Structuring Code

Control flow has allowed us to create dynamic programs that can use conditions within the code to create branches and loops. This has given you a lot of capability, but the code can quickly get out of hand without a better way of organising things. In this module, learn about how to structure your code to better organise the logic and your thoughts.

Learning Goals

To complete this task, you need to demonstrate that you can do the following:

- Create, work with, and describe functions and procedures and how they operate.
- Use parameters to accept data into functions and procedures.
- Return data from functions.

Focus

As you work through this task, focus on the following aspects of the unit:

- **Programming concepts:** Focus on how functions and procedures allow you to modularise your code, and how parameters allow you to send data around your program.
- **Programming process**: Focus on how we can break-down larger programs into smaller functions and procedures, allowing each to have a single purpose and a small amount of logic.
- **Coding**: Focus on memorising the language rules for creating functions and procedures, parameters, and returning data.
- **Professional Characteristics:** Continue focusing on building your attention to detail, and in this topic, how you can supercharge compartmentalisation using functions and procedures.

Your Task

For this task you will need to submit the following:

- A PDF document containing:
  - Summaries and reflections.
  - At least three (3) screenshots:
    - One each for the change calculator, guess that number, and the test your knowledge activity
  - Learning Journey and Resources
- Source code for:
  - Revised Change Calculator
  - Code for Guess that Number
  - Your chosen program from the Test Your Knowledge activities

1. Complete Learning Activities

Work through these steps to develop and demonstrate your understanding. Aim to demonstrate, to yourself and others, that you have achieved the learning goals.

1. Everything your need is in Chapter 2 Structuring Code from Part 2 of the Programmers Field Guide.
2. Build at least two small programs to test out how functions and procedures work
   a. Create a program with a procedure – test out calling it multiple times. See this age for examples.
   b. Create functions to read a string, read an integer, and read an integer within a range (e.g. an integer between 1 and 10) and a program to test these. Include parameters for the prompt and the range in read integer range. (For help with the logic see the Read Functions page)

3. Build the revised Change Calculator, Guess that Number, and the updated Fly Catch program. **Capture** notes on your learning as you progress, indicating if and where you need to review the existing solutions.
4. Complete one of the Test Your Knowledge activities.
5. Prepare your summary, making sure to cover all learning goals and related concepts. Remember that this is a personal summary that demonstrates your understanding of the concepts.
6. Prepare your reflections by responding to the following:
    - How do you know you have achieved the learning goals?
    - What is the most important thing you learned from this and why?
7. Capture your learning journey and collate your evidence of study and practice.

2. Upload Your Submission & Engage with Feedback

Mark the task as **Ready for Feedback** and upload the required files. Make sure to keep copies of these in case you need to resubmit. Then engage with the feedback you receive and get the task Complete!