

Edbot Dream

Python

Teacher's Guide Sample



Table of Contents

Introduction	2
Suitability	2
Differentiation	2
Preparation needed by the teacher	2
Assessing without Levels	3
Assessing Progress	3
Lesson 1 of 4 - Crocodile	4
Lesson 2 of 4 - Scorpion	7
Lesson 3 of 4 - Puppy 1	9
Lesson 4 of 4 - Puppy 2	12

Introduction

This unit is suitable for students in KS2 and KS3 and covers the following criteria:

- 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

These lessons are not intended as a beginner's Python course but provide a chance to practice Python coding techniques already learned. These lessons are based on the Edbot Dream crocodile, scorpion and puppy robots. There are lots of other models that you can build and write programs for. These lessons are just a starting point to get your students going.

Number of lessons

Recommend 4 x 1 hour teaching lessons, although this will work with slightly shorter or longer lessons.

The timings shown are colour coded to help you split up the lesson timings to fit your length of lesson.

Pink	Short activities
Yellow	Medium length activities
Blue	Longer activities

Suitability

This unit is suitable for mixed ability classes.

Differentiation

Lessons are differentiated by outcome and this is reflected in the success criteria and 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 Dream Python API. The documentation is available via the Edbot Software by selecting "Documentation..." in the "Help" menu. The teacher can use the edbot_dream_python_intro.pptx presentation to introduce the Python API. It could be provided as a reference document for the students.

The Edbot Software should be installed on the teacher's PC 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 connection in the Edbot Software to give them control of the Edbot Dream. You need to ensure the Edbot Software is configured correctly on the teacher's PC with the Edbot Dream connected via Bluetooth.

Make sure the Edbot Dream's battery is fully charged and installed correctly.

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

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.

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 5, 6, 7, 8 or 9 and the students can still show they have made progress.

Edbot Dream Python Mark Sheet			
Name: _____		Class: _____	
Assessment Objective	Foundation Essentials	Mainstream Learners	Extended Experts
A.O.1	<input type="checkbox"/> I can program the crocodile robot to open and close its mouth by controlling a motor's speed and direction.	<input type="checkbox"/> I can program the crocodile robot to repeat a series of actions and react.	<input type="checkbox"/> I can program the crocodile robot to react when you sleep.
A.O.2	<input type="checkbox"/> I can make the scorpion robot's tail move by controlling a motor's speed and direction.	<input type="checkbox"/> I can program the scorpion robot to move and react to an object.	<input type="checkbox"/> I can program the scorpion robot to talk and use a function.
A.O.3	<input type="checkbox"/> I can make the puppy robot move.	<input type="checkbox"/> I can make the puppy robot move and react using functions.	<input type="checkbox"/> I can use an if...and statement. <input type="checkbox"/> I can create an efficient program with arbitrary pre-defined code.
A.O.4	<input type="checkbox"/> I can make the puppy robot react to the IR sensor.	<input type="checkbox"/> I can make the puppy robot move and stay inside the circle.	<input type="checkbox"/> I can make the puppy robot constantly move and stay in the circle.

Copyright © 2020 Robots in Schools Ltd. All rights reserved. Edbot is a registered trademark of Robots in Schools Ltd.

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 Dream Python

Lesson 2 of 4 - Scorpion

Lesson objective:	I can control multiple outputs with a sensor. I can use “while” loops, “if” statements and functions.	
All will be able to: Make the scorpion’s tail move by controlling a motor’s speed and direction.	Most will be able to: Program the scorpion to move and react to an object.	Some will be able to: Program the scorpion to talk and use a function.

