

# Contents

Introduction .....	6
Scheme of Work (SoW) .....	7
Cross-Curricular Links .....	8

## Computational Thinking and Programming

### I) Understanding Algorithms and Programming On-Screen

1. Programming Inputs .....	10
2. Programming Objects and Events .....	11
3. Instructions and Buttons .....	12
4. Introduction to Scratch .....	13
5. Motion Blocks .....	14
6. Looks and Backgrounds .....	15
7. Programming On-Screen .....	16
8. Debugging and Programming with an App .....	17

## Communication, Collaboration and Creativity

### II) Researching on the Web and E-Safety

1. Mind-Mapping a Research Topic .....	19
2. Research and Investigation .....	20
3. Using Britannica Digital Learning .....	21
4. Creating a Presentation .....	22
5. Refining and Delivery .....	23
6. Staying Safe Online .....	24
7. Creating a Storyboard .....	26
8. Creating an E-Book .....	28
9. Sharing and Presenting .....	29
10. E-Safety Scenarios .....	30

## Computer Networks and Productivity

### III) Producing an E-Postcard and Representing Information Graphically

1. Bar Graphs .....	33
2. Binary Tree .....	36
3. Digital Binary Tree .....	38
4. Questionnaires .....	40
5. Creating a Database .....	42
6. Exploring Simulations .....	45
7. Population Growth .....	46
8. Simulation Game .....	47

Glossary .....	48-49
----------------	-------

Resources .....	50
-----------------	----

# Digital Literacy in the primary classroom – Year 2



## Introduction

### Introduction:

Becoming digitally literate and understanding how the technological world works is becoming increasingly relevant for children in the 21<sup>st</sup> century. In some instances it is as important as core life skills, such as reading, writing and maths.

Digital literacy is commonly defined as the collection of knowledge and skills used on digital devices such as PCs, laptops, tablets and the use of apps across computers.

In primary school, digital literacy enables learners to use technology strategically, to:

- Find and evaluate information
- Create and share content
- Connect and communicate with peers
- Achieve personal goals.

The aim of ***Digital Literacy in the primary classroom*** is to move beyond the basics of Information and Communications Technology (ICT) and introduce a broader concept of technology, supporting teachers in developing digital literacy attributes and preparing young learners to embark upon their future with new skills.

### About the series:

***Digital Literacy in the primary classroom*** is a series of 7 books – from Reception to Year 6 – written specifically to offer accessible and comprehensive material to complement the computing curriculum. The series is designed to follow a structured yet flexible approach to adapt to the requirements of the classroom, including ideas for differentiation, and is an ideal resource for both experienced teachers and non-specialists alike.

It covers the computing curriculum objectives and focuses in particular on the following areas:

- Computational Thinking and Programming
- Communication and Collaboration
- Computer Networks and Productivity.

The focus of digital literacies includes elements of digital learning, innovation, communication, collaboration, e-safety, information and data understanding.

The books are made up of units including lesson plans, lists of resources and worksheets. Each unit comprises a broad range of creative and practical ideas for making learning more hands-on, with step-by-step instructions and learning objectives. Each book also includes a glossary to facilitate the learning of general and more specialised digital literacy terms.

### About this book:

This book is for **Year 2** children and aims to:

- Develop digital literacy competencies in areas of computational thinking and programming, communication and collaboration with creativity, and the study of computer networks and productivity.
- Support teachers by providing step-by-step instructions and activity ideas based on whole class, group, paired and individual experiences.
- Encourage children to recognise the importance of staying safe online and creating a positive digital identity whilst using the internet.

