Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

Subject: Purple Mash unit 5.1 and 6.1 Coding

Key Concept/ Theme: Coding

Prior Learning links:

	Cycle 1	Cycle 2
Year 1/2	Unit 1.1 Exploring Purple mash • General use of Purple Mash • Design: avatar creation • Paint Projects: use of the simple paint tools Unit 2.6 Creating Pictures • 2Paint a Picture: art effects, collage effects	Unit 1.1 Exploring Purple mash • General use of Purple Mash • Design: avatar creation • Paint Projects: use of the simple paint tools Unit 1.6 Animated Story Books • 2Create a Story: Painting tool. • Animating images using built in effects • Concept of background (static) and foreground (can move)
Year 3/4		Unit 4.6 Animation • Create a stop motion animation using 2Animate • Use of art tools to create backgrounds and effects
Year 5/6	Unit 5.5 Game creator • Themed art • Art in 3D • Animating 3D characters	

Vocabulary:

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

Key Vocabulary

Abstraction
A way of de-cluttering and removing unnecessary details to get a program functioning.

Action
The way that objects change when programmed to do so. For example, move or change a property.

Algorithm
A precise step by step set of instructions used to solve a problem or achieve an objective.

Concatenation
The action of linking a mixture of strings, variable values and numbers together in a series.

Debug\ Debugging Fixing code that has errors so that the code will run the way it was designed.

Decomposition
A method of breaking down a task into manageable components. This makes coding easier as the components can then be coded separately and then brought back together in the program.

Efficient
In coding, simplified code runs faster and uses less processing memory, it is said to be more efficient.

Flowchart
A diagram that uses specifically shaped, labelled boxes and arrows to represent an algorithm as a diagram.

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

Event
An occurrence that causes a block of code to be run. The event could be the result of user action such as the user pressing a key (**when Key**) or clicking or swiping the screen (**when Clicked, when Swiped**) or when objects interact (collision). In 2Code, the event commands are used to create blocks of code that are run when events happen.

Nesting
When coding commands are put inside other commands. These commands only run when the outer command runs.

Physical System
In this context, this is any object or situation that can be analysed and modelled. For example modelling the function of a traffic light,

Function
A block or sequence of code that you can access when you need it, so you don't have to rewrite the code repeatedly. Instead, you simply '**call**' the function each time you want it.

Object
Items in a program that can be given instructions to move or change in some way (action). In 2Code Gorilla, the **object type**s are button number, input, text, shape turtle, character, object, vehicle, animal.

Properties
These determine the look and size of an object. Each object has properties such as the image, scale and position of the object.

Input
Information going into the computer. This could be the user moving or clicking the mouse, or the user entering characters on the keyboard. On tablets there are other forms such as finger swipes, touch gestures and tilting the device.

Output
Information that comes out of the computer e.g. **sound**, **prompt**, **alert** or **print to screen**.

Repeat
This command can be used to make a block of commands run a set number of times, until a condition is met or forever.

Sequence
This is when a computer program runs commands in order.

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

modelling friction of cars moving down surfaces or modelling the functions of a home's security system.

Selection

A conditional decision command. When selection is used, a program will choose which bit of code to run depending on a condition. In 2Code selection is accomplished using '**if**' or '**if/else**' statements.

Simplify

In coding this is used to describe modifying the code to complete the same process with less lines of code.

Timer

Use this command to run a block of commands after a timed delay or at regular intervals.

Variable

A named area in computer memory. A variable has a **name** and a **value**. The program can change this variable value. Variables are used in programming to keep track of things that can change while a program is running. In 2Code, variables can be **strings**, **numbers** or **computer-generated** variables to control objects of a type.

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

Key Images

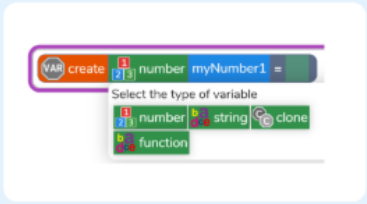
Open design mode in 2Code.

Switch to code mode in 2Code.

Add a new Tab to your code

A change variable block.

Example of combining variables and strings to print to the screen



Creating a variable in 2Code

Creating a function in 2Code

Calling a function in 2Code

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

Key Vocabulary

Action
The way that objects change when programmed to do so. For example, move or change a property.

Algorithm
A precise step by step set of instructions used to solve a problem or achieve an objective.

Co-ordinates
Numbers which determine the position of a point, shape or object in a particular space.

