 **Class Activity** – Students complete activities 11 – 19.

**Instruction** – Individualized, as needed

**Assessment** – Activity 20 Exit Ticket

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 20 **Class Activity** – Log: “What shapes make up this house? Can you tell what shapes go with what code?”; Codesters Lesson #3: students complete activities 3 – 8.

**Instruction** – Vocabulary; Discussion of parameters

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 21 **Class Activity** – Students complete activities 9 and 10.

**Instruction** – Individualized, as needed

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 22 **Class Activity** – Students complete activity 11.

**Instruction** – Individualized, as needed

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 23 **Class Activity** – Students complete activities 12 – 16.

**Instruction** – Explanation of how to change the name of a shape by changing the word before the equal sign

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 24 **Class Activity** – Students complete activity 17.

**Instruction** – Individualized, as needed

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 25 **Class Activity** – Students complete activities 18 – 25.

**Instruction** – Individualized, as needed

**Assessment** – Activity 26 Exit Ticket

**Extension** – Students add additional parts to their house (chimney, etc.), change aspects of the shapes, and/or add background and sprite.

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 26 **Class Activity** – Log: “What lines collect user input? What lines use the information that the user provided?”; Codesters Lesson 4; Students complete activities 3 – 11.

**Instruction** – “If” statements

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 27 **Class Activity** – Students complete activity 12.

**Instruction** – Storage of user-input

**Assessment** – Interactive activity to check for understanding

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 28 **Class Activity** – Students complete activities 14 and 15.

**Instruction** – Individualized, as needed

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 29 **Class Activity** – Students complete activities 16 – 20.

**Instruction** – Individualized, as needed

**Assessment** – Activity 21 Exit Ticket

**Extension** – Students create a build a program that promotes a local animal shelter and



pet adoption.

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 30 **Class Activity** – Log: “In what order do the recycling sprites appear on the stage? Can you find the line in the editor where you first see those image names in that order?”; Codesters Lesson #5; Students complete activities 2 – 10.

**Instruction** – Vocabulary; loops

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 31 **Class Activity** – Students complete activities 11 and 12.

**Instruction** – Individualized, as needed

**Assessment** – Quick check question for understanding

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 32 **Class Activity** – Students complete activities 13 and 14.

**Instruction** – Individualized, as needed

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 33 **Class Activity** – Students complete activities 15 and 16.

**Instruction** – Explanation of looping

**Resource** – [www.codesters.com](http://www.codesters.com)

Day 34 **Class Activity** – Students complete activities 17 – 20.

**Instruction** – Adding commands to a loop

**Assessment** – Activity 21 Exit Ticket

**Extension** – Students create an animated logo to promote their school or favorite sports team.

**Resource** – [www.codesters.com](http://www.codesters.com)

## 4. STEM 2

Day 1 **Class Activity** – Open Python; TechRocket Chapter 1

**Instruction** – What is Python and why we’re learning it.

**Assessment** – Pre-assessment

**Resource** – [python-for-kids](#); Python-for-Kids chapter 1

Day 2 **Class Activity** – Print command in Python; TechRocket Chapter 2

**Instruction** – Using the print command in Python

**Resource** – [python-for-kids](#); [TechRocket](#)

Day 3 **Class Activity** – Tech Rocket Chapter 3: Variables; Python-for-kids Chapter 2: Calculations and Variables

**Instruction** – Via TechRocket

**Resource** – [TechRocket](#); [python-for-kids](#)

Day 4 **Class Activity** – Tech Rocket chapter 3.2 and 3.3: All about variables

**Assessment** – Formative Assessment #1 (see Teacher Resource Document); Project 1

**Instruction** – Via TechRocket

**Resource** – [TechRocket](#)

Day 5 **Class Activity** – Tech Rocket chapter 3.4: Concatenation

**Instruction** – Via TechRocket

**Resource** – [TechRocket](#)

Day 6 **Class Activity** – Tech Rocket chapter 4.1: String and Numeric Data Types; 5.1 – 5.3: Operators

**Instruction** – Via TechRocket

**Resource** – [TechRocket](#)

Day 7 **Class Activity** – Python-for-kids chapter 3: Strings, Lists, Tuples, and Maps

**Assessment** – Greetings! Program Puzzle

**Instruction** – Individualized, as needed

**Resource** – [python-for-kids](#)

Day 8 **Class Activity** – Python-for-kids chapter 4: Drawing with Turtles

**Assessment** – Rectangle Puzzle (bronze level), Triangle Puzzle (silver level), or Box without Corners (gold level)

**Instruction** – Individualized, as needed

**Resource** – [python-for-kids](#)

Day 9 **Class Activity** – TechRocket 6.1-6.2: Commenting

**Instruction** – Via TechRocket

**Resource** – [TechRocket](#)

Day 10 **Class Activity** – Tech Rocket chapter 6.3: Debugging/Quarantining

**Assessment** – Formative Assessment #2 (see Teacher Resource Document)

**Instruction** – Via TechRocket

**Resource** – [TechRocket](#)

Day 11 **Class Activity** – Python-for-kids chapter 5: Asking Questions with If and Else (conditionals)

