



LẬP TRÌNH WEB CHẠY Ở SERVER

CHƯƠNG V

Nội dung

- Lập trình Web với CGI
- Lập trình Web với Servlet
- Lập trình Web với JSP
- Lập trình Web với ASP (???)
- Lập trình Web với PHP (tự học)

Lập trình web với CGI

- Trang HTML chỉ chứa các nội dung cố định hoặc có thể thao tác thay đổi qua DHTML ở client.
- Nhu cầu sử dụng các trang web như một ứng dụng (giao tiếp với user, tìm kiếm, nội dung thay đổi, tương tác các ứng dụng, với database...)=> ứng dụng web.
- CGI (Common Gateway Interface) là một chuẩn (standard) để viết ứng dụng web.

Lập trình web với CGI

- **Các đặc điểm**

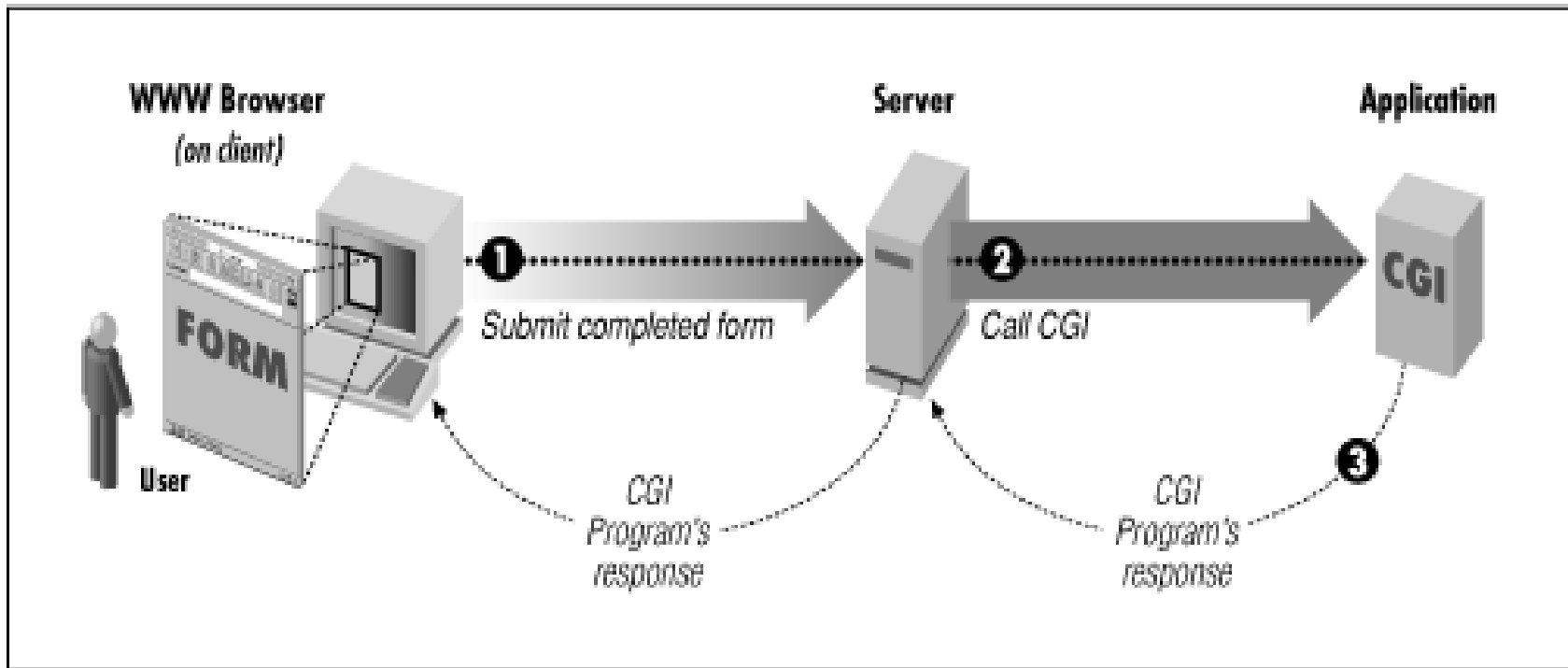
- Là ứng dụng chạy trên nền web server.
- Nhận thông tin từ web browser và xuất thông tin để web browser hiển thị thông qua web server.
- Ứng dụng có thể được viết với bất cứ ngôn ngữ lập trình có standard input và standard output. (C/C++, Perl, Shell script, Dos shell...)

