CCT396, Fall 2011

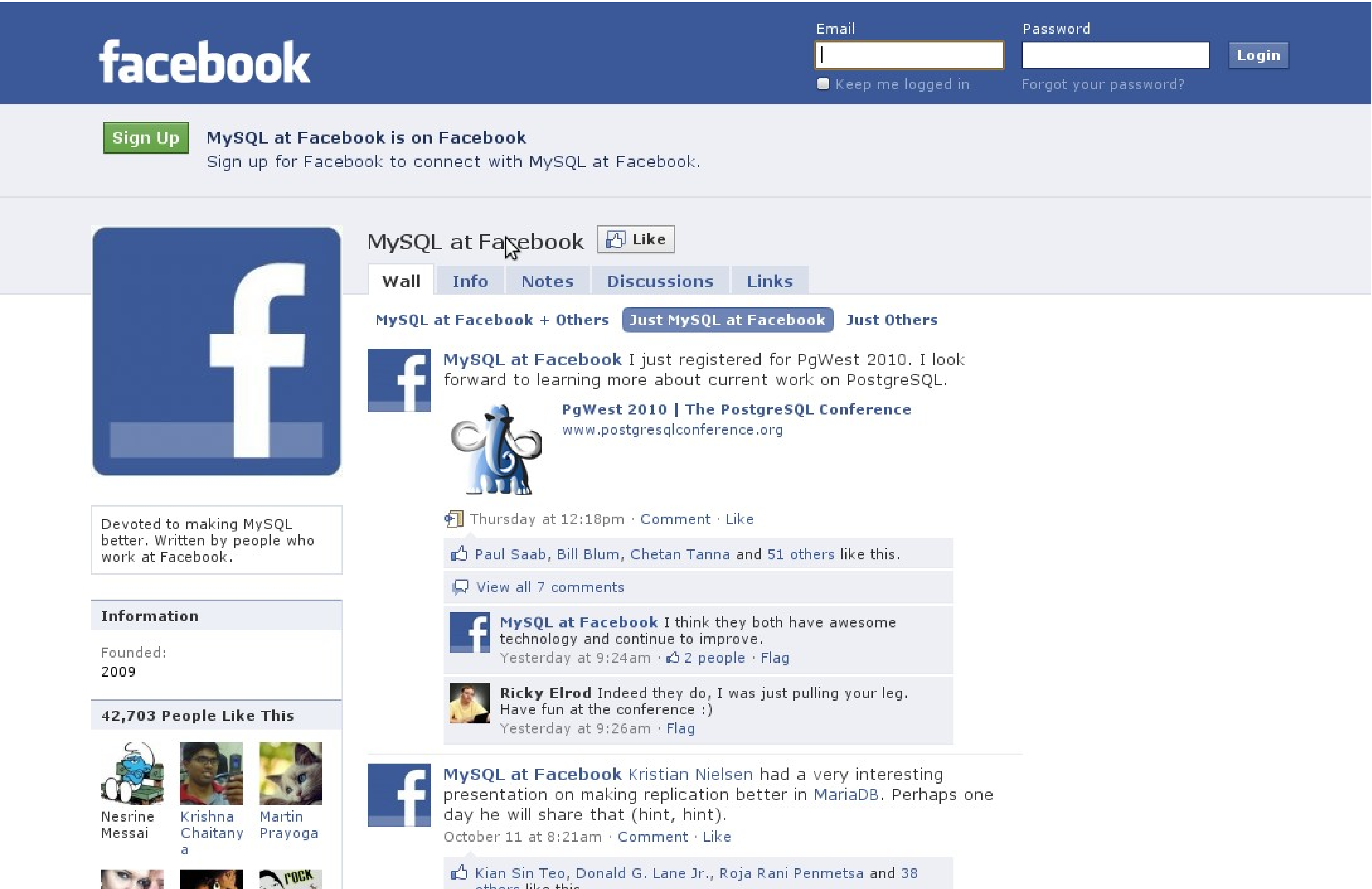
Database Design and Implementation

Yuri Takhteyev
University of Toronto

Week 8

Embedded SQL

Facebook runs on MySQL



MySQL at Facebook is on Facebook
Sign up for Facebook to connect with MySQL at Facebook.

MySQL at Facebook

Devoted to making MySQL better. Written by people who work at Facebook.

Information

Founded:
2009

42,703 People Like This

Nesrine Messai
Krishna Chaitanya
Martin Prayoga

MySQL at Facebook I just registered for PgWest 2010. I look forward to learning more about current work on PostgreSQL.

