COURSE 811M

PYTHON FOR DATA SCIENTISTS

EXERCISE MANUAL

The following course materials are copyright protected materials. They may not be reproduced or distributed and may only be used by students attending the *Python for Data Scientists* course.

Table of Contents

This page intentionally left blank.

Exercise 2.1: Array Creation

The aim of this exercise is to gain some experience of working with NumPy arrays.

1. Start IPython.

2. Define an `ndarray` containing the integer numbers 0 to 9.

3. Print the type of the array to the console.

4. Print the following properties of the array (they are accessible in the same way as `ntype`):

   a. `ndim`
   b. `shape`
   c. `size`
   d. `itemsize`

5. Define a 3 x 3 NumPy array containing all 1's and display it to the console.

6. Print the four properties of Step 4 on the array defined in Step 5.

This page intentionally left blank.

Exercise 2.2: Array Basic Operations

The aim of this exercise is to gain some experience of working with basic operations on NumPy arrays.

1. Define a 3 x 3 array with the integers 1 through 9 named `array1`.

2. Define a second 3 x 3 array with the number 2 in each cell named `array2`.

3. Now, perform the following operations using the two arrays:

    a. `array1+array2`
    b. `array1-array2`
    c. `array1/array2`
    d. `array1*5`

4. Print elements 4 to 6 of array 1 using a slice operation.

5. Create a new single-dimensioned array named `array3` with the numbers 0 through 19 in it.

6. Take a slice of elements 5 to 15 of array 3 and assign the slice to a variable named `aslice` and print the variable.

7. Modify the contents of the first and last elements of the slice by writing the value 99 into these elements.

8. Print the contents of the slice `aslice` and the array `array3`. Are the contents what you expect of both arrays?

   _____

This page intentionally left blank.

Exercise 3.1: Pandas `Series`

The aim of this exercise is to gain some experience of working with the Pandas `Series` data structure.

1. Define a `Series` object holding the values 1 to 10.

2. Display the data values of the `Series` object defined in Step 1.

3. Display the index values of the `Series` object defined in Step 1.

4. Define a new `Series` object holding the values 1 to 10, with the corresponding index values set 'a' through to 'j'.

5. Display the data values and index of the `Series` object of Step 4.

6. Access the third and fifth elements of the `Series` object using their index.

7. Define the following dictionary: `{'Dublin': 200000, 'Athlone': 15000, 'Galway': 700000}`.

8. Define the following array: `['Dublin', 'Athlone', 'Waterford']`.

9. Now, construct a `Series` object using the dictionary in Step 7 and the index in Step 8.

10. Display the `Series` object defined in Step 9.

11. Use the `Series notnull()` and `isnull()` methods to display which elements are not null and null, respectively, for the `Series` object defined in Step 9.

This page intentionally left blank.

Exercise 3.2: Pandas `DataFrame`

The aim of this exercise is to gain some experience of working with Pandas `DataFrame` data structure.

1. Use the following dictionary to create a `DataFrame`:

   a. `{'team':['Leicester', 'Manchester City', 'Arsenal'], player':['Vardy', 'Aguero', 'Sanchez'],'goals':[24,22,19]}`

2. Display the above `DataFrame` to the console.

3. What values are assigned for the index and columns?

   _____

4. Use the dictionary from Step 1 and create a second `DataFrame` with index values `'one'`, `'two'`, `'three'`, respectively, and columns `team`, `player`, `goals`, `played`. Display the `DataFrame` to the console.