****

**Mawlana Bhashani Science and Technology University**

Santosh, Tangail-1902.

|  |
| --- |
| **Lab Report** |

**Department of Information and Communication Technology**

**Report No:** 05

**Course Code : ICT-3208**

**Course Title:** Network Planning and Designing Lab

|  |  |
| --- | --- |
| Submitted By | Submitted To |
| Name: **Md. Abul Basar**  ID: **IT-17026**  Session: 2016-17  3rd Year 2nd Semester  Dept. of Information & Communication Technology, MBSTU. | **Nazrul Islam**  **Assistant Professor**  Dept. of Information & Communication Technology, MBSTU. |

**Python functions:**

Functions are reusable pieces of programs. They allow you to give a name to ablock of statements, allowing you to run that block using the specified name anywhere in the program and any number of times. This is known as calling the function.

**Local Variables:**

Variables declared inside a function definition are not related in any way to othervariables with the same names used outside the function (variable names are local to the function). This is called the scope of the variable. All variables have the scope of the block they are declared in starting from the point of definition of the name.

**The global statement:**

Variables defined at the top level of the program are intended global. Globalvariables are intended to be used in any functions or classes). Global statement allows defining global variables inside functions as well.

**Modules:**

Modules allow reusing a number of functions in other programs.

* **TCP**:

TCP stands for transmission control protocol. It is implemented in the transport layer of theIP/TCP model and is used to establish reliable connections. TCP is one of the protocols that encapsulate data into packets. It then transfers these to the remote end of the connection using the methods available on the lower layers. On the other end, it can check for errors, request certain pieces to be resent, and reassemble the information into one logical piece to send to the application layer.

* **UDP:**

UDP stands for user datagram protocol. It is a popular companion protocol to TCP and is alsoimplemented in the transport layer.

The fundamental difference between UDP and TCP is that UDP offers unreliable data transfer. It does not verify that data has been received on the other end of the connection. This might sound like a bad thing, and for many purposes, it is. However, it is also extremely important for some functions.Because it is not required to wait for confirmation that the data was received and forced to resend data, UDP is much faster than TCP. It does not establish a connection with the remote host, it simply fires off the data to that host and doesn't care if it is accepted or not. Because it is a simple transaction, it is useful for simple communications like querying for network resources. It also doesn't maintain a state, which makes it great for transmitting data from one

machine to many real-time clients. This makes it ideal for VOIP, games, and other applications that cannot afford delays.