**Assessment** – Are You Rich Puzzle; Project 2

**Instruction** – Conditionals

**Resource** – [python-for-kids](#)

Day 12 **Class Activity** – Assessment/Catch-Up Day

**Instruction** – Individualized, as needed

**Resource** – [TechRocket](#)

Day 13 **Class Activity** – Python-for-kids chapter 6: Going Loopy (Loops)

**Assessment** – The Hello Loop Puzzle

**Instruction** – Loops

**Resource** – [python-for-kids](#)

Day 14 **Class Activity** – Catch-Up/Extension Day

**Assessment** – Project 3

**Instruction** – Individualized, as needed

Day 15 **Class Activity** – Tech Rocket chapter 7.1 and 7.2: Organizing Projects and Declaring Variables

**Assessment** – Formative Assessment #3 (see Teacher Resource Document)

**Instruction** – Via TechRocket

**Resource** – [TechRocket](#)

Day 16 **Class Activity** – Tech Rocket chapter 7.3 and 7.4: Random Values; Import; Modules; Random Range

**Instruction** – Via TechRocket

**Resource** – [TechRocket](#)

Day 17 **Class Activity** – Tech Rocket chapter 8.1: User Input

**Assessment** – Project 4

**Instruction** – Via TechRocket

**Resource** – [TechRocket](#)

Day 18 **Class Activity** – Tech Rocket chapter 8.2: Casting Input

**Instruction** – Via TechRocket

**Resource** – [TechRocket](#)

Day 19 **Class Activity** – Tech Rocket chapter 9.1

**Instruction** – Via TechRocket

**Resource** – [TechRocket](#)

Day 20 **Class Activity** – Tech Rocket chapter 9.2: Recap

**Instruction** – Via TechRocket

**Resource** – [TechRocket](#)

Day 21 **Class Activity** – Python-for-kids chapter 12: Using Tkinter for Better Graphics

**Instruction** – Tkinter usage

**Resource** – [python-for-kids](#)

Day 22 **Class Activity** – Bounce! Python-for-kids chapter 13: Beginning Your First Game

**Instruction** – Individualized, as needed

**Resource** – [python-for-kids](#)

Day 23 **Class Activity** – Bounce! Python-for-kids chapter 14: Finishing Your First Game

**Instruction** – Individualized, as needed

**Resource** – [python-for-kids](#)

Day 24 **Class Activity** – Final Assessment/Project

**Assessment** – Project 5

**Instruction** – Individualized, as needed

**Resource** – [TechRocket](#) and [python-for-kids](#)

Day 25 **Class Activity** – Arduino Introduction

**Assessment** – Pre-assessment

**Instruction** – Circuit boards, microcontrollers, sensors, etc.

**Resource** – [What is Arduino?](#)

Day 26 **Class Activity** – Fundamentals of Electronics

**Instruction** – Circuits, voltage, current, resistance

Day 27 **Class Activity** – Vocabulary

**Instruction** – Electronics vocabulary

Day 28 **Class Activity** – Blink an LED

**Instruction** – Individualized, as needed

**Resource** – learn.sparkfun.com: [Experiment 1](#)

Day 29 **Class Activity** – Reading a Potentiometer

**Assessment** – Logbook check

**Instruction** – Individualized, as needed

**Resource** – learn.sparkfun.com: [Experiment 2](#)

Day 30 **Class Activity** – Driving an RGB LED

**Instruction** – Individualized, as needed

**Resource** – learn.sparkfun.com: [Experiment 3](#)

Day 31 **Class Activity** – Driving Multiple LEDs

**Instruction** – Individualized, as needed

**Resource** – learn.sparkfun.com: [Experiment 4](#)

Day 32 **Class Activity** – Reading a Button Press

**Instruction** – Individualized, as needed

**Resource** – learn.sparkfun.com: [Experiment 5](#)

Day 33 **Class Activity** – Reading a Photoresistor

**Instruction** – Individualized, as needed

**Resource** – learn.sparkfun.com: [Experiment 6](#)

Day 34 **Class Activity** – Reading a Temperature Sensor

**Assessment** – Logbook check

**Instruction** – Individualized, as needed

**Resource** – learn.sparkfun.com: [Experiment 7](#)

Day 35 **Class Activity** – Using a Servo Motor

**Instruction** – Individualized, as needed

**Resource** – learn.sparkfun.com: [Experiment 8](#)

Day 36 **Class Activity** – Driving a Motor with an H-Bridge

**Instruction** – Individualized, as needed

**Resource** – learn.sparkfun.com: [Experiment 9](#)

Day 37 **Class Activity** – Controlling a Motor with Inputs

**Instruction** – Individualized, as needed

**Resource** – learn.sparkfun.com: [Experiment 10](#)

Day 38 **Class Activity** – Reading Serial Data

**Instruction** – Individualized, as needed

**Resource** – learn.sparkfun.com: [Experiment 11](#)