Coursera Course Notes - Python for Everyone (Using Python to Acess Web Data)

Module 4

Unicode Characters and Strings:

Character Encoding Basics

- Computers understand **numbers**, not letters → need mapping.
- **ASCII** (128 characters): maps numbers to letters (e.g., H=72, e=101, newline=10).
- Early systems: **only uppercase** letters, 1 character = 1 byte (8 bits).

Limitations of ASCII

- Only covers English + basic symbols.
- Different countries built **incompatible character sets**.

Unicode & UTF Encodings

- **Unicode**: universal mapping for all world languages.
- **UTF-32**: 4 bytes/char (inefficient).
- **UTF-16**: 2 bytes/char.
- **UTF-8**: variable length (1–4 bytes), backward compatible with ASCII → **most widely used**.

Python Strings

- **Python 2**: two string types → "string" (ASCII/bytes) vs u"string" (Unicode).
- **Python 3**: all strings = Unicode; separate **bytes type** (b"string").

Encoding & Decoding

- Inside Python: always Unicode.
- Outside world (files, networks, databases): data must be **encoded/decoded**.
  - **Encode**: string → bytes (for sending).
  - **Decode**: bytes → string (for receiving).
- Default encoding: **UTF-8** (compatible with ASCII).

Retrieving Web Pages

Using urllib in Python

- Python makes web requests simple → use urllib instead of raw sockets.
- urllib.request.urlopen(URL) opens a connection → returns a **file-like handle**.
- Works like open(filename) but for web URLs.

Reading Web Data

- Use for line in handle: to loop through web page lines.
- Data comes as **bytes**, not strings → must **decode** (usually UTF-8).
- Once decoded → normal Python string methods apply.

Headers vs Data

- urlopen hides HTTP headers by default → can retrieve headers separately if needed.
- Default loop only gives the **body/content**.

Processing the Content

- Treats a webpage like a file → easy to split, strip, count words, etc.
- Can handle text files or HTML.

Building a Simple Web Crawler

- Example: search for href="..." links in HTML.
- Fetch each link → repeat → basic web crawler logic.
- This is essentially how **Google's crawler** works (at large scale).