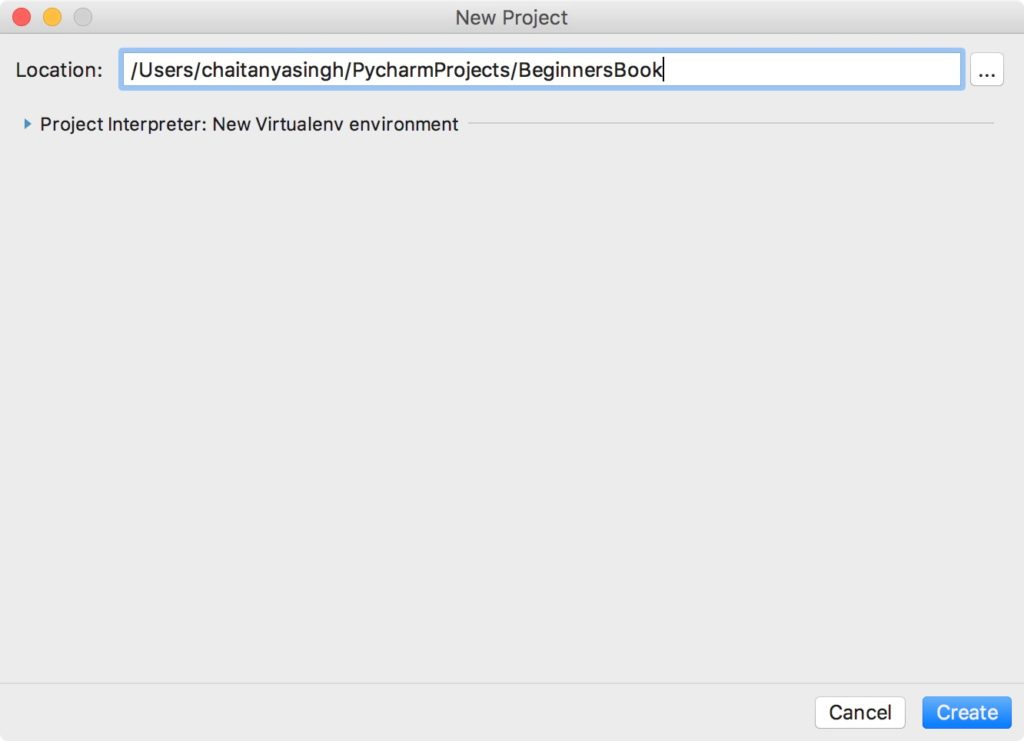
**Exercises:**

**Exercise 4.1.1: Create a python project using with Computer Network Lab**

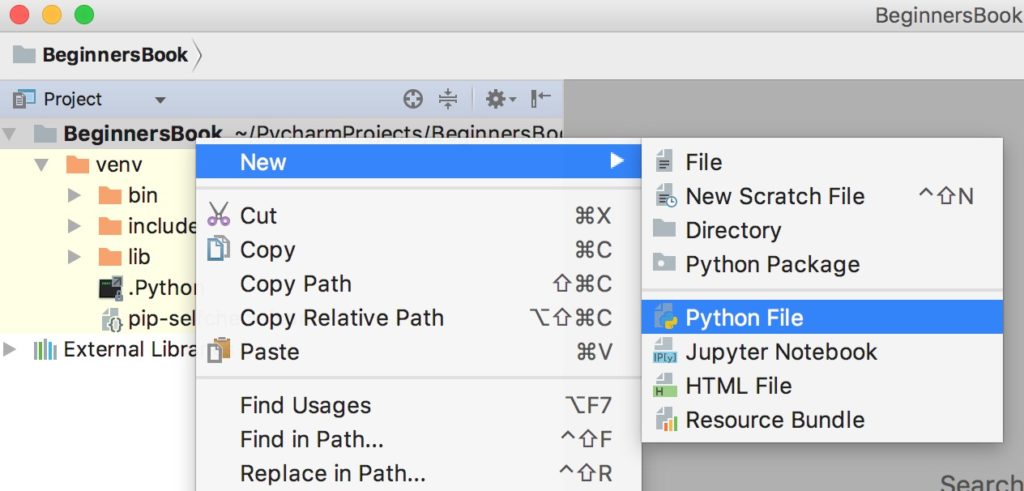
1. **Click “Create New Project” in the PyCharm welcome screen.**



