6.009
Fundamentals of Programming

Interlude:
Python, Sockets, and the Web

Adam Hartz
hz@mit.edu

CAT-SOOP

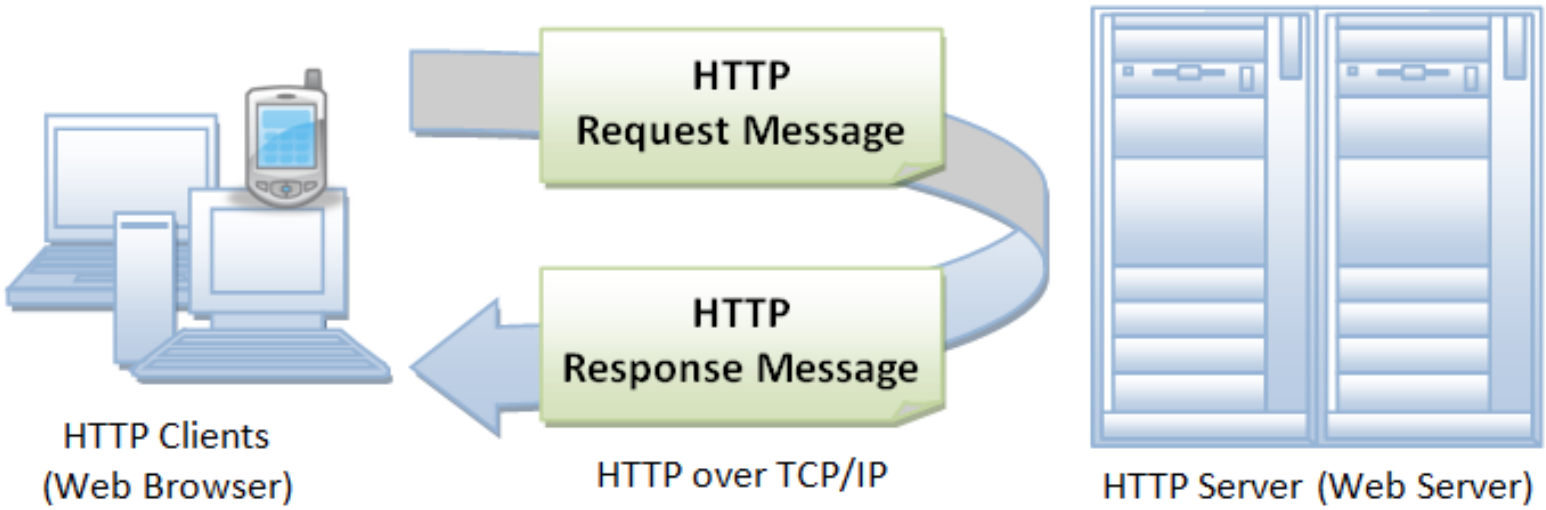
An "automatic tutor": collect and assess online exercises
Written in Python, 2011-now, rewritten completely multiple times

Based on a program called "the tutor" by
Tomás Lozano-Pérez (written in Scheme!)

CAT-SOOP is a fairly standard web application. What does it do?

- Receive request from user
  ("show me a page," "submit this answer," etc)
- Find relevant information
  (page content, user info, history of submissions, etc)
- Log new information if necessary, and
- Send response
  (typically, HTML to be displayed in the browser)

The World Wide Web



https://www.ntu.edu.sg/home/ehchua/programming/webprogramming/HTTP_Basics.html

CAT-SOOP

```
`255` white and vice versa).  For example, here is a photograph of Adam H's cat.  On
the left side is the original image, and on the right is an inverted version.

<center>
![Stronger the cat](CURRENT/stronger.jpg)
</center>


Most of the implementation of the inversion filter has been completed for you
(it is invoked by calling the method called `inverted`), but some pieces have
not been implemented correctly.  Your task in this part of the lab is to fix
the implementation of the inversion filter.

Before you do that, however, let's add a simple test case so that we can test
whether our code is working.

Let's start with a 1x4 image that is defined with the following parameters:

<python>
tutor.init_random()
img1_lo = cs_random.randint(0, 32)
img1_mid = cs_random.randint(64, 96)
img1_mid2 = cs_random.randint(128, 160)
img1_high = cs_random.randint(192, 224)
img1_pixels = [img1_lo, img1_mid, img1_mid2, img1_high]
</python>

* height: `1`
* width: `4`
* pixels: `@{img1_pixels}`


<question pythonliteral>
csq_prompt='''
If we were to run this image through a working inversion filter, what would the
expected output be?  In the box below, enter a Python list representing the
expected `pixels` key in the resulting image:<br/>
'''

csq_soln = [255-i for i in img1_pixels]
def csq_check_function(sub, sol):
    if not(isinstance(sub, list)):
        return (0.0, 'Please enter a Python list.')
    if len(sub) != len(sol):
        return (0.0, 'Double-check the length of the list you provided.')
    elif not all(type(i)==type(j) and i==j for i,j in zip(sub, sol)):
        return (0.0, 'Make sure that all of the elements in the given list are integers.')
    return 1.0
</question>

<subsection>Adding a Test Case</subsection>

Let's also add this test case to the lab's regular tests so that it is run when we
execute `test.py`.  If you open `test.py` in a text editor, you will see that
it is a Python file that makes use of Python's [`unittest`
module](https://docs.python.org/3/library/unittest.html) for unit testing.

Each class in this file serves as a collection of test cases, and each method
within a class represents a particular test case.

Running `test.py` will cause Python to run and report on _all_ of the tests in
the file.  However, you can make Python run only a subset of the tests by
running, for example:

```
:nohl
```

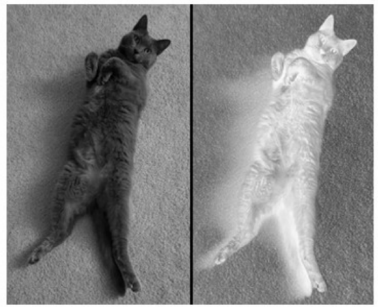
10 Haste The Day - Su | 6.009 Fall 2017

https://eecs6009.mit.edu/fall17/labs/lab1

6.009 Home / Lab 1: Image Processing Homepage Information Quizzes Labs Help Log Out

3) IMAGE FILTERING VIA PER-PIXEL TRANSFORMATIONS

As our first task in manipulating images, we will look at an *inversion* filter, which reflects pixels about the middle gray value (`0` black becomes `255` white and vice versa). For example, here is a photograph of Adam H's cat. On the left side is the original image, and on the right is an inverted version.



Most of the implementation of the inversion filter has been completed for you (it is invoked by calling the method called `inverted`), but some pieces have not been implemented correctly. Your task in this part of the lab is to fix the implementation of the inversion filter.

Before you do that, however, let's add a simple test case so that we can test whether our code is working.

Let's start with a 1x4 image that is defined with the following parameters:

- height: `1`
- width: `4`
- pixels: `[12, 70, 154, 222]`

If we were to run this image through a working inversion filter, what would the expected output be? In the box below, enter a Python list representing the expected `pixels` key in the resulting image:

Save | Submit | View Answer

As staff, you are always allowed to submit. If you were a student, you would see the following:

You have submitted this assignment 0 times.

3.1) Adding a Test Case
```

What Is A Server?

What Is A Server?

Sockets

Sockets allow communication across processes (on the same machine or different machines).

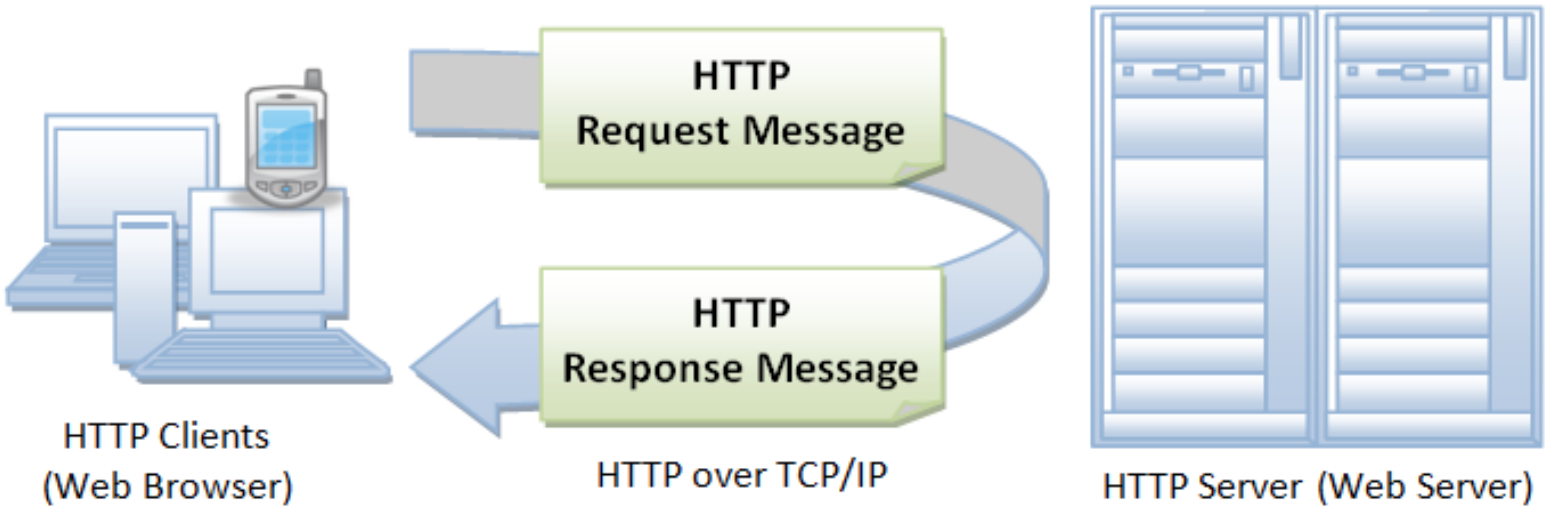
Typically, a *server* will wait for a *client* to make a connection on a designated *port* (a virtual endpoint for a connection).