PgWest 2010 | The PostgreSQL Conference
www.postgresqlconference.org

Thursday at 12:18pm · Comment · Like

Paul Saab, Bill Blum, Chetan Tanna and 51 others like this.

View all 7 comments

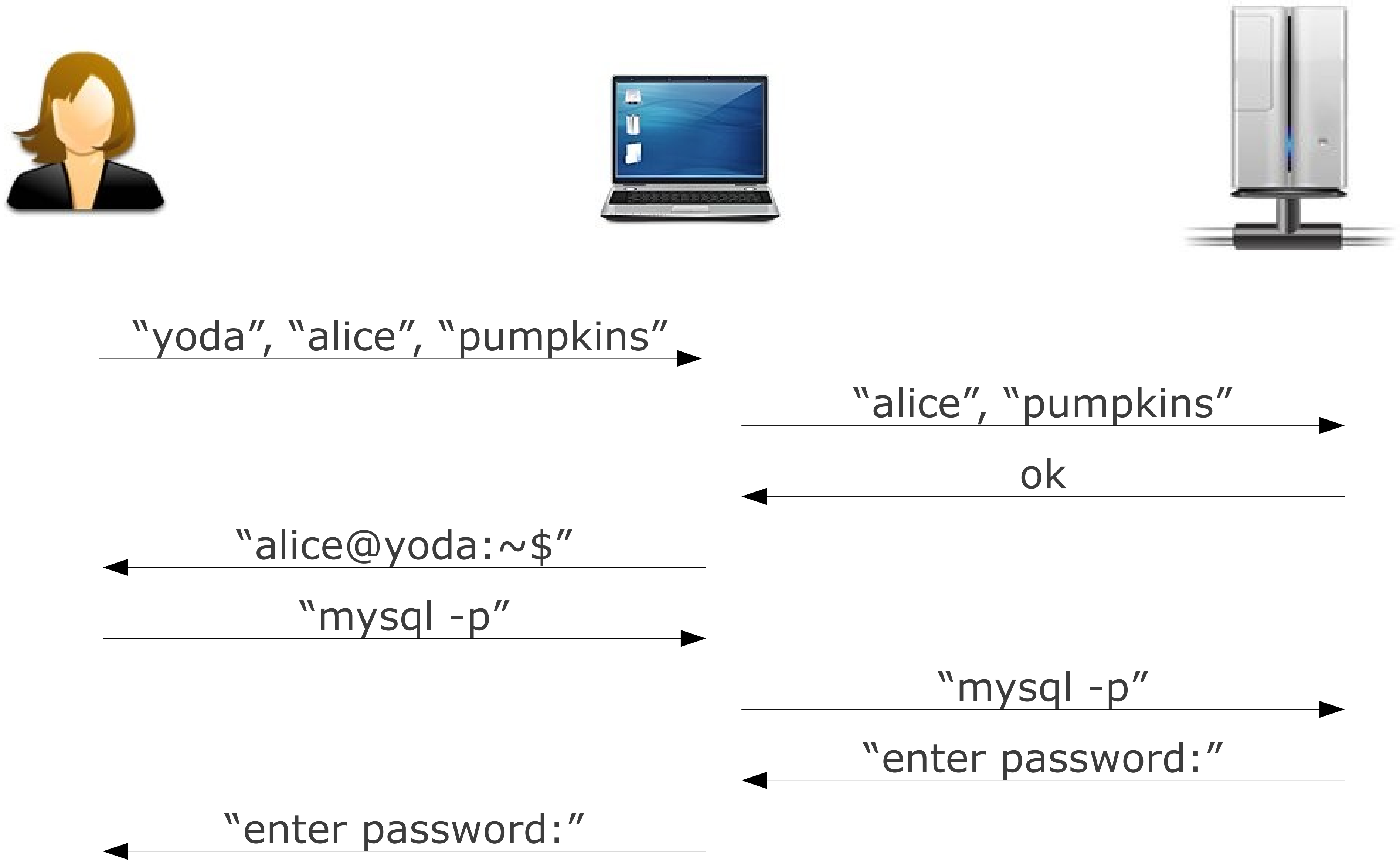
MySQL at Facebook I think they both have awesome technology and continue to improve.
Yesterday at 9:24am · 2 people · Flag

Ricky Elrod Indeed they do, I was just pulling your leg. Have fun at the conference :)
Yesterday at 9:26am · Flag