Event
An occurrence that causes a block of code to be run. The event could be the result of user action such as the user pressing a key (**when Key)** or clicking or swiping the screen (**when Clicked**, **when Swiped**) or when objects interact (**collision**). In 2Code, the event commands are used to create blocks of code that are run when events happen.

Execute\Run
Clicking the Play button to make the computer respond to the code. Execute is the technical word for when you run the code. We say, 'the program (or code) executes.'

Command
A single instruction in a computer program.

Decomposition
A method of breaking down a task into manageable components. This makes coding easier as the components can then be coded separately and then brought back together in the program.

Debug/Debugging
Fixing code that has errors so that the code will run the way it was designed to.

Flowchart
A diagram that uses specifically shaped, labelled boxes and arrows to represent an algorithm as a diagram.

Need more support? Contact us:

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

Function
A block or sequence of code that you can access when you need it, so you don't have to rewrite the code repeatedly. Instead, you simply **call** the function each time you want it.

Object
Items in a program that can be given instructions to move or change in some way (action). In 2Code Gorilla, the **object types** are button number, input, text, shape turtle, character, object, vehicle, animal.

Procedure
An independent code module that fulfils a task and is referenced within a larger body of code. In 2Code a procedure might be coded as a function.

Input
Information going into the computer. This could be the user moving or clicking the mouse, or the user entering characters on the keyboard. On tablets there are other forms such as finger swipes, touch gestures and tilting the device. In 2Code the commands **prompt for input** and **get input** are used to prompt the user to enter typed input and then use this input.

Properties
These determine the look and size of an object. Each object has properties such as the image, scale and position of the object.

Sequence
This is when a computer program runs commands in order.

Launch Command
This command will open another Purple Mash file or an external website that you specify when it is called.

Output
Information that comes out of the computer e.g. **sound**. **prompt**, **alert** or **print to screen**.

Predict
Use your understanding of a situation to say what will happen in the future or will be a consequence of something

Repeat
This command can be used to make a block of commands run a set number of times or forever.

Repeat Until

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

order.

Selection
Selection is a decision command. When selection is used, a program will choose which bit of code to run depending on a condition. In 2Code selection is accomplished using '**if**' or '**if/else**' statements.

Tab
In 2Code, this is a way to organise a program into separate pages (tabs) of code.

Simulation
A model that represents a real or imaginary situation. Simulations can be used to explore options and to test predictions.

Timer
Use this command to run a block of commands after a timed delay or at regular intervals.

Repeat Until
In 2Code this command will repeat a block of commands until a condition is met.

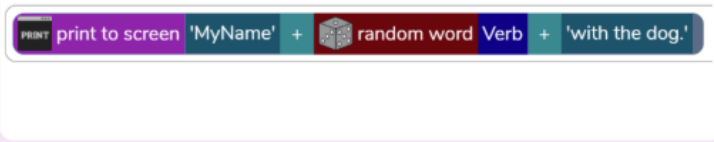
Variable
A named area in computer memory. A variable has a **name** and a **value**. The program can change this variable value. Variables are used in programming to keep track of things that can change while a program is running.

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

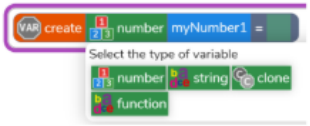
Open design mode in 2Code.

Switch to code mode in 2Code.

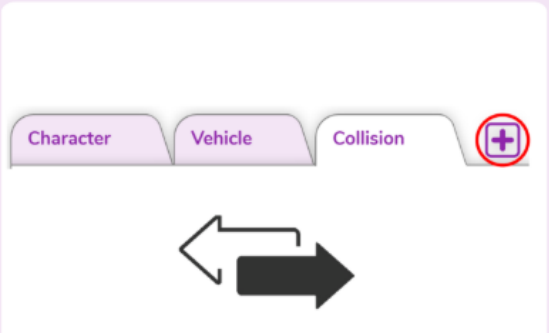
A change variable block.

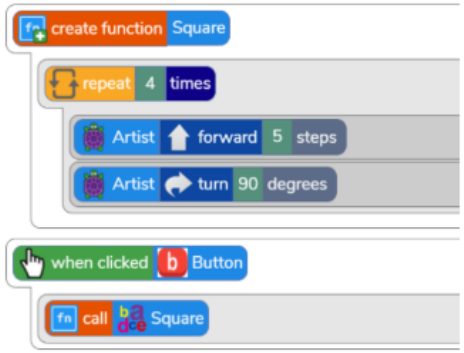


Example of combining variables and strings to print to the screen



Creating a variable in 2Code





Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

Key Images:

Resources needed for each lesson – 2dos to set.