# Scheme of Work (SoW)

Year Group	Computational Thinking	Communication and Collaboration	Computer Networks
	<i>Programming</i>	<i>Creativity</i>	<i>Productivity</i>
<b>Reception</b>	Understanding instructions <i>Logical thinking</i>	Introduction to a computer/tablet <i>Using drawing tools: e-safety theme</i>	Combining texts, graphics and sounds <i>Creating digital stories</i>
<b>Year 1</b>	Algorithms and logical reasoning <i>Programming toys</i>	Digital information around us <i>Illustrating e-safety posters</i>	Creating an electronic postcard <i>Representing information graphically</i>
<b>Year 2</b>	Understanding algorithms <i>Programming on-screen</i>	Researching on the web <i>Illustrating an e-safety e-book</i>	Questions and answers <i>Exploring simulations</i>
<b>Year 3</b>	Sequence and animation <i>Programming on-screen</i>	Using email <i>Illustrating an e-safety comic strip</i>	Survey and analysis <i>Introduction to databases</i>
<b>Year 4</b>	Variables, repetition and loops <i>Programming on-screen</i>	Producing a wiki <i>E-safety research and presentation</i>	Databases, charts and graphs <i>Introduction to spreadsheets</i>
<b>Year 5</b>	Direction and simulation <i>Programming interactive games</i>	Creating a blog <i>Creating an e-safety website</i>	Electrical circuits <i>Spreadsheet modelling</i>
<b>Year 6</b>	Complex variables and properties <i>Visual programming</i>	Collaborative research on e-safety topic <i>Presentation techniques</i>	Enterprise and Development <i>Revision of Digital Skills</i>

# Lesson 1: Programming Inputs

## Setting the scene: individual activity

### Learning objectives

- To program an object to move across the screen
- To input events and create more advanced programs

### Resources

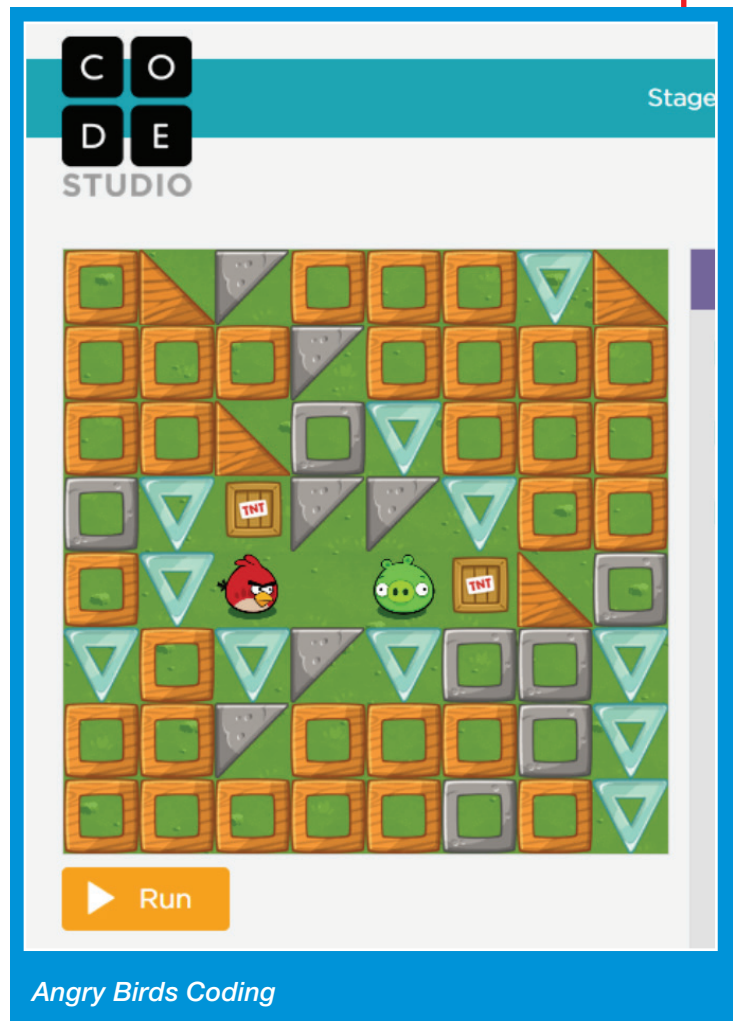
- Angry Birds Online: <http://studio.code.org/s/course2>
- Espresso Coding: <http://www.espressocoding.co.uk/espresso/coding/index.html>
- Optional: Headphones

### Small group activity

This lesson will focus on teaching children the fundamental principles of programming. For the first three lessons, the teacher will introduce the children to Angry Birds Online. Encourage the children with Stage 3: Maze Sequence. Tell the children to start with step 1 and progress. The teacher will have to demonstrate the first few lessons so that children understand what is required of them. Show the children that when they have applied the correct algorithm, they can also 'view code'. The code they see is how it is actually written with HTML computer programming. HTML stands for Hyper Text Markup Language and is a standard language for creating web pages.