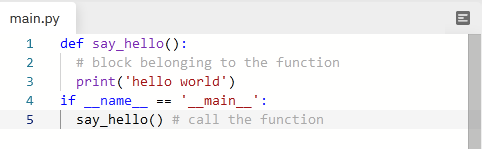
1. **Give a meaningful project name.**

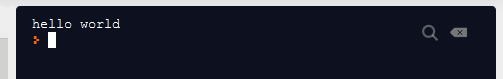


**Writing and running your first Python Program**

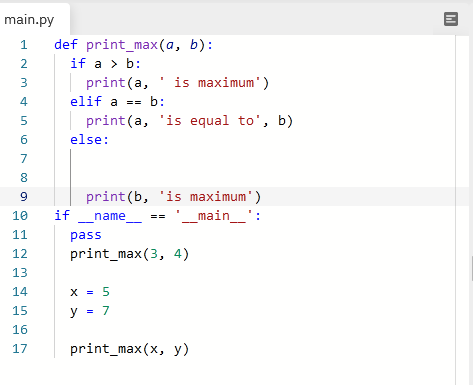


**Exercise 4.1.2: Python function (save as main.py)**



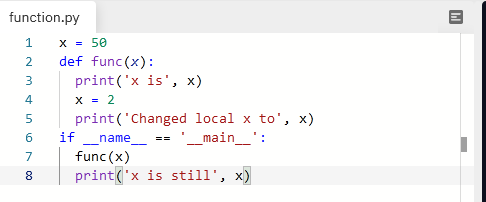


**Exercise 4.1.3: Python function (save as main.py)**



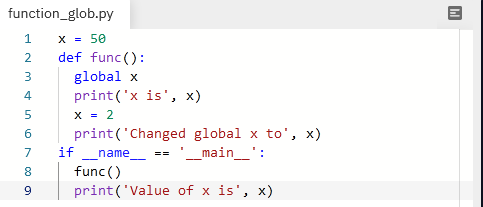


**Exercise 4.1.4: Local variable (save as function.py)**



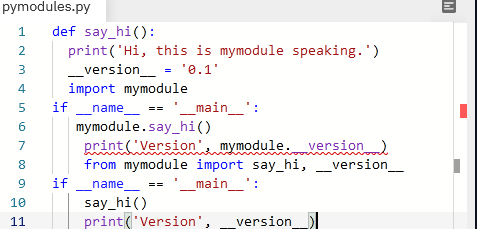


**Exercise 4.1.5: Global variable (save as function\_glob.py)**

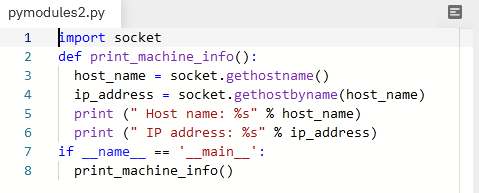


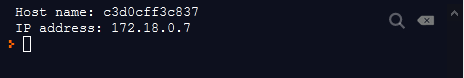


**Exercise 4.1.6: Python modules**

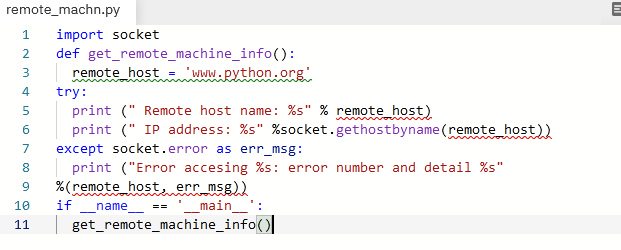


**Exercise 4.2.1: Printing your machine's name and IPv4 address**

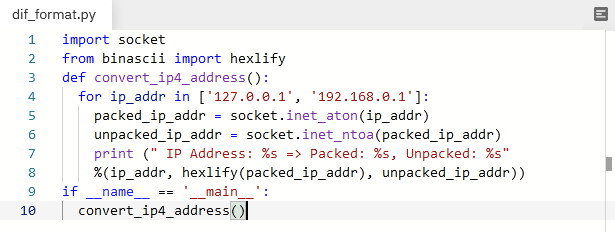


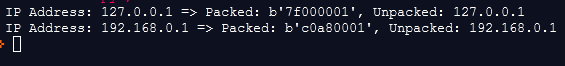


**Exercise 4.2.2: Retrieving a remote machine's IP address**

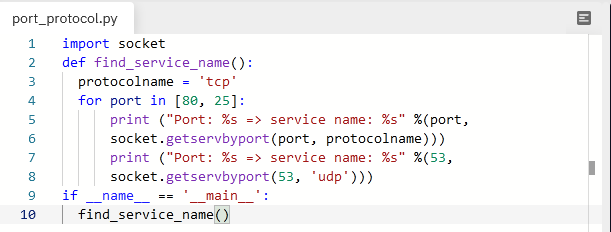


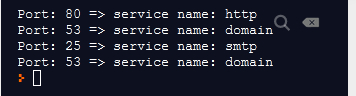
**Exercise 4.2.3: Converting an IPv4 address to different formats**



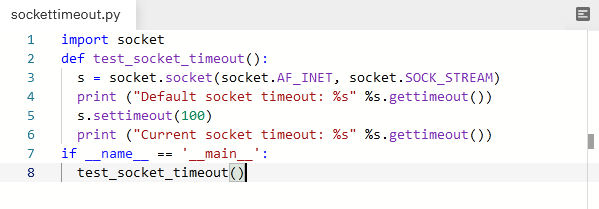


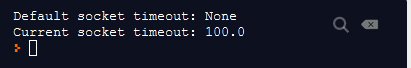
**Exercise 4.2.4: Finding a service name, given the port and protocol**