Lesson 1:

Unless otherwise stated, all resources can be found on the main unit 5.1 page. From here, click on the icon to set a resource as a 2Do for your class. Use the links below to preview the resources; right-click on the link and 'open in new tab' so you don't lose this page. • Y5 Coding Vocabulary Quiz. • Animal Race 1 • Animal Race 2 • Catching Game. • Free Code Gorilla. This is on the main 2Code Page (scroll down to the Gorilla activities). • Optional) Vocabulary flash cards. The teacher flash cards have been created so you can print them on A4 paper, cut them to size, fold them in half and glue them together. You can display and use these throughout coding lessons to support use of vocabulary. Preparation • Set Catching Game as a 2Do. You can select the following objectives when setting the 2Dos to make future assessment easier

Lesson 2

Unless otherwise stated, all resources can be found on the main unit 5.1 page. From here, click on the icon to set a resource as a 2Do for your class. Use the links below to preview the resources; right-click on the link and 'open in new tab' so you don't lose this page. • Traffic Light Algorithm vocabulary. • Traffic Light Algorithm writing frame. • Traffic Light Sequence Flowchart. • Free Code Gorilla. This is on the main 2Code Page (scroll down to the Gorilla activities). • 2Chart Preparation • Open the video in browser tab. • Print Traffic Light Algorithm Vocabulary and put it on display. • Set the Traffic Light Algorithm writing frame OR Traffic Light Sequence Flowchart as a 2Do. You can select the following computing objectives when setting the 2Dos to make future assessment easier: • Set Free Code Gorilla as a 2D

Lesson 3

Unless otherwise stated, all resources can be found on the main unit 5.1 page. From here, click on the icon to set a resource as a 2Do for your class. Use the links below to preview the resources; rightclick on the link and 'open in new tab' so you don't lose this page. • Football Game. This is on the main 2Code Page (scroll down to the Gorilla activities). • Friction Example. • Physical Football (Optional) Preparation • Set Football Game as a 2Do. You can select the following objectives when setting the 2Dos to make future assessment easier: • Create a display board for the class to share their programs to. Details of how to do this are given in Appendix 1. • Physical Football (Optional

Lesson 4

Unless otherwise stated, all resources can be found on the main unit 5.1 page. From here, click on the icon to set a resource as a 2Do for your class. Use the links below to preview the resources; rightclick on the link and 'open in new tab' so you don't lose this page. • Alien Blast Game. • Alien Blast Scene. Preparation • Set Alien Blast Scene as a 2Do. You can select the following objectives when setting the 2Dos to make future assessment easier: • Create a display board for the class to share their programs to. Details of how to do this are given in Appendix 1

Lesson 5

Unless otherwise stated, all resources can be found on the main unit 5.1 page. From here, click on the icon to set a resource as a 2Do for your class. Use the links below to preview the resources; rightclick on the link and 'open in new tab' so you don't lose this page. • Free Code Gorilla. This is on the main 2Code Page (scroll down to the Gorilla activities). • Concatenation. This is on the main 2Code Page (scroll down to the Coding Principles activities). • Code Snippet Example 1. • Code Snippet Example

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

2. Preparation • Set Concatenation as a 2Do. You can select the following objectives when setting the 2Dos to make future assessment easier: • Set Free Code Gorilla as a 2Do Lesson 6 Unless otherwise stated, all resources can be found on the main unit 6.1 page. From here, click on the icon to set a resource as a 2do for your class. Use the links below to preview the resources; right-click on the link and 'open in new tab' so you don't lose this page. • Alien cards. • Aliens Database. • Guess the Alien. • Is It Raining and Reginald Rocket Examples from Y4. • User Input Example 1. • User Input Example 2. • Free Code Gorilla found on the main 2Code screen in the Gorilla section. Preparation • Set Guess the Alien as a 2Do. Add in Aliens Database and Alien Cards as supporting documents – you will need to save these documents to your work folder before you set the 2Do. You can select the following objectives when setting the 2Dos to make future assessment easier: • Print/ copy Alien cards enough for one each for the children	
1. Unit Lesson 1	**LO:** To review existing coding knowledge. • To begin to simplify code. • To create a playable game. **Activity:**

