**Python for Everyone: Notes**

**Chapter 12: Networked Programs**

* Transport Control Protocol (TCP)
  + Built on top of IP (internet protocol)
  + Assumes IP might lose some data – stores and retransmits data if seems to be lost
  + Handles flow control using a transmit window
  + Provides a nice reliable pipe
* TCP connections / sockets
  + In computer networking, an internet socket or network socket is an endpoint of a bidirectional interprocess communication flow across an internet protocol based computer network such as the internet
* TCP port numbers
  + A port is an application specific or process specific software communications endpoint
  + It allows multiple networked applications to coexist on the same server
  + There is a list of well known TCP port numbers
* Common TCP ports
  + Telnet (23) – login
  + SSH (22) – secure login
  + Http (80)
  + Https (443) – secure
  + SMTP (25) – mail
  + IMAP (143 / 220 / 993) – mail retrieval
  + POP (109/110) – mail retrieval
  + DNS (53) – Domain name
  + FTP (21) – file transfer
* Sockets in python
  + Python has built in support for tcp socket
* Application protocol
  + Since TCP and python gives us reliable socket, what do we want to do with the socket, what problem do we want to solve
  + Application protocols
    - Mail
    - World Wide Web
* HTTP – hypertext transfer protocol
  + The dominant application layer protocol on the internet
  + Invented for the web to retrieve HTML, images, documents, etc
  + Extended to be data in addition to documents, rss, web services, etc. Basic concept, make a connection, request, a document, retrieve the document, close the connection
* HTTP
  + The hypertext transfer protocol is the set of rules to allow browsers to retrieve web documents from servers over the internet
* What is a protocol
  + A set of rules that all parties follow so we can predict each other’s behavior
  + And not bump into each other
    - On two way roads in the USA, dive on the right hand side of the road
    - On two way roads in the UK, drive on the left hand side of the road
* Getting data from the server
  + Each time the user clicks on an anchor that with an href= value to switch to a new page, the browser makes a connection to the web server and issues a “GET” request to get the contents of the page at the specified url
  + The server returns the html document to the browser, which format and displays the document to the user
* Internet standards
  + The standards for all of the internet protocols (inner workings) are developed by an organization
  + Internet engineering task force (IETF)
  + Standards are called “RFCs” – request for commands
* Making an http request
  + Connect to server like [www.dr-chuck.com](http://www.dr-chuck.com)
  + Request a document (or the default document)
    - Get [http://www.dr-chuck.com/page1.htm HTTP/1.0](http://www.dr-chuck.com/page1.htm%20HTTP/1.0)
* ASCII
  + American standard code for information interchange
* Representing simple strings
  + Each character is represented by a number between 0 and 256 stored in 8 bits of memory
  + We refer to “8 bits of memory as a “byte” of memory
  + The ord() function tells us the numeric value of a simple ascii character
* Multi byte characters
  + To represent the wide range of characters computers must handle we represent characters with more than one byte
    - UTF-16 fixed length 2 bytes
    - UTF-32 fixed length 4 bytes
    - UTF-8 1-4 bytes
      * Upward compatiable with ascii
      * Automatic detection between ascii and utf-8
      * Utf-8 is recommended practice for encoding data to be exchanged between systems
* Python 3 and Unicode
  + In python 3, all strings interally are Unicode
  + Working with string variables in python programs and reading data from files usually just works
  + When we talk to a network resource using sockets or talk to a database we have to encode and decode data usually to utf-8
* Python strings to bytes
  + When we talk to an external resource like a network socket we send bytes, so we need to encode python 3 strings into given character encoding
  + When we read data from external resource, we must decode it based on the character set so it is properly represented in python 3 as a string
* Using urllib in python
  + Since http is so common, we have a library that does all the socket work for us and makes webpages look like a file
* What is webscraping
  + When a program or script pretends to be a browser and retrieves webpages, looks at those web pages, extract information and then looks at more web pages
  + Search engines scrape web pages – called spidering the web or web crawling
* Why scrape
  + Pull data, particularly social data, who links to who
  + Get your own data back out of some system that has no export capability
  + Monitor a site for new information
  + Spider the web to make a database for a search engine
* Scraping web pages
  + There is controversy about web scraping and some sites are a bit snippy about it
  + Republishing copyrighted information is not allowed
  + Violating terms of service is not allowed
* The easy way – beautiful soup
  + You could do string searches the hard way
  + Or use free software library called beautiful soup
* Summary
  + The tcp/ip gives us pipes/sockets between applications
  + We designed application protocols to make use of these pipes
  + Hypertext transfer protocol (http) is simple yet powerful protocol
  + Python has good support for sockets, http, and html parsing