

Reading Web Data From Python

TOTAL POINTS 13

1.Question 1

Which of the following Python data structures is most similar to the value returned in this line of Python:

```
x = urllib.request.urlopen('http://data.pr4e.org/romeo.txt')
```

- ☒ file handle
- ☐ list
- ☐ socket
- ☐ regular expression
- ☐ dictionary

1 point

2.Question 2

In this Python code, which line actually reads the data?

```
import socket
mysock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
mysock.connect(('data.pr4e.org', 80))
cmd = 'GET http://data.pr4e.org/romeo.txt HTTP/1.0\n\n'.encode()
mysock.send(cmd)
while True:
    data = mysock.recv(512)
    if (len(data) < 1):
        break
    print(data.decode())
mysock.close()
```

- ☒ mysock.recv()
- ☐ socket.socket()
- ☐ mysock.close()
- ☐ mysock.connect()
- ☐ mysock.send()

1 point

3.Question 3

Which of the following regular expressions would extract the URL from this line of HTML:

```
<p>Please click <a href="http://www.dr-chuck.com">here</a></p>
```

- ☒ href="(.)"
- ☐ href=".+"
- ☐ http://.*
- ☐ <.*>

1 point

4.Question 4

In this Python code, which line is most like the open() call to read a file:

```
import socket
mysock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
mysock.connect(('data.pr4e.org', 80))
cmd = 'GET http://data.pr4e.org/romeo.txt HTTP/1.0\n\n'.encode()
mysock.send(cmd)
while True:
    data = mysock.recv(512)
    if (len(data) < 1):
        break
    print(data.decode())
mysock.close()
```

- ☒ mysock.connect()
- ☐ import socket
- ☐ mysock.recv()
- ☐ mysock.send()
- ☐ socket.socket()

1 point

5.Question 5

Which HTTP header tells the browser the kind of document that is being returned?

- ☐ Document-Type:
- ☒ Content-Type:
- ☐ Metadata:
- ☐ HTML-Document:
- ☐ ETag:

1 point

6.Question 6

What should you check before scraping a web site?

- ☐ That the web site returns HTML for all pages
- ☐ That the web site supports the HTTP GET command
- ☐ That the web site only has links within the same site
- ☒ That the web site allows scraping

1 point

7.Question 7

What is the purpose of the BeautifulSoup Python library?

- ☐ It optimizes files that are retrieved many times
- ☐ It builds word clouds from web pages
- ☒ It repairs and parses HTML to make it easier for a program to understand
- ☐ It allows a web site to choose an attractive skin
- ☐ It animates web operations to make them more attractive

1 point

8.Question 8

What ends up in the "x" variable in the following code:

```
html = urllib.request.urlopen(url).read()
soup = BeautifulSoup(html, 'html.parser')
x = soup('a')
```

- ☒ A list of all the anchor tags (<a..) in the HTML from the URL
- ☐ True if there were any anchor tags in the HTML from the URL
- ☐ All of the externally linked CSS files in the HTML from the URL
- ☐ All of the paragraphs of the HTML from the URL

1 point

9.Question 9

What is the most common Unicode encoding when moving data between systems?

- ☐ UTF-16
- ☐ UTF-128
- ☐ UTF-32
- ☒ UTF-8
- ☐ UTF-64

1 point

10.Question 10

What is the decimal (Base-10) numeric value for the upper case letter "G" in the ASCII character set?

- ☒ 71
- ☐ 7
- ☐ 103
- ☐ 25073
- ☐ 14

(Hint: try in python `print(ord('G'))`)

1 point

11.Question 11

What word does the following sequence of numbers represent in ASCII:

108, 105, 110, 101

- ☐ lost

- ☐ ping
- ☐ tree
- ☒ line
- ☐ func

(Hint: In python try `print(ord('l'))` and so on to get full answer)

1 point

12.Question 12

How are strings stored internally in Python 3?

- ☐ UTF-8
- ☐ EBCDIC
- ☒ Unicode
- ☐ ASCII
- ☐ Byte Code

1 point

13.Question 13

When reading data across the network (i.e. from a URL) in Python 3, what method must be used to convert it to the internal format used by strings?

- ☐ `trim()`
- ☐ `find()`
- ☒ `decode()`
- ☐ `encode()`
- ☐ `upper()`

1 point