Lập trình web với CGI

- **Cách thức hoạt động của ứng dụng web CGI**
 - Web server nhận request ứng dụng CGI từ browser.
 - Web server gọi ứng dụng CGI, truyền các thông số bằng các biến môi trường(thông qua standard input).
 - Ứng dụng web CGI xử lý, giao tiếp với các ứng dụng khác (database, mail...), xuất kết quả dạng HTML qua standard output đến web server.
 - Web server nhận kết quả và trả về cho browser.

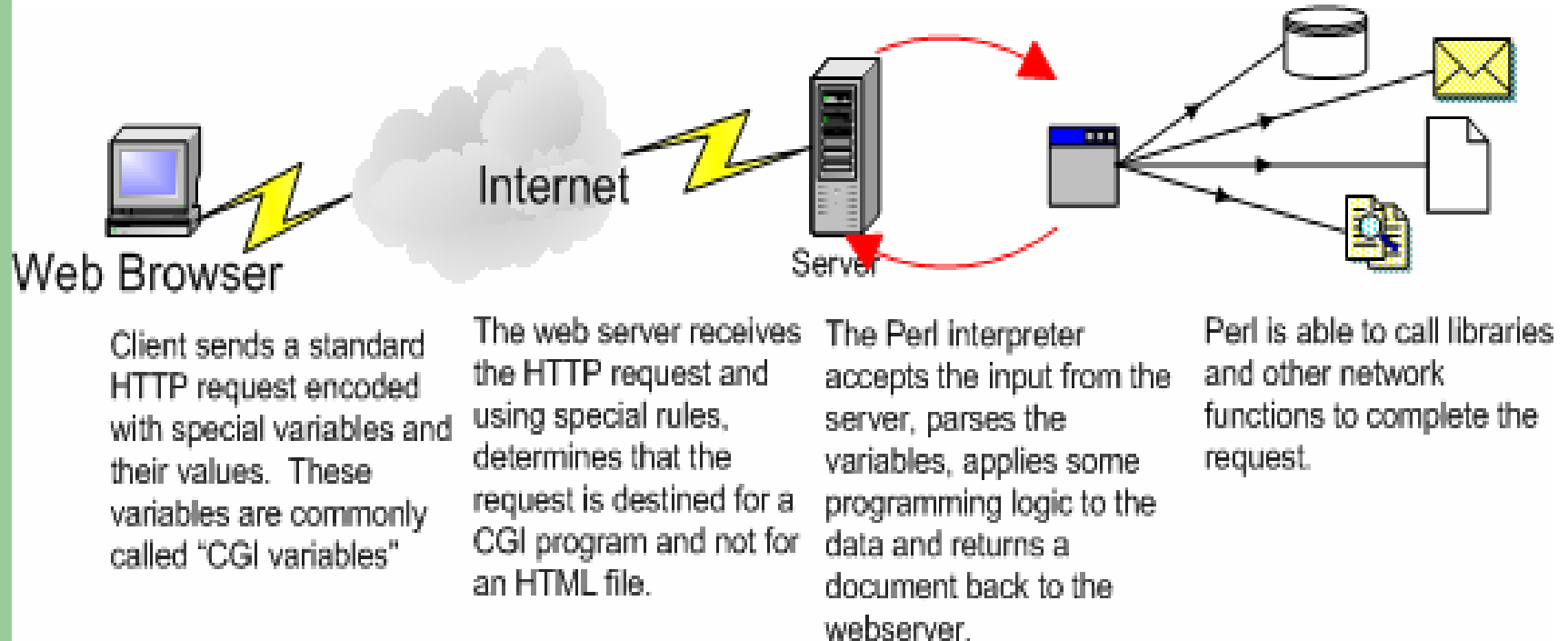
Lập trình web với CGI

- Hình vẽ minh họa hoạt động CGI



Lập trình web với CGI

- Hình vẽ minh họa hoạt động CGI (Perl)



Lập trình web với CGI

- Ví dụ (Perl)

```
#!/perl/bin/perl
#Remember : this path will vary depending on
#where Perl is located
print "Content-type:text/html\n\n";
print "<html><head><title>HELLO!</title></head>";
print "<body>\n";
print "<h2>Hello!</h2>\n";
foreach $key (sort(keys %ENV)) {
    print "VARIABLE $key = $ENV{$key}<br>\n";
}
print "</body></html>\n";
```

