Other Notes

**String methods:**

.isupper()/.islower() – return whether the string is all uppercase/lowercase or not, respectively.

.upper()/.lower() – returns the string converted to all uppercase/lowercase, respectively.

.isalpha()/.isalnum() – returns whether the string is alphabetical/alphanumeric only, respectively.

.isdecimal() – returns whether the string has a decimal value.

**Everything else:**

enumerate(iterable) – returns an iterator of tuples (i, val) where i is the index and val is a value from iterable.

Separate input by a number rather than by a case separation character:

x **=** x**.**splitlines**()**

res **=** **list()**

**while** x**:**

# extract the number of lines from the input

nlines **=** **int(**x**[**0**])**

# append the case to res

res**.**append**(**x**[**1**:**nlines **+** 1**])**

# remove the case from x

x **=** x**[**nlines **+** 1**:]**

Using empty classes to store information in a way that is easy to remember and versatile:

**class** **person:** **pass**

# you don’t need to define anything inside the class, just give it a name

ross **=** person**()**

ross**.**name **=** 'Ross'

ross**.**age **=** 21

keegan **=** person**()**

keegan**.**name **=** 'Keegan'

# instance variables can still be compared

**if** ross**.**name **==** keegan**.**name**:** **print(**"same name"**)**

# this will crash because keegan.age is undefined

# if ross.age > keegan.age: print("older")

# these object act just like a dictionary, and can be converted to one with vars()

**print(vars(**ross**))** # {'name': 'Ross', 'age': 21}