Imperative Languages - Reading

Imperative languages are a style of language that you are most likely to encounter.

In English, an imperative sentence is one which has a **command**. For example, "sit down". In the same way, imperative languages are built from commands in programming which state how the program must operate.

Many languages use an imperative style. Examples are: C, Java, C#, PHP, Python and Visual Basic. Languages that ask questions, such as **SQL**, are **query languages** and are called **declarative languages**.

There are three key principles that are used in imperative languages:

1. **Sequences** – the ability to run instructions/commands in order
2. **Iteration/Repetition** – the ability to repeat code until some condition is met
3. **Selection/Choice** – the ability to run different branches of instructions based on a condition

Sequence

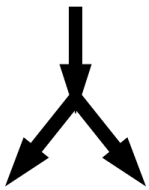
```
age1 ← 15
age2 ← 14
ageTotal ← age1 + age2
avgAge ← ageTotal / 2
OUTPUT avgAge
```

Iteration

```
FOR i ← 1 TO 10
   OUTPUT i
ENDFOR
```

Selection

```
gameRunning ← True
IF gameRunning THEN
   OUTPUT 'Playing game'
ELSE
   OUTPUT 'Game over'
ENDIF
```



Other features of imperative languages are given in the table below.

Imperative language feature	What it does
Variables	Store a value in memory which **can be changed** while the program is running. The variable will use an **identifier**, known as a **variable name** to refer to the value stored.
Constants	Constants are the same as variables, except the value **cannot be changed** when the program is running.
Assignment	This is the operation of placing a value into a variable.
Mathematical operations	The ability to do calculations such as addition or subtraction.
Subroutine	Defining a block of code with its own name so that it can be called and run at various points in the program. When the block of code **returns** a value it is called a **function**.
Comparison operations	The ability to compare values and variables. This is often used as part of selection or iteration.