Lập trình web với CGI

- Các biến môi trường

Variable	Data
DOCUMENT_ROOT	The root directory of the web server
HTTP_COOKIE	The visitor's cookie, if one is set
HTTP_HOST	The name of your server
HTTP_REFERER	The page that called the script
HTTP_USER_AGENT	The browser the visitor is using
HTTPS	Set "on" if it is called through a secure server
PATH	The system path your server is running on
QUERY_STRING	See the GET method of receiving data below
REMOTE_ADDR	The visitors IP address
REMOTE_HOST	The hostname of the visitor
REMOTE_PORT	The port on your web server the visitor is connected to
REMOTE_USER	The visitors username
REQUEST_METHOD	GET or POST - see below
REQUEST_URI	The interpreted pathname of the requested document or CGI
SCRIPT_FILENAME	The full path name of the CGI
SCRIPT_NAME	The interpreted pathname of the CGI

Lập trình web với CGI

- **Các biến môi trường**

SERVER_ADMIN	The email address of the webmaster
SERVER_NAME	Your domain name
SERVER_PORT	The port the server is set to
SERVER_SOFTWARE	Your server type eg. Apache 1.3.11

- **Ví dụ**

Để in ra địa chỉ IP của máy khách:

```
print "IP Address = $ENV{'REMOTE_ADDR'}\n";
```

- **Một số các ví dụ cụ thể**

- Dùng phương thức GET: biến môi trường lưu ý là QUERY_STRING
- Dùng phương thức POST.
- Lưu các biến dùng COOKIE hoặc Session.
- Lập trình CGI kết nối với CSDL.

Lập trình web với CGI

- Tài liệu tham khảo

- Perl Programming
- http://inconnu.isu.edu/~ink/perl_cgi/
- <http://www.bewley.net/perl/>
- http://jan.netcomp.monash.edu.au/ecommerce/perl_session.html

