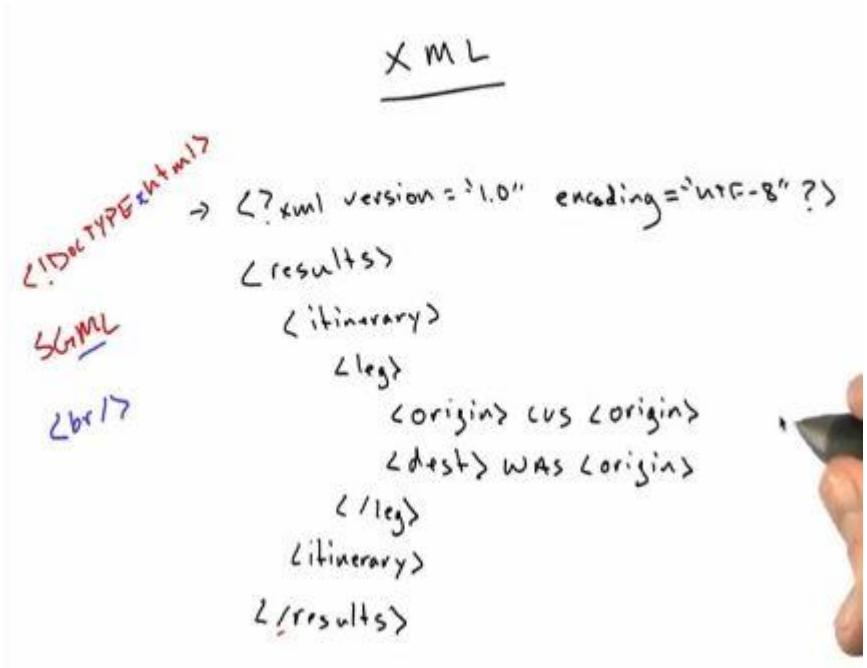
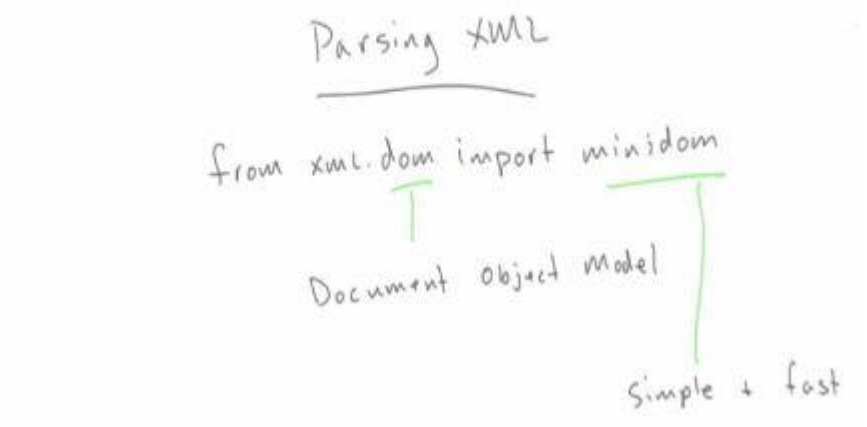
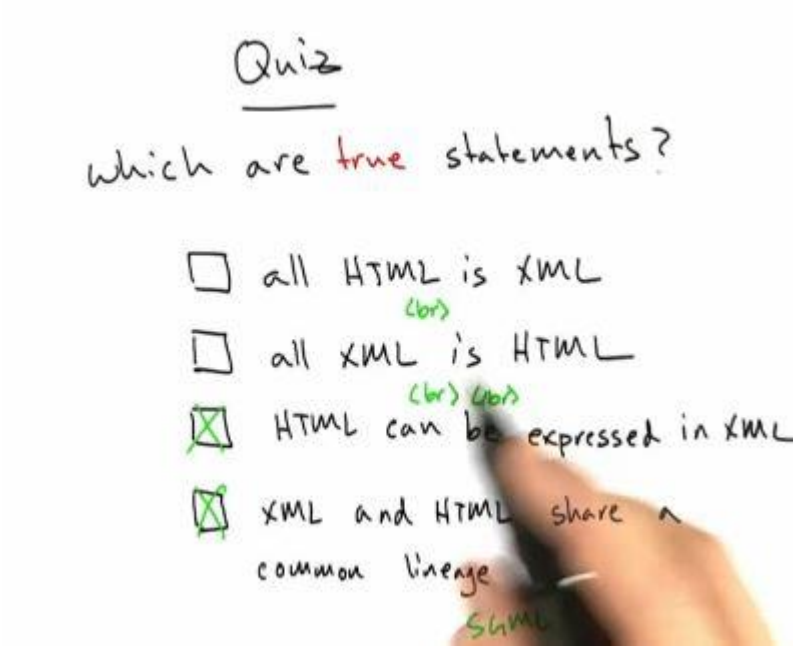
XML

