Python

**Core Functions**

**Open(path):** opens file and returns a file object

**ord:** Returns unicode of char

**isinstance(object, class):** checks if object is ofType

**is:** Checks if two objects are the same

**type(myvar):** returns the object type

**str**(someNumber) converts to string

**float**(someVar) converts to float

**bool**(someVar) converts to bool

**int**(someVar) converts to int

Sequence functions

**len(anySequence):** gets length of seq, whether its a string, list, dict…

**in:** sees if an element is in a sequence

**sorted(anySequence):** returns a sorted seq but leaves original alone

**reversed(anySequence):** returns a reversed seq but leaves original alone

**List functions**

**list(string):** converts string to a list, one for each char

**sum(someList):**sums items in list(if number)

**del someList[index]:** deletes from a sequence at index. Can use slice methods

**Dictionary functions**

**newemptydict = dict():** creates a new empty dictionary

**Tuple functions**

newtuple = tuple()

**Conversion**

To **int** from **float** = **int(float**(number))

**Split techniques [wheretostart : wheretoend:step]**

'banana'[0:5:2] = 'bnn'sd

**newList = someList[1:-1]** Creates new list without 1 and last entries of old list

**reversed= smeList[::-1]** reverses a list

Other!

**raise LookupError():** raises an exception

**<within method> global some\_global\_var = ‘new value’:** reassigns a global var. Note this is happening within a method

If t not in ‘sequence’

**String object**

**Methods** *Most string methods return new string and leave original alone*

**Len(s):** length of string

**s.isLower():** True if all of s is lower

**s.isUpper():** True if all of s is upper

**s.Upper():** Converts s to all Upper

**s.count(char,start):** Returns # of occurrences of char in s

**s.find(char,start):** Returns index of char if found

**s.strip():** Trims string

**s.capitalize():** Capitilizes first char

**s.split(‘ ‘):** converts string to a list, one for each word

**s.join(‘ ‘):** inverse of split

**List object []**

**Methods** *Most list methods modify the argument and return None.*

**l.append(item):** adds to end of list

**l.extend(list):** adds new list to end of list

**l.remove(item):** deletes item

**l.pop(index):** deletes item at index and returns item

**l.pop():** deletes last item and returns item

**l.sort()** sorts list

**emptylist= []** Creates empty list

**newlist = list1 + list2** Creates newlist which is same as **extend** method

**[8] \* 4** = new list of [8,8,8,8]

**Dictionary object {}**

**Methods** *newdict= {'one': 1, 'two': 2, 'three': 3}*

**v = d.values():** returns values of a dictionary

**d.get(1,some default value):** returns value at index or default value

**seq = d.items():** Returns a sequence of tuples

**emptydict={}**

**Tuple object ()**

**Methods** *newtuple= 1,2,3*

**v = d.values():** returns values of a dictionary

**d.get(1,some default value):** returns value at index or default value

**emptytuple=()**

**Traversing Sequences**

**By index:** for i in range(len(values))

**By value:** for r in results:, for k,v in dict.items():

**File object**

**Methods**

**f.readline():** reads a line from file