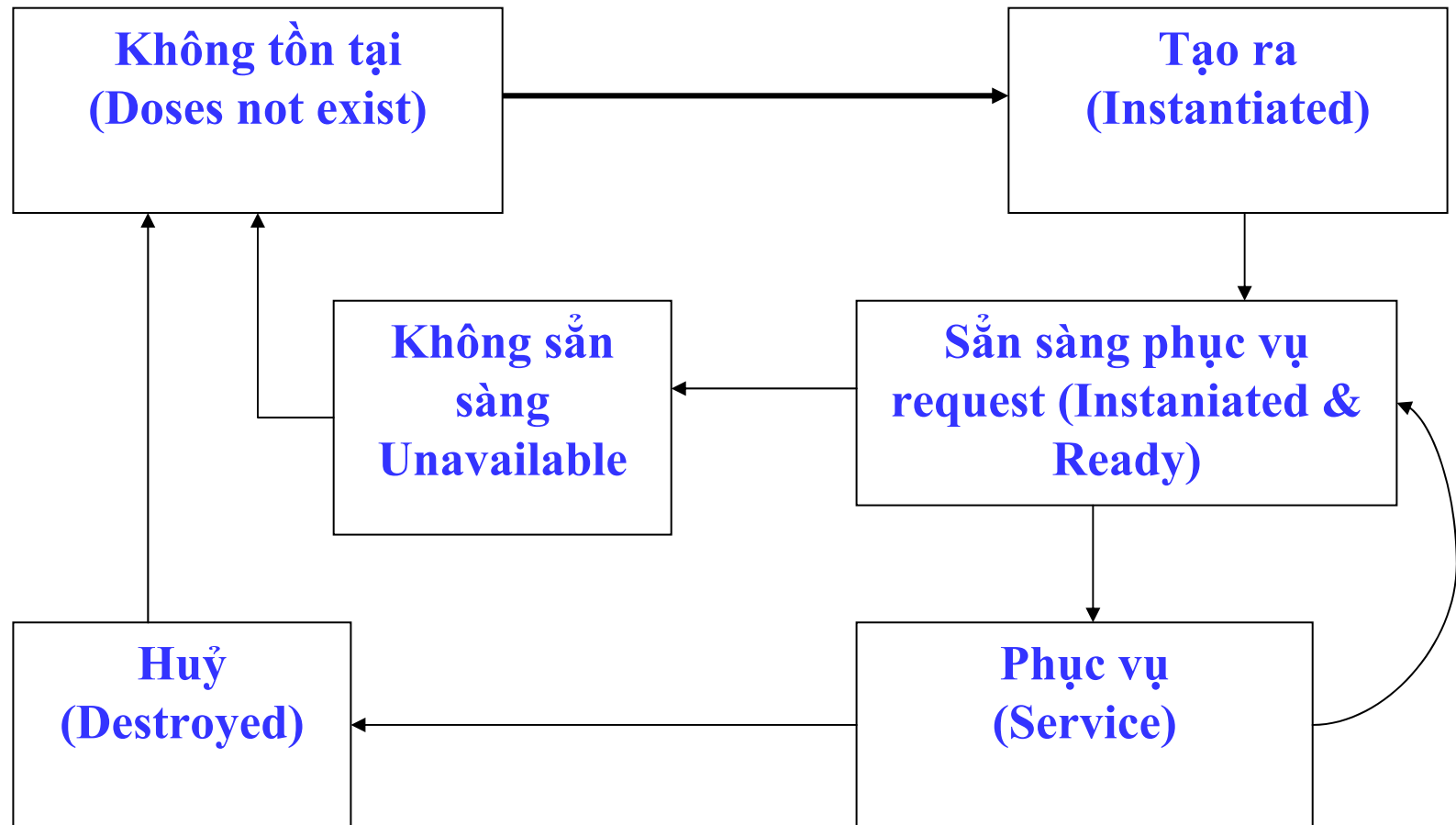
Giới thiệu Servlet

- Java servlet là những chương trình độc lập platform và nhỏ chạy ở phía server được lập trình để mở rộng chức năng của web server(Web server phải hỗ trợ Java).
- Cơ chế hoạt động theo mô hình CGI mở rộng.
- Chương trình servlet:
 - Thường extends class `HttpServlet`. Không có method *main*.
 - Phải được dịch ra ở dạng byte-code(.class) và khai báo với web server.

Giới thiệu Servlet

- Cần có package servlet.jar để biên dịch
(<http://java.sun.com/products/servlet/>)
- Các server hiện hỗ trợ Servlet:
 - Apache Tomcat (<http://jakarta.apache.org>)
 - Sun's Java Web Server, free, hiện không cho download
(<http://www.sun.com/software/jwebserver/>)
 - New Atlanta's ServletExec, tích hợp ServletEngine vào các web server(<http://newatlanta.com>)
 - <http://www.macromedia.com/software/jrun/trial/>
 - ...
- Tham khảo các tài liệu về Servlet:
<http://java.sun.com/products/servlet/docs.html>

Vòng đời của Servlet



Lập trình với Servlet API (t.t)

- **Java Servlet API gồm:**
 - **javax.servlet.***
 - Chứa các class và interface độc lập với HTTP
 - **javax.servlet.http.***
 - Chứa các class cụ thể cho HTTP
- Tất cả servlet đều phải hiện thực giao diện **javax.servlet.Servlet**.
- Tuy nhiên do chúng ta dùng giao thức HTTP nên **javax.servlet.http.HttpServlet** là lớp hiện thực giao diện Servlet, nên
- Vì vậy ta thường phải kế thừa từ lớp này và override các method **doGet()/doPost()**.

Lập trình với Servlet API (t.t)

Mục đích	Class/Interface
Hiện thực servlet	<code>javax.servlet.Servlet</code> <code>javax.servlet.SingleThreadModel</code> <code>javax.servlet.GenericServlet</code> <code>javax.servlet.http.HttpServlet</code>
Cấu hình sevlet	<code>javax.servlet.ServletConfig</code>
Servlet Exception	<code>javax.servlet.ServletException</code>
Requests	<code>javax.servlet.ServletRequest</code> <code>javax.servlet.http.HttpServletRequest</code> <code>javax.servlet.ServletInputStream</code>

Lập trình với Servlet API (t.t)

Mục đích	Class/Interface
Responses	<code>javax.servlet.ServletResponse</code> <code>javax.servlet.http.HttpServletResponse</code> <code>javax.servlet.ServletOutput</code>
Session tracking	<code>javax.servlet.http.HttpSession</code> <code>java.servlet.http.HttpSessionBindingListener</code>
Servlet context	<code>javax.servlet.ServletContext</code>
Servlet collaboration	<code>javax.servlet.RequestDispatcher</code>
Khác	<code>javax.servlet.http.Cookie</code> <code>java.servlet.http.HttpUtils</code>

Cơ chế hoạt động của servlet

- Web server nhận yêu cầu triệu gọi servlet từ client.
- Nếu servlet chạy lần đầu, web server load file servlet tương ứng, khởi tạo các thông số bằng qua method **init()**
- Nếu servlet đã được khởi tạo, tạo một thread để xử lý yêu cầu.
- Gọi methods **doXxx()** để xử lý các request tương ứng theo giao thức HTTP:
 - **doGet** cho HTTP GET, **doPost** cho HTTP POST

Cấu trúc đơn giản của một Servlet

```
import java.io.*;
import java.servlet.*;
import java.servlet.http.*;
public class Sample extends HttpServlet{
    public doGet(HttpServletRequest request,
                    HttpServletResponse response)
        throws ServletException, IOException{
        //dùng đối tượng "request" để đọc dữ liệu từ client
        //dùng đối tượng "response" để xuất dữ liệu cho client
        PrintWriter out = response.getWriter();
        //dùng đối tượng out để ghi (method print) dữ liệu cho client
    }
}
```