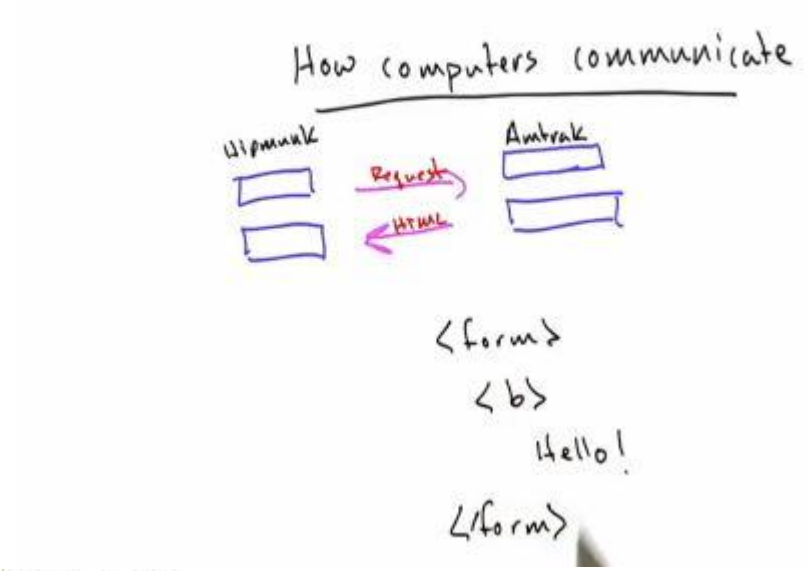


Parsing XML





XML



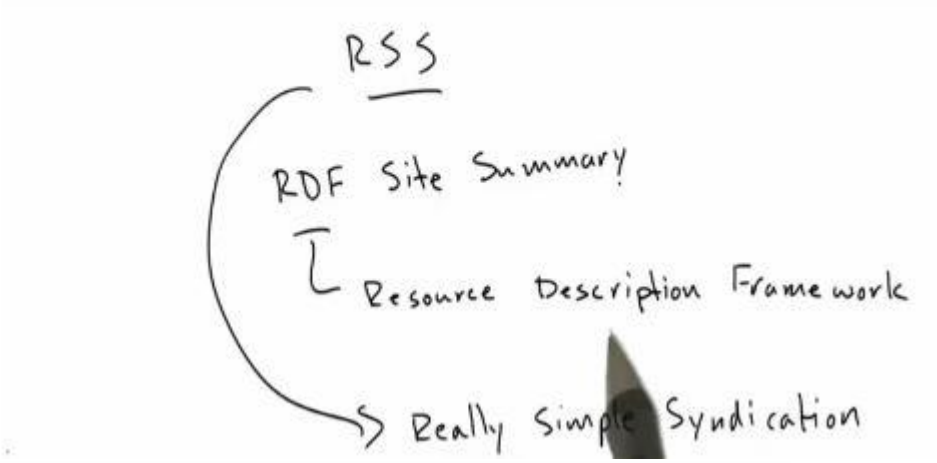

```
In [1]: import urllib2

In [2]: import urllib

In [3]:

In [3]: p = urllib2.urlopen("http://www.google.com")

In [4]: p
Out[4]: <addinfourl at 4423695464 whose fp = <socket._fileobject object at 0x107a8505@>>

In [5]: c = p.read()

In [6]: c
```

```
In [3]: p = urllib2.urlopen("http://www.google.com")

In [4]: p
Out[4]: <addinfourl at 4423695464 whose fp = <socket._fileobject object at 0x107a8505@>>
```

```
In [12]: p.url
Out[12]: 'http://www.google.com'

In [13]: p.headers
Out[13]: <httplib.HTTPMessage instance at 0x107ac3ab8>

In [14]: p.headers.items()
Out[14]:
[('x-xss-protection', '1; mode=block'),
 ('set-cookie',
  'PREF=ID=cc75876badc59c23:FF=0:TM=1335376838:LM=1335376838:S=bOFnPUY9merOhDs8; expires=Fri, 25-Apr-2014 18:00:38
GMT; path=/; domain=.google.com, NID=59=H8jmgYTb6JfkcQoHDcieGXe6iybzu1DrS4aH01JRhkJmeeLMbaC6ZhErYZeVKzESndAFv3YtMMk
AiTTHjS7TFUPXJGnQ4DJvnkn9DuSQXi63bfvnCRTIMO6lxY0XaNKj; expires=Thu, 25-Oct-2012 18:00:38 GMT; path=/; domain=.googl
e.com; HttpOnly'),
 ('expires', '-1'),
 ('server', 'gws'),
 ('connection', 'close'),
 ('cache-control', 'private, max-age=0'),
 ('date', 'Wed, 25 Apr 2012 18:00:38 GMT'),
 ('p3p',
  'CP="This is not a P3P policy! See http://www.google.com/support/accounts/bin/answer.py?hl=en&answer=151657 for m
ore info."'),
 ('content-type', 'text/html; charset=ISO-8859-1'),
 ('x-frame-options', 'SAMEORIGIN')]

In [15]: p.headers['content-type']
Out[15]: 'text/html; charset=ISO-8859-1'
```

SOAP

RSS



Quiz

which of those are good habits to get into?

- [x] Sending proper user-agents
- [ ] writing custom protocols
- [ ] using SOAP
- [x] rate-limiting yourself
- [x] using common protocols and data formats

```python
import json

reddit_front = r"""{"kind": "Listing", "data": {"modhash": "", "children": [{"kind": "t3", "data": {"dor