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

Introduction	Display **slide 2** and outline the lesson aims. Display **slide 3** and outline the success criteria.
Vocabulary	Display **slide 4**. Use the Y5 Coding Vocabulary Quiz as a class to help refresh coding knowledge from previous years. It is set up so that you attempt all questions and then click the  button to check the answers. Click 'OK' to see which are correct and incorrect: Run through the answers to the questions together. You could use the vocabulary cards to find the answers and display in the classroom or use slide 5 which has definitions.
	Slide 5 can be used to review previous vocabulary. The use of this vocabulary is recapped during the lesson. The vocabulary is repeated at the end of the lesson where it can be used to review new vocabulary.
Activity 1: Animal Race	Display **slide 6.** Ask the children to look at the design and read the code, can they **predict** what will happen when the program is **run**? Use the slide to open Animal Race 1, click on play to **run** the program and click on the animals to see if their predictions were correct. Recap **event** – **object** – **action**, identifying each in this code.
	Display **slide 7**. The design in this program is the same, but the code is different. Can children **predict** what will happen when this program is **run**? Use the slide to open Animal Race 2, click on play to **run** the program and click on the animals to see if their predictions were correct. Explain to children that in this lesson they will revise some of the vocabulary
	and concepts they have learnt in Year 5, and start being able to **simplify** code to make their programming more **efficient**. Discuss what it might mean to make things more efficient. Return to **slide 6** and begin to look at how this code works.

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

	Display **slides 8-9**. Click through the slides and use them to help you explain how the code in Animal Race 2 works. In year 5 children will use Free Code Gorilla. At the end of **slide 9** you could open Free Code Gorilla at this point and use it to recap children's understanding of different object types.
Catching Game	Use **slide 10**. Display Catching Game on the board. The following slides go through and explain points about this activity. Complete it on the board together or get children to open it as well and complete it with you one step at a time using the following slides. Refer to key vocabulary and concepts as you go through it with the children including **selection**, **IF/ELSE Statements**, **prompt**, **number variable**, **variable**, **timer** (**after/ every**), **event**, **object**, **action**, **co-ordinates**.
Catching Game: Stage 1	**Slide 11: Catching Game Stage 1** Start by clicking on 'Design' and looking at the **scene** together, there is a score**,** a catcher and some food **objects**. Click on the food **objects** and set a speed in the Attribute Value box for each one– around 1-3 is sensible!
Catching Game: Stages 2 and 3	**Slide 12: Catching Game Stages 2 and 3** Stage 2: Create arrow key press events so the player can control the catcher. Stage 3: Add a collision detection event so that when the catcher collides with a weight, a sound plays and the game starts again. Explain that because there are 2 objects with the same tag (avoidables), 2Code gives you the option of using 'Any avoidable' to save you having to program a collision detection for each one. It has automatically generated a variable for this object type.

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

			NB weight is a custom object type that is not usually available in 2Code but was created for this example only. This puts into practise simplifying the code. When you test this code children might notice that the sound doesn't play because the program restarts straight away (re-run and watch when the code highlights orange to notice this) – discuss with them how that might be overcome later when we are able to fix it.	

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

Catching Game: Stages 4 and 5	**Slide 13: Catching Game Stages 4 and 5** Stage 4: Write code that increases the score by a value of 1 and plays a sound when the catcher collides with the food. Step 5: Write code so that the food hides when the catcher collides with it. The individual food object that needs to hide is the value of the 2Code generated variable, 'Any catchable'. 'Any catchable' is a variable generated by 2Code which could contain any of the food items – whichever is clicked on. This variable value will be set by the collision detection event, so they will need to use 'change variable' for this. Once they drag 'change variable' into the collision detection event they can select 'Any catchable'. Stage 6: Ask the children for ideas for how the game can be improved and together fix the problem of the sound not playing when the catcher bumps into a weight – add in a timer so the game restarts after a second.
Activity: Improve Catching Game	**EITHER (Slide 14)** Ask children to open Purple Mash and work through the Catching Game 2Do. They should work through all the steps you've been through and then try and improve the game in the final stage – challenge them to include x and y **co-ordinates** and an **IF statement** in their code. Challenge them to fix the problem where the sound doesn't play before the game restarts when the catcher collides with a weight. Remind children they can click on the instruction to bring the video back up. **OR (Slide 15)** Ask children to use Free Code Gorilla to have a go at creating their own game. OR both!
How did you get on?	Display **slide 16**. Share children's work 2Displayboard (see Appendix 1) and allow them some time to play each other's games. Review their work and celebrate achievements.
Vocabulary Overview	**Slide 17** can be used to review lesson vocabulary. Click on the words to reveal the definitions.
Review Success Criteria	Display **slide 18**. Review the success criteria from **slide 3.** Children could rate how well they achieved this using a show of hands.