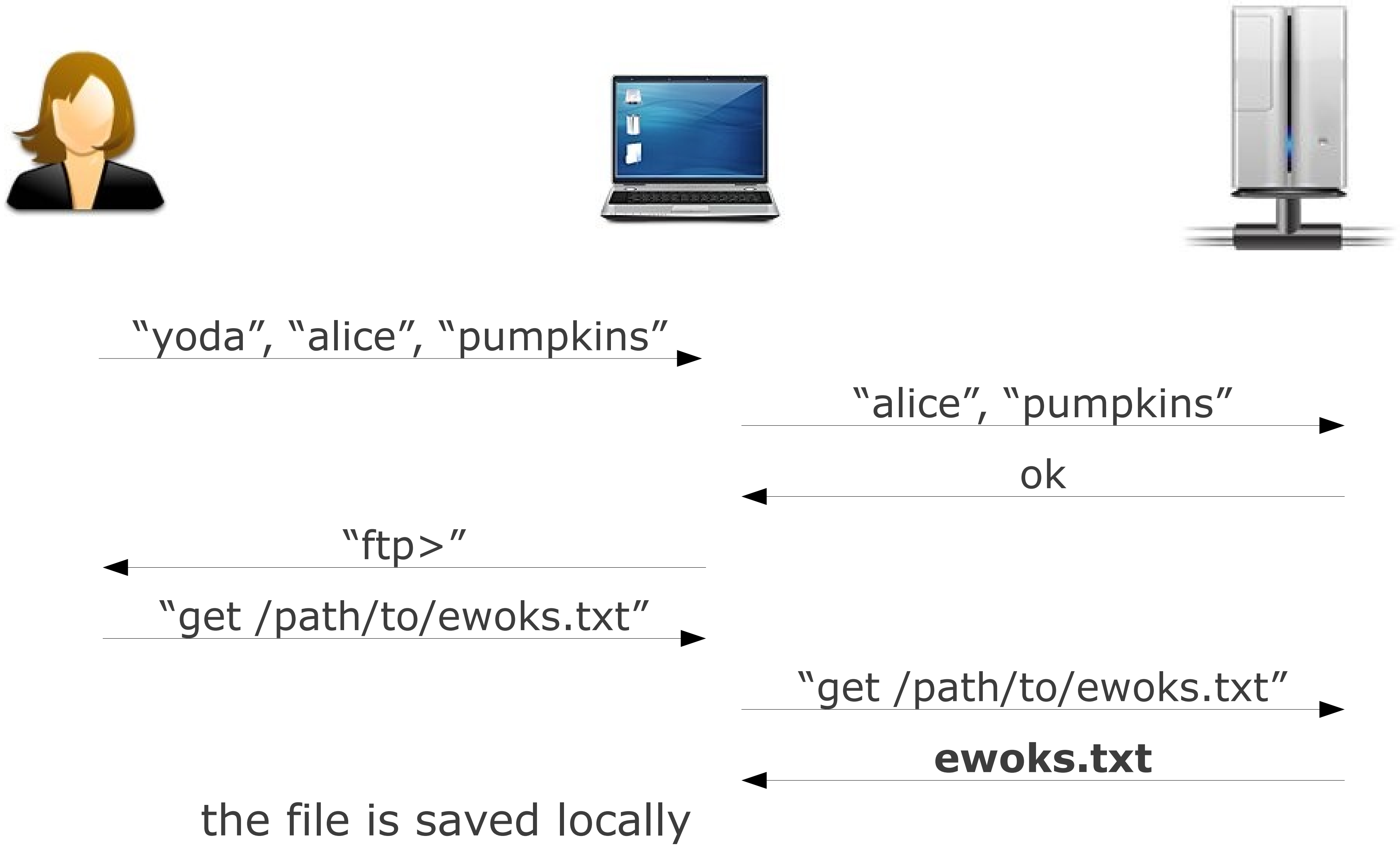
MySQL at Facebook Kristian Nielsen had a very interesting presentation on making replication better in MariaDB. Perhaps one day he will share that (hint, hint).
October 11 at 8:21am · Comment · Like

