

*Your submission for this tutorial must include your full name and you nine-digit student number as a comment at the top of every source file you submit. All source code files must be written using the Python 3 programming language and must run on the course's official virtual machine.*

---

**Exercise A: "Matrix Validity Test"**

---

For this exercise you will design and implement a function that takes one argument and returns a single Boolean value indicating whether or not the argument is a "valid" matrix. The operational definition for a "valid" matrix (for this activity) is a rectangular list of lists of numeric data elements, so your function will need to test each of these three properties separately.

In order to complete this task, you will need to:

- recognize that the two data types in Python that are actually numeric are "int" and "float"
- read about Python's built-in "isinstance" function

Your submission for this exercise:

- must be a source code file with filename<sup>1</sup> `'comp1405_f21_#####_tutorial_07_a.py'`
- must use one or more loops to check that every element in the argument is, itself, a list
- must use one or more loops to check that every "row" of the matrix is the same length
- must use one of more loops to check that every element in the matrix is a numeric value

---

**Exercise B: "Matrix Addition"**

---

For this exercise you will design and implement a function that takes two matrix arguments and computes their sum (if and only if a sum can actually be computed, returning an empty list otherwise). Two matrices can only be added together if they have exactly the same dimensions. If the dimensions are acceptable, then the matrix sum will have the same dimensions as either of the operands, and the value of each element in the matrix sum is itself the sum of the corresponding elements in the operand matrices. And although it does make sense to test rectangularity before attempting addition, you need not perform this step (since you did it above).

In order to complete this task, you will need to:

- ensure you know how matrix addition can be performed<sup>2</sup>

Your submission for this exercise:

- must be a source code file with filename `'comp1405_f21_#####_tutorial_07_b.py'`
- must return an empty list if the two operands (i.e., the arguments) have different dimensions
- must create a new matrix that holds the sum (without overwriting either of the arguments)

---

<sup>1</sup> You must replace the number signs in the filename with your official nine-digit student identification number.

<sup>2</sup> You may wish to visit [https://en.wikipedia.org/wiki/Matrix\\_addition](https://en.wikipedia.org/wiki/Matrix_addition) if you are uncertain about this operation.