

# Please don't print these slides

Think of the trees!

# Playing with Blocks

Blockly-based coding games



- Talk given August 2017 at the Games For Learning conference in Wellington

# Who am I?

- Rachel Fenichel
- Software Engineer
- Kids Coding team at Google
- [fenichel@google.com](mailto:fenichel@google.com)



All opinions are my own and not those of my employer.

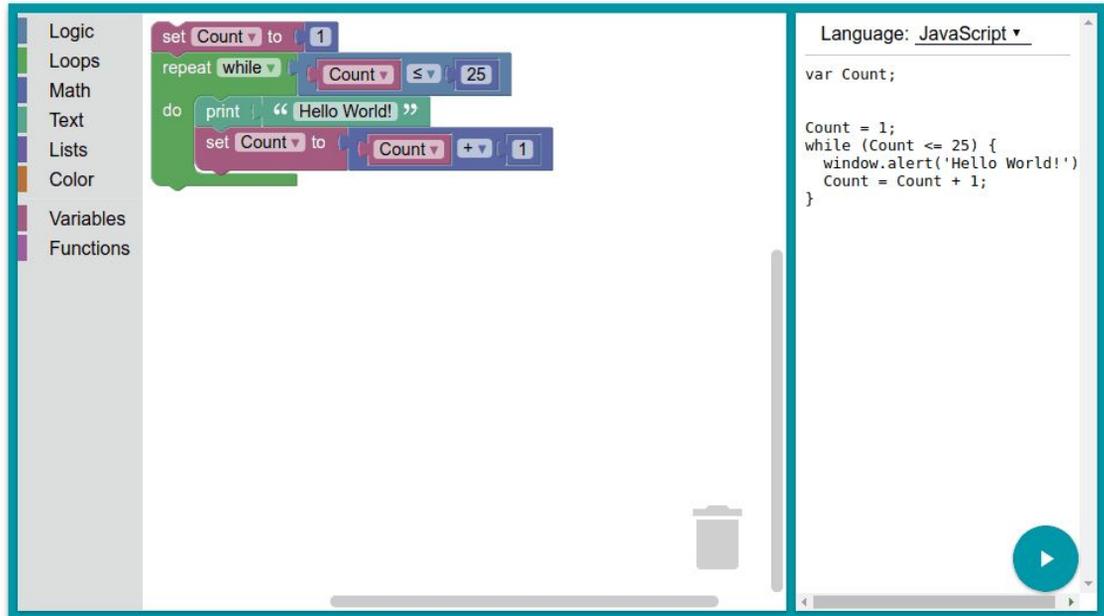
# Why am I here?

- My job: make it easier to teach kids to code
  - By working on an open-source project called Blockly
- Blockly is used to make educational toys and games
- Teachers: I want to share tools you can use in the classroom
- Game developers: I want to show you what Blockly can do

# What is Blockly

Blockly is a tool for building drag-and-drop code editors

Developers use Blockly to make educational games, toys, and more



- This is a demo of Blockly, aimed at developers
- Main components: workspace, toolbox, output
- Workspace: drag and drop code.
- Toolbox: see what you can do.
  - Helps with blank-slate fear.
- Output: code
  - Blockly is always used in an environment that can run that code
  - Can generate multiple languages
  - Code is well-formatted, to help users make the connection between the blocks and the text code.

# What isn't Blockly

Blockly is not a programming language or an app for end users

# Built with

- Scratch
- Gamefroot
- Code.org
- Blockly Games
- BBC Micro:bit

- I'm here to talk about things that are built with Blockly
- Teachers will not use Blockly directly.
- They are more likely to use one of these programs.
- I'll talk about each one

# Questions I'll answer

Cost	?
Lesson plans	?
Community	?
Build games with code	?
Play games with code	?
Text-based coding available	?
Ages	?

- For each of these project I'll aim to answer these questions
- There will be a summary slide for easy comparison

# Questions I'll answer

- Why use it?
- Why not use it?

- I'll cover pros and cons of each program
- These are only suggestions. Evaluate everything in the context of your own classroom.

Starting from



- Very well-known block-based coding website
- Used worldwide
- Used by kids ages 8-18
- In the process of designing and programming Scratch projects, young people learn to think creatively, reason systematically, and work collaboratively. Scratch is a project of the Lifelong Kindergarten group at the MIT Media Lab. It is available for free at <http://scratch.mit.edu>

Scratch Create Explore Tips About Search Join Scratch Sign in

Create stories, games, and animations  
Share with others around the world

TRY IT OUT SEE EXAMPLES JOIN SCRATCH (it's free)

A creative learning community with **24,250,299** projects shared

[ABOUT SCRATCH](#) | [FOR EDUCATORS](#) | [FOR PARENTS](#)

Featured Projects

Framed Triangle  
raucket

Go to Trash Can!  
yuki384

Micro Dungeon  
-TSTD-

Clean Water For All  
RainbowCloud525

Chickens!!!  
scrappymommy

- Let's take a look at the Scratch home page
- "Create stories, games, and animations"
- "Share with others around the world"
- Two core ideas here: creativity and community
- Low floor, wide walls, high ceiling: it's easy to get started, you can make a wide variety of projects, and you can make arbitrary complicated projects.
- Featured projects: one project per day. Scratchers care about getting here.
- What happens when we click on one of those projects?

**Adrift**  
by -TorpedoTurtle-

28 scripts  
7 sprites [See inside](#)



**Instructions**

This game took so long to make so I hope you enjoy!  
In this game the world is ending and you must return a star to a pedestal while jumping on broken wooden planks and dodging meteors!

~Controls~  
WASD or Arrow Keys

**Notes and Credits**

Art: -TorpedoTurtle-  
Music: Tobu  
Fonts: Endor and Bauhaus93  
OMG FEATURED THX SO MUCH EVERYONE!!!!!!! :D  
SO MANY MESSAGES THIS NEVER HAPPENED BEFORE!!! XD

Shared: 25 Jul 2017 Modified: 19 Aug 2017

★ 913 ♥ 1260 👁 17678 🌸 59

**Comments (1383)**

Leave a comment

You have 500 characters left.

[Post](#) [Cancel](#)

**Remixes (55)** [View all](#)



- This is what you see when you click on a project.
- This one is a platformer game that someone built.
- Immediately playable: click on the green flag.
- The maker can give you instructions and credit other Scratchers who helped out.
- Stars, hearts, views, and comments: it's very social.
- Community is a big part of Scratch. Scratch is well-moderated and is a good place to learn to be a good internet citizen.

**Adrift**  
by -TorpedoTurtle-

28 scripts  
7 sprites [See inside](#)



**Instructions**

This game took so long to make so I hope you enjoy!  
In this game the world is ending and you must return a star to a pedestal while jumping on broken wooden planks and dodging meteors!

~Controls~  
WASD or Arrow Keys

**Notes and Credits**

Art: -TorpedoTurtle-  
Music: Tobu  
Fonts: Endor and Bauhaus93  
OMG FEATURED THX SO MUCH EVERYONE!!!!!!! :D  
SO MANY MESSAGES THIS NEVER HAPPENED BEFORE!!! XD

© Shared: 25 Jul 2017 Modified: 19 Aug 2017

★ 913 ❤️ 1260 👁️ 17678 🌸 59

**Comments (1383)**

Leave a comment

You have 500 characters left.

[Post](#) [Cancel](#)

**Remixes (55)** [View all](#)



Adrift colorful f...  
by Stonefursfan

- On the lower right you can see a list of remixes of the project.
- “Remixing” is when someone takes the project and makes it their own.
- Sometimes this is just changing colors. Sometimes it’s much more than that.
- Remixing is a core part of Scratch.
- From early on, teach that you don’t have to write every line of code to make it your own.
- Scratchers can make really complicated projects by building on work that others have done.
- One path into Scratch is to find a project that you like and decide to tweak it.

**Adrift**  
by -TorpedoTurtle-

28 scripts  
7 sprites

[See inside](#)



**Instructions**

This game took so long to make so I hope you enjoy!  
In this game the world is ending and you must return a star to a pedestal while jumping on broken wooden planks and dodging meteors!