Kian Sin Teo, Donald G. Lane Jr., Roja Rani Penmetsa and 38

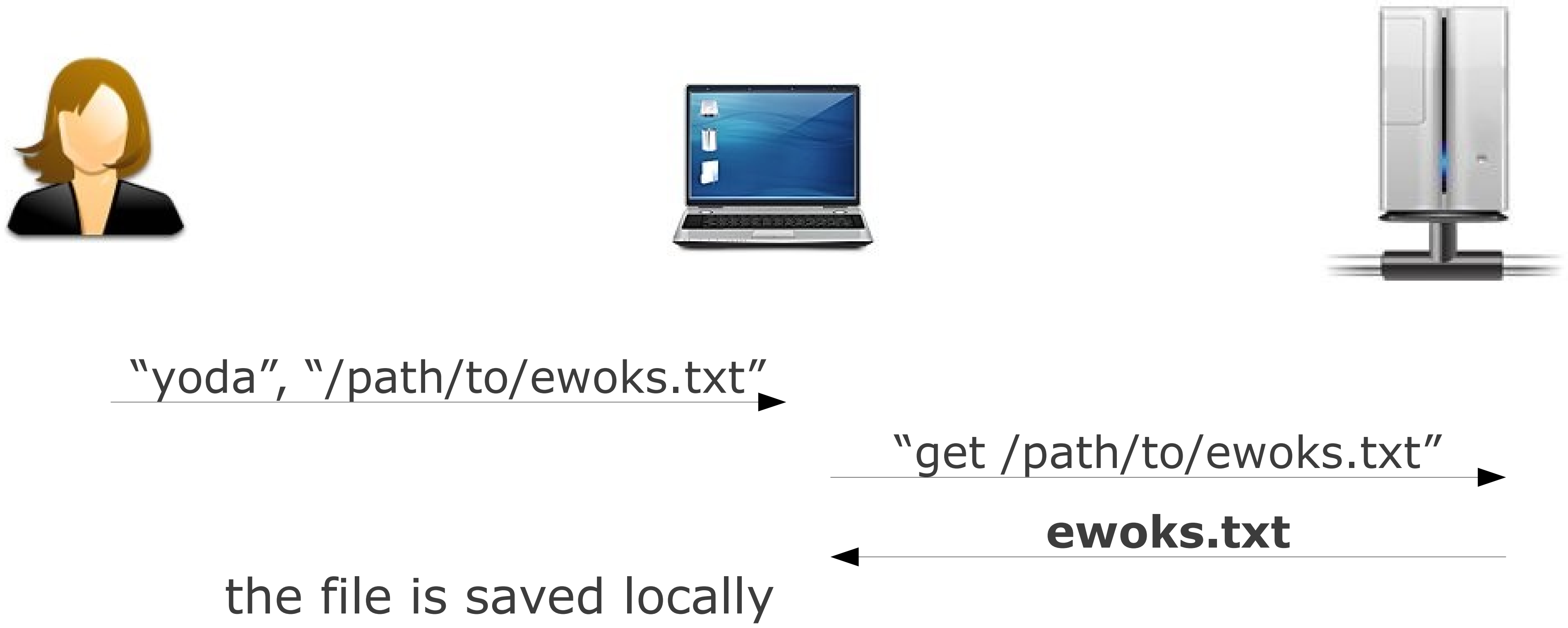
Telnet (1969)


"yoda", "alice", "pumpkins"
"alice", "pumpkins"
ok
"alice@yoda:~$"
"mysql -p"
"mysql -p"
"enter password:"
"enter password:"

FTP (1971)