Alternatively, if the school has access to Espresso Coding, begin with the Year 2 Starter Unit. In Espresso Coding, there are three lessons in total in which there are a series of steps to follow. If necessary, you can view the video tutorials. It will soon become obvious that the children will know what to do and will carry on step by step. It is crucial that children do not skip lessons. The teacher may have to provide the children headphones so that they can work independently. In the Year 2 Espresso Coding unit, children will learn how to program codes so that they can move an object around the screen when specific keys are pressed.

It is important that the teacher explains to the children what an algorithm is, what a program is and how they can go about debugging and solving problems. Explain to the children that an algorithm is a sequence of instructions that creates a program, simple or complex. Debugging means to fix or solve a problem so that an algorithm works. The focus of this lesson is for children to understand



programming inputs. They will learn how to input specific instructions and movements to provide for specific outputs.

### Plenary session

Discuss and share the responses from the activity. Recap the terms algorithm, program and debugging and emphasise the need for precise instructions. Did the children find that they needed to provide specific instructions in order for their algorithm to work? What did they think? Can they remember any particular code? Can the children offer an example of code in which a specific action was produced?

### Differentiation

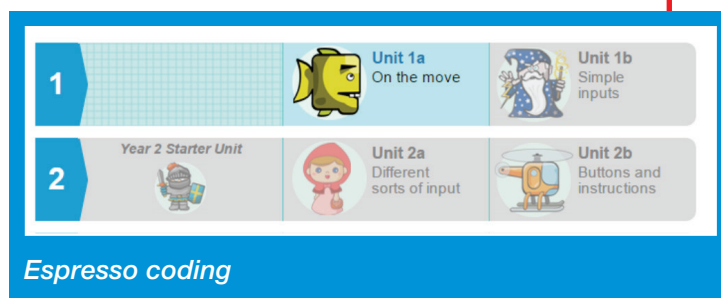
Some children will move quickly through these lessons. If they are using Angry Birds, encourage them to click on the two icons that ask them to create their own app. If they are using Espresso Coding, encourage them to move on to Unit 2a.

# Lesson 2: Programming Objects and Events

## Setting the scene: individual activity

### Learning objectives

- To program an object to move across the screen
- To input events to create more advanced programs
- To create a program which responds to different inputs
- To understand that the keyboard can be used to control different commands on-screen



### Resources

- Angry Birds Online: <http://studio.code.org/s/course2>
- Espresso Coding: <http://www.espressocoding.co.uk/espresso/coding/index.html>
- Optional: Headphones

### Individual activity

This lesson will continue to focus on teaching children the fundamental principles of programming. If they are using Angry Birds Online, allow the children who started Stage 3: Maze Sequence to continue and complete the lessons. For children that have completed the Maze Sequence, encourage them to start on Stage 4: Artist Sequence. This lesson will continue to focus on sequence and repetition. Children will learn to decompose large activities into a series of small events, and arrange a sequence of events into a logical order to create a small program. By now the children will have come to understand what is expected of them and the teacher may want to hand out the headphones again. Again, remind the children to 'view code' and pay more attention to the code. Tell them that you will ask for a list of codes that they remember which can be listed on the IWB, as the lessons continue. The teacher can start making a list on the board.

However, if the school is using Espresso Coding, then they can look to start the children on Unit 2a. In Unit 2a, there are six different lessons. The children will require headphones as they will all be at different stages. In Unit 2a, children will learn that a character is an object which can be controlled to do actions when keys are pressed. The children will begin to develop an understanding of algorithms and how a sequence of events can produce a desired outcome. Encourage the children to go through each lesson and complete all the exercises before they move on to the next.