~Controls~  
WASD or Arrow Keys

**Notes and Credits**

Art: -TorpedoTurtle-  
Music: Tobu  
Fonts: Endor and Bauhaus93  
OMG FEATURED THX SO MUCH EVERYONE!!!!!!! :D  
SO MANY MESSAGES THIS NEVER HAPPENED BEFORE!!! XD

Shared: 25 Jul 2017 Modified: 19 Aug 2017

★ 913    ♥ 1260    👁 17678    ❄ 59

**Comments (1383)**

Leave a comment

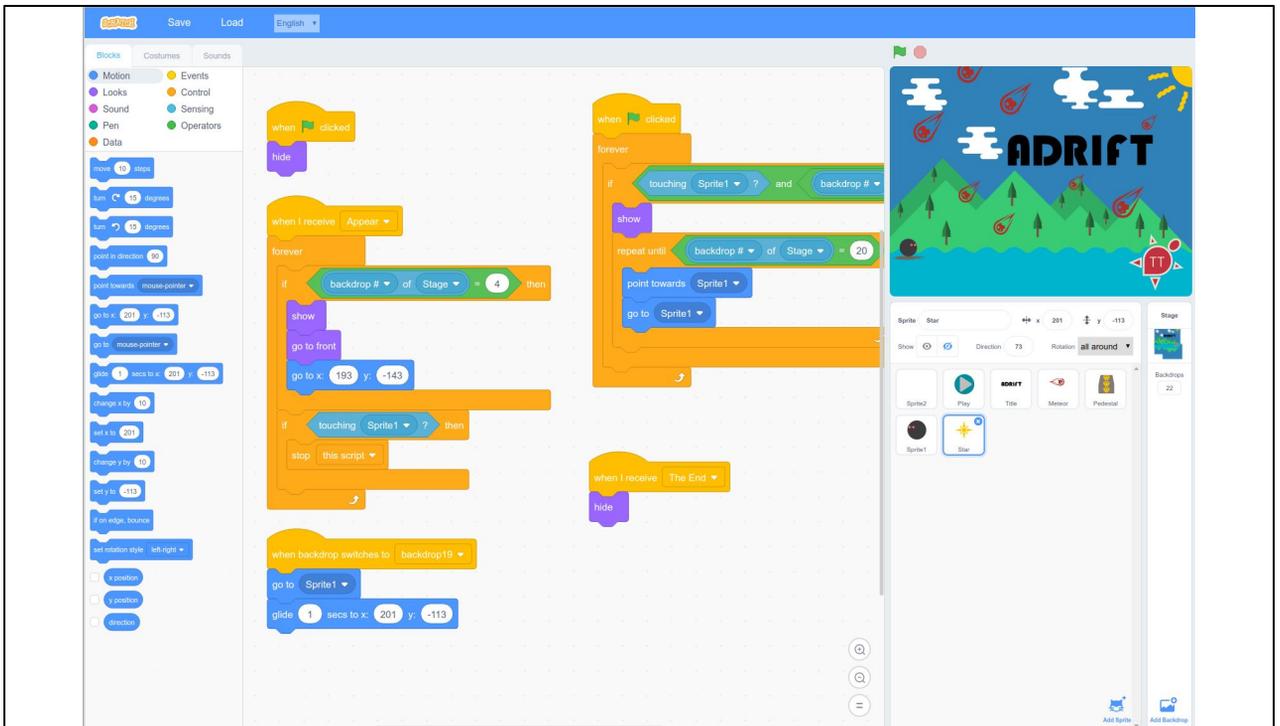
You have 500 characters left.

[Post](#) [Cancel](#)

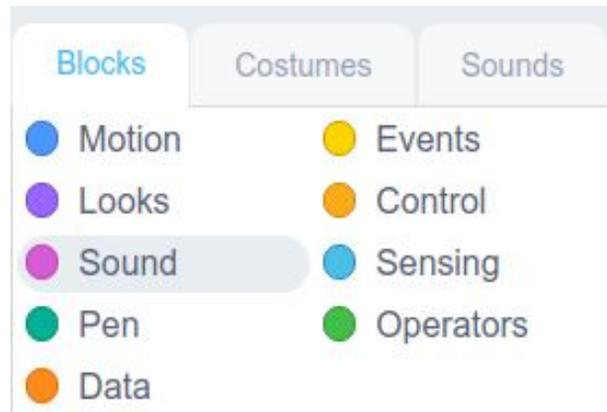
**Remixes (55)** [View all](#)



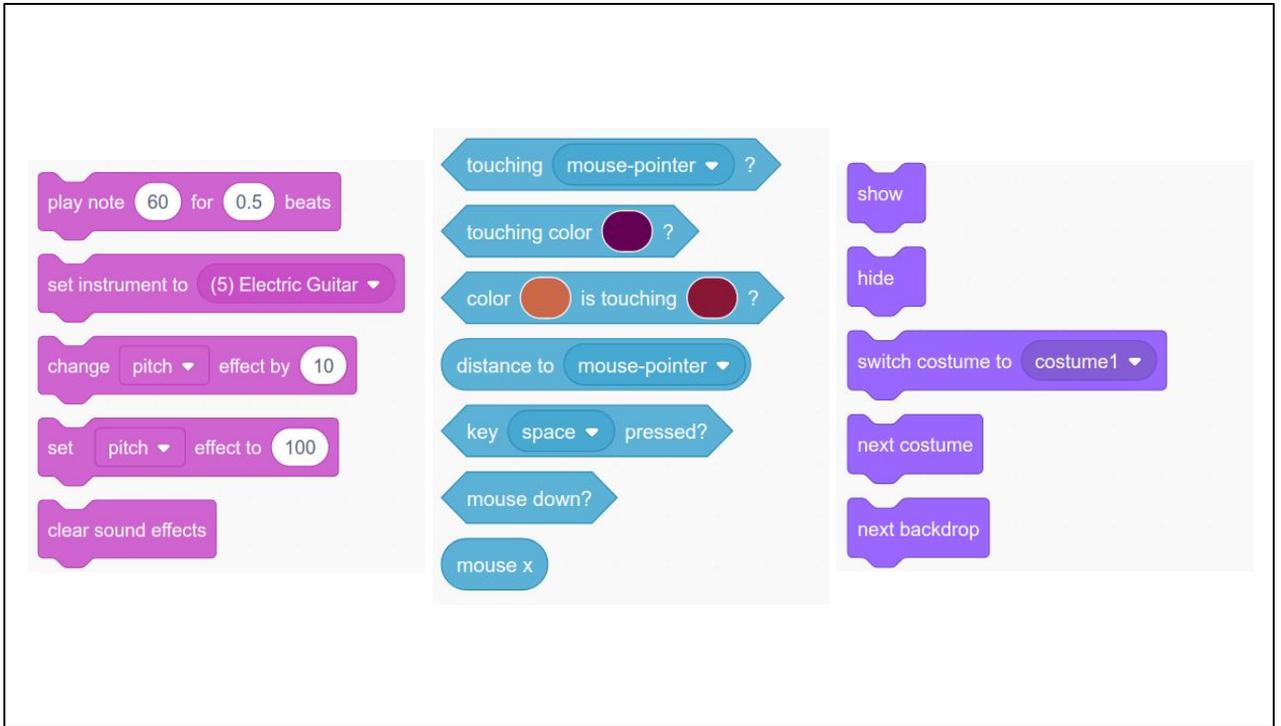
- On the top right there's a "see inside" button.
- You can look at the code and see how the project works.
- Let's see what happens when we click that button.



- This is the block editor
  - In Scratch 3.0
  - Current Scratch has a different editor, but with all of the same components.
- On the left we have the workspace and toolbox
- On the right, instead of text code, there's a stage.
- Changes that you make in code are instantly reflected in the stage.
- The blocks in the toolbox determine what things you can do.



- Scratch has three editors: Blocks, Costumes, and Sounds.
- In the block editor, these are the categories that you get.



- Blocks for sound, sensing, and looks.
- Form the basis of a lot of games and animations.

## Educator Resources

### A Community for Educators

[ScratchEd](#) is an online community where Scratch educators [share stories](#), exchange resources, ask questions, and find people. ScratchEd is developed and supported by the Harvard Graduate School of Education.

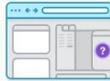
### In-Person Gatherings

[Scratch Educator Meetups](#) are gatherings of Scratch Educators who want to learn with and from each other, sharing their ideas and strategies for supporting computational creativity in all its forms.

### Guides & Tutorials



The [Things to Try page](#) offers a variety of tutorials, activity cards, and educator guides.



The [Tips Window](#) provides help for creating projects in Scratch.



The [Creative Computing Curriculum Guide](#) provides plans, activities, and strategies for introducing creative computing.

- Scratch has very solid educator resources: <https://scratch.mit.edu/educators>
- Ranges from getting started guides to lesson plans to communities of teachers using Scratch.
- There are also lesson plans through CS First: <https://www.cs-first.com/en/home>

## Complete Themes

These themes are intended for in- or after-school use over several days or weeks, and lead students through building 8 complete projects while introducing them to core computer science concepts.



### Storytelling

In Storytelling, students use computer science to tell fun and interactive stories. Storytelling emphasizes creativity by encouraging club members to tell a unique story each day.

[Try Now](#)

[View Lesson Plans](#)



### Friends

In Friends, students are encouraged to sign up with a friend or make a new friend in the club. Friends emphasizes teamwork by allowing club members to tell the story of how their friendship started and imagine a company together.

