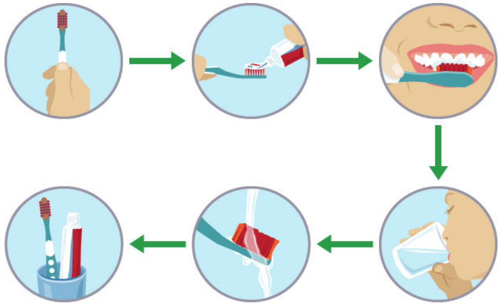
Knowledge Organiser: Programming — Procedures and Functions

Procedures and Functions

Summary—When writing programs, we should avoid long, repetitive code. Procedures and functions help to keep our programs simple and short.

A **procedure** is a small section of a program that performs a specific task. Procedures can be used repeatedly throughout a program.

A **function** is also a small section of a program that performs a specific task that can be used repeatedly throughout a program, but the task is usually a calculation. Functions perform the task and return a value to the main program.



What is a Procedure

A **procedure** is a small section of a program that performs a specific task. Procedures can be used repeatedly throughout a program.

Why use a Procedure

Program code is easier to read and understand when it is broken up into smaller sections. By breaking a program up into these sections, or procedures, code can be made shorter and simpler.

Writing a Procedure

Writing a **procedure** is extremely simple. Every procedure needs:
a name

the **program** code to perform the task

Running a Procedure

Once a **procedure** is named and written, it can be **called** at any point in the **program**.

To call a procedure in **Python**, simply use its name (include the brackets)

What is a Function

A **function** is also a small section of a program that performs a specific task that can be used repeatedly throughout a program, but the task is usually a calculation. Functions perform the task and return a value to the main program**.**

Both functions and procedures are small sections of code that can be repeated through a program. The difference between them is that functions return a value to the program where procedures perform a specific task.

Why do we use Functions

Program code is easier to read and understand when it is broken up into smaller sections. Functions and procedures can make code shorter, simpler and easier to write.

A function is just like a procedure, except that you wait to see what the result is.

Running a Function

Once a **function** is named and written, it can be **called** at any point in the **program**. 'Calling' a function means running it.

To call a function in **Python**, simply use its name (include the brackets) and the value it needs to use for calculation, plus a variable that the returned value will be stored in.

Key Vocabulary

Algorithm	A sequence of logical instructions for carrying out a task. In computing, algorithms are needed to design computer programs.
Calling	Starting or running a function or procedure.
Code	Instructions in a computer program.
Debug	The process of finding and correcting programming errors.
Function	A section of code that, when programming, can be called by another part of the program with the purpose of returning one single value.
Procedure	A section of computer code that performs a specific task.
Program	Sequences of instructions for a computer.
Python	A high-level programming language.
Statement	The smallest element of a programming language which expresses an action to be carried out.

Knowledge Organiser: Programming — Writing Error Free Code

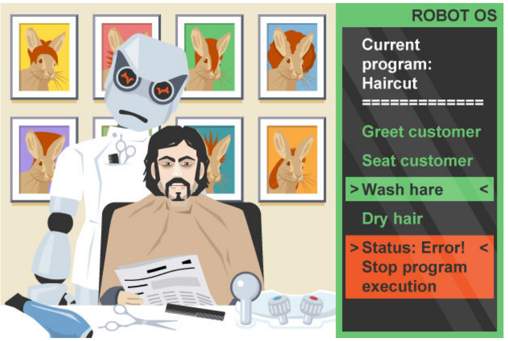
Writing Error Free Code

Summary

When writing programs, code should be as legible and error free as possible. **Debugging** helps keep code free of errors and documenting helps keep code clear enough to read.

When programs are written, it is likely that at least some errors will creep in. Errors in programs are often referred to as **bugs**.

It is vital that programs are as free of errors as possible. Errors can cause a program to produce unexpected results, or **crash**.



Syntax Error

Syntax errors will cause a program to crash or not run at all. The **program** may run until it encounters a syntax

Types of Syntax Error

More than one type of syntax error may exist. There may be:
incorrectly spelled **statements**
incorrectly spelled **variables**
missing punctuation (quotes, brackets, etc)

Any one or more of these errors may exist in a program, and each will cause the program to crash or not run at all.
Fixing syntax errors is simple. Spotting them is more difficult. Some of the most common syntax errors are **things that come in pairs**. Always check for speech marks and brackets first if your program

Fixing syntax errors is simple.
Spotting them is more difficult.

Logic Errors

Logic errors occur when there is a **fault** in the **logic** or **structure** of the problem. Logic errors do not usually cause a program to **crash.** However, logic errors can cause a program to produce **unexpected results.**

Types of Logic Error

More than one type of logic error may exist. Parts of the program may:
be in the wrong **sequence**
have the wrong **Boolean** expression
use the wrong **data type**
be missing altogether

Any one or more of these errors may exist in a pro-

Documenting Code

Code that is hard to read makes it difficult to understand what the **program** is trying to do. It also makes it difficult to understand the purpose of
any **variables**, **procedures** and **functions**. Errors are easier to fix when we understand the code we are reading through.

When we **document** code, we make it easier to read and understand. We also document code in case:
we need to come back to it at a later date

Key Vocabulary

Boolean	A data type in computing which only has two possible values, true or false.
Bug	An error in a program.
Code	Instructions in a computer program.
Commenting	Adding one or more sentences to explain the purpose of a section of code.
Crash	When an application or operating system no longer responds. Crashes can be caused by software and hardware.
Data Type	The format in which a variable or constant holds data, such as 'integer' or 'string'.
Debug	The process of finding and correcting programming errors.
Documenting	Making code easier to understand by giving meaningful names for variables, procedures and functions and by adding comments to explain the purpose of instructions.
Execute	To run a computer program.
Function	A section of code that, when programming, can be called by another part of the program with the purpose of returning one single value.
Instruction	A single action that can be performed by a computer processor.
Logic error	A fault in the logic or structure of the problem.
Procedure	A section of computer code that performs a specific task.
Program	Sequences of instructions for a computer.
Programming Language	A language used by a programmer to write a piece of software.
Python	A high-level programming language.
Sequence	In computer programming, this is a set of instructions that follow on one from another.
Statement	The smallest element of a programming language which expresses an action to be carried out.