EDBOT

ROBOTIS

PLAY&CODE

Scratch 3.0

Teacher’s Guide

Table of Contents

Introduction 2
Number of lessons 2
Suitability 3
Differentiation 3
Preparation needed by the teacher 3
Assessing without Levels 4
Assessing Progress 4
Lesson 1 - Windmill 1: Make it & make it go! 5
Lesson 2 - Windmill 2: Your first program 6
Lesson 3 - Puppy 1: Time for a walk 7
Lesson 4 - Puppy 2: Going Loopy 8
Lesson 5 - Puppy 3: If only… 9
Lesson 6 - Puppy 4: No escape! 10
Lesson 7 - Car 1: Learning to drive 11
Lesson 8 - Car 2: Reverse parking 12
Lesson 9 - Car 3: The big race 13
Lesson 10 - Scorpion 1: Walking speed 14
Lesson 11 - Scorpion 2: Follow that line 15
Lesson 12 - Scorpion 3: Quiz time! 16

Introduction

This unit is suitable for students in KS2/KS3 and covers the following objectives from the National Curriculum for Computing:

- design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems
- solve problems by decomposing them into smaller parts
- use sequence, selection and repetition in programs; work with variables and various forms of input and output
- use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs

The unit is designed to be an introduction to various Scratch programming techniques. It is assumed that the students will already have been introduced to the Scratch programming environment and will know how to combine Scratch blocks together and execute them.

The lessons cover the following key concepts:

- Controlling a robot with Edbot Play&Code Scratch blocks
- Loops (repeat, forever and repeat until) and nested loops
- Variables and lists
- If statements
- Comparison and mathematical operators
- Joining strings
- Use of sprites, keyboard input and user input to control a robot
- Use of different sensors to control a robot

The aim of each lesson is to allow the students to control their Edbot Play&Code robot with a Scratch program. Students will be asked to modify existing programs or write their own programs and correct them for errors.

For each lesson there is an accompanying PowerPoint presentation that the teacher can use to guide the class through the lesson. The presentations guide the students through the programming concepts listed above, with plenty of examples and opportunities for the students to have a go at programming.

All the presentations have slides with recap questions to test the students' understanding. The recap answers are not provided as they are generally based on the lesson content.

Most lessons have extension coding tasks for higher ability students, so that they will always have something to keep them busy.

Number of lessons

There are 12 lessons which can be split into 2 blocks of 6. Most of the programming techniques listed above are introduced in the first 6 lessons and are based on the windmill and puppy robots. Lessons 7-12 provide an opportunity to further practice these techniques using the car and scorpion robots. Recommend a minimum of 12 x 1 hour teaching lessons. Some of the lessons have extension ideas and further activities that can fill longer lessons. The timings shown below are colour coded to help you split up the lesson timings to fit your length of lesson.

Pink	Short activities (under 5 minutes each)
Yellow	Medium length activities (between 6 and 12 minutes long)
Blue	Longer activities (over 12 minutes long)

Suitability

This unit is suitable for mixed ability classes.

Differentiation

Lessons are differentiated by outcome and this is reflected in the success criteria. Part of each lesson involves the students working independently, freeing up the teacher to offer more one-to-one help to the students who need it. There are also instructions in the lesson plans on how to adapt the lesson for lower or higher ability students.

Preparation needed by the teacher

The teacher needs to make sure that they are familiar with the contents of each lesson and the activities and questions involved which will help them with assisting the students in the lesson.

The teacher will need to be familiar with the Edbot Play&Code Scratch blocks. The documentation is available via the Edbot Software by selecting "Documentation…" in the "Help" menu.

Each student should be issued with a mark sheet at the beginning of the unit.

The Edbot Software should be installed on all the PCs and it is a good idea to have the students log in with their individual user names if they have them. This makes it easier to find their PC in the Edbot Software to give them control of the Edbot Play&Code robot. You also need to ensure the Edbot Software is configured correctly on the teacher's PC with the Edbot Play&Code robots connected via Bluetooth.

Make sure the Edbot Play&Code robots have been fitted with batteries that are installed correctly.

Before the first lesson you will need to convert the Scratch projects in the ***Teacher Files*** folder so they reference the particular name you've given to your Edbot Play&Code demonstration robot. Use the Edbot Software to do this. You'll find the project converter in the Scratch menu. Copy the converted student files into a shared area that the students will be able to access.

Four of the lessons include a time slot to build the Edbot Play&Code robots. The teacher will need to decide whether the construction of the Edbot Play&Code robots will be part of the lesson, or whether this will be an activity given to the students at another time. This may depend of the number of Edbot Play&Code kits available to the class. Building the robots provides the students with an opportunity to better understand how the robots work. It also provides an opportunity for developing fine motor skills, teamwork, communication and organisational skills.

