INTRODUCTION TO SETS

LEARNING OBJECTIVES:

* What are sets?
* Why sets?
* Intro to working with sets (union, intersections, in/not in, for)

WHAT ARE SETS?

In Python, a set is an object representing an unordered collection of unique elements (that is, no duplicates). Sets are iterable and mutable.

Individual elements of sets must be “hashable”, meaning their value will not change during their lifetime. In other words, only immutable objects (numbers, strings, tuples, or frozen sets, and not lists, dictionaries, or sets) can be elements of sets.

, and support (operations).

Sets may be “frozen sets”, which are immutable

WHY USE SETS?

For one, speed! Unlike lists, sets are unordered. Thus, sets Because (\_\_\_\_), checking to see if a value is included in a set is significantly faster than checking to see if a value is included in a list.

For two, memory! Unlike dictionaries, sets do not have (indexes and keys). Thus, sets take up significantly less memory than dictionaries.

Thus, sets are useful whenever (you need to \_\_\_\_ or \_\_\_\_ but not \_\_\_\_).

WORKING WITH SETS