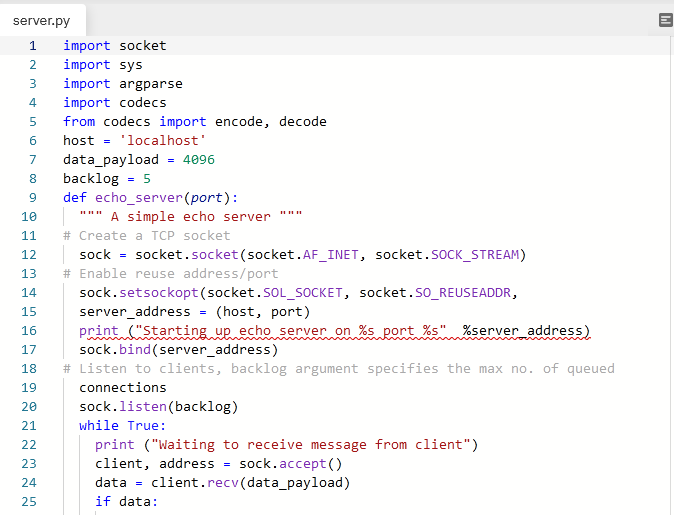
**Exercise 4.2.5: Setting and getting the default socket timeout**

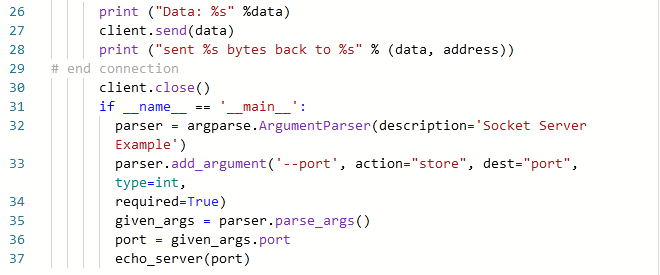




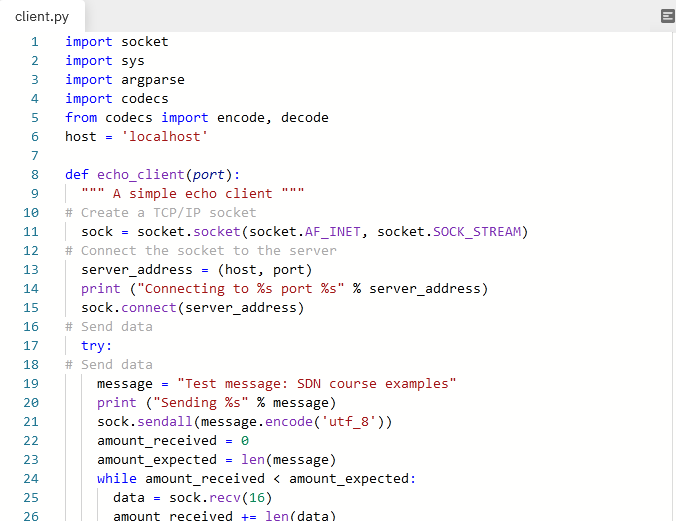
**Exercise 4.2.6: Writing a simple echo client/server application (Tip: Use port 9900)**

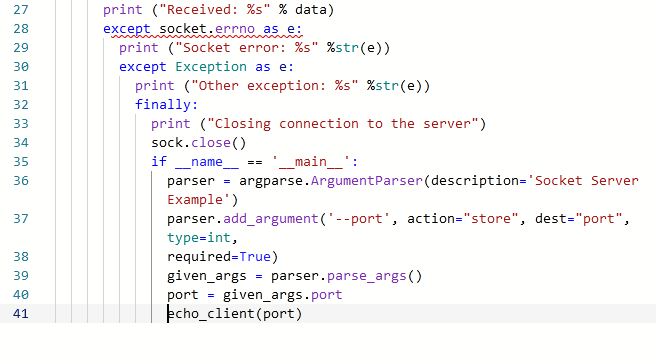
**Server code:**





**Client code :**





**Conclusion:**

Python plays an essential role in network programming. The standard library of Python has full support for network protocols, encoding, and decoding of data and other networking concepts, and it is simpler to write network programs in Python than that of C++. There are two levels of network service access in Python.

In the first case, programmers can use and access the basic socket support for the operating system using Python's libraries, and programmers can implement both connection-less and connection-oriented protocols for programming.

Application-level network protocols can also be accessed using high-level access provided by Python libraries. These protocols are HTTP, FTP, etc.