[Try Now](#)

[View Lesson Plans](#)



### Fashion & Design

In Fashion & Design, students learn how computer science and technology are used in the fashion industry while building fashion-themed programs, like a fashion walk, a stylist tool, and a pattern maker.

[Try Now](#)

[View Lesson Plans](#)

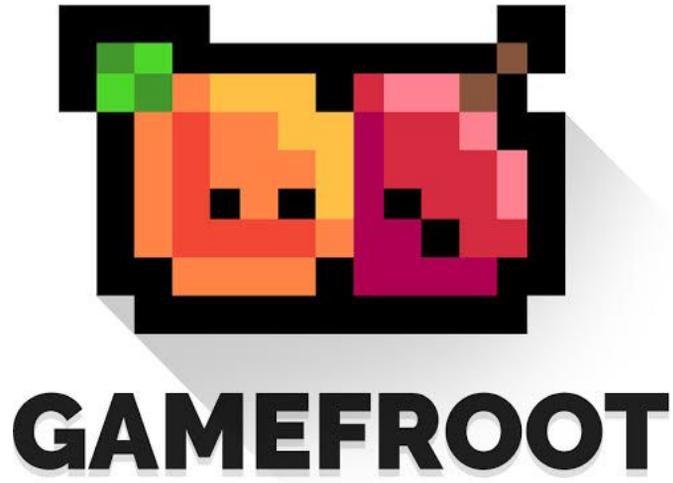
- CS First is a set of resources, lesson plans, etc.
- Made by Google
- Lesson plans use Scratch
- <https://www.cs-first.com/materials>

# Summary: Scratch

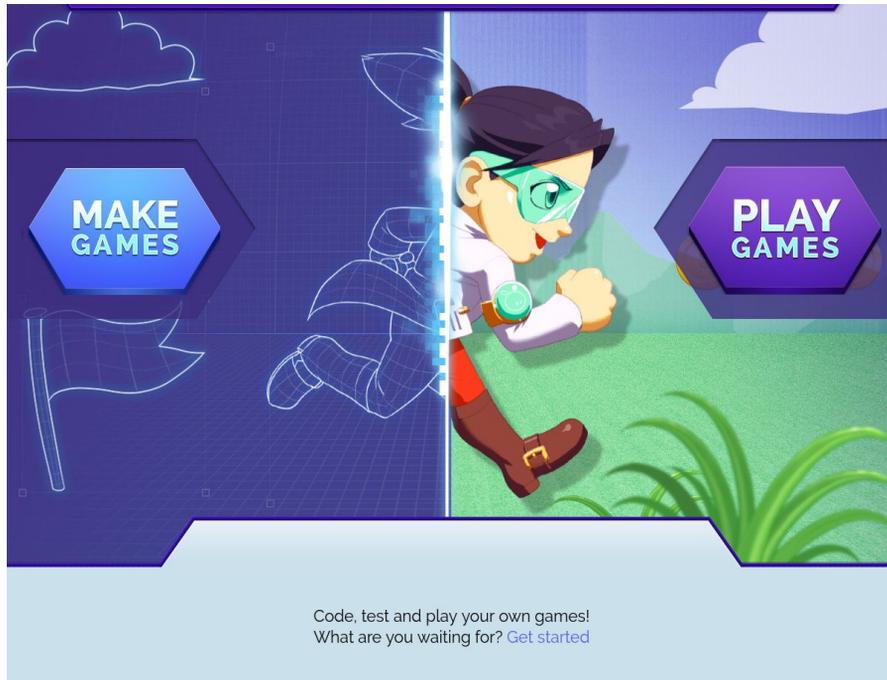
Cost	Free
Lesson plans	Yes
Community	Yes
Build games with code	Yes
Play games with code	No
Text-based coding available	No
Ages	Elementary school and up

# Summary: Scratch

- Why use it:
  - Excellent online community for kids
  - Remixing and sharing are core ideas
  - Low floor, wide walls, high ceiling
- Why not use it:
  - Open-ended nature makes it difficult to use in large classrooms



- Gamefroot is a New Zealand company, and instrumental in making this conference happen
- Dan Milward from Gamefroot is here showing how to use it.



- Let's take a look at the Gamefoot home page
- "Make games" and "Play games" are the important ideas here
- Immediate call to action: get started!
- Scrolling down, we'll see featured projects.

Code, test and play your own games!  
What are you waiting for? [Get started](#)

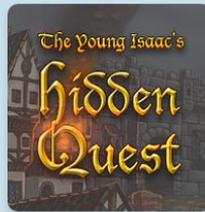
STAFF PICKS



Super Bounce  
by Gamefroot



Running for Ages  
by Gamefroot



The Young Isaac's  
Hidden Object Game  
by Gamefroot\_Dave



Asteroids  
by ryanc1256



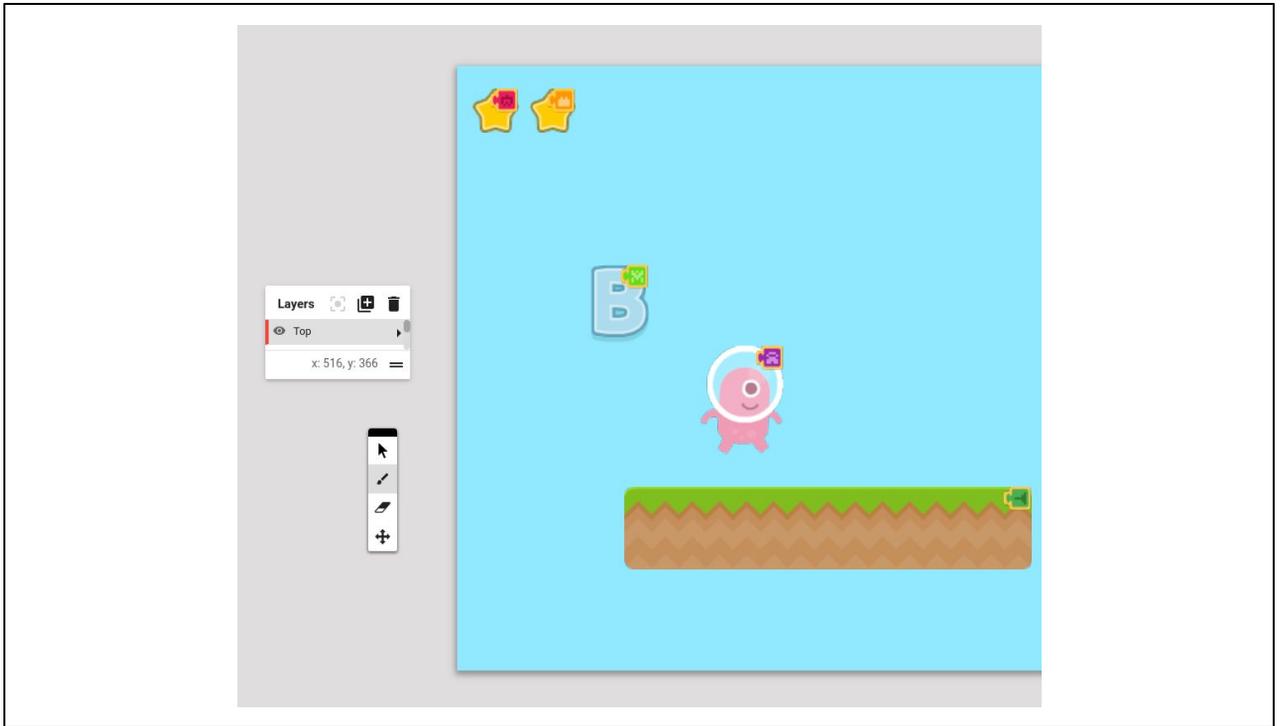
Vensian Vengeance  
by Dr. Grordbort's

Home page, continued:

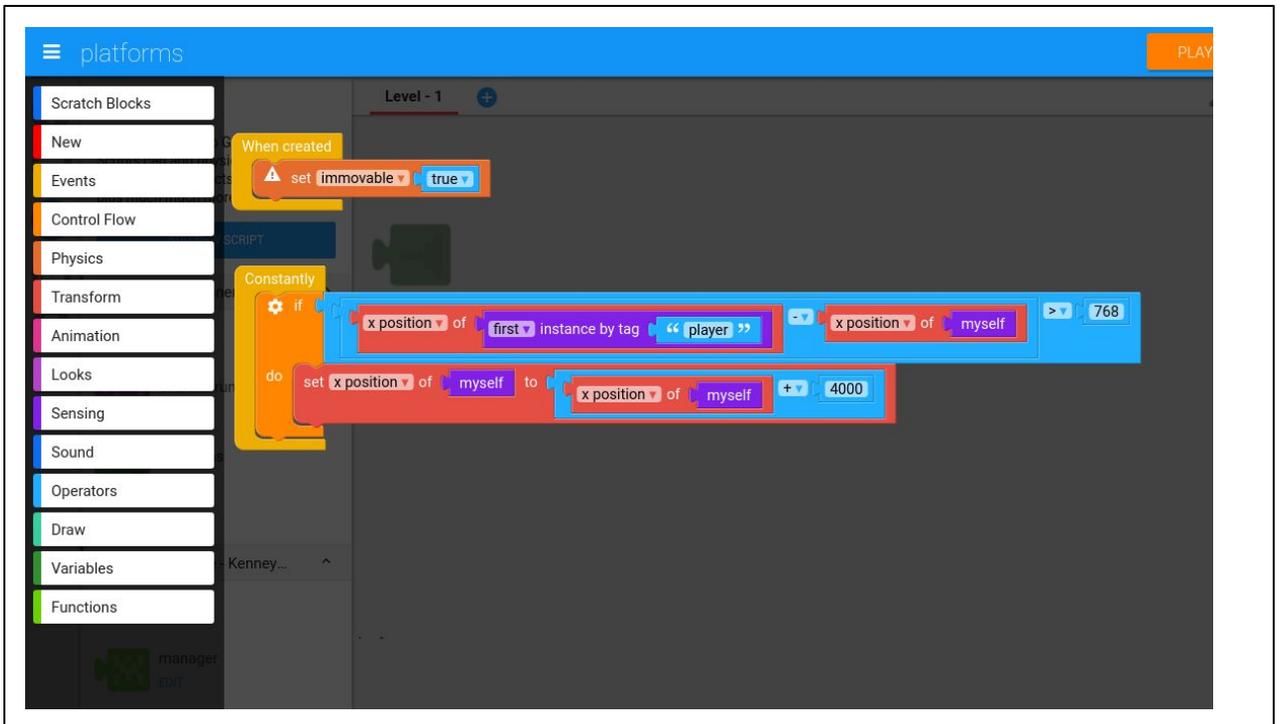
- Featured projects
- Seeing what other people are doing is really important.
- Let's look at one of these projects.



- Here's the basic view of a game.
- Very similar to Scratch
- Remixing is an important idea here as well.
- Hit remix and we'll see what's inside.



- Here's part of the level editor.
- It's powerful, but can be confusing.
- There are tutorials to help kids get started for the first time, so don't be too worried.
- Each thing on the level has a little block icon.
- Clicking that gets us to the code editor.



- Here's the code editor!
- Again, there's a workspace and a toolbox
- In this case it's overlaid on top of the level editor, but that's not a big difference.
- The set of categories in the toolbox is different, and more focused on games.

The image displays the Scratch block palette on the left, listing categories: Scratch Blocks, New, Events, Control Flow, Physics, Transform, Animation, Looks, Sensing, Sound, Operators, Draw, Variables, and Functions. To the right, two code snippets are shown. The first snippet, titled 'Camera Lock', contains the following blocks: 'center camera on myself', 'is camera locked', 'Camera Boundaries' (a section header), 'set camera x minimum to 0', 'set camera x maximum to 0', 'camera x minimum', and 'camera x maximum'. The second snippet contains: 'set immovable true', 'set reacts to gravity true', 'set mass 0', 'physics enabled', and 'moves'.

- There are some powerful blocks here:
- You can control what the “camera”--that is, the viewport--is focused on, so that it can move with your player.
- You can also add mass or enable gravity
- You can do these things in Scratch, but you have to code them yourself instead of having blocks for them.
- Reflects a different focus.

## Gamefroot and Curriculum

Resources are designed around the New Zealand curriculum and NCEA Units and can be used by teachers to award credits that can go towards student education but they can also be used as a lesson plan for teachers in other countries.

We also have a bunch of [fun and helpful tutorial resources](#) that you can use to guide your students through using Gamefroot to make games.

### Level 1 Units

**Unit: Game to Think – Design a Gamefroot arcade game!**

#### Games for Learning

- Teacher Training
- Student Workshops
- Curriculum & Resources

#### Need Help?

- FAQ
- Knowledgebase
- Support Request

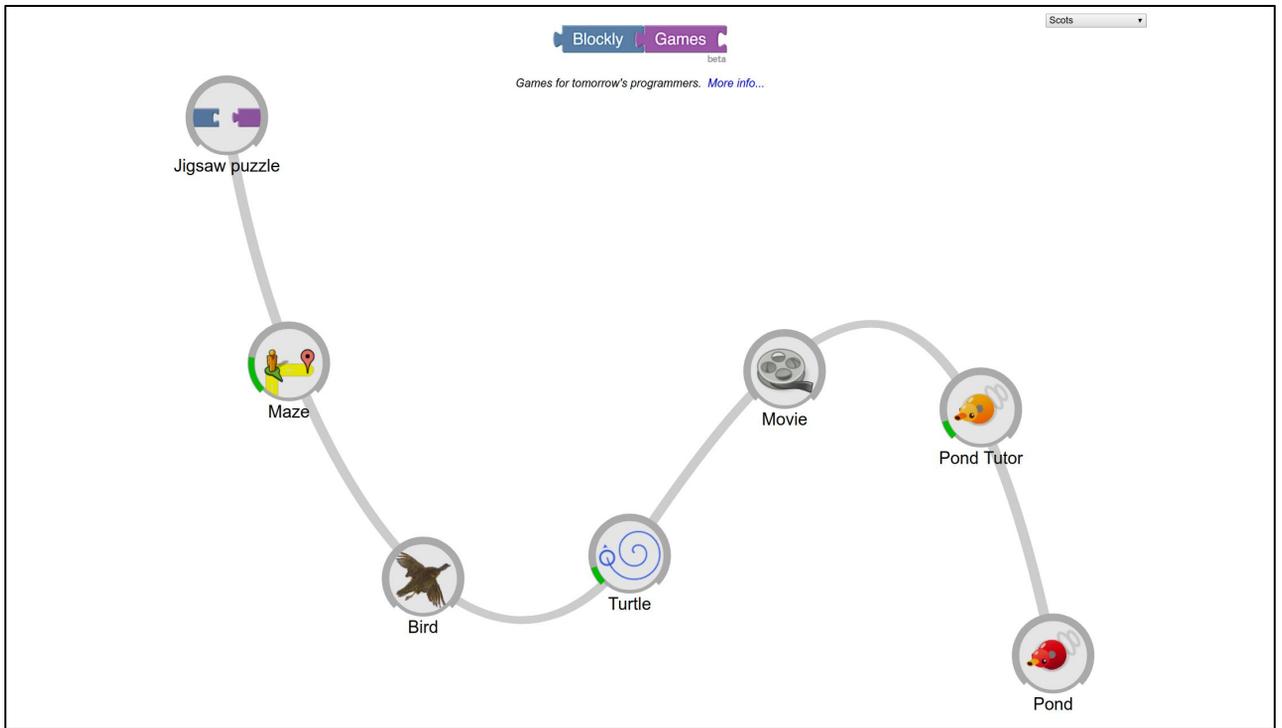
- Gamefroot for educators: <http://gamefroot.com/for-teachers/>
- Units built for NZ curriculum.
- Teacher training available.