Approximate robot assembly times (for an adult):

Edbot Play&Code robot	Approximate time to make (minutes)	Approximate time to take apart (minutes)
Windmill	15	5
Puppy	17	6
Car	20	7
Scorpion	20	7

Assessing without Levels

“As part of our reforms to the national curriculum, the current system of ‘levels’ used to report children’s attainment and progress will be removed. It will not be replaced.

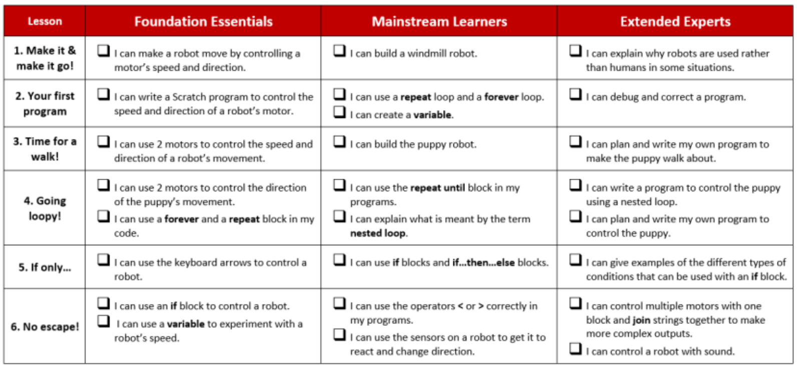
We believe this system is complicated and difficult to understand, especially for parents. It also encourages teachers to focus on a pupil’s current level, rather than consider more broadly what the pupil can actually do. Prescribing a single detailed approach to assessment does not fit with the curriculum freedoms we are giving schools.” [1]

With this in mind, we have developed a three-tier system which can easily be adapted to any system your school has implemented. We have referred to these bands as

- Foundation Essentials
- Mainstream Learners
- Extended Experts

Assessing Progress

Each student should be given a mark sheet on which they will need to write their name, so that they can get the same sheet back each lesson and could be kept in a work folder which they can refer to every lesson.



Lesson	Foundation Essentials	Mainstream Learners	Extended Experts
1. Make it & make it go!	❑ I can make a robot move by controlling a motor’s speed and direction.	❑ I can build a windmill robot.	❑ I can explain why robots are used rather than humans in some situations.
2. Your first program	❑ I can write a Scratch program to control the speed and direction of a robot’s motor.	❑ I can use a **repeat** loop and a **forever** loop. ❑ I can create a **variable**.	❑ I can debug and correct a program.
3. Time for a walk!	❑ I can use 2 motors to control the speed and direction of a robot’s movement.	❑ I can build the puppy robot.	❑ I can plan and write my own program to make the puppy walk about.
4. Going loopy!	❑ I can use 2 motors to control the direction of the puppy’s movement. ❑ I can use a **forever** and a **repeat** block in my code.	❑ I can use the **repeat until** block in my programs. ❑ I can explain what is meant by the term **nested loop**.	❑ I can write a program to control the puppy using a nested loop. ❑ I can plan and write my own program to control the puppy.
5. If only…	❑ I can use the keyboard arrows to control a robot.	❑ I can use **if** blocks and **if…then…else** blocks.	❑ I can give examples of the different types of conditions that can be used with an **if** block.
6. No escape!	❑ I can use an **if** block to control a robot. ❑ I can use a **variable** to experiment with a robot’s speed.	❑ I can use the operators **<** or **>** correctly in my programs. ❑ I can use the sensors on a robot to get it to react and change direction.	❑ I can control multiple motors with one block and **join** strings together to make more complex outputs. ❑ I can control a robot with sound.

The assessment sheet is based on “A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom’s Taxonomy of Educational Objectives”[2] and avoids use of the old national curriculum levels altogether. This means that the same unit can be used in years 3, 4, 5, 6, 7, 8 or 9 and the students can still show they have made progress.

The assessment should be completed at the end of every lesson by the students as a form of self-assessment and the last slide in each presentation tells the students the skills that they have covered. The students tick the box next to the objective if they feel they have fully met that criteria. The teacher can then use this as a basis to help them assess the students’ ability along with class observations, questioning students and viewing the students’ work.

[1] Taken from www.education.gov.uk/schools/teachingandlearning/curriculum/nationalcurriculum2014/a00225864/assessing-without-levels downloaded on 5th March 2014

[2] Anderson, L.W. (Ed.), Krathwohl, D.R. (Ed.), Airasian, P.W., Cruikshank, K.A., Mayer, R.E., Pintrich, P.R., Raths, J., & Wittrock, M.C. (2001). A taxonomy for learning, teaching, and assessing: A revision of Bloom’s Taxonomy of Educational Objectives (Complete edition). New York: Longman.