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

	Extension:
2. Unit Lesson 2	**LO:** To understand what a simulation is. • To program a simulation using 2Code **Activity:** Go over new vocab for lesson

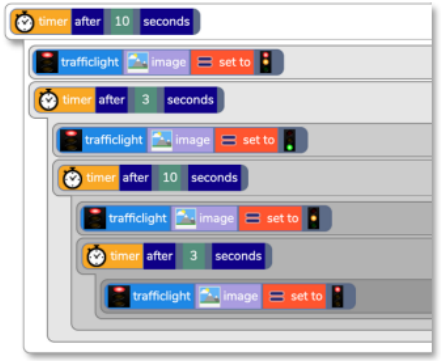
Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

Introduction	Display **slide 2** and outline the lesson aims. Display **slide 3** and outline the success criteria.
Vocabulary	**Slide 4** can be used to review previous vocabulary. The use of this vocabulary is recapped during the lesson. The vocabulary is repeated at the end of the lesson where it can be used to review new vocabulary.
	Use **slide 5** to remind the children of the word algorithm and simulation.
Simulating a Physical System	Display **slide 6**. Use the slide to explain to children they will be writing an algorithm for a program that simulates a physical system.
Traffic Light Algorithm Vocabulary	Use **slide 7** to share the vocabulary that could be included for a traffic light simulation or share the PDF.
Video of a UK Traffic Light Sequence	Use **slide 8** to watch the Traffic Light Sequence video together. Ask the children to take notes that will enable them to write the **algorithm**. The algorithm is a **sequence** of instructions – discuss why it is so important in this case that the **sequence** is correct. You may need to play the video more than once.
Planning the Algorithm	Display **slide 9**. Give the children around 10 minutes to complete the following task (choose which one you set) *EITHER* Traffic Lights Flowchart: ask them to have a go at creating the flow chart for the traffic lights. *OR* Traffic Lights Algorithm: Ask them to use the writing template to formulate the **algorithm** for the traffic lights.
	If using 2Chart show children how to change the text and add a title by double-clicking on the existing text. It should look something like these examples:

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

		<table><tr><td>Turn on red</td></tr><tr><td>Wait 10 seconds</td></tr><tr><td>Turn on amber</td></tr><tr><td>Wait 3 seconds</td></tr><tr><td>Turn off red</td></tr><tr><td>Turn off amber</td></tr><tr><td>Turn on green</td></tr><tr><td>Wait 10 seconds</td></tr><tr><td>Turn on amber</td></tr><tr><td>Turn off green</td></tr><tr><td>Wait 3 seconds</td></tr><tr><td>Turn off amber</td></tr><tr><td>Repeat all forever</td></tr></table> 
	Activity 1: Setting up the Scene	Display **slide 10**. Review how children got on and then reveal the slide to them with the first task of setting the scene. Children to open their 2Dos – Free Code Gorilla.	
	Activity 2: Making the Lights Change	Display **slide 11**. Children to look at planned algorithm to help them create code that make the lights changes. They will need to ensure that the **timers** are all **nested** (inside each other). Here is an example using the algorithm:	

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

	
Making the Sequence Repeat Forever	Reveal **slide 12**. Use the slide to get children thinking about how we get the sequence to repeat forever.
Activity 3: Making the Sequence Repeat Forever	Display **slide 13.** Children to have a go at making the sequence last forever.

Activity 4: Adapting your Traffic Light Sequence	Display **slide 14**. Share some of the simulations so far made with the class. Children to have a go at adapting their code for the scenarios on the slide.
Activity 5: Extension	Display **slide 15**. Children try to simulate a pedestrian crossing.
Vocabulary Overview	**Slide 16** can be used to review lesson vocabulary. Click on the words to reveal the definitions.
Review Success Criteria	Review the success criteria from **slide 3.** Children could rate how well they achieved this using a show of hands.

Go over vocab and success criteria.

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

3.