# QUIZ - reddit_front is a JSON string of reddit's front page. Inside it is a
# list of links, each of which has an "ups" attribute. From this dataset, what
# is the total number of ups of all the links?
#
# Implement the function total_ups(), and make it return the total number of ups

def total_ups():
    j = json.loads(reddit_front)
    return sum(c['data']['ups'] for c in j['data']['children'])

print total_ups()
```

RUN

Quiz

How many <item> elements are at this URL?

http://www.nytimes.com/services/xml/rss/nyt/GlobalHome.xml

use urllib2 and minidom

16

getElementsByTagName()

what server does www.example.com use? (use the Server header) urllib2

Apache/2.2.3

BigIP

```
In [2]: from xml.dom import minidom

In [3]: minidom.parseString
Out[3]: <function xml.dom.minidom.parseString>

In [4]: minidom.parseString("<mytag>contents!<childrden><item>1</item><item>2</item></children></mytag>")
```

Escaping JSON in Python

```
blog — Python — 116×29
--> 360         obj, end = self.raw_decode(s, idx=_w(s, 0).end())
    361         end = _w(s, end).end()
    362         if end != len(s):

/System/Library/Frameworks/Python.framework/Versions/2.7/lib/python2.7/json/
    374         """
    375         try:
--> 376             obj, end = self.scan_once(s, idx)
    377         except StopIteration:
    378             raise ValueError("No JSON object could be decoded")

ValueError: Expecting , delimiter: line 1 column 17 (char 17)

In [23]: json.loads('{"story":"once u\\"on a time"}')
Out[23]: {u'story': u'once u"on a time'}

In [24]: json.loads(r'{"story":"once u\"on a time"}')
Out[24]: {u'story': u'once u"on a time'}

In [25]: json.dumps([1,2,3])
Out[25]: '[1, 2, 3]'

In [26]: json.dumps({"one": 1, "two": 2})
Out[26]: '{"two": 2, "one": 1}'

In [27]: json.dumps({"one": 1, "two": 'the main said, "cool!"'})
Out[27]: '{"two": "the main said, \\"cool!\\"", "one": 1}'
```

Quiz

what is the valid JSON for this Python datastructure?

