# Lesson 29 – Testing and Development

|  |  |
| --- | --- |
| The Big Picture – Why is this Relevant? | Learning Objectives |
| * This lesson gives Learners the opportunity to demonstrate the skills that they have developed. They will initially be given code that they trace through to determine what it does * Learners will then add to the code through completing a number of differentiated activities | * Understand how to trace through code to determine the functionality * Understand how to determine the outputs from a number of given inputs * Add functionality to a given program independently |
| Engagement – How Can I Engage Learners? | Assessment for Learning |
| * Learners will enjoy realising how much they have learned throughout the course so far * They will gain a real sense of achievement through being able to determine the outputs from the provided code whilst also having the opportunity to add further functionality | **Expected Progress:**   * Learners will understand that test plans can be used to check that code produces the correct outputs for any valid input * Learners will be able to interpret the purpose of a program from looking at the code   **Good Progress:**   * Learners will understand how to use selection statements to decide whether a line of code should be executed * Learners will be able to complete the code challenges to add extra functionality to the provided program   **Exceptional Progress:**   * Learners will understand how to create their own test plan to check the functionality of their code * Learners will be able to write a range of valid tests and will execute those tests to check for any errors |
| Key Concepts: | Key Words |
| * Test plans are used to check that correct outputs are generated for a set of valid inputs * The activity covers a range of code structures that have been covered throughout the course. Learners will investigate the use of input, output and selection statements. * Learners will develop their own code which will be executed by pressing Button B. They will understand that different blocks of code can be executed depending upon which input is entered by the user. | * Test plan * Input * Output * Selection |
| Differentiation | Resources |
| A range of differentiated activities have been included as part of the challenge. More capable Learners should create their own test plans for the additional code that they have produced. | * 1 micro:bit per learner * 1 USB cable to connect the micro:bit to a PC * A PC * Access to <https://makecode.microbit.org> * Lesson 29 ppt * Test Plan worksheet * Code Challenge worksheet * Lesson 29.hex * 1 enviro:bit per user |
| Lesson Flow | |
| * This lesson follows the PRIMM methodology. Predict - Learners will initially be provided with a set of code which they will attempt to explain in their own words. The code has been included on the ‘Test Plan’ worksheet. * Explain to Learners that one way to check that code works as expected is to write a test plan and then run each test to check that valid outputs are generated. Explain that the test plan included on the worksheet was written before the code was written. Learners should complete the test plan without running the code. They should try to work out the outputs from looking at the code. * Run – Learners should now run the code on their micro:bit. Ask them to think about whether it performs as expected. * Investigate - Learners should then complete each of the tests. Does the program perform as expected? Did any of the tests fail? More capable learners may be able to establish that the second IF statement contains a < instead of a > symbol. * Modify – Once they have identified the error, they should try to fix the problem and re-run the test. Explain that this is the type of process that happens when developing computer programs in industry. * Make - learners should now try to complete the challenges on the ‘Code Challenge’ worksheet. They should initially design their own icons for the different temperatures. The appropriate icon should be shown for the different temperatures. The next challenge is to create code that executes when Button B is pressed and measures humidity. This challenge requires learners to use the enviro:bit. They will need to remember how to install the enviro:bit library. When a reading is taken they should display the following appropriate messages based upon the humidity reading. Learners can choose their own message or use the suggestions below:   + 40 – Perfect Summer Day   + 50 – Comfortable   + 60 – A little uncomfortable   + 65 – Very uncomfortable   + 75 - Unbearable * More capable learners should create a test plan to check that their new code functions as expected | |
| Making | |
| There are no making activities in this lesson. | |