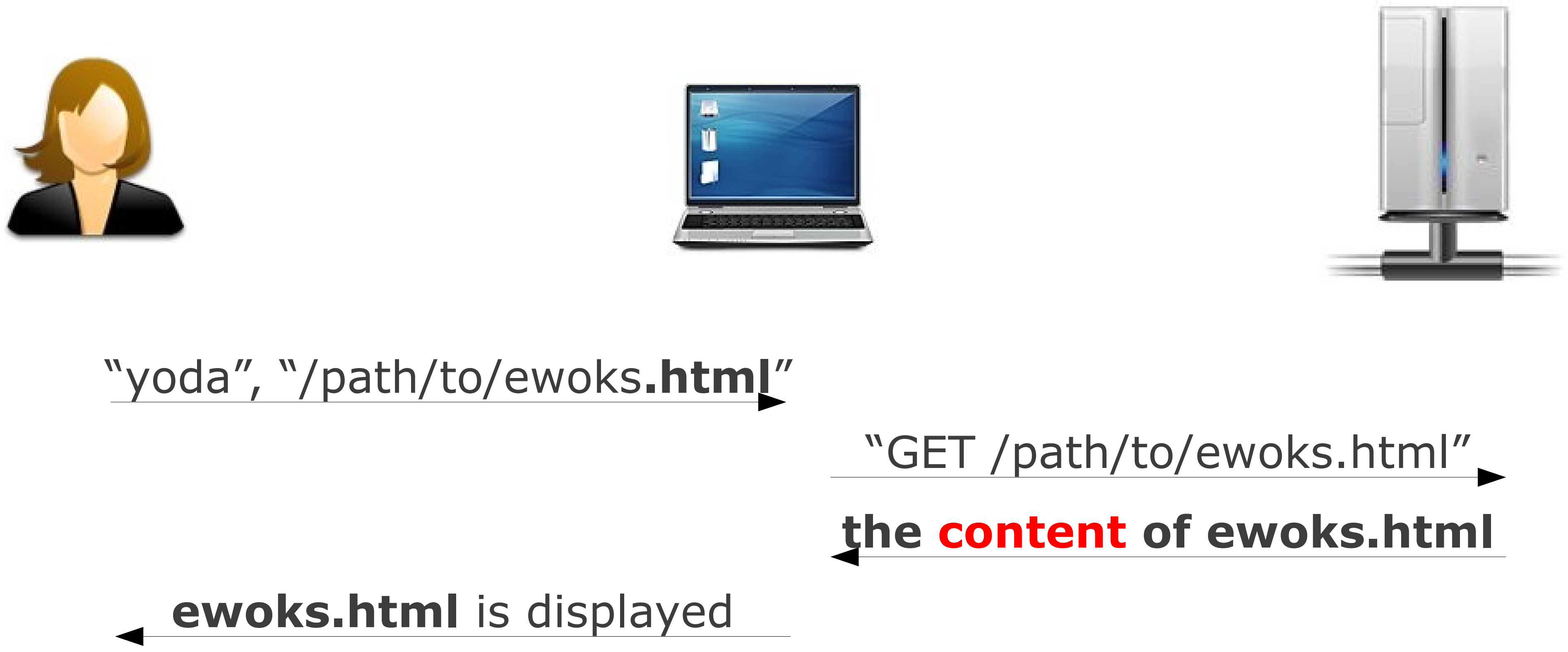

"yoda", "alice", "pumpkins"
"alice", "pumpkins"
ok
"ftp>"
"get /path/to/ewoks.txt"
"get /path/to/ewoks.txt"
ewoks.txt
the file is saved locally

Anonymous FTP



ftp**://**yoda**/**path/to/ewoks.txt

HTTP (1991)



“yoda”, “/path/to/ewoks**.html**”

“GET /path/to/ewoks.html”

the content of ewoks.html

ewoks.html is displayed

http**://**yoda**/**path/to/ewoks.html

HTML

```
<html xmlns="http://www.w3.org/1999/xhtml">
 <head>
  <title>
   Ewoks
  </title>
 </head>
 <body>
  <h1>
   The List of Ewoks
  </h1>
  ...
```

