

# DRAWING AND CODING WITH WHIO

## Activity Description

Introduce students to coding! Students create a whio/blue duck digital character and start to use coding programs.

## Teaching Information

The Whio Forever digital learning activities can be used as part of the Whio Forever Education Resource, see: <http://whioforever.co.nz/teachers>. This activity supports learning in **Activity 1. Meet the whio** or **Activity 3. Whio features** which are about whio/blue ducks' appearance and special features.

### Curriculum links

Levels 3-4 (Year5-8)

**Technology: Digital Technologies**

**Computational Thinking: Progress outcome 1:** In authentic contexts, students use their decomposition skills to break down simple non-computerised tasks into precise, unambiguous, step-by-step instructions (algorithmic thinking).

**Designing and Developing Digital Outcomes (DDDO): Progress outcome 1**

In authentic contexts and taking account of end-users, students participate in teacher-led activities to develop, manipulate, store, retrieve and share digital content in order to meet technological challenges. In doing so, they identify digital devices and their purposes and understand that humans make them. They know how to use some applications, they can identify the inputs and outputs of a system, and they understand that digital devices store content, which can be retrieved later.

**English:** Speaking, writing, presenting

**Key competencies:** Thinking, Managing self.

### Learning intention

Students are learning to create a character or sprite using a digital coding application/platform based on their knowledge of the physical features of a whio/blue duck.

### Success criteria

Students can follow instructions to draw a character or sprite that has the features of a whio/blue duck.

### Vocabulary

sprite, character, actor, coding block, event, motion, stage, costumes.

**New to learning about digital technology or have limited devices?**

Start with the [Create a whio/blue duck algorithm unplugged activity](#)

### Context for learning

The whio (Blue Duck) is endemic to New Zealand, and one of only four duck species in the world that lives in fast flowing water. Whio are also indicators of a river's health. If they are present the river or stream is considered to be in a very healthy state.

The Whio Forever project is funded through a partnership between the Department of Conservation and Genesis. This partnership is currently in its second five-year term, with Genesis funding \$400,000 per year to support the work done through the Whio Forever project.

For more information on the Whio Forever project see [www.whioforever.co.nz](http://www.whioforever.co.nz).

## Learning experience

**Sprite: A character or visual representation of an object in a computer game, simulation or application. In Tynker a sprite is also known as an 'actor'**

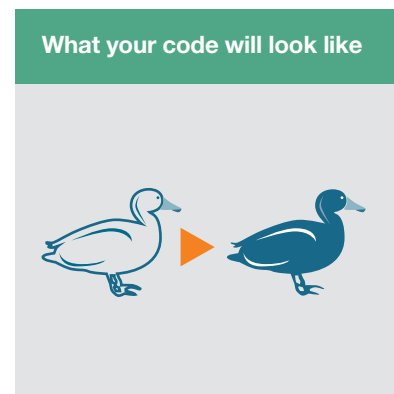
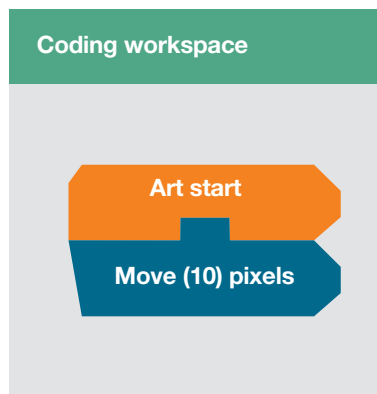
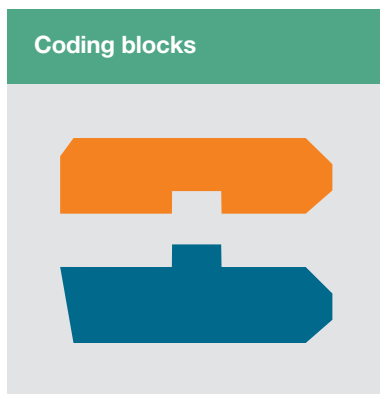
### Introduction

- Explain that students are going to create a who/blue duck character (or sprite) in a coding application of their choice.
- Discuss why we need characters for coding with digital devices.
- Review what a who/blue duck looks like using activities and resources from the Whio Forever education resources, see: <https://whioforever.co.nz>

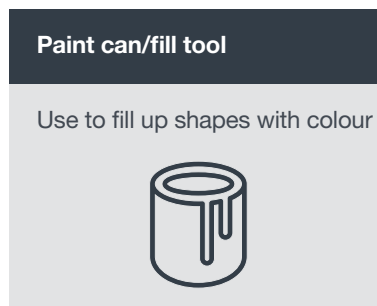
- Examples of how you can create a character in Tynker and Scratch (free online coding applications) are given below. Students can alternatively use other programmes and/or apps if they are more familiar with them.

