

# Python Network Programming Cheat Sheet

## Python Hints

**myString = 'this is a string.'**

This example shows how to declare a string.

**myNumber = 432**

This example shows how to declare a number.

**MyBool = True**

This example shows you how to declare a boolean value like True or False. Notice there's no single or double quotes?

**myList = ['string 1','string 2',53,42]**

This example shows how to declare a list containing two strings, and two numbers.

**myList[0]**

Piggybacking on the example above, myList[0] would print 'string 1'. This is because it is the 0 index of the list. myList[1] would print 'string 2', and so on.

**""" Check out my long  
comment!!!  
"""**

3 quotation marks opens a comment block and 3 quotation marks closes them. Use this to document your code! This is a very useful technique so you understand what your code is doing. Additionally you may use a # sign to comment.

## Socket Library Hints

Review the socket library documentation here!

<https://docs.python.org/2/library/socket.html>

**import socket:** Enhances your script with socket communication capabilities. This line needs to be at the top of your script to make sure it can be used.

**socket.gethostbyname(string):** Pass a hostname as a string like 'dsu.edu' into this method to have the socket library resolve the domain name dsu.edu into an IP address.

**socket.gethostbyaddr(string):** Pass an IP address as a string like '138.247.65.57' into this method to have the socket library find the hostname for the IP address.

**socket.socket(address\_type, protocol\_type):** Use this method to define a socket object. Is is important

to define the address type, and the protocol to be used for communication. An example to define an IPv4 TCP socket would be: **myTCPSocket = socket.socket(socket.AF\_INET,socket.SOCK\_STREAM)**

**myTCPSocket.connect((ip\_address,port\_number))**: piggybacking off of the above example. Once you have defined what type of socket you want to utilize, you need to connect to the destination. Here, you pass the ip\_address and port\_number together as a pair to connect. The (( and )) are not typos and you must type this to successfully connect. A working example to connect to the DSU.EDU Webserver would be: **myTCPSocket.connect(('138.247.65.57',80))**

**myTCPSocket.sendall(string)**: piggybacking off of the above example. Once the myTCPSocket.connect() method has been called, you can now send any string you want through the socket. This literally means you are sending data to the other machine!!! A working example of this would be: **myTCPSocket.sendall('Is anyone there?')**

**myTCPSocket.recv(#ofdatapackets)**: piggybacking off of the above example. Once the myTCPSocket.sendall() method succeeds, you now have the opportunity to capture what message is sent back from the destination. Make sure that the #ofdatapackets is a number between 1 and 65535 A working example of this would be: **receivedData = myTCPSocket.recv(1024)**

**socket.AF\_INET** (IPv4), **socket.AF\_INET6** (Ipv6): The common values used in socket.socket() method for the address\_type variable.

**socket.SOCK\_STREAM** (TCP), **socket.SOCK\_DGRAM** (UDP), **socket.SOCK\_RAW** (Special): The common values used in the socket.socket() method for the protocol\_type variable.