See: http://www.w3schools.com/html/default.asp

On ~~Yoda~~ ZZ9

1. Put the file in ~/public_html/

2. Let the server see it:

```
chmod a+r ewoks.html
```

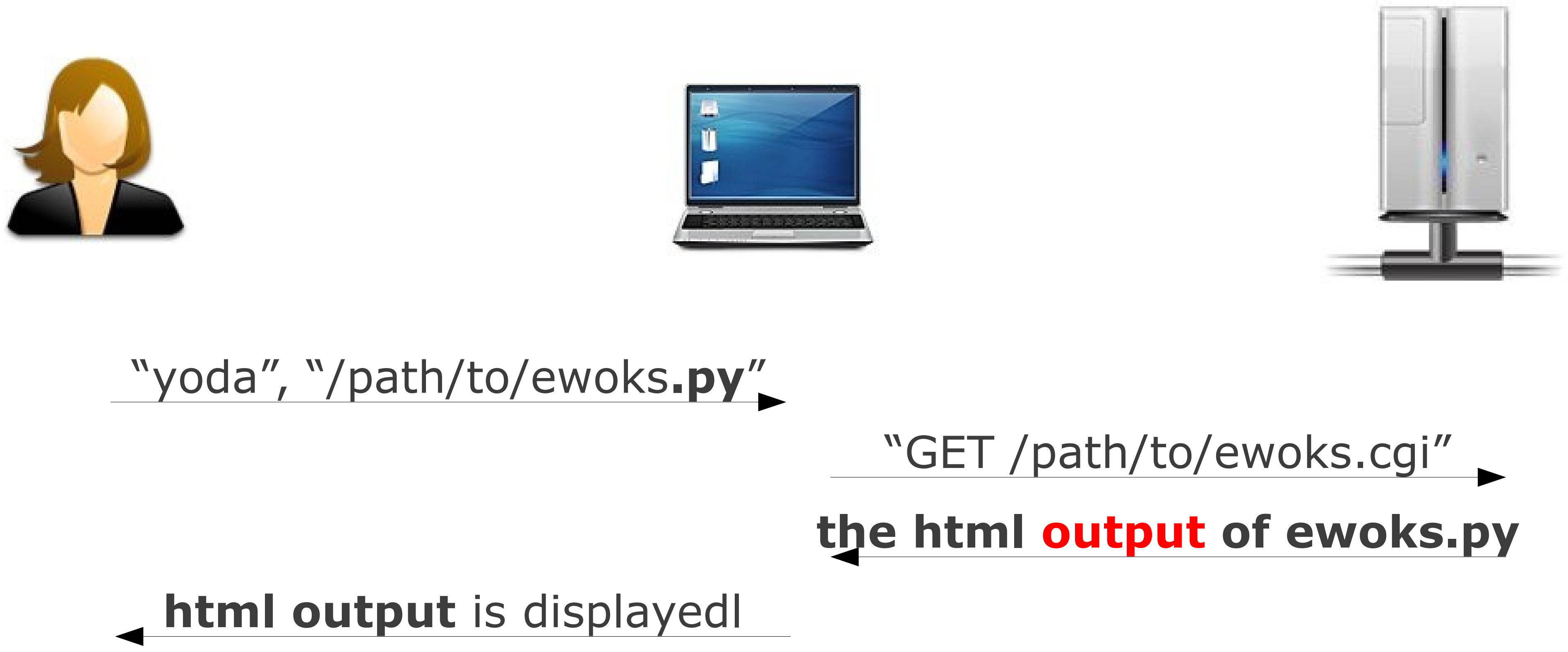
Locally:

```
scp ewoks.html okenobi@zz9.ischool.utoronto.ca:~/public_html/
```

Remotely:

```
cd ~/public_html/
chmod 644 ewoks.html
```

CGI


"yoda", "/path/to/ewoks.**py**"
"GET /path/to/ewoks.cgi"
the html output of ewoks.py
html output is displayedl

ewoks.cgi contains **code**, but **returns** HTML

ewoks.py

$$\underbrace{\texttt{http:}}_{\text{protocol}}\texttt{//}\underbrace{\texttt{zz9.ischool.utoronto.ca}}_{\text{server address}}\texttt{/}\overbrace{\texttt{~okenobi}}^{\text{user name}}\texttt{/}\underbrace{\texttt{dynamic/ewoks.py}}_{\text{path}}$$

ewoks.py

```
#!/usr/bin/python

print "Content-Type: text/html"
print
print "<h1>Ewoks</h1>"
```

Set permissions to a+rx

Remotely:

```
cd ~/public_html/dynamic/
cp /play/ewoks.py .
chmod 700 ewoks.py
```

Python Console

On ZZ9:

type “python”

Locally:

Install from http://python.org/

Use Python **2.x**, not Python **3** (“2000”).

Elements of Python

Constants and Expressions:

```
5
"Ewoks"
```

Variables:

```
count = 5
count = count + 1
species = "Ewoks"
"Hello, " + species + "!"
```

Elements of Python

Print:

```
print count
print species, count
```

Functions:

```
print str(count)
print str(count)+" "+species
```

Python is “Procedural”

(We describe the steps rather than the results.)

CGI with Padawan

```
#!/usr/bin/padawan

print "Content-Type: text/html"
print

species = "Ewoks"
count = 5
print "<h1>", species, "</h1>"
print count, species, "<br/>"
count = count + 1
print count, species, "<br/>"
```

Always the Same?

Parameters
- additional data from the user

Database
- information that we store

Using the DB

```
connect_to_db("starwars", "okenobi")
query = "select * from persona"
execute_query(query)
print get_row_count()
```

Padawan gives you a subset of the DB functionality. For more options you can try Python's module MySQLdb (this is what Padawan uses behind the Scenes).