{'blah': ['one", 2, 'th\"r"ee']}

{'blah': ['one", 2, 'th\"r\"ee"]}

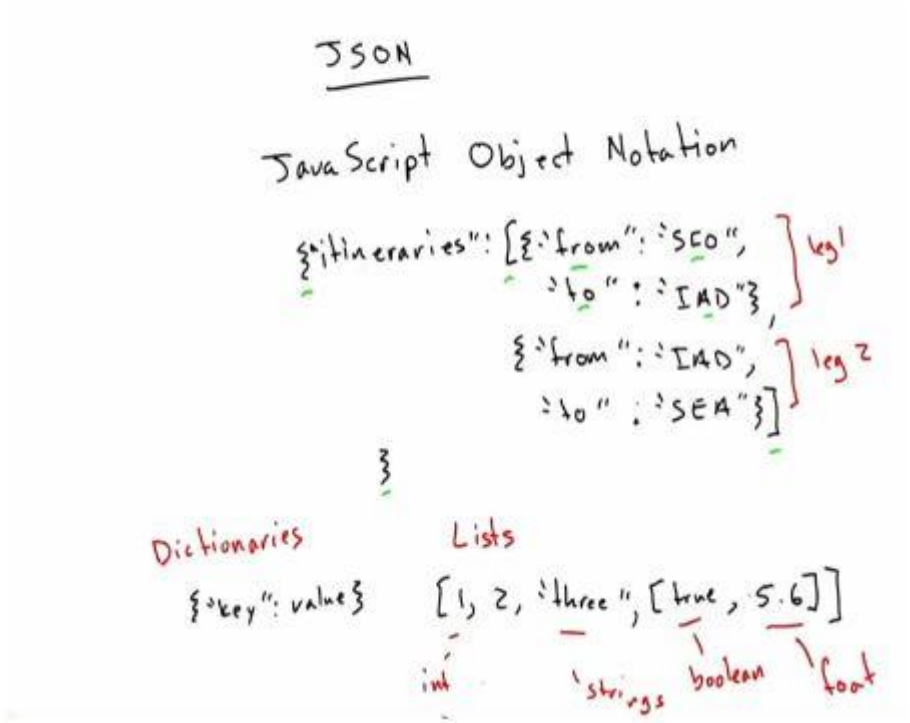
JSON Escaping

JSON Escaping

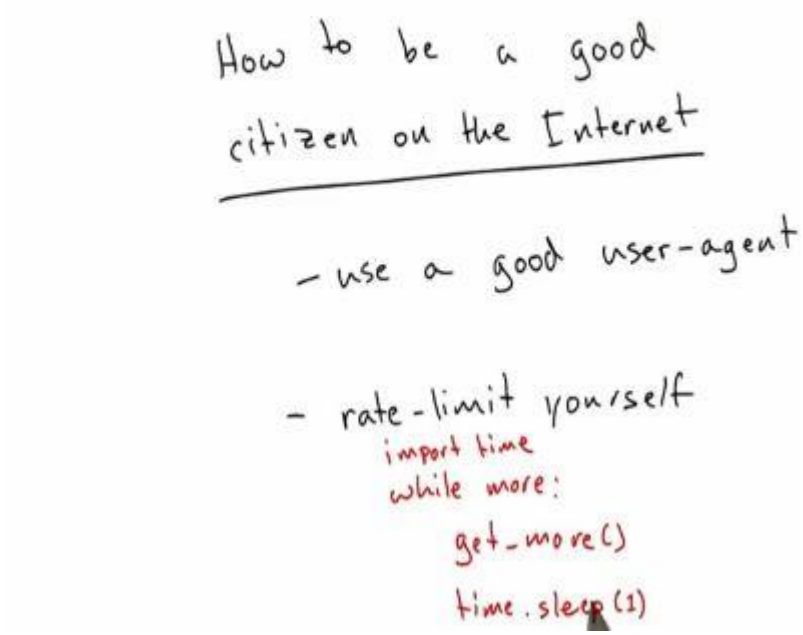
{'story': 'once u\"on a time"}

```
In [1]: import json

In [2]: j = '{"one": 1, "numbers": [1,2,3.5]}'

In [3]: json.loads(j)
Out[3]: {u'numbers': [1, 2, 3.5], u'one': 1}

In [4]: d = json.loads(j)

In [5]: d['numbers']
Out[5]: [1, 2, 3.5]

In [6]: d['one']
Out[6]: 1

In [7]: eval(j)
Out[7]: {'numbers': [1, 2, 3.5], 'one': 1}

In [8]:
```

