

Coding, Computational Thinking, & Maker Education

Please Note: There are several hyperlinks throughout the document.

Introduction

Learners' explorations and learning include several tactile, visual, and auditory experiences. The tools and ideas shared build on the efforts of educators who provide environments that are conducive for exploration and learning.

We hope these lesson plans assist you in teaching coding and computational thinking in the form of maker education in your classroom.

Lesson Ideas

Choose from the activities found in the links below. We recommend choosing at least 3 centres, either using a rotational schedule, or over a period of 3 days.

1. **Coding Unplugged - No Tech Required**
2. **Micro:Bit - Micro:Bit kit & Device Required**
3. **Scratch - Device Required**
4. **Makey Makey and Scratch**
5. **Ozobot Racers**

Suggested Time

1. **Settling In:** 5 minutes
2. **Thinking About:** 5 minutes
3. **Demonstrating:** 2-3 minutes
4. **Exploring:** 10 minutes
5. **Challenges:** 20 minutes
6. **Consolidating:** 10 minutes
7. **Exit Tickets** 3-5 minutes

Teacher Tips

- Where the number of materials are sufficient, work on the same activity in small groups.
- **Suggested group size:** 4-6 students.
- If you are using centres/stations, it is important to have an adult at each station to monitor the tech being used. A group discussion can be used to introduce/close the station meaningfully
- You can choose to complete centres in a library, gym, classroom, or other communal space.
- View students as apprentices to help them develop transferable



skills. Ask students to think of which job would relate to the tasks they are working on (i.e. designer, media, scientist, engineer, mathematician, modelling, etc.).

- You may view your role as a facilitator or supervisor in a workplace or lab. This will allow students to engage, observe, explore, question, and play in a productive learning environment.

Exit Ticket & Sharing Beyond the Classroom

In these activities, we have included a variety of exit tickets students can complete to consolidate their learning. We also suggest you include discussions at the beginning and end of your lesson. This will help learners focus on the curriculum, transferable skills, and technology skills they have learned. Please adapt these to your students' needs.

Students will be excited and proud to talk about what they have learned, the ideas they imagined, and the levels/challenges they completed. This is why we have included a printable [Letter to Parents/Guardians](#) so that they can continue to share and learn outside of the classroom.

Overall Curricular Connections:

- Review geometry math (e.g. polygons, angles, perimeter)
- Motion math (lines, directions, grids)
- Turns/rotations, angles, interior and exterior, benchmark angles
- Flips/Reflections, Slides/Translations
- Scale/Size
- Rates math (e.g. speed, estimate elapsed time and distance, and quantities such as percentages)
- Locations, coordinates
- Measurement of distance, length, perimeter, area, angles.
- Shapes as pathways simulated by a robot.

Coding & Technology concepts

As the following coding concepts and processes arise during the activities, you may explore them with students:

- Sequence
- Logical organization of: algorithms, programs, commands, code
- Study tasks, data, controls, operations, variables, conditions
- Develop interface knowledge, and design interactive interfaces and simulations.
- Create steps and organize code to solve a problem
- Navigate mazes/puzzles
- Output and effectors such as light and sound.
- Input, sensors

- Ideate, prototype, design, analyze, revise.
- Work with block codes (e.g. create, order, understand, remix)
- Animate and simulate math characters, shapes, backgrounds, and apps.
- Problem solve and think, animate and simulate, represent with code.