Once the client connects, the socket allows for communication between the server and the client.

Client and server can each send/receive data via the socket.

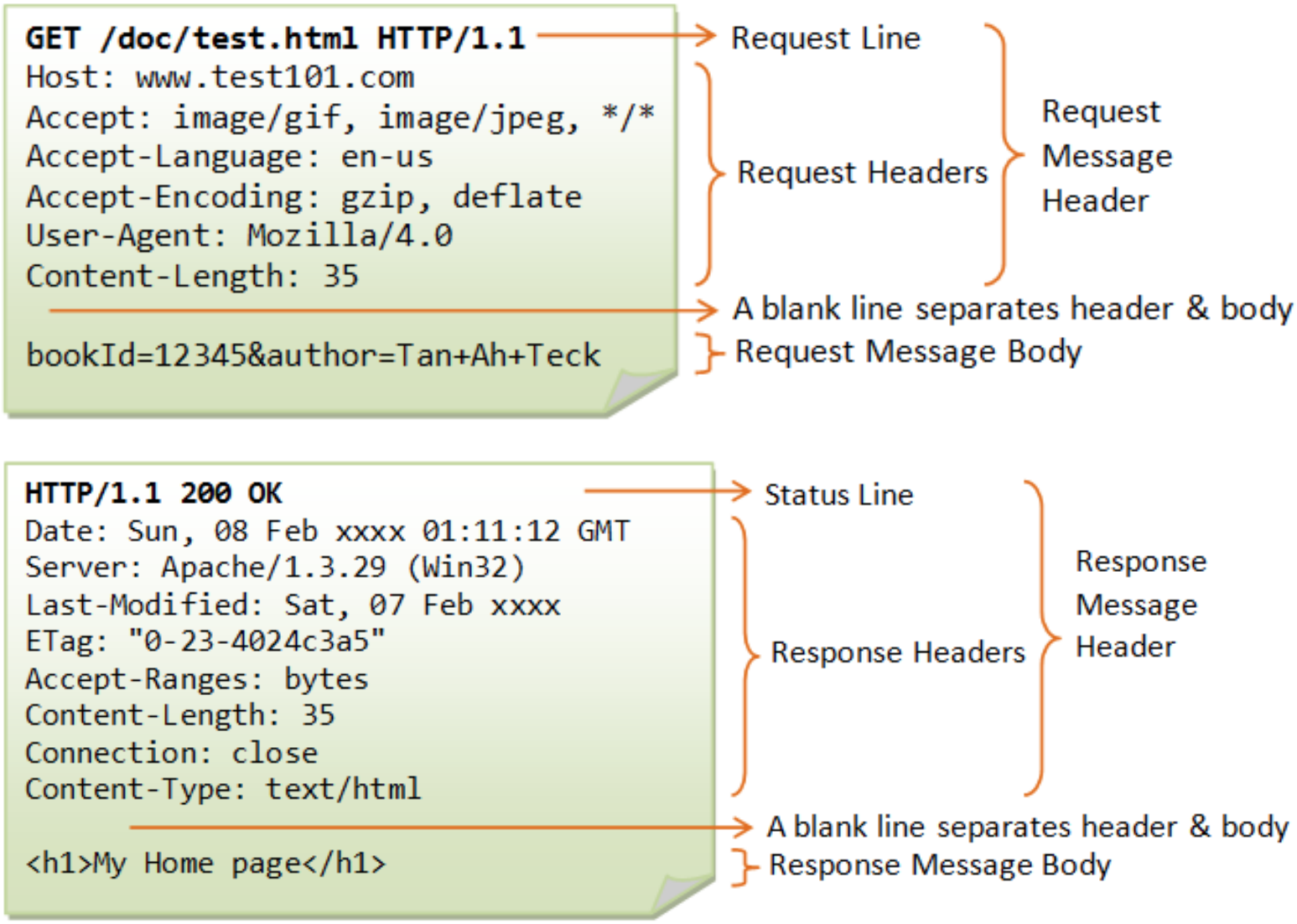
Example: yelling echo server.

The World Wide Web



https://www.ntu.edu.sg/home/ehchua/programming/webprogramming/HTTP_Basics.html

HTTP Request and Response



https://www.ntu.edu.sg/home/ehchua/programming/webprogramming/HTTP_Basics.html

The Rest of Today

Designing a:

- Web Server
- Web Application Framework
- Web Application