import – import module/library

as – change library name

<module>.<function>() – use module function

def <function name>(<parameter>) – make function

<variable> = lambda <parameter> : <function> - anonymous function

continue – skip loop till end but don’t break, start over

assert – make an error message over a condition, can’t be used in Lambda

in – something is in an array

print – print(“<text>”+<variable>+”<text>”)

if, else, elif – elif is else if

class –

#<comment> - comment

try – check something

except <name> - <exception block>/<print> - alert if fails on this specific error

else – if all works

finally – after everything, do this even if it doesn’t work

iterator

make iterable, list or array or tuple, something with things inside it.

a=[1,6,5]

b=iter(a)

print(next(b) -> 1

print(next(b) -> 6

print(next(b) -> 5

print(next(b) -> error, no items left

or b.\_\_next\_\_()

<list or array>.remove(<item>) – removes item by name

<list or array>.pop(<index>) – removes item by index

<list or array>.clear() – remove all items

del <list>[index] – same as pop

copy list - <new list> = <list>.copy() or <new list>=list(<list>)

sets

<set name> = {<items>}

print(<set>-<other set>} – show only things that are unique to the first set

& - both ^ - only unique parts of both sets | - both sets

dictionary – <name>= {“<key>”:”<value>”,}

<dictionary>[<key>] or <dictionary>.get(“<key>”)– get value

<variable>=input("<text>") – ask for a string input.

<variable> = int(<old variable>) – turn old variable into different type