As in the previous lesson, it is important to explain to the class definitions of an algorithm, debugging and what it means to solve problems in programming. The teacher may wish to offer extra points to those children who can offer a good understanding of the terms. Ask one or two children to come up to the interactive screen and demonstrate their work. Ask them to show you what makes an algorithm, what is its input and output, and how it is made into a program. Encourage the children to help each other if necessary.

### Plenary session

Discuss and share the responses from the activity. Explain that this lesson focuses on the use of an object and its direction which, when coded, can provide an output. This is the basis of a simple program. Illustrate the inputs and outputs of the lesson.

### Differentiation

Some children will move very quickly through these lessons. If using Angry Birds, encourage them to click on the two icons that ask them to create their own app continue to extend the creation of their app. This is an extension. If using Espresso Coding, encourage them to move on to Unit 2b.

# Lesson 3: Instructions and Buttons

## Setting the scene: individual activity

### Learning objectives

- To program an object to move across the screen
- To create a program which responds to different inputs
- To program objects to move in a particular direction

### Resources

- Angry Birds Online: <http://studio.code.org/s/course2>
- Espresso Coding: <http://www.espressocoding.co.uk/espresso/coding/index.html>
- Headphones
- Daisy the Dinosaur: <https://itunes.apple.com/gb/app/daisy-the-dinosaur/id490514278?mt=8>

### Individual activity

This is the last set of instructional activities using Angry Birds Online. Encourage the children to begin Stage 4: Artist Sequence. Explain to them that the next set of lessons will focus on setting instructions and clicking on buttons for the program to run. Encourage the children to finish the set of lessons and encourage them to change their settings to watch their code function in turtle slow mode to a faster fox run. Encourage them to explore the various options of how to code the same set of instructions in different ways.

This is a more advanced level of instruction. What patterns do they see repeating? What instructions are they using to solve the problem? Can the same result be achieved using a different set of instructions?

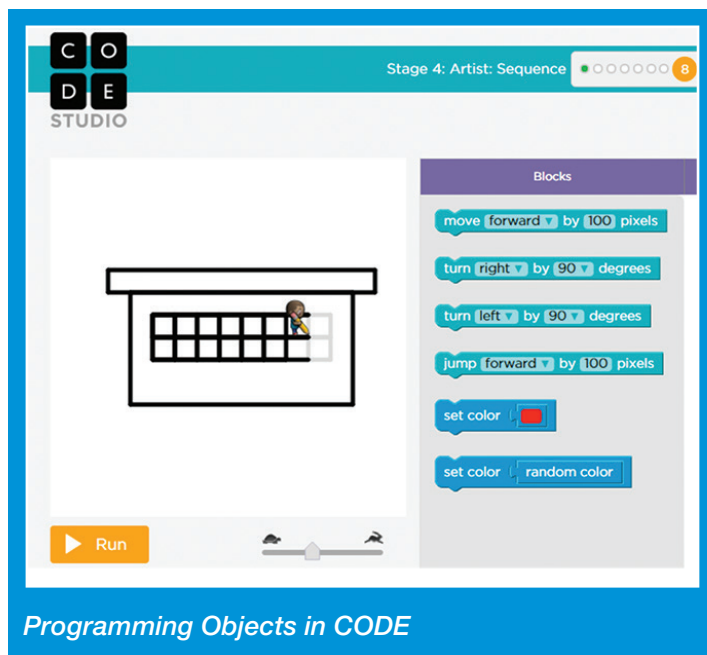
If using Espresso Coding, encourage the children to begin Unit 2b. Children will learn to move objects in a particular direction and will learn to use one object to control another object in a program. Together as a class, ask the children what they think will take place when they drag the icons into the function box and the start key is pressed. Encourage the children to predict the result first by studying the code first before the action is played back when the Run option is clicked.

### Plenary session

Discuss and share the responses from the activity. Ask the children what they have learned in the past few weeks. Recap the computational terms of algorithm, input, output and debugging. Ask the children what it means to debug a program. Encourage a couple of children to explain this to the class using the interactive white board.

### Differentiation

Some children may require further extension. Encourage them to use the Daisy the Dinosaur app on their tablets to experiment and create their own programming sequences.



Programming Objects in CODE