Unit

Lesson 3

LO: • To understand how to use friction in code. • To begin to understand what a function is and how functions work in code

Activity:

Introduction	Display **slide 2** and outline the lesson aims. Display **slide 3** and outline the success criteria.
Vocabulary	**Slide 4** can be used to review previous vocabulary. The use of this vocabulary is recapped during the lesson. The vocabulary is repeated at the end of the lesson where it can be used to review new vocabulary.
Football	Remind children of the lesson where they made the **simulation** of a traffic light **sequence** – in that lesson they made a program that simulated a **physical system**. Explain that often when coding we want to **simulate physical systems** – program objects to behave in a realistic way. Gently kick a football across a space in front of the class. Watch it roll, then come to a stop. Ask children what would happen if you'd kicked it harder? And what if you were on a wooden floor and not carpet? Display **slide 5**. Explore this slide together as a class. Discuss the code and questions.
Football Friction	Display **slide 6**. Open the Friction Example. Go through the slide together as class. You might choose to let the children explore the Friction Example in small groups and feed back to the class.
	Display **slide 7**. Discuss with the children the code on the slide. Explore how friction has been set for each surface.
Football Game	Display **slide 8**. Children to open Football Game from their 2Dos. Explain the stages and what they are simulating. Display **slides 9 to 11** respectively working from stages 1 to 4 as a class, modelling each of the stages (1 to 4). • **Stage 1:** Choose 'any' so the ball can be swiped in any direction. A

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

	speed of 1-4 is sensible. • **Stage 2:** Add **friction** to the ball, friction of between 1-4 is about right, test and change it and see what children want to set. • **Stage 3:** Involves altering existing code so click on what you want to change – the speed, click on the value for the current speed and select swipe speed so the speed of the ball will match the speed of the swipe. Drag in football and set the angle in the same way. • **Stage 4:** Introduces children to **functions**.
	• Watch the video together. • At the start of the program the ball is at X=3, Y=8 and the speed = 0. • Creating this **function** will apply those properties to the ball whenever it is used (**called**). • Use 'create function' to create the function – a **function** is a type of **variable.** This is a difficult concept to comprehend until you are creating more complex code (e.g. Java or Python coding) when the reasoning becomes clearer. • And then use 'call function' to add your **function** into your collision detection **event.**
Activity 1: Football Game	Display **slide 12**. Use this to set the children off on completing the Football Game from their 2Dos from start to finish. Children to carry on until they have made the football game. Review children's progress and ask them how many times they called their **function**. The **function** contained 3 pieces of code: - The X Co-ordinate - The Y Co-ordinate - The speed Notice together how calling the **function** saved them coding all this each time. **Functions** also help us **simplify** code and make our programming more efficient.

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

	How did you get on?	Display **slide 13**. Share children's work on a Displayboard (see appendix 1) and give them some time to play each other's games.
	Vocabulary Overview	**Slide 14** can be used to review lesson vocabulary. Click on the words to reveal the definitions.
	Review Success Criteria	Review the success criteria from **slide 3.** Children could rate how well they achieved this using a show of hands.
	Extension:	
Unit Lesson 4	**LO:** • To understand what the different variable types are and how they are used differently. • To understand how to create a string. **Activity:**	

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

	Introduction	Display **slide 2** and outline the lesson aims. Display **slide 3** and outline the success criteria.	
	Vocabulary	**Slide 4** can be used to review previous vocabulary. The use of this vocabulary is recapped during the lesson. The vocabulary is repeated at the end of the lesson where it can be used to review new vocabulary.	
		Display **slide 5**. Go through the definition of variables on the slide and use it to remind the children of where they have encountered variables before.	
	Variables	Display **slide 6.** Use the slide to help children understand that you can select either: number, string or function.	
		Display **slides 7-8.** These demonstrate the importance of setting an initial value for a variable. Click to reveal examples and learning points.	
	Alien Blast	Display **slide 9**. Introduce strings by opening 'Alien Blast Game'. Click on 'Design' and explain that in this game the rocket blasts the aliens when it collides with them.	
		Display **slide 10**. Click on 'See Code' and explain to children that you are going to look through the code tabs with them to see how it works. Start by looking at the 'Instruction Tab'. **Instruction Tab** The code in here will trigger an alert at the start to tell the player how to control the rocket.	
		Display **slide 11**. Use the slide to review the 'Progress Tab' **Progress Tab** The progress tab contains code to keep the player informed of their progress. There is an initial message that will print to the screen at the start. A string is sequence of characters, which could form words, phrases or even	

	whole sentences. There is a 'describe' string initialized to the value 'x'. The value will be changed when the program executes to contain a word. Where will the value for this variable come from? What will it be? Explain that variables that are strings can also be called text variables.
	Display **slide 12**. Use the slide to continue to review the 'Progress Tab'.
	Read down to see that the value of 'describe' is set to a different **random** adjective every quarter second. These adjectives will be used to describe the aliens as they are blasted. • Click on the **Run** button to **run** the code. • Click on OK on the alert, then look for the 'Variable Watch' box. • Look for 'describe'. • Watch it change to a different **random** adjective every quarter second.
	Display **slide 13**. Click on the **stop** button and return to looking at the code: **Collision** The code in here programs what happens each time an alien is blasted. Where does it use the 'describe' string? What will this look like in the game?
	Display **slide 14**. Review the 'Rocket Tab'. **Rocket** The code in here will enable the player to control the rocket. It uses speed and angles. **NB: If children are using tablets, the when key event is not available. Instead, they could try using when swiped events.**  Play the game together as a class and explain that they will be making their own version of this game.