Một Servlet đơn giản

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class HelloWorld extends HttpServlet {
    public void doGet(HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException, IOException {
        PrintWriter out = response.getWriter();
        out.println("Hello World");
    }
}
```

Lập trình Web với Servlet

- **API cho HttpServletRequest**

1. Object `getAttribute(String name)`
2. void `setAttribute(String name, Object att )`
3. Enumeration `getAttributeNames()`
4. int `getContentLength()`
5. String `getParameter(String name)`
6. Enumeration `getParameterNames()`
7. Enumeration `getParameterValues()`
8. String `getRemoteHost()`
9. RequestDispatcher `getRequestDispatcher( String path )`
10. boolean `isSecure()`
11. ServletInputStream `getServletInputStream()`
12. int `getContentLength()`
13. ...

Lập trình Web với Servlet

- **Lấy dữ liệu từ web client gửi đến bằng servlet**
 - Dùng đối tượng của class *HttpServletRequest*
 - Các methods để lấy thông số:
 - *getParameter("para-name")*
 - *getParameterValues("para-name")*

```
String username=  
    request.getParameter("username");  
String[] choice =  
    request.getParameterValues("comments");
```

- Dùng đối tượng của class *HttpServletRequest* để lấy các thông tin HTTP header
- Ví dụ: Request Parameter

Lập trình Web với Servlet

- **Lấy các thông số HTTP request header**
 - class *HttpServletRequest* cung cấp các method để lấy các thông số request header.
 - *String getHeader(header-name)*: lấy nội dung của header-name
 - *Enumeration getHeaderNames()*: lấy tất cả các header-name.
 - Một số method điển hình:
 - *Cookie[] getCookies()*: dãy Cookie từ client
 - *int getContentLength()*: trả giá trị Content-Length
 - *int getContentType()*: trả giá trị Content-Type
 - *int getRemoteUser()*: giá trị username nếu có authenticate
 - Ví dụ: Request Header

Lập trình Web với Servlet

- **Lấy các thông số HTTP request header (t.t)**
 - Lấy các giá trị của biến môi trường CGI:
 - QUERY_STRING: *getQueryString()*
 - REMOTE_ADDR: *getRemoteAddr()*
 - REMOTE_HOST: *getRemoteHost()*
 - REQUEST_METHOD: *getMethod()*
 - PATH_INFO: *getPathInfo()*
 - SCRIPT_NAME: *getServletPath()*
 - SERVER_NAME: *getServerName()*
 - SERVER_PORT: *getServerPort()*
 - HTTP_XXX_YYY: *getHeader("Xxx-Yyy")*
 - ...
 - Ví dụ : [Request Info](#)

Lập trình Web với Servlet

- **API cho HttpServletResponse**

1. `String getCharaterEncoding()`
2. `ServletOutputStream getOutputStream()`
3. `PrintWriter getWriter()`
4. `setContentLength( int len )`
5. `setContentType( String type )`
6. `setBufferSize( int len )`
7. `int getBufferSize()`
8. `flushBuffer()`
9. ...

Lập trình Web với Servlet

- **Gửi dữ liệu cho web client**

- Dùng đối tượng của class *HttpServletResponse*.
- Tạo đối tượng *PrintWriter* để ghi dữ liệu gửi
 - `PrintWriter out = response.getWriter();`
- Xử lý các mã HTTP trả về với các method của class *HttpServletResponse*:
 - `void setStatus(int statusCode)`: gửi các mã response
 - `void sendError(int errorCode, String msg)`: gửi mã lỗi theo giao thức HTTP và message
 - `void sendRedirect(String URL)`: chuyển đến một trang URL khác
 - Các mã có thể theo giao thức HTTP hoặc dùng các hằng số trong class *HttpServletResponse*.

Lập trình Web với Servlet

- **Xử lý Cookie với web browser**

- Chức năng Cookie

- Kết hợp với web browser để lưu các thông số cần thiết.
 - Thông tin có thể dùng để thiết lập phiên làm việc (session) trong các ứng dụng thương mại điện tử(e-commerce).
 - Lưu trữ username, password
 - Thông tin để customize web site cho user hay dùng cho cơ chế personalization
 - ...

Lập trình Web với Servlet

- Ghi thông tin Cookie lên máy client
 - Thông tin được truyền đi trong field `Set-Cookie` HTTP header
 - Dùng method *`addCookie(Cookie cookie)`* của class *`HttpServletResponse`*.
 - Các thuộc tính quan trọng trong class *`Cookie`*:
 - Cookie name: *`setName(String name)`* – *`getName()`*
 - Cookie value: *`setValue(String value)`* – *`getValue()`*
 - Max Age: *`setMaxAge(int seconds)`* – *`getMaxAge()`*
 - Ví dụ: Cookies

Lập trình Web với Servlet

- Ví dụ