Edbot Play&Code

Lesson 1 - Windmill 1: Make it & make it go!

Lesson objective:	Understand how to write simple commands to control a robot.	
All will be able to: Make a robot move by controlling a motor's speed and direction.	**Most will be able to:** Build a windmill robot.	**Some will be able to:** Explain why robots are used rather than humans in some situations.

Differentiation

Low Ability:	**High Ability:**
Partner them with another student who might be able to help them.	Encourage them to think about how robots may be used in real-world situations.

Starter

Time	Description	Resources
Medium Activity	Display the starter slide and give the students a short time to talk about the questions together as a pair and then ask random pairs for their answers. Use slide 3 to explain what the unit is about.	edbot_play_scratch3_lesson1.pptx Slides 2 – 3

Main Activities

Time	Description	Resources
Short Activity	Explain the objectives to the class and go through slides 5 & 6 to explain what a robot is.	Slides 4-6
Long Activity	The students build the windmill robot.	Slide 7
Long Activity	Go through the slides to explain how the robot can be controlled using Scratch commands. Encourage the students to hold and examine the robot whilst you go through slides 9 & 10, explaining how to control the robot's motors.	Slides 8 - 14
Medium Activity	The students now have a go at using the Edbot Play&Code Scratch blocks to control the windmill's motor. When they are ready to turn the windmill robot on, they must hold the power button down until it beeps **twice**. When they raise their hand, select their computer from the dropdown list in the Edbot Software which gives them control of the Edbot Play&Code robot.	Slide 15

Review

Time	Description	Resources
Medium Activity	Go through the questions on the slide with the whole class. This will help them recap what they have learnt in the lesson.	Slide 16

Self-Assessment

Time	Description	Resources
Short Activity	Give out the mark sheet for lessons 1 to 6 from ***edbot_play_scratch3_mark_sheet.pdf*** and make sure your students write their name on it. They read through the highlighted objectives and if they feel they have met the criteria fully they need to tick the box. If they do not feel they have met the objective they should not tick the box.	edbot_play_scratch3_mark_sheet.pdf Slide 17

Edbot Play&Code

Lesson 2 - Windmill 2: Your first program

Lesson objective:	To write a program to control a robot and learn how to repeat code.	
All will be able to: Write a Scratch program to control the speed and direction of a robot's motor.	**Most will be able to:** Create a variable and use a **repeat** and **forever** loop.	**Some will be able to:** Debug and correct a program.

Differentiation

Low Ability: Provide a Scratch file which already has the blocks added and ask them to rearrange the blocks.	**High Ability:** Get them to complete the debugging exercise on slide 10.

Starter

Time	Description	Resources
Short Activity	Remind the students what they learnt in the last lesson and explain the objectives of this lesson.	edbot_play_scratch3_lesson2.pptx Slides 2 - 3

Main Activities

Time	Description	Resources
Short Activity	Explain what a computer program is.	Slide 4
Long Activity	The students now have a go at creating a program to control the windmill (slide 5). Make sure they can see the Edbot Play&Code blocks. Remind them to save their program. When they raise their hand, select their computer from the dropdown list in the Edbot Software which gives them control of the Edbot Play&Code robot. Slide 6 asks them to modify their program to use the repeat block. Ask them to predict what this block does.	Slides 5 – 6 Windmill robot edbot_play_scratch3_lesson2_windy_day_blocks.sb3 (for low ability)
Short Activity	Go through the example answer: ***edbot_play_scratch3_lesson1_windy _day.sb3***.	Slide 7
Long Activity	Go through these slides with the class, explaining and demonstrating how to create a variable. If time allows get everyone to write their code as you go through the slides. Slide 8 – pick someone to demonstrate their code with the windmill. Slide 9 introduces the forever block. Before you show the answer, can the students add the forever block to their code correctly? Let them run their code on the windmill to test it out.	Slides 8 - 9
Medium Activity	Extension - slide 10: A program has been incorrectly coded. Can the students use the storyboard shown to debug and change it so that it follows the storyboard correctly? The code is also long-winded. Can the students make the code more efficient by using a repeat function?	Slides 10 – 11 edbot_play_scratch3_lesson2_debug.sb3
Medium Activity	These are extension ideas to use the windmill robot with students who have completed the rest of the course.	Slides 12 - 13

Review

Time	Description	Resources
Medium Activity	Go through the questions on the slide with the whole class. This will help them recap what they have learnt in the lesson.	Slide 14

Self-Assessment

Time	Description	Resources
Short Activity	Give out the mark sheet from the previous lesson. They read through the highlighted objectives and if they feel they have met the criteria fully they need to tick the box. If they do not feel they have met the objective they should not tick the box.	edbot_play_scratch3_mark_sheet.pdf Slide 15