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

<table>
<tr><td></td><td>
<table>
<tr><td>Activity 1: Alien Blast</td><td>Display slide 15. Ask children to open Alien Blast Scene (a starting point for the program you've just looked at – with a rocket, and alien and no code) from their 2Dos.
Challenge children to use the 'clone' button in Design view to help create their scene, they should change and rename the objects, they might want to change the rocket.
Check code for how the rocket moves – you might need to recap how angles are used.
They must create and use a text variable and should also be encouraged to include a timer and a score.</td></tr>
<tr><td>How did you get on?</td><td>Display slide 16. Use this as an opportunity to share children's games to a display board (see appendix 1). Children could critique each other's games.</td></tr>
<tr><td>Vocabulary Overview</td><td>Slide 17 can be used to review lesson vocabulary. Click on the words to reveal the definitions.</td></tr>
<tr><td>Review Success Criteria</td><td>Review the success criteria from slide 3. Children could rate how well they achieved this using a show of hands.</td></tr>
</table>
</td></tr>
<tr><td>Lesson 5</td><td>L.O

To begin to explore text variables when coding. • To understand what concatenation is and how it works.

Activities</td></tr>
</table>

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

Introduction	Display **slide 2** and outline the lesson aims. Display **slide 3** and outline the success criteria.
Vocabulary	**Slide 4** can be used to review previous vocabulary. The use of this vocabulary is recapped during the lesson. The vocabulary is repeated at the end of the lesson where it can be used to review new vocabulary.
	Display **slide 5**. Use the slide to introduce the word 'Concatenation' and go through the discussion points. Can they think of any coding they have done where they may have used concatenation? They may say they have added a score into a 'Well done' message at the end of a game.
Activity 1: Concatenation	Display **slide 6**. Tell them you have set a 2Code concatenation task as a 2Do for them in Purple Mash. Ask them to work through it and see how they get on. Before setting them off it you may wish to recap nouns, verbs and adjectives and ask them for some examples of a phrase that includes all three.
Code Snippet Example 1	Display **slide 7**. Display the Code Snippet Example. But before running the code, can the children suggest what the code will do? Click to reveal the explanation, talk through it slowly with the children: • There is a myName string variable initialized to the value, 'Archie'. • There is a myAnimal string variable initialized to the string 'animal' – where does the value for this variable get changed? – Ask children to read down and see if they notice that it is set to randomly generate a value every second.
	Run it to show how the value of myAnimal is set to a different random animal every second.
Code Snippet Example 2	Display **slide 8**. Open Code Snippet Example 2 and see if children can read the code and tell you what would happen if you clicked run now.

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

		Run it and notice the errors, see if they can help you debug.  **(Solution)** To add the space, you need to click on the '+' where you want to add it and then type a space into the box that comes up. Things to note and demonstrate with the children: Use of the Random function – what does random mean? What does the + sign do to text? What would 'Hello + World' produce? (For strings (text) the + will concatenate the pieces of text.) The way 2Code can select a random animal, noun, verb or adjective in order to build sentences. The importance of spaces (the spaces were not correct when you opened it!)	
	Text Variables	Display **slide 9**. Start a new 2Code Gorilla document and set a string **variable** to '1' (the quote marks are required) and + 2 to it. Can children guess what this will make?	
	Number Variables	Display **slide 10**. Start a new 2Code Gorilla document and set a number variable to 1 (no quote marks this time) and + 2 to it. Can children guess what this will make?	

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

Number Variables	Display **slide 10**. Start a new 2Code Gorilla document and set a number variable to 1 (no quote marks this time) and + 2 to it. Can children guess what this will make?
Explore Text and Number Variables	Display **slide 11**. Give children some time using Free Code Gorilla to explore text and number variables. What ideas can they come up with to make funny messages?
Vocabulary Overview	**Slide 12** can be used to review lesson vocabulary. Click on the words to reveal the definitions.
Review Success Criteria	Review the success criteria from **slide 3.** Children could rate how well they achieved this using a show of hands.

Lesson 6

L.O

To understand the different options of generating user input in 2Code. • To understand how user input can be used in a program.