# Summary: Gamefroot

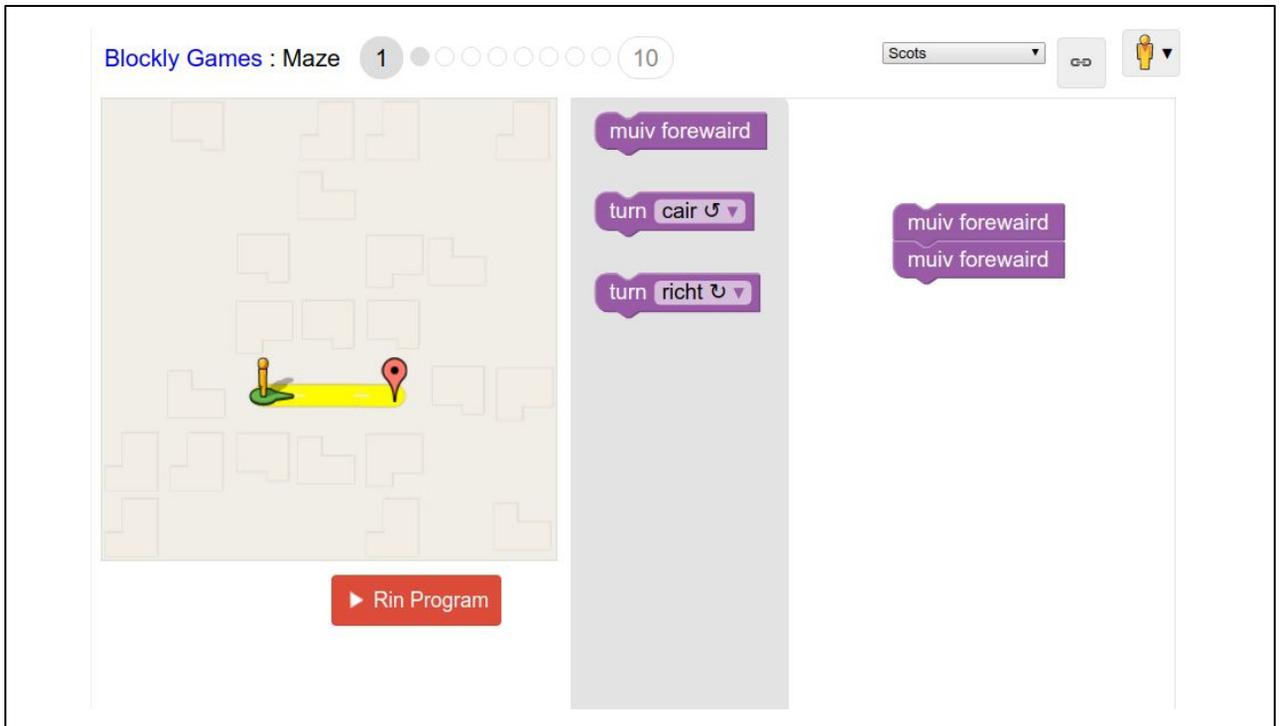
Cost	Free, with additional paid resources
Lesson plans	Yes
Community	Yes
Build games with code	Yes
Play games with code	No
Text-based coding available	No
Ages	Middle school and up

# Summary: Gamefroot

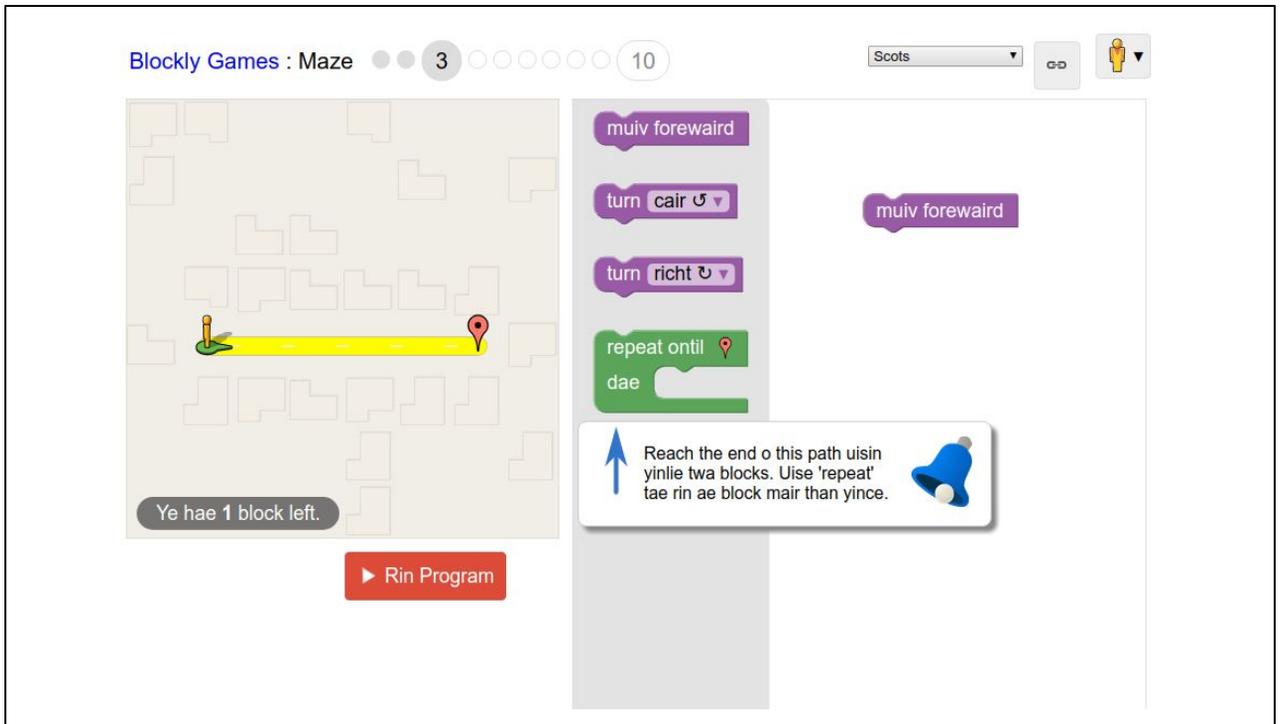
- Why use it:
  - Lesson plans designed around New Zealand curriculum
  - Remixing and sharing are core ideas
  - Powerful tools for game creation
- Why not use it:
  - May be daunting for beginners



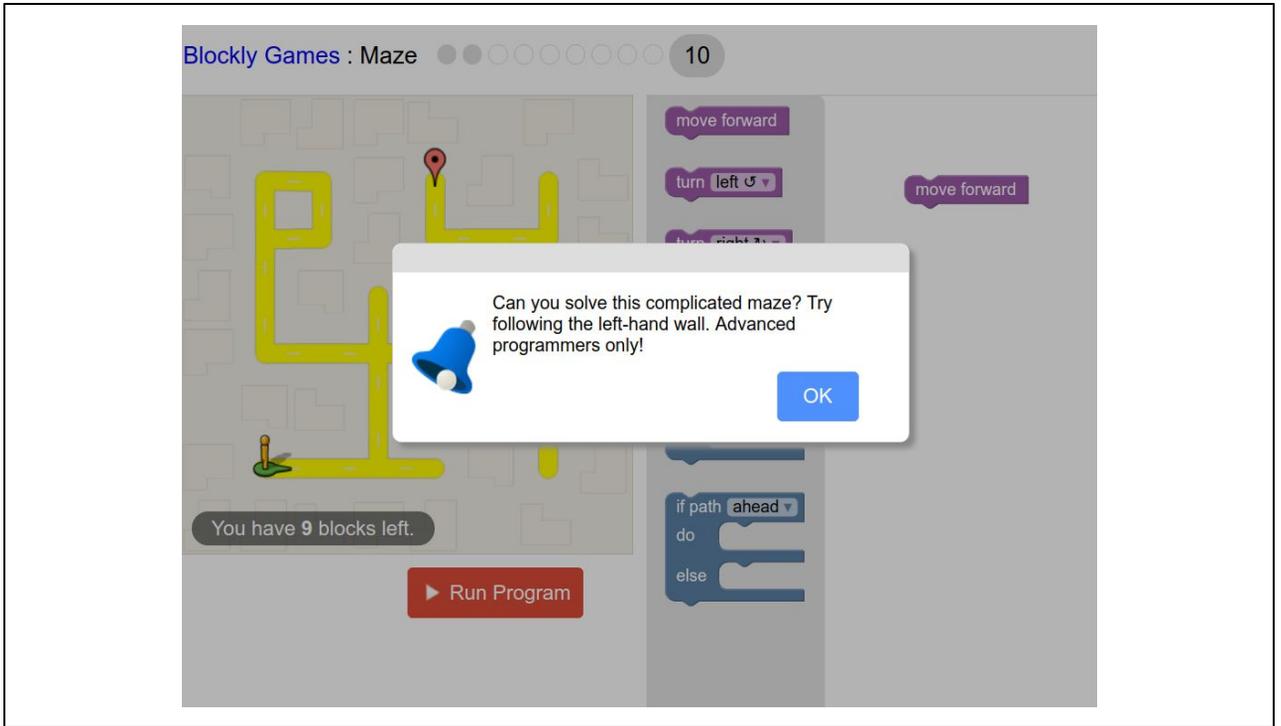
- Blockly Games is a project built by the Blockly team
- It's a series of games that you play using Blockly
- For example, there's the maze
- Website: <https://blockly-games.appspot.com/>