```
String user="",pass="";
Cookie[] cookies = request.getCookies();
if(cookies.length==0){
    user = request.getParameter("username");
    pass = request.getParameter("password");
    Cookie name_cookie = new Cookie("username",user);
    response.addCookie(name_cookie);
    Cookie pass_cookie = new Cookie("password",pass);
    response.addCookie(pass_cookie);
}
else{
    for(int i=0;i<cookies.length;i++){
        Cookie cookie = cookies[i];
        if(cookie.getName().equals("username"))
            user=cookie.getValue();
        if(cookie.getName().equals("password"))
            pass=cookie.getValue();
    }
}
```

Lập trình Web với Servlet

- Lưu thông tin về phiên làm việc của user
 - class *HttpSession*.
 - Có thể dùng để lưu bất kỳ đối tượng nào.
 - Đối tượng của class *HttpSession* được trả về từ method *getSession()* của class *HttpServletRequest*.
 - Các method thường sử dụng:
 - *Object getValue(String name)* [2.2: *getAttribute*]
 - *void putValue(String name, Object object)* [2.2: *putAttribute*]
 - *void removeValue(String name)* [2.2: *removeAttribute*]
 - *String[] getValueNames()* [*Enumeration getAttributeNames()*]
 - *String getId()*
 - *void setMaxInactiveInterval(int seconds)*
 - Ví dụ: Session

Lập trình Web với Servlet

- Ví dụ lưu ShoppingCart vào session

```
HttpSession session = request.getSession(true);  
ShoppingCart cart =  
    (ShoppingCart) session.getValue("ShoppingCart");  
if (cart == null) {  
    cart = new ShoppingCart();  
    session.putValue("ShoppingCart", cart);  
}  
//process(cart)
```

Lập trình Web với Servlet

- **Kết nối cơ sở dữ liệu**

- Dùng JDBC (Java DataBase Connectivity) để kết nối và thao tác với database.
- Quy trình xử lý:
 - Tạo JDBC driver và URL database.
 - Thiết lập connection đến URL database.
 - Tạo đối tượng statement.
 - Thực thi các câu lệnh SQL.
 - Xử lý kết quả thực thi.
 - Đóng kết nối đến database.

Lập trình Web với Servlet

- **Kết nối đến database thông qua ODBC.JDBC trên Windows**

- Tạo DataSourceName trong ODBC.JDBC

```
Connection con = null;
Statement stmt = null;
ResultSet rs = null;
String driver = "sun.jdbc.odbc.JdbcOdbcDriver";
String databaseURL = "jdbc:odbc:DataSourceName";
try{
    Class.forName(driver);
    con = DriverManager.getConnection(databaseURL);
    stmt = con.createStatement()
    rs = stmt.executeQuery("SELECT * FROM Products");
    while(rs.next()){
        out.println(rs.getString[1]+"-"+
        getInt("quantity")); //...
    }
    con.close();
}catch(SQLException se){ con.close(); }
```

Lập trình Web với Servlet

- **Kết nối database server bất kỳ**

- Cần có driver hỗ trợ.
- Kết nối đến Oracle Database Server:

```
driver="oracle.jdbc.driver.OracleDriver"  
databaseURL ="jdbc:oracle:thin@localhost:1521:"+dbName  
con = DriverManager.getConnection(databaseURL,user,password)
```

- Kết nối đến Sysbase:

```
driver="com.sysbase.jdbc.SysDriver"  
databaseURL=  
    "jdbc:sysbase:Tds:localhost:1521?SERVICENAME="+dbName  
con = DriverManager.getConnection(databaseURL,user,password)
```

Bài tập: Ứng dụng tra cứu DBĐT bằng servlet

- Viết một **InputForm** servlet để tạo một HTML form nhập để nhận dữ liệu tra cứu danh bạ điện thoại.
- Viết một **SearchAccount** servlet để tra cứu thông tin danh bạ.

Viết servlet InputForm

```
1. import javax.servlet.*; import javax.servlet.http.*;
2. import java.io.IOException;

3. public class InputForm extends HttpServlet {
4.     static final private String CONTENT_TYPE = "text/html";
5.     //Initialize global variables
6.     public void init() throws ServletException {
7.         // TO DO ...
8.     }