Getting Values

```
connect_to_db("starwars", "okenobi")
query = "select * from persona"
execute_query(query)
print get_row_count()

fetch_all_rows()
row = get_row(0)
print get_row_value(row, "species")
```

Loops

```
connect_to_db("starwars", "okenobi")
query = "select * from persona"
execute_query(query)
print get_row_count()

for row in fetch_all_rows() :
    print get_row_value(row, "species")
```

$\underbrace{\qquad\qquad}$

4 spaces. Do **not** use tabs.

Longer Loops

```
connect_to_db("starwars", "okenobi")
query = "select * from persona"
execute_query(query)
print get_row_count()

for row in fetch_all_rows() :
    print get_row_value(row, "species")
    print ","
print "done!"
```

Formatting a Table

```
print "<table>"
for row in fetch_all_rows() :
    print "<tr>"
    print "<td>"
    print get_row_value(row, "name")
    print "</td>"
    print "<td>"
    print get_row_value(row, "species")
    print "</td>"
    print "</tr>"
Print "</table>"
```

Conditions

```
for row in fetch_all_rows() :
    name = get_row_value(row, "name")
    if name=="Jabba" :
        print "<tr>"
        species = get_row_value(row,
                      "species")
        print "He is a " + species + "!"
```

Parameters

- additional data from the user

Database

- information that we store

Passing Parameters

Option 1, “GET” (in the URL)

http://zz9.ischool.utoronto.ca/~okenobi/dynamic/personas.py?species=Human

.../personas.py?species=Human

Option 2, “POST” (invisibly)

(we'll get back to this)

Accessing Parameters

```
print_form()

if form_has_field("species") :
    print "Species is defined!"
else :
    print "Species is not defined. :("

s = get_form_field_value("species")
print "The species is " + s
```

Dynamic SQL

```
s = get_form_field_value("species")
print "The species is " + s
```

Dynamic SQL (Wrong)

```
if form_has_field("species") :
    s = get_form_field_value("species")
    query = ( "select * from persona"
            + " where species="
            + "'" + s + "'" )
    print query
    execute_query(query);
    count = get_row_count()
    print "Got ", count, "results!"
```

Dynamic SQL (Wrong)

```
rows = fetch_all_rows()
for r in rows :
   n = get_row_value(r, "name")
   print n, ","
```

Dynamic SQL (Wrong)

```
if form_has_field("species") :
    s = get_form_field_value("species")
    query = ( "select * from persona"
              + " where species="
              + "'" + s + "'" )
    print query
    execute_query(query);
    count = get_row_count()
    print "Got ", count, "results!"
```

Never do this!

More Elegant

Still wrong!

```
template = """
select * from persona
where species='%s';
"""
if form_has_field("species") :
    s = get_form_field_value("species")
    query = fill_template(template, s)
    print query
    execute_query(query);
    count = get_row_count()
    print "Got ", count, "results!"
```

You can use python’s “%” operator instead of fill_template().

Here Is Why

http://zz9.ischool.utoronto.ca/~okenobi/dynamic/ewoks2.py?species=Human'%3Binsert+into+persona+(name,species)+values+('X','Human')%3Bselect+*+from+persona+where+'

Right

```
template = """
select * from persona
where species='%s';
"""
if form_has_field("species") :
    s = get_form_field_value("species")
    safe_s = escape_string(s)
    query = fill_template(template,
                          safe_s)
    print query
    execute_query(query);
    count = get_row_count()
    print "Got ", count, "results!"
```

Modifying Data

```
template = """
insert into films (title, director)
values ('%s','%s');
"""
t = get_form_field_value("title")
t = escape_string(t)
d = get_form_field_value("directory")
d = escape_string(d)
query = fill_template(template, t, d)
print query
execute_query(query)
commit()
```