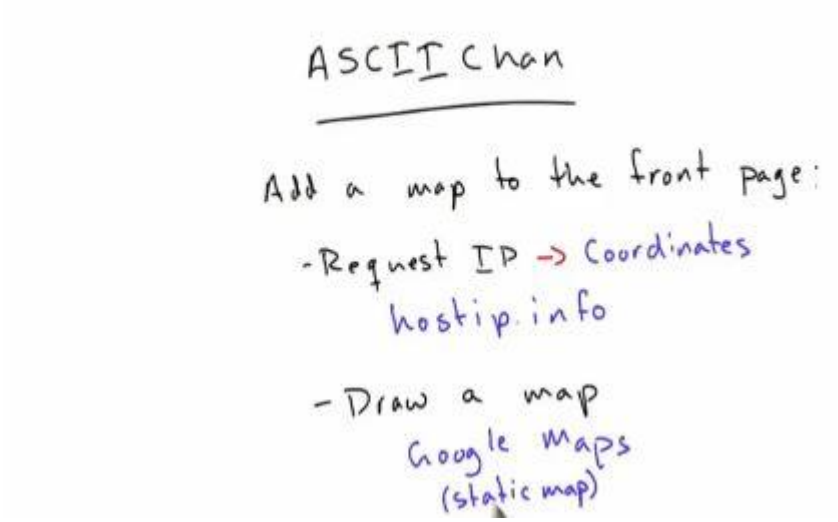
JSON



Being a Good Citizen



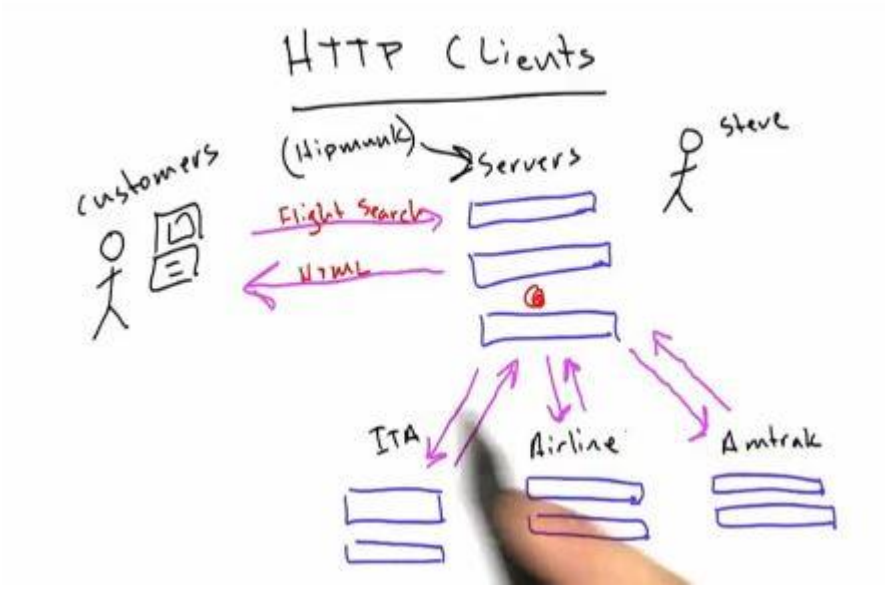
ASCII Chan 2

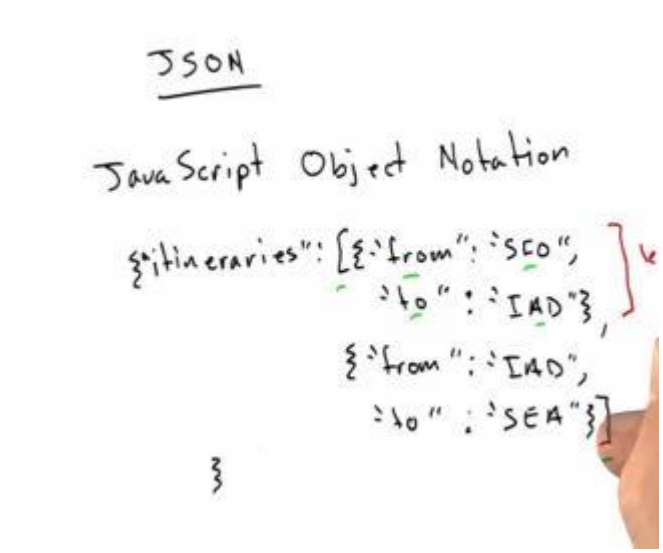


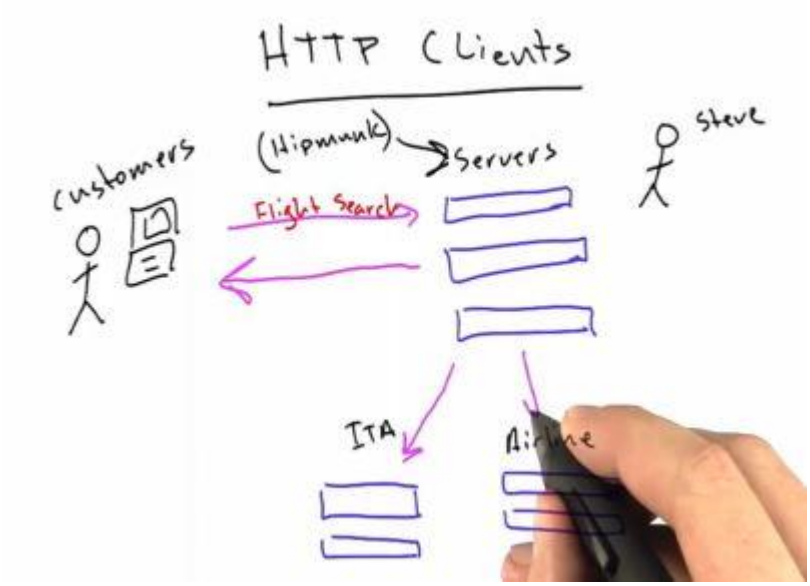
```python
from collections import namedtuple

# make a basic Point class
Point = namedtuple('Point', ["lat", "lon"])
points = [Point(1,2),
          Point(3,4),
          Point(5,6)]

# implement the function gmaps_img(points) that returns the google maps image
# for a map with the points passed in. A example valid response looks like
# this:
#
# http://maps.googleapis.com/maps/api/staticmap?size=380x263&sensor=false&marker=1,2&marker=3,4

GMAPS_URL = "http://maps.googleapis.com/maps/api/staticmap?size=380x263&sensor=false&"

def gmaps_img(points):
    markers = '&'.join('markers=%s,%s' % (p.lat, p.lon)
                       for p in points)
    return GMAPS_URL + markers

print gmaps_img([Point(100,200)])
```

RUN

```
http://maps.googleapis.com/maps/api/staticmap?
size=380x263&sensor=false&markers=1,2&markers=3,4&markers=5,6
```




JSON
JavaScript Object Notation
{"itineraries": [{"from": "SFO",
"to": "IAD"},
{"from": "IAD",
"to": "SEA"}]
}


HTTP Clients
customers
(Hipmunk)
Servers
Steve
Flight Search
ITA
Airline