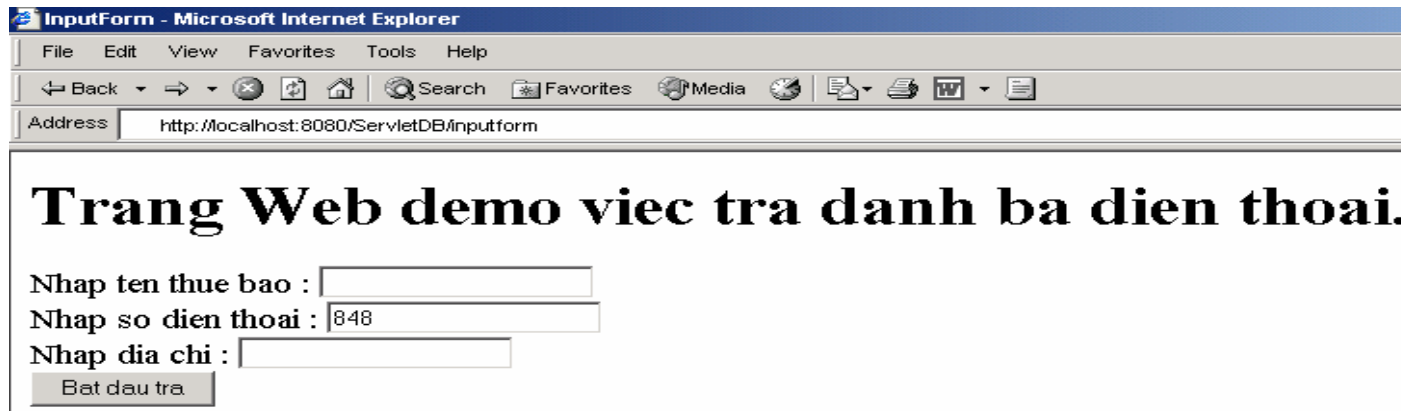
9.     public void doGet (HttpServletRequest request,
10.                        HttpServletResponse response)
11.        throws ServletException, IOException {
12.         // TO DO ...
13.     }
14. }
```

Viết servlet InputForm (t.t)

```
1. //Process the HTTP Get request
2. public void doGet ( HttpServletRequest request, HttpServletResponse
   response)
3.             throws ServletException, IOException {
4.     response.setContentType(CONTENT_TYPE);
5.     PrintWriter out = response.getWriter(); // Write to client

6.     out.println("<html><head><title>InputForm</title></head>");
7.     out.println("<body>");
8.     out.println("<h1>Trang Web demo viec tra danh ba dien thoai.</h1>");
9.     out.println("<form action='searchaccount' method='post'>");
10.    out.println("Nhap ten thue bao: <INPUT NAME='ttbao'><br>");
11.    out.println("Nhap so dien thoai: <INPUT NAME='sodthoai'><br>");
12.    out.println("Nhap dia chi: <INPUT NAME='diachi'> <br>");
13.    out.println("<input type='submit' name='Submit' value='Bat dau tra'>");
14.    out.println("</form>"); out.println("</body></html>");
15. }
```

Kết quả hiển thị trên trình duyệt



The screenshot shows a Microsoft Internet Explorer window titled "InputForm - Microsoft Internet Explorer". The address bar displays "http://localhost:8080/ServletDB/inputform". The main content area features a heading "Trang Web demo viec tra danh ba dien thoai." followed by a form with three input fields: "Nhap ten thue bao :", "Nhap so dien thoai : 848", and "Nhap dia chi :". A "Bat dau tra" button is located at the bottom of the form.

```
<html>
<head><title>InputForm</title></head>
<body> <h1>Trang Web demo viec tra danh ba dien thoai.</h1>
<form action=searchaccount method='post'>
    Nhap ten thue bao : <input name='ttbao' ><br>
    Nhap so dien thoai : <input name='sodthoai' ><br>
    Nhap dia chi      : <input name='diachi' ><br>
    <input type='submit' name='Submit' value='Bat dau tra'>
</form>
</body></html>
```

Viết servlet SearchAccount