HTML Forms

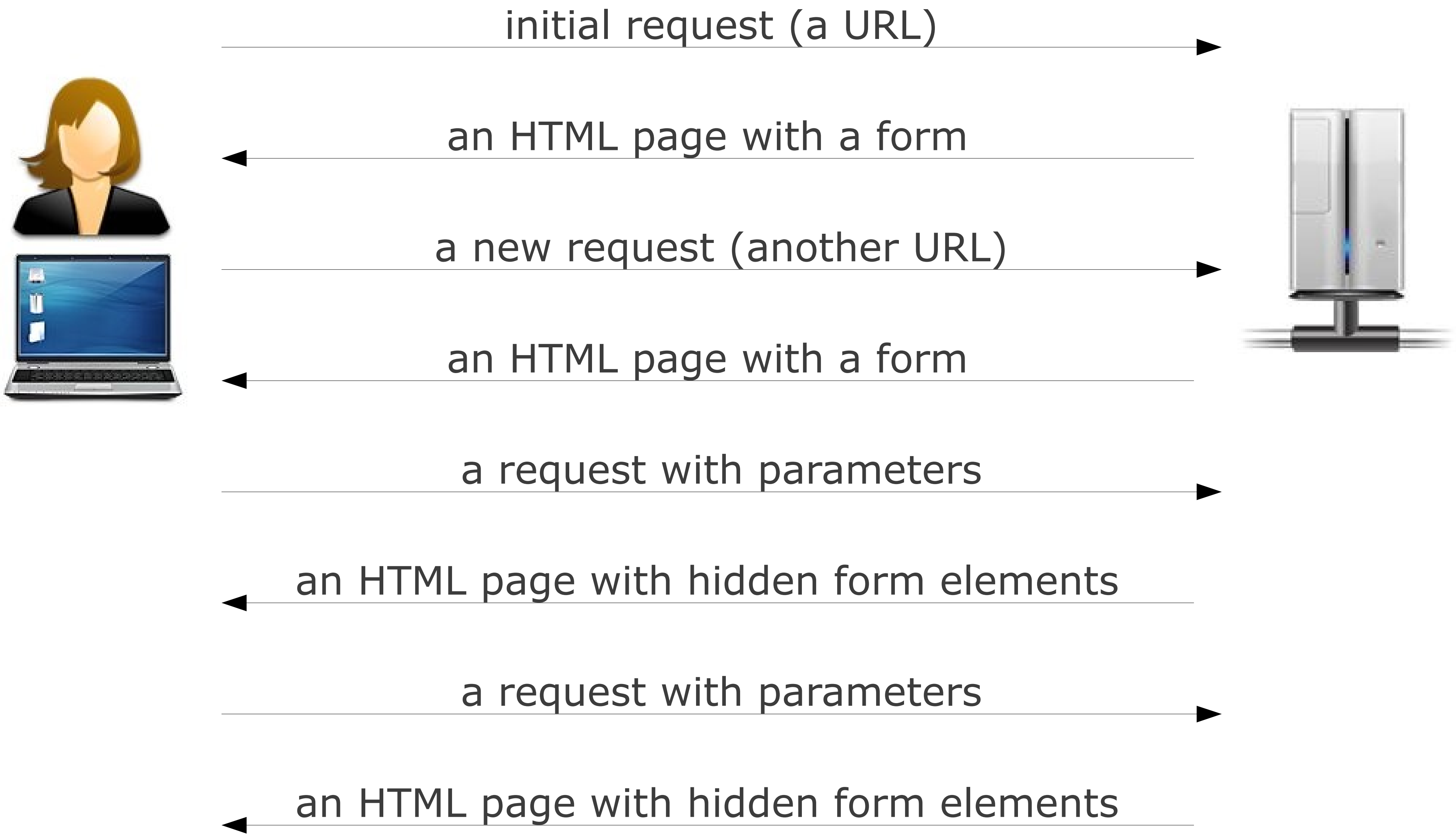
```
<form action="/~okenobi/dynamic/personas.py"
      method="get">
  Please select a species:
  <input type="text" name="species" />
  <input type="submit" />
</form>
```

HTML Forms

```
<form action="/~okenobi/dynamic/personas.py"
      method="get">
  Please select a species:
  <select name="species">
    <option value="Human">Human</option>
    <option value="Ewok">Ewok</option>
    <option value="Wookiee">Wookiee</option>
  </select>
  <input type="submit" />
</form>
```

HTML Forms: POST

```
<form action="/~kenobio1/cgi-bin/personas3.cgi"
      method="post">
  Please select a species:
  <select name="species">
    <option value="Human">Human</option>
    <option value="Ewok">Ewok</option>
    <option value="Wookiee">Wookiee</option>
  </select>
  <input type="submit" />
</form>
```


initial request (a URL)
an HTML page with a form
a new request (another URL)
an HTML page with a form
a request with parameters
an HTML page with hidden form elements
a request with parameters
an HTML page with hidden form elements

Prefilled Forms

```
Please select a species:
  <input type="text" name="species"
   value="Human"/>
```

Hidden Elements

```
Please select a species:
  <input type="hidden" name="species"
   value="Human"/>
```

Or Just in Links

```
<a href="dynamic/personas.py?species=Human">
Click here for the Humans
</a>
```

Questions?