Lists

### General Teacher Notes

#### Explanation of Lists

A list is an example of a ‘[data structure](https://www.geeksforgeeks.org/data-structures/)’. This is the name for a number of different programming techniques used to organise and manipulate data in programs.

So far to store data in the program we have used variables, which hold one piece of data and use an identifier (variable name) to represent it in the code.

Lists work in the same way as variables, but they store multiple items of data using one identifier.

Lists are created in Python like this:

countries = [“UK”, “USA”, “Chad”, “Australia”, “Thailand”]

primeNumbers = [1,3,5,7,11,13]

Notice that the list containing string (text) data has each item in speech marks. The list containing integer (whole number) data doesn’t need them.

To identify individual items in a list, the data is **indexed** (given a number). Indexing starts at 0, meaning that the first item in a list will be number 0, the second number 1 and so on.

#### Lists - Key Concepts/Misconceptions

You may hear lists referred to as ‘arrays’ in other programming languages. There are small differences but at this stage we will treat them as essentially the same thing.

A list can store multiple items of data.

Items in a list are surrounded by square brackets. Each item is separated by a comma.

Lists are indexed - each item is given a numerical position in the list.

Lists are named in the same way as variables - the programmer decides what they are called. Using [camel case](https://en.wikipedia.org/wiki/Camel_case) is best practice

Indexing starts at 0, students often forget this as they are in the habit of starting to count with 1.

A list is changeable - each individual item can be edited, added or removed. Extra items can be added to the list.

A list allows duplicate items.

**Errors To Watch Out For**

Square brackets for items in the list

Commas between each item in the list.

Range - the end value is NOT included in the output.

Search - not initialising counter OUTSIDE the loop.

Search - not incrementing counter INSIDE the loop.

### 

### 1 - Output A Range Of Items

#### TEACHER NOTES

It is possible to output some consecutive items from a list by customising the print function to add a start and end index for the output.

There are a couple of traps to be careful of here.

The start value IS included in the output.

The end value IS NOT included in the output.

countries = [“UK”, “USA”, “Chad”, “Australia”, “Thailand”]

# Outputs USA, Chad, Australia

print(countries[1:4])

# Outputs from the beginning of the list to Chad.

print(countries[:3])

# Outputs from USA to the end of the list.

print(countries[1:])

#### Output a Range of Items - Tasks

##### Predict And Run

Task and instructions - <https://repl.it/@MrAColley/251-Output-Range-Predict-Run>

Example solution - <https://repl.it/@MrAColley/251-Output-Range-Predict-Run-Solution>

##### Investigate And Modify

Task and instructions - <https://repl.it/@MrAColley/252-Output-Range-Investigate-Modify>

Example solution - <https://repl.it/@MrAColley/252-Output-Range-Investigate-Modify-Solution>

##### Make

Task and instructions - <https://repl.it/@MrAColley/253-Output-Range-Make>

Example solution - <https://repl.it/@MrAColley/253-Output-Range-Make-Solution-1>

Example solution 2 (with functions) - <https://repl.it/@MrAColley/253-Output-Range-Make-Solution-2>

### 2 - Search A List

#### TEACHER NOTES

These activities teach students how to use loops to search a list using the [linear search](https://www.youtube.com/watch?v=TwsgCHYmbbA) technique.

The slides guide you through 3 ways to do this, the first two are more ‘traditional’ methods, the third is a method that is built into Python. This 3rd method is a huge shortcut but I’ve included the other two so that students start to appreciate the actual workings of how the search takes place.

There’s a lot to unpack with the code at first, rather than put it here I’ve laid it out as step by step examples on the slides. Take your time and model lots of examples to help your class understand it. Please check out the slide notes and work through with your class.

#### Search a List - Tasks

##### Predict And Run

Task and instructions - <https://repl.it/@MrAColley/254-Search-Lists-Predict-Run>

Example solution - <https://repl.it/@MrAColley/254-Search-Lists-Predict-Run-Solution>

##### Investigate And Modify

Task and instructions - <https://repl.it/@MrAColley/255-Search-List-Investigate-Modify>

Example solution - <https://repl.it/@MrAColley/255-Search-List-Investigate-Modify-Solution>

##### Make

Task and instructions - <https://repl.it/@MrAColley/256-Search-List-Make>

Example solution - <https://repl.it/@MrAColley/256-Search-List-Make-Solution>