

```
# James Roesemann
# CSCI379
# Project 1
# Documentation
#
# The programs TCP_client.py and TCP_server.py use the TCP protocol to
send and receive messages.
# They both use the port number 9999 to transmit messages.
#
# the program TCP_server must be running before TCP client in order to
operate.
#
# the default hostname of the TCP_server is the loopback address.
# If an argument is passed to TCP_client the value of the first
argument will be TCP_server hostname.
# if an invalid or unreachable ip address is passed as an argument
TCP_client will not run.
# The user of TCP_client enters a sentence to be modified. TCP client
transmits this sentence in the form of a string to TCP_server on port
9999.
# The user of TCP_client entered a command to modify the sentence.
TCP_client transmits this command in the form of a string to
TCP_server on port 9999.
#
# TCP_server receives a sentence in the form of a string and an
command from TCP_client in the form of a string.
# if the command value is valid the sentence is modified based on the
value.
# If the command value is invalid a message stating so replaces the
sentence.
#
# the valid commands are listed as follows:
# Upper: changes all letters in the string to upper case.
# Lower: changes all letters in the string to lower case.
# Swap: Change all the upper case letters to lower case letters and
change all the lower case letters to uppercase letters.
# Rev: reverse the order of the words in the string.
# Cap: capitalize the first letter of the string. change all other
letters to lower case.
# KILL: Kills the process running on the server. Returns a message
confirming this.
#
# TCP_server transmits the modified sentence back to TCP_client using
port 9999.
# In most cases TCP_server stays online and waits for the process to
repeat.
# If the KILL command was received, then the socket closes and the
program terminates.
```