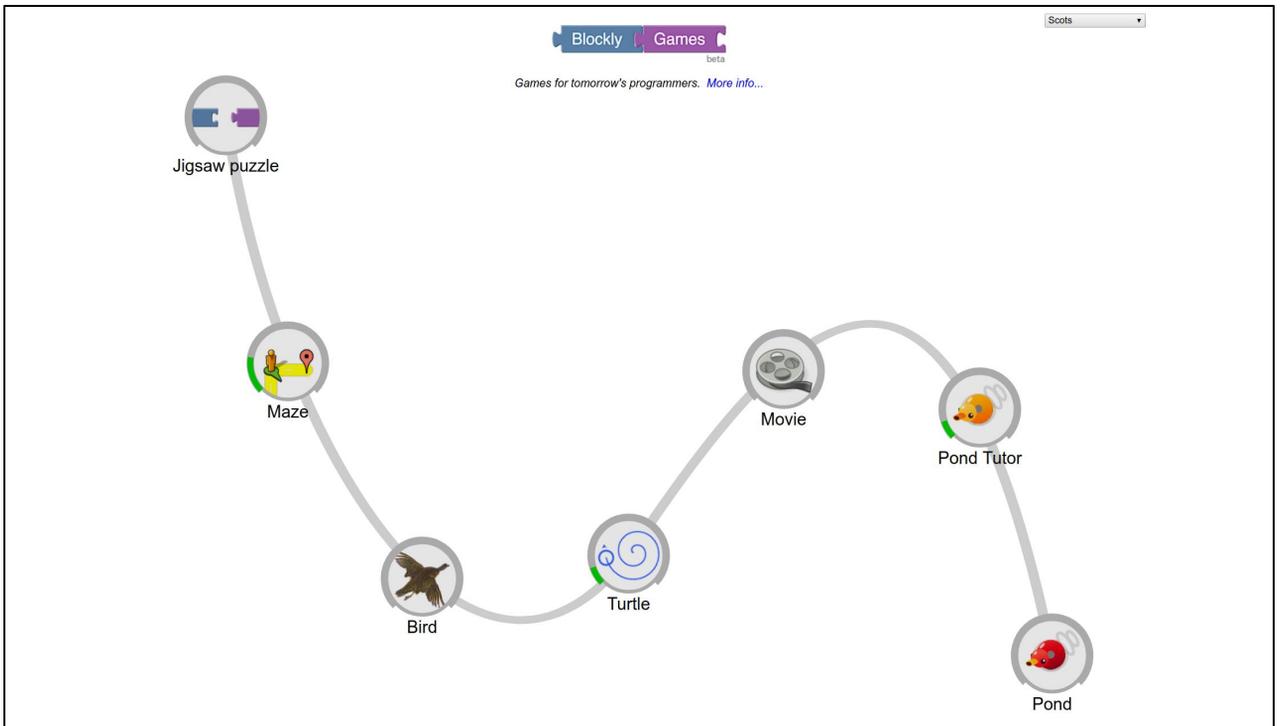
- Blockly Games: maze
- The usual layout: workspace, toolbox, etc.
- Code runs on the left.
- Notice that it's in Scots: Blockly generally does a good job with translation, and Blockly Games is published in a lot of languages.



- Maze, level 3
- There's a progression here, where it teaches the student how to use a new block or concept, then asks them to apply that.
- There are inline instructions, so that a student can be fully self-guided
- This makes it useful in large groups
- There's a limit on the number of blocks you can use.
- These artificial constraints make Blockly Games less creative, but helps with teaching specific concepts without teacher intervention.



- Maze level 10: Ending with a more open-ended challenge



- Blockly games is made up of a series of games.
- Each game has explicit progression between levels.
- Each game teaches a specific concept.
- There's also progression between games
- The first one teaches the basics of manipulating blocks, the next teaches sequences and loops, etc.

# Summary: Blockly Games

Cost	Free
Lesson plans	No*
Community	No
Build games with code	No
Play games with code	Yes
Text-based coding available	Yes
Ages	Middle school and up

\* No explicit lesson plans, but the games have clear progressions and instructions built in

<https://blockly-games.appspot.com/>

# Summary: Blockly Games

- Why use it:
  - Little teacher experience needed
  - Each game teaches a specific concept
  - Clear progression between levels and between games
- Why not use it:
  - Not open-ended

C

O

D

E

## Learn Computer Science



### Code.org courses

Code.org is the most popular coding platform in K-12 education. We offer fun introductory courses you can learn alone or in a classroom, all the way through full year computer science curriculum for middle or high school.

[I'm a student](#)

[I'm a teacher](#)

### Hour of Code

Get started by choosing a tutorial from over 100 one hour computer science activities from Code.org and partners. Join 10% of the world's students; it's your turn to try!

[Start](#)



### Courses from 3rd parties

Check out some of the 3rd party online courses we recommend, or find a local workshop, school, or summer camp to learn in a classroom.

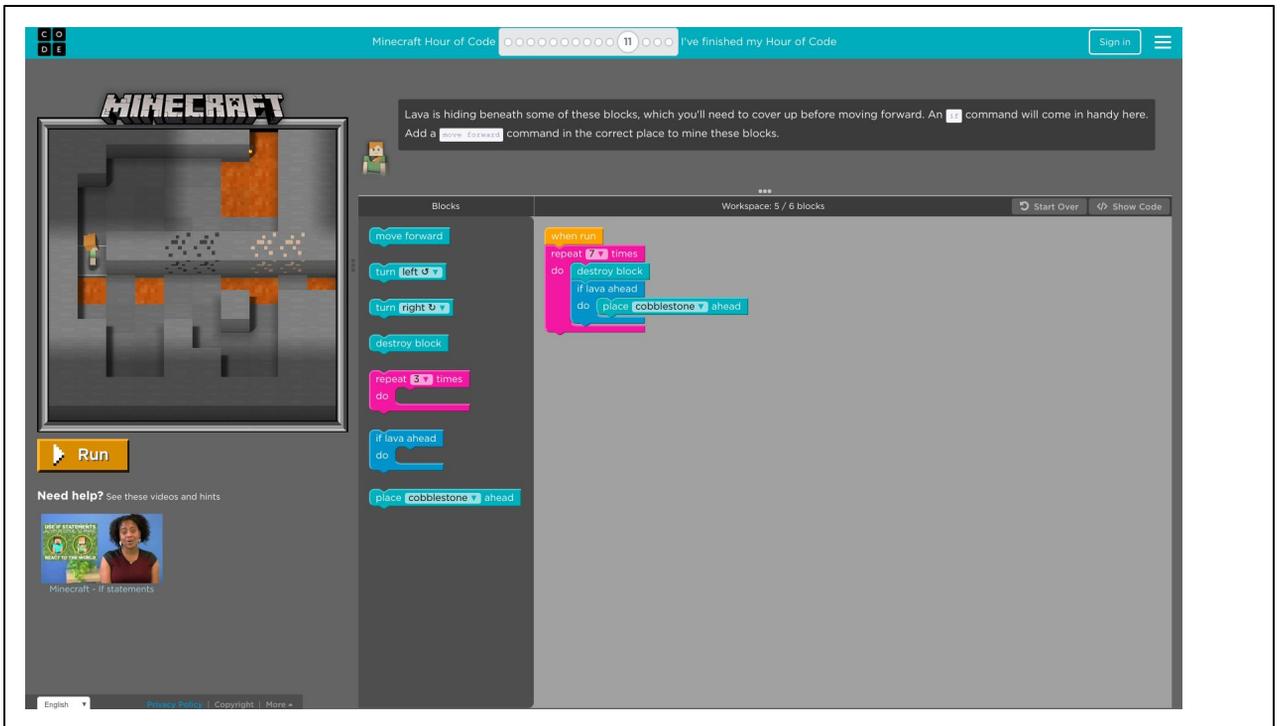
[Learn online](#)

[Local classes](#)

- Student portal: take full courses, or just do an hour of code.
- Also a jumping-off point for courses that other groups have made.

 <p><b>Minecraft</b> Program animals and other Minecraft creatures in your own version of Minecraft.</p>	 <p><b>Star Wars</b> Learn to program droids, and create your own Star Wars game in a galaxy far, far away.</p>	 <p><b>Frozen</b> Let's use code to join Anna and Elsa as they explore the magic and beauty of ice.</p>
 <p><b>Sports</b> Make a basketball game or mix and match across sports.</p>	 <p><b>Flappy Code</b> Wanna write your own game in less than 10 minutes? Try our Flappy Code tutorial!</p>	 <p><b>Classic Maze</b> Try the basics of computer science. Millions have given it a shot.</p>
 <p><b>Infinity Play Lab</b> Use Play Lab to create a story or game starring Disney Infinity characters.</p>	 <p><b>Play Lab</b> Create a story or make a game with Play Lab!</p>	 <p><b>Artist</b> Draw cool pictures and designs with the Artist!</p>

- Hour of Code: US initiative
- These activities all cover the same basic concepts
- Students can pick the wrapper that appeals to them the most



- Hour of code editor
- Similar to Blockly Games: inline instructions
- Clear progression (across the top) within a lesson
- Video and text resources when a student gets stuck
- As a result, can be self-guided.

## Recommended Code.org courses

[View my recent courses >](#)

Courses from Code.org for students in grades K-12 and professional learning for teachers.