Differentiation

Low Ability: Will be given a program file which has all lines of coded they will need already included. Focus on getting the tail and legs to move only. Students will experiment with lines of code and combine them correctly.	High Ability: They should be using functions in their program and attempt to make the scorpion talk.
--	--

Starter

Time	Description	Resources
Short Activity	<p><i>The lesson presumes that the scorpion robot is already built and ready to use. The robots take between 45-60 minutes to build depending on ability. Factor in 15 minutes or so to disassemble old robots.</i></p> <p>Explain the objectives to the class, slide 2.</p> <p>Demo the scorpion with the default program (<i>edbot_dream_python_lesson2_scorpion_demo.py</i>). What are the inputs and outputs being used? Students to discuss. Inputs – IR Sensor, Outputs - 2 motors and speakers from computer.</p>	<p>Scorpion robot</p> <p>edbot_dream_python_lesson2_scorpion_demo.py</p> <p>edbot_dream_python_lesson2.pptx</p> <p>Slides 2 - 4</p>

Main Activities

Time	Description	Resources
Medium Activity	<p>Set students a challenge - Can you write a program to make the scorpion’s tail sting? Discuss what that means? Tail moves and then stops.</p> <p>Students to write a program that makes the tail sting. The code below is included in the python file (<i>edbot_dream_python_lesson2a_scorpion.py</i>). For higher ability students you could remove it to make it more challenging. Remind them that they need to stop the motor moving after the tail sting.</p> <pre>ec.set_servo_speed(name, "2/-100") # Set motor on port 2 to speed 100 clockwise ec.set_servo_speed(name, "2/100") # Set motor on port 2 to speed 100 anti-clockwise time.sleep(1) # wait for 1 second</pre> <p><i>The tail/sting motor is plugged into port 2 according to the instructions.</i></p> <p>Higher ability students – can they write a function to make the scorpion’s tail sting?</p>	<p>edbot_dream_python_lesson2a_scorpion.py</p> <p>Slide 5</p>

Short activity	<p>Review solutions with the class and show how to turn the code into a function. Get the class to help you code a solution to the sting. Solutions will vary. See <i>edbot_dream_python_lesson2_scorpion_demo.py</i> for a possible solution.</p> <p>Why does the motor need to move clockwise and anti-clockwise each time? Get the students to examine how the scorpion's tail is constructed. The lever on the scorpion's tail limits the movement of the motor to 180° so the motor needs to reset to the initial position each time. When the motor moves in a clockwise direction the tail will sting. The tail will sting again when the motor moves in an anti-clockwise direction.</p>	Slide 6
Long Activity	<p>Can you make the scorpion react when someone gets too close?</p> <p>You must:</p> <ul style="list-style-type: none"> - make it move - make it sting - make the scorpion react to something getting too close <p>Discuss with the students, decomposing the problem. What have we already worked out? Sting. What is the next step? Movement. This could be changed into a function.</p> <p>Show demo again if needed. Note, the file <i>edbot_dream_python_lesson2b_scorpion.py</i> already includes the code for testing if something is too close to the scorpion. This code could be removed. This program also includes some hints for possible lines of code. These can be removed for higher ability students.</p> <p>Extension</p> <ul style="list-style-type: none"> - Can you make it speak as well? - Can you add functions to your code? 	<p><i>edbot_dream_python_lesson2b_scorpion.py</i></p> <p><i>edbot_dream_python_lesson2_scorpion_demo.py</i></p> <p>Slide 7 - 8</p>
Medium Activity	<p>Students should comment their code. Can you add a note to explain what it does? Summarise it. Remind them - comments always start with a # symbol.</p>	Slide 9

Review

Time	Description	Resources
Medium Activity	<p>Get students to share their solutions.</p> <p>Get them to explain their code, by summarising what it does.</p> <p>If needed show possible solution, <i>edbot_dream_python_lesson2_scorpion_demo.py</i></p> <p>Go through the other questions on the slide.</p>	<p><i>edbot_dream_python_lesson2_scorpion_demo.py</i></p> <p>Slide 10</p>

Self-Assessment

Time	Description	Resources
Short activity	<p>Give out the mark sheet from the previous lesson. The students 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.</p>	<p><i>edbot_dream_python_mark_sheet.pdf</i></p> <p>Slide 11</p>