### Background knowledge that may help with using coding programmes or applications

In most coding applications or platforms, the code is written on the left of the screen and on the right is where you can see the result of your coding.



### Useful tool icons to recognise:



## Drawing your actor

Make sure your who/blue duck is blue/grey and has a white bill. Who also have brown spots on their chests, pale webbed feet and black rubbery lips.

### Using Tynker to create a new who actor

#### Setting up your coding space

- Go to <https://tynker.com>
- 1. Select 'Play' in the top menu bar then 'create' from the menu on the left.
- 2. Select 'drawing' then 'blank project.'
- 3. Click on 'add actor' and select 'draw your own'. In Tynker sprites/characters are called actors.

Note: See Tynker tutorials [here](#)

#### Drawing your actor or character

4. Select the pencil or brush (1st or 2nd options) from left of screen. Then click on the cursor and hold the left mouse key down in the drawing space to draw. Change the colour of the pen by clicking on the colour you want at the top of the screen.
5. Draw the shape of a who/blue duck. If you make a mistake use the eraser tool (third tool down) on the left.

#### Colouring in your actor

6. Choose the fill tool (4th down) bucket on the left. Select your colour and then click within the area you want to make that colour. The shape you fill must be closed and have lines all around it.
7. Keep drawing until you're happy that your character has enough features of a who/blue duck. Then hit 'save' in top left of screen. Your character will then become a sprite/actor.

### Using Scratch to create a new sprite

#### Setting up your coding soace

1. OpenScratch:  
<https://beta.scratch.mit.edu> (there are also video tutorials here to help you).
2. Choose 'start creating' or choose 'start' in top menu bar.
3. Choose to add a new sprite (click on the orange cat icon with a plus sign next to it, on the right-hand side of the

#### Drawing your who sprite

- screen).
4. Click on the brush icon when you hover over the cat to paint your sprite. On the left of the screen a drawing canvas will come up (this is the costume tab). Click on your new sprite, e.g. Sprite 2 to edit that sprite.
5. Select the pencil or brush (1st or 2nd options) from left of screen. Then click on the cursor and hold the left mouse key down in the drawing space to draw. Change the colour of the pen by clicking on the fill colour you want at the top of the canvas.
6. Draw the shape of a who/blue duck with the brush tool. If you make a mistake use the eraser tool (usually 7th tool down) on the left. Or hit the undo button above the canvas.

#### Colouring in your actor

7. Choose the fill tool bucket. Select your colour and then click within the area you want to make that colour. The shape you fill must be closed and have lines all around it.
8. Keep drawing until you're happy that your character has enough features of a who/blue duck. Then hit 'save'. Your character will then become a sprite/actor.
9. To delete unwanted sprites, click on them and then click on the x in the top right corner of the sprite.

## Editing an existing duck sprite

- For an easier edit of a sprite, select the duck sprite and then edit it using the costume tab at the top of the screen. You can then change the colours and outline to make it look more like a whoio/blue duck.
- Or start with this Scratch project where you can edit our sprite outline to look like a whoio: <https://scratch.mit.edu/projects/301764490/>. Click 'see inside' to edit the Whoio sprite you can also upload a picture of a duck and edit that.

## Other options for drawing

If you do not have access to the internet you can also try to draw a character in a digital drawing program such as Paint or Google Drawings.

## Start coding(Tynker or Scratch): make your duck sprite move

1. Go to the code tab at the top of your screen.
2. Drag your duck sprite to where you want it to start in your workspace. (See page2 and explanation of coding workspace)
3. Click on the sprite you want to code for.  
Go to the events tab on the left top of your screen. An event is when your code will start. Events are usually yellow or orange blocks. Choose an event block such as 'when this sprite is clicked'. Drag this into your middle coding window.  
Choose a motion or sound block to add to your 'when this sprite is clicked' tab. Drag this block under the event tab so that they fit together.
4. Add another motion or movement tab to the code.
5. Click play to try out your code. Or you may need to click the green flag or your sprite (it depends which event started your code). The result of your coding will show on the right-hand side of your screen.
6. Where did your sprite move to? What did it do? Is this what you intended? If the coding doesn't do what you wanted it to then there is a 'bug' in your code. Edit your code (drag blocks in and out of the coding window) until your sprite does what you want it to do. This is called debugging.
7. Well done! You're coding!

## Extra for experts:

- Try another sequence for another sprite/actor.
- Recreate the movement and instructions/algorithm from the [Create a whoio/blue duck algorithm unplugged activity](#) in your coding programme.