Activities

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

Introduction	Display **slide 2** and outline the lesson aims. Display **slide 3** and outline the success criteria.
Vocabulary	**Slide 4** can be used to review previous vocabulary. The use of this vocabulary is recapped during the lesson. The vocabulary is repeated at the end of the lesson where it can be used to review new vocabulary.
	Display **slide 5**. Reveal the content of the slide and go through 'User Input'. Open Reginald Rocket and Is it Raining in separate tabs. Look at the code, predict and then run both.
Example Code	Display **slide 6**. Reveal the slide's contents to look at the example code and discuss concatenation. Open User Input Example Code 1 and with the children, add the missing full stop. Run the code.
	Display **slide 7**. Show children User Input Example 2. Discuss setting the input to a variable (number or string) and then manipulating it. What happens if you give a word answer for this last example?
Activity 1: Using Input	Display **slide 8**. Ask the children to Start Free Code Gorilla and explore using the **get input** command and the **prompt for input** command. Challenge them to have a go at adding code so that the program uses the **input** in a **string** when it is collected.
Alien Game	Use **slide 9**. Explain to children that in this lesson they will be making an interactive game using **input** from the user. Display the Alien cards or slide 8 on the board and find out about the aliens.

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

		Ask the children to come up with questions to identify them, you could ask them to do this with talking partners.
	Guess the Alien	Display **slide 10**. Open Guess the Alien Game on the board and read the code in the first **tab** – the code for the 'Introduction'. Ask children to explain how they think this game starts – they could discuss it and then feedback. Click on Run to run the game, choose the alien called Zinky (at the moment Zinky will work but none of the others have been

		programmed yet). Work through the questions until the program guesses that the alien is Zinky.
		Display **slide 11**. Look at the code in the second tab – the code for 'Red Aliens' and discuss how it works with the children. What do they think the next steps to completing the program are? Add code for the red alien with 2 eyes. Add code for the other colours of aliens; after colour, the questions do not have to be about number of eyes, but they do have to be about appearance.
	Activity 2: Guess the Alien	Display **slide 12**. Ask the children to open Guess the Alien from their 2Dos and see if they can add more to it. Give them a copy of the Alien Cards or have the pictures of the Aliens on the board so they have a point of reference. Remind them that they can use the Aliens Database to find out more detail about the aliens. Remind them about decomposition and using tabs.
	How did you get on?	Display **slide 13**. Share children's work 2Displayboard (see Appendix 1) and play a couple of the games together to share children's programs and celebrate achievements.
	Vocabulary Overview	**Slide 14** can be used to review lesson vocabulary. Click on the words to reveal the definitions.
	Review Success Criteria	Review the success criteria from **slide 3.**

End Points:

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

Key Questions

How can you use Tabs in 2Code Gorilla?

Tabs are used to organise you code and make it more readable. This also makes it easier to debug. Give the Tabs useful names to help with this.

What is a function in coding? Give an example that you have used in 2Code Gorilla.

A function is a block of code that you can access when you need it, so you don't have to rewrite the same block repeatedly. You call the function each time you want it. In a turtle program you could have a button that will make the turtle draw a square each time you click it. In the text adventure, there were functions for each room that were called when the user navigated to the room.

In 2Code Gorilla, how can a program receive user input?

When the user clicks on an object, when the user presses keys or swipes the screen with the mouse, the 'Get Input' and 'Prompt for input' commands.
On a touchscreen: when the screen is touched or swiped.

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT

What does simulating a physical system mean?

Creating a program where the objects behave as they would in the real world. For example, a football program that uses angles, speed and friction to simulate kicking a football. When simulating a physical system, you first must break the system down into parts that can be coded (decomposition). The different parts will come together to make the full simulation.

Describe how you would use variables to make a timer countdown and a scorepad for a game.

Timer countdown:
Create a timer variable and set it to the starting number of seconds. Add a Timer command that repeats and subtracts 1 every second. Add a text object in design view to display this number.

Score:
Create a variable to store the score, each time the user gains a point, change and display the value of the variable.

Give examples of how you could use the Launch command in 2Code.

Clicking on a button or other object in the program to opens another 2Code program or a webpage.

What do the terms decomposition and abstraction mean? Use examples to explain them.

Decomposition is breaking a task into its component parts so that each part can be coded separately.
If you were coding a game of chess, you could decompose into the moves of the different pieces and the setup of the playing space.

Evaluation:

See quiz on purple mash

Pioneer Federation
Medium term plan
Cycle 1, Term 5
ICT