THE UNIVERSITY FIREWALLS

You will remember that we needed to add the port used by our Tornado server to the firewall that Azure creates for our Ubuntu Virtual Machine (and similarly if we run sshd on a second port).
e.g. from the Workshop

```
...add the port numbers you use to the inbound security rules in Azure:
Azure: Resource groups:
   My-Resource-Group
      Icon that looks like a shield
         Inbound security rules
            + Add
               Name: MyTestServer
               Destination port range: 43210
```

You may have noticed that some of the computer labs at the University also have their own firewall, and this is not something we can adjust ourselves (it required a request to ICT). We do however know that port 80 is open (this is the default port for http traffic, and so is generally open). In order to use this port (instead of 43210) for our Tornado server one way is as follows (do not do this!):

1) Open up port 80 on the Azure firewall (i.e. as above, but with 80 instead of 43210)

2) Change our python script to use port 80

3) Run our python script with root privileges (since any port <1024 needs admin rights to use)
THIS STEP IS HIGHLY PROBLEMATIC, SO I DO NOT SUGGEST YOU USE THIS METHOD!

Running your own programs with root privileges is generally a bad idea, you can wipe your entire Virtual Machine with a single mistyped command, so it's best reserved for systems administration tasks.

So, how then do we make our Tornado traffic go to port 80 without running our python script as root?

The standard method is to adjust the Virtual Machine's own firewall table to redirect as follows:

```
$ sudo iptables -A INPUT -i eth0 -p tcp --dport 80 -j ACCEPT
$ sudo iptables -A INPUT -i eth0 -p tcp --dport 43210 -j ACCEPT
$ sudo iptables -A PREROUTING -t nat -i eth0 -p tcp --dport 80 -j REDIRECT --to-port 43210
```

(note, these changes are temporary and will be lost if you reboot your machine).

This will effectively link ports 80 and 43210 together, meaning you can run your Tornado server on port 43210 (so keeping your port number and without requiring root privileges to run), and access it on either.

You will likely find you can access your server on either port at home, but only on port 80 in the Labs. Also, since port 80 is the default for http you can actually miss off (":80") when you provide the url to your browser as it will be assumed anyway.[1]

1 Note, I have used this method on the test service for your assignment, so you can access it in the labs via port 80.