Appendix D: Working with Dictionaries

	Description	Sample Code	Output

clear( )	Removes all elements of the dictionary, returning an empty dictionary	dic1 = {1: ‘one’, 2: ‘two’}
		print (dic1)
		
		dic1.clear()
		print (dic1)
			{1: 'one', 2: 'two'}
			
			{ }

del	Delete the entire dictionary	dic1 = {1: ‘one’, 2: ‘two’} del dic1 print (dic1)	NameError: name 'dic1' is not defined

get( )	Returns a value for the given key.
	
	If the key is not found, it’ll return the keyword None.
	
	Alternatively, you can state the value to return if the key is not found.
		dic1 = {1: ‘one’, 2: ‘two’}
		dic1.get(1)
		
		dic1.get(5)
		
		dic1.get(5, “Not Found”)
			‘one’
			
			None
			
			‘Not Found’

in	Check if an item is in a dictionary	dic1 = {1: ‘one’, 2: ‘two’} # based on the key 1 in dic1 3 in dic1 # based on the value ‘one’ in dic1.values() ‘three’ in dic1.values()	True False True False

items( )	Returns a list of dictionary’s pairs as tuples	dic1 = {1: ‘one’, 2: ‘two’}
		dic1.items()
			dict_items([(1, 'one'), (2, 'two')])

keys( )	Returns list of the dictionary's keys	dic1 = {1: ‘one’, 2: ‘two’}
		dic1.keys()
			dict_keys([1, 2])

len( )	Find the number of items in a dictionary	dic1 = {1: ‘one’, 2: ‘two’} print (len(dic1))	2

update( )	Adds one dictionary’s key-values pairs to	dic1 = {1: ‘one’, 2: ‘two’}
		dic2 = {1: ‘one’, 3: ‘three’}	

	another. Duplicates are removed.	dic1.update(dic2)
		print (dic1)
		print (dic2) #no change
			{1: 'one', 2: 'two', 3: 'three'}
			{1: ‘one’, 3: ‘three’}

values( )	Returns list of the dictionary's values	dic1 = {1: ‘one’, 2: ‘two’}
		dic1.values()
			dict_values(['one', 'two'])