CS1116/CS5018
Web Development 2

Dr Derek Bridge

School of Computer Science & Information Technology

University College Cork

Sessions

- A **session** is a sequence of requests from the same client within a short period of time
- Although the user may access different pages from the web site, we may want to:
  - track the client, i.e. identify which client is making requests and
  - keep user-specific data across the pages visited
- Example: a shopping cart

Session ids

- A session will have a unique identifier — its unique **session id (SID)**
- A hash of the server time will be a unique SID:

```python
from hashlib import sha256
from time import time

sid = sha256(repr(time()).encode()).hexdigest()
```

- The SID is:
  - created by the server
  - sent to the client and
  - sent back to the server whenever the client makes a request

  A cookie is ideal for this

Session state

- **Session state:** other data that we keep about this session (e.g. contents of the shopping cart)
- Session state is not sent back-and-forth in cookies:
  - store it in a file or database
- The shelve module:
  - opens a file whose filename can be based on the SID
  - read and write to the file as you would a dictionary

E.g.

```python
from shelve import open

session_store = open('sess_' + sid, writeback=True)

# Examples of writing to the file
session_store['firstname'] = 'Hugh'
session_store['surname'] = 'Jeegoh'

# Example of reading from the file
print(session_store.get('firstname'))

session_store.close()
```
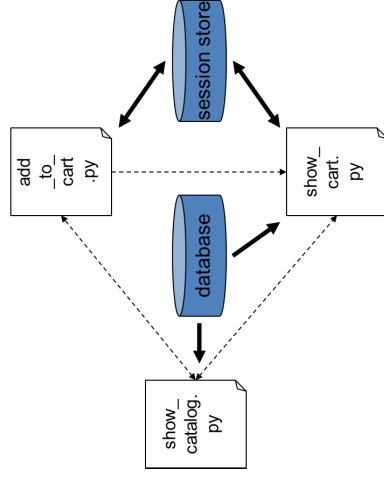
Case study 2: a very, very, very simple shopping cart

Suppose our company *Shrine of Bacchus Wines* sell wines, listed in this database:

```
CREATE TABLE wines
(
    wine_id INT AUTO_INCREMENT,
    name VARCHAR(255) NOT NULL,
    price DECIMAL(5, 2) NOT NULL,
    PRIMARY KEY (wine_id)
);
```

Overview of the programs

show.catalog.py: displaying our wine catalog

```python
#!/usr/local/bin/python3

from cgitb import enable
enable()

import pymysql as db
print('Content-Type: text/html')
print()

result = ''
try:
    connection = db.connect('localhost', 'userid', 'password', 'database_name')
    cursor = connection.cursor(db.cursors.DictCursor)
    cursor.execute("""SELECT * FROM wines ORDER BY wine_id""")
    result = """<table>
    <tr><th colspan="3">Our Wines</th></tr>
    <tr><th>Name</th><th>Price</th><th></th></tr>"""
    for row in cursor.fetchall():
        result += """<tr>
        <td>%s</td>
        <td>%s</td>
        <td><a href="add_to_cart.py?wine_id=%s">Add to cart</a></td>
        </tr>""" % (row['name'], row['price'], row['wine_id'])
    result += '</table>'
    cursor.close()
    connection.close()
except db.Error:
    result = '<p>Sorry! We are experiencing problems at the moment. Please call back later.</p>'
print("""<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8" />
        <title>Shrine of Bacchus Wines</title>
    </head>
    <body>
        %s
        <p>
            <a href="show_cart.py">Show cart</a>
        </p>
    </body>
</html>""" % (result))
```

add_to_cart.py: adding to the shopping cart

```python
#!/usr/local/bin/python3

from cgitb import enable
enable()

from cgi import FieldStorage
from os import environ
from hashlib import sha256
from time import time
from shelve import open
from http.cookies import SimpleCookie

result = ''
try:
    cookie = SimpleCookie()
    http_cookie_header = environ.get('HTTP_COOKIE')
    if not http_cookie_header:
        sid = sha256(repr(time()).encode()).hexdigest()
        cookie['sid'] = sid
    else:
        cookie.load(http_cookie_header)
        if 'sid' not in cookie:
            sid = sha256(repr(time()).encode()).hexdigest()
            cookie['sid'] = sid
        else:
            sid = cookie['sid'].value

    session_store = open('sess_' + sid, writeback=True)

    # Get the id of the item being added to the cart
    form_data = FieldStorage()
    wine_id = form_data.getfirst('wine_id')

    # If this item is not in the cart already, then quantity is 1; otherwise, increment the quanti
    qty = session_store.get(wine_id)
    if not qty:
        qty = 1
    else:
        qty +=1
    session_store[wine_id] = qty
    session_store.close()

    print(cookie)
    result = '<p>Item successfully added to your cart.</p>'
except IOError:
    result = '<p>Sorry! We are experiencing problems at the moment. Please call back later.</p>'

print('Content-Type: text/html')
print()
print("""
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8" />
        <title>Shrine of Bacchus Wines</title>
    </head>
    <body>
        %s
        <p>
        <a href="show_cart.py">Show cart</a>
        </p>
        <p>
        <a href="show_catalog.py">Show catalog</a>
```

```
        </p>
    </body>
</html>""" % (result))
```

show_cart.py:
displaying the shopping cart

```python
#!/usr/local/bin/python3

from cgitb import enable
enable()

from os import environ
from hashlib import sha256
from time import time
from shelve import open
from http.cookies import SimpleCookie
import pymysql as db

result = ''
try:
    cookie = SimpleCookie()
    http_cookie_header = environ.get('HTTP_COOKIE')
    if not http_cookie_header:
        sid = sha256(repr(time()).encode()).hexdigest()
        cookie['sid'] = sid
    else:
        cookie.load(http_cookie_header)
        if 'sid' not in cookie:
            sid = sha256(repr(time()).encode()).hexdigest()
            cookie['sid'] = sid
        else:
            sid = cookie['sid'].value

    session_store = open('sess_' + sid, writeback=True)

    if len(session_store) == 0:
        result = '<p>No items in shopping cart.</p>'
    else:
        connection = db.connect('localhost', 'userid', 'password', 'database_name')
        cursor = connection.cursor(db.cursors.DictCursor)
        result = """<table>
        <tr><th colspan="2">Your Cart</th></tr>
        <tr><th>Wine</th><th>Quantity</th></tr>"""

        for wine_id in session_store:
            cursor.execute("""SELECT name FROM wines
                              WHERE wine_id = %s""", (wine_id))
            row = cursor.fetchone()
            result += '<tr><td>%s</td><td>%s</td></tr>' % (row['name'], session_store.get(wine_id))
        result += '</table>'
        cursor.close()
        connection.close()

    session_store.close()
    print(cookie)
except (db.Error, IOError):
    result = '<p>Sorry! We are experiencing problems at the moment. Please call back later.</p>'

print('Content-Type: text/html')
print()
print("""
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <title>Shrine of Bacchus Wines</title>
  </head>
  <body>
    %s
    <p>
      <a href="show_catalog.py">Show catalog</a>
    </p>
  </body>
</html>""" % (result))
```