# Exercises

## Lesson 4

**Exercise**: write a small piece of code which will accept two numbers from the user and print out a suitable message indicating which is the larger or whether or not they are equal.

**Exercise:** Write a for loop to print the 5 times table. Starting at 1x5 and ending with 12x5.

**Exercise**: Write a while loop which accepts words from the user and stops when the uses type ‘End’.

## Lesson 6

**Exercise**: For the flowchart shown at the end of Lesson 2, create a test plan to test the possible combinations of the 3 variable A, B and C.

## Lesson 7

**Exercise**: Modify the Reading and Writing to a file example (files.py) to use your local copy of the Rio\_medals\_table.csv file and count the number of records in it.

**Exercise**: Modify the Reading and Writing to a file example (files.py) to use your local copy of the Rio\_medals\_table.csv file and print the full record to the screen

**Exercise**: Modify the Reading and Writing to a file example (files.py) to use your local copy of the Rio\_medals\_table.csv file. Accept a country code from the user and output the contents of the record containing that country code.

**Exercise**: Modify the Reading and Writing to a file example (files.py) to use your local copy of the Rio\_medals\_table.csv file. Accept a country code from the user and using the split() function, output the contents of the record containing that country code as a list of items.

**Exercise**: Modify the Reading and Writing to a file example (files.py) to use your local copy of the Rio\_medals\_table.csv file. Accept a country code from the user and using the split() function, output the Gold, Silver and Bronze values for the given country code.

**Exercise:** Repeat the last exercise, but now accept 2 country codes, record the Gold, silvers and bronzes for each in variables and then when you have finished reading the file, print out the Gold, Silver and Bronze values for each country code

## Lesson 8

**Exercise**: Extend the xtimesprint function to add an optional parameter called sep to indicate the separator between the string repetitions. The default value for sep should be the null string ""

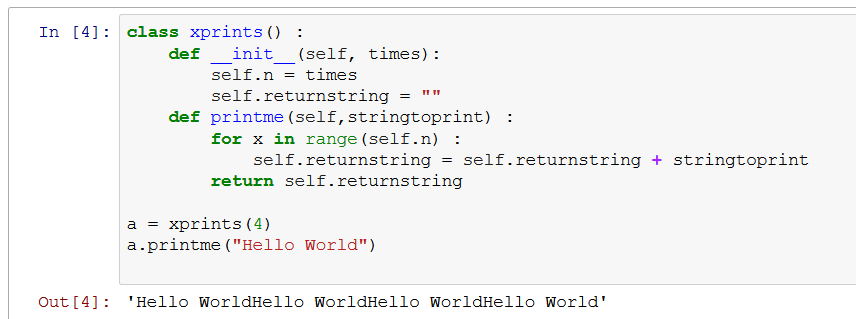
If you wanted the repetitions to be on separate lines, what value would you pass to sep?

Provide an appropriate docstring entry.

**Exercise**: Write a function which returns the product of the numbers 1 to n. The value of n should be provided as a parameter and not be > 20.

## Lesson 9

**Exercise**: The code below creates a class called xprints and gives an example of it being used. It is based on the xtimesprint function we used in lesson 8.



Create a new class based on this class which include the functionality of providing a separator character when an instance is created and allows it to be overridden in a call to the method printme.