Elementary school						Middle school			High school			
K	1	2	3	4	5	6	7	8	9	10	11	12
									CS Principles			
							CS Discoveries					
CS Fundamentals												
Pre-reader Express		CS Fundamentals: Express										

- Code.org for educators: <https://code.org/educate/curriculum>
- From an educator perspective, Code.org offers both full courses and short activities.
- Courses vary in length and age target, and include full lesson plans.
- Lesson plans are free.

### Grades

---

- Pre-reader
- Grades 2-5
- Grades 6-8
- Grades 9+

### Educator experience

---

- Beginner
- Comfortable

### Student experience

---

- Beginner
- Comfortable

### Classroom technology

---

- Computers
- Android
- iPad/iPhone
- Poor or no internet
- No computers or devices

### Topics

---

- Science
- Math
- Social Studies
- Language Arts
- Art, Media, Music
- Computer Science only

### Activity type

---

- Self-led tutorial
- Lesson plan

### Length

---

- One hour
- One hour with follow-on
- A few hours

### Language

---

- Blocks
- Typing
- Other

- When selecting activities, you can filter by a variety of parameters.
- Age and student experience are basic filters, but educator experience also matters.
- Classroom technology: Code.org includes fully unplugged activities and activities that work with spotty internet.

# Summary: Code.org

Cost	Free
Lesson plans	Yes, more than you can imagine
Community	No
Build games with code	Yes
Play games with code	Yes
Text-based coding available	Yes
Ages	Elementary school and up

# Summary: Code.org

- Why use it:
  - Tons of educator resources
  - Lessons for a wide variety of ages and interests
- Why not use it:
  - Takes extra work to find the right set of lessons for your class and available time



- The last topic is the BBC micro:bit
- The BBC micro:bit is a pocket-sized codeable computer with motion detection, a built-in compass and Bluetooth technology, which was given free to every child in year 7 or equivalent across the UK in 2016.
- <https://www.microbit.co.uk/home>

# Start your micro:bit adventure!

Meet micro:bit

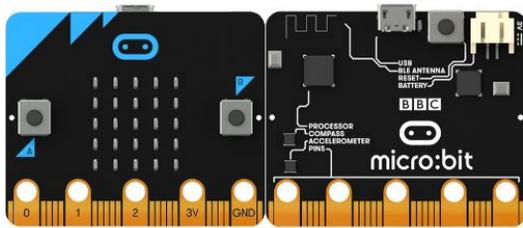
Quick Start

Apps

Hardware

Safety

Get Involved



## Hello!

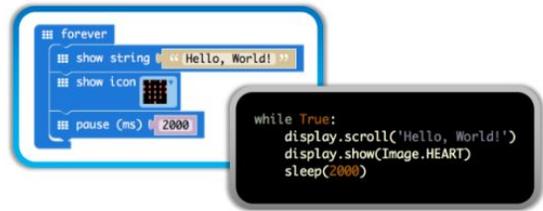
You can use your BBC micro:bit for all sorts of cool creations, from robots to musical instruments – the possibilities are endless.

- Home page is enthusiastic: let's get started!
- You can do quick start or jump into more complicated hardware challenges.

## Easy Peasy

It can be coded from any web browser in Blocks, Javascript, Python, Scratch and more; no software required.

Let's Code

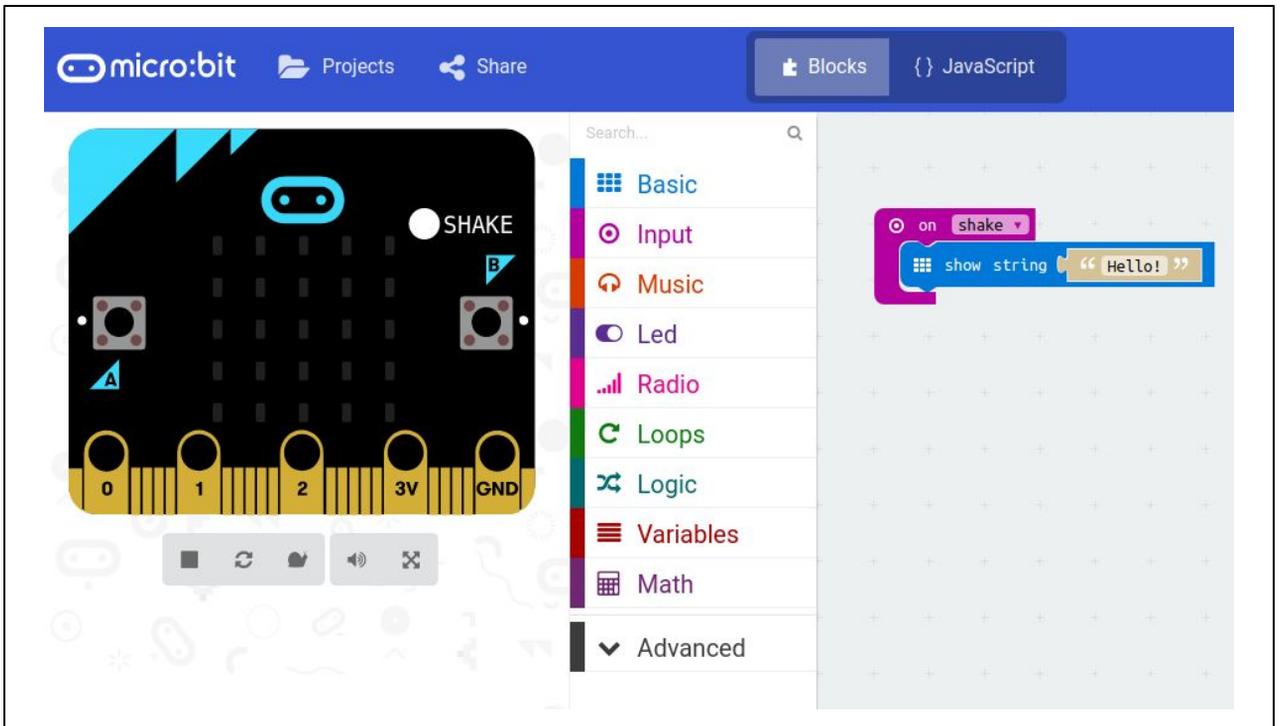


## Get Inspired

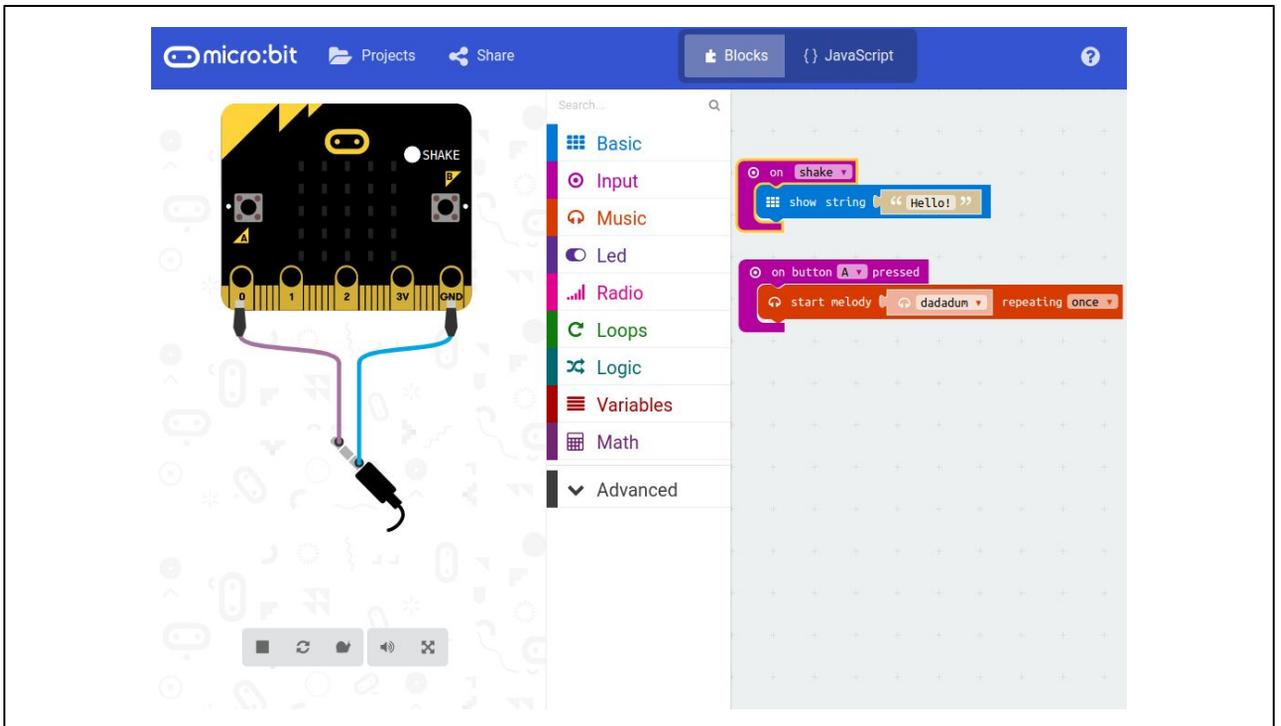
We have over 200 different activities and resources to try, from easy experiments to creative coding challenges.

