

Practical Exercise 2

(1) Write a function that returns information about different countries. The function should take the name of a country as a single string input, and return a single structure named 'Info' that contains the following fields as an output: 'Population', with a value equal to the population of that country; 'Capital', with a string that contains the name of the capital of that country; 'Language', with a string that contains the name of the official language of that country; and 'Area', with a value equal to the area of that country in square kilometres. Choose any five or more countries that this function will return data for. If the input string does not match the name of any of your chosen countries, then ask the function to print the following text on screen:

"Sorry, I do not have any information about that country..."

(2) Write a function that converts a distance in metres to one of several other units. The function should take two inputs – one numerical value that corresponds to a length, in metres; and one string that corresponds to another unit of length - either inches, in; feet, ft; millimetres, mm; or centimetres, cm. The function should return one numerical output, corresponding to the input distance converted into the chosen units, and print text describing the conversion on screen. For example, if I were to enter a distance of 1 as an input and a unit of 'feet', the function would return an output variable with a value of 3.281 and print the following string on the screen:

"1 metre is equal to 3.281 feet"

(3) Write a script that first generates a 100 row x 101 column matrix named 'Data' containing random numbers that are normally distributed with a mean of 700 and a standard deviation of 100, and then completes the following steps:

- Replace any values that are less than 500 with a value of 500
- Add 100 to the 20th, 25th, 30th and 35th rows
- Multiply the 90 to 100th rows by 1.5
- Change the first column to the numbers 1 to 100
- Compute the mean of each row, excluding the first column, and store it as a variable named 'MeanRT'
- Generate a 100 row x 1 column variable named 'Conditions' that contains fifty values of 1 and fifty values of 2 in a randomised order
- Imagine that the 'MeanRT' variable contains mean reaction time data for 100 participants in a behavioural task who were divided into two conditions, corresponding to the two different values in 'Conditions'. Compute the standard error of the 'MeanRT' values for each condition separately and store them as variables named 'StdErr1' and 'StdErr2', respectively
- Perform a two sample t-test comparing the values in 'MeanRT' between participants in each condition. Store the t-statistic as a variable named 't_stat' and the corresponding p value as a variable named 'p_value'