Get ideas

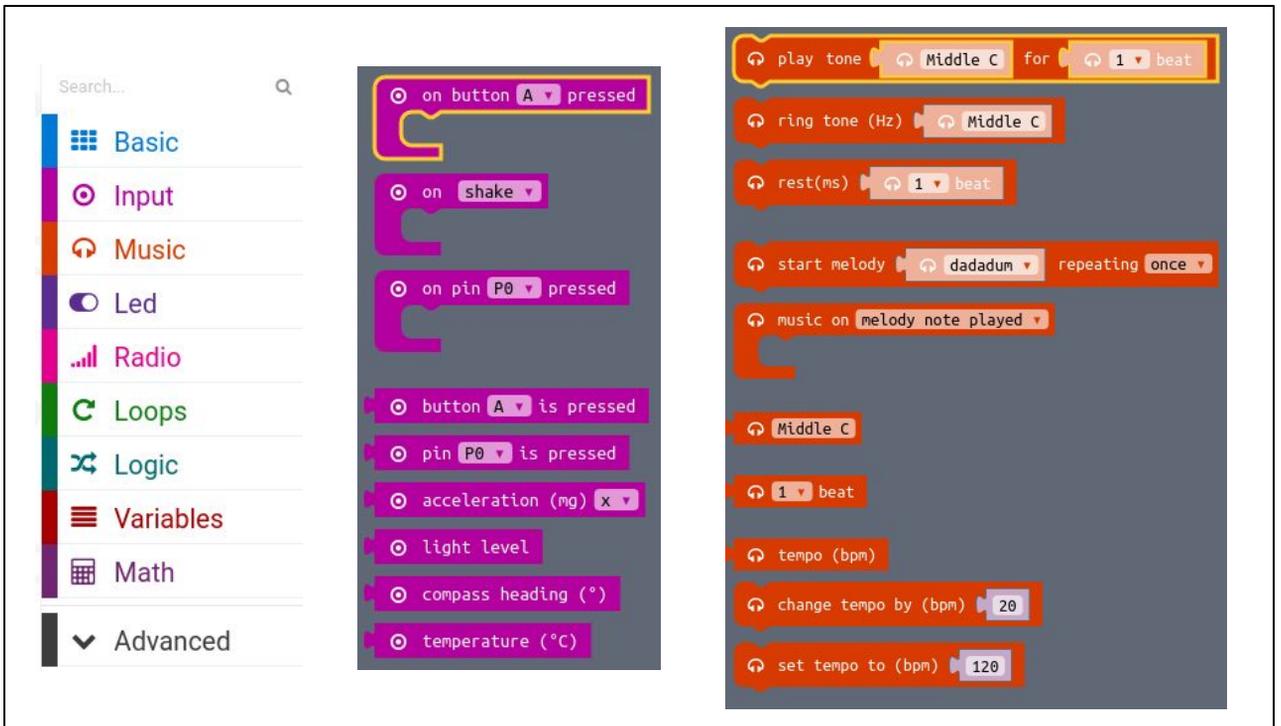
- There are multiple ways to code it, one of which is based on Blockly
- Students are presented with “activities”, “ideas”, and “challenges”
- Even if a given activity is very focused, the general idea is to promote open-ended creativity.



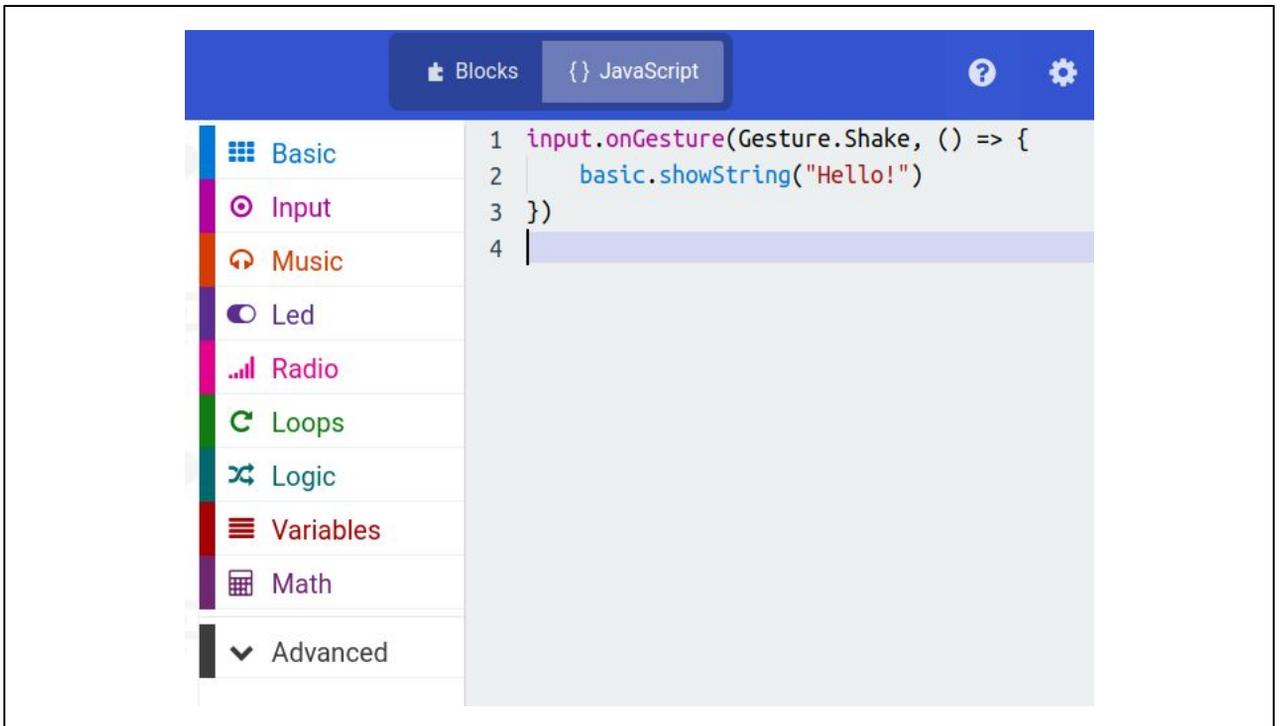
- Here's the basic block editor.
- Code on the right, and a simulator of the board on the left.
- You can use the simulator to test your programs if you don't have hardware, or if you don't have hardware for every student.



- Some blocks require different hardware.
- The simulator shows that, and shows how to hook that hardware up



- Available block categories include radio, input, music, and LEDs
- Works at multiple levels:
  - You can work with individual pins, or play specific notes
  - Or you can use higher level ideas, like sensing the acceleration and playing melodies.



- The micro:bit lets you switch to JavaScript.
- You can see what JavaScript your code turns into
- You can also edit that JavaScript

```
onGesture(Gesture.Shake, () => {  
  })
```

```
onPinPressed(TouchPin.P0, () => {  
  })
```

```
buttonIsPressed(Button.A)
```

```
pinIsPressed(TouchPin.P0)
```

```
acceleration(Dimension.X)
```

- You can still drag in blocks in the JavaScript editor, but they turn into text when you drop them

## Teaching Resources

Curriculum

Get started

Computer science

Design technology

Science

- Micro:bit for educators: <http://microbit.org/teach/#resources-section>
- Micro:bit doesn't make their own teaching resources, mostly, but there are plenty of lesson plans and resources built around the micro:bit
- Being used by a whole country has some advantages.

# Summary: BBC Micro:bit

Cost	Starts at 25 NZD
Lesson plans	Yes
Community	No*
Build games with code	Yes
Play games with code	No
Text-based coding available	Yes
Ages	Middle school and up

\* You can definitely find maker communities built around the micro:bit

# Summary: BBC Micro:bit

- Why use it:
  - Fun, cheap introduction to coding and electronics
  - Tangible inputs and outputs
- Why not use it:
  - Not free\*

\* Free simulator. Hardware is cheap, but not free

# But wait, there's more!



- Look at the "Built with Blockly" section <https://developers.google.com/blockly/> for more

## 5 minutes: discuss

- Have you used any of these before?
  - Are you willing to be a resource for other teachers?
- Would you use any of these in your classroom?
  - Why or why not?
- Would your students be excited about any of these?



Questions for me?



Thanks!