```
1. import java.io.*; import java.util.*; import java.sql.*;
2. public class SearchAccount extends HttpServlet {
3.     static final private String CONTENT_TYPE = "text/html";
4.
5.     // Process the HTTP Post request
6.     public void doPost (HttpServletRequest request,
7.                         HttpServletResponse response)
8.         throws ServletException, IOException
9.     {
10.         // TODO...
11.         // Lay gia tri tu cac textbox tren InputForm
12.     }
13. }
```

Viết servlet SearchAccount (t.t)

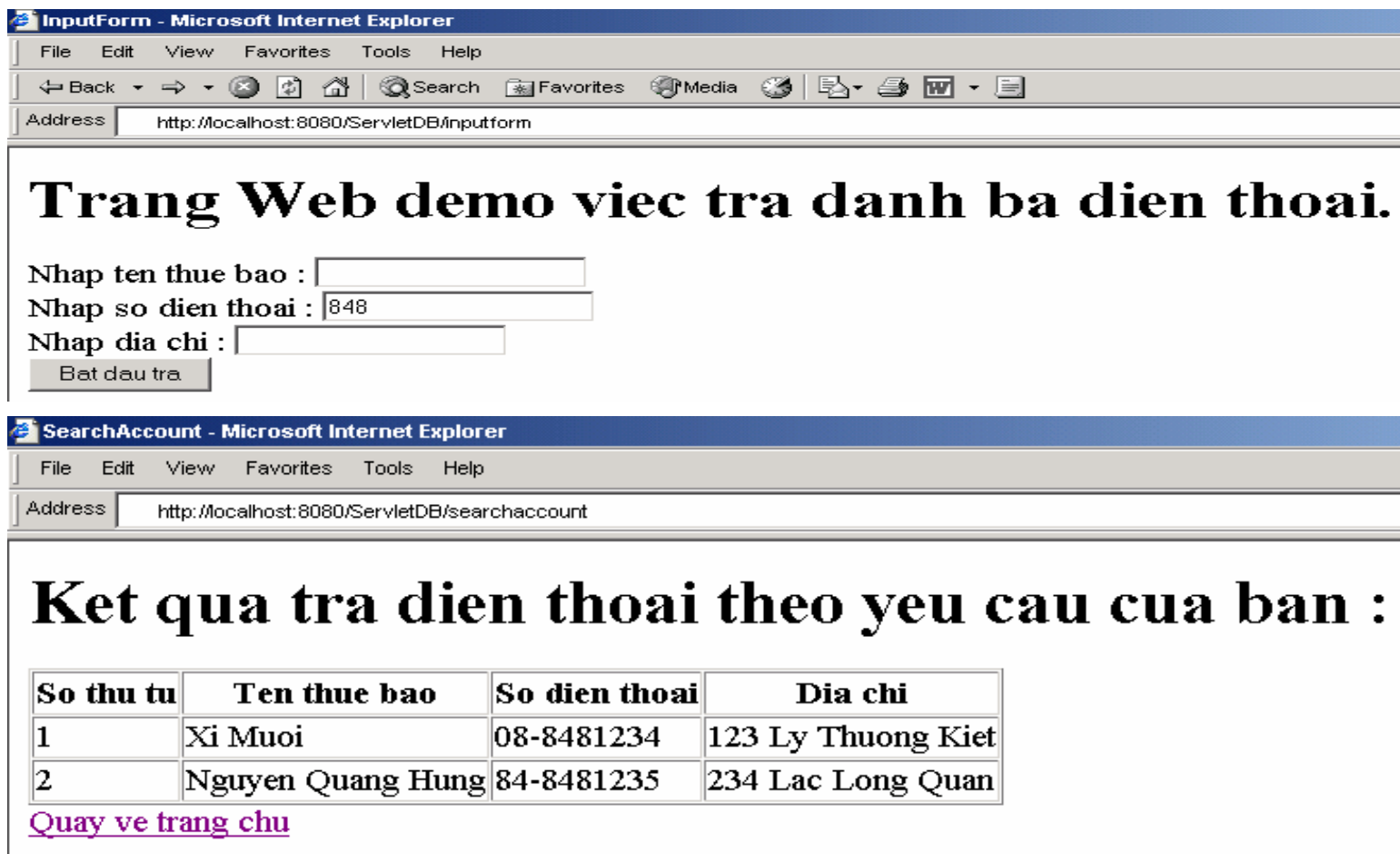
```
1. //Process the HTTP Post request
2. public void doPost (HttpServletRequest request, HttpServletResponse response)
3.     throws ServletException, IOException {
4.     response.setContentType(CONTENT_TYPE);
5.     PrintWriter out = response.getWriter();
6.     out.println("<html><head><title>SearchAccount</title></head>");
7.     out.println("<body><h1>Ket qua tra dien thoai theo yeu cau cua ban :</h1>");
8.     out.println("<table border=1 cellPadding=1 cellSpacing=1>");
9.     String tentbao = request.getParameter("ttbao");

10.    // xay dung lenh SQL
11.    String newSQL = "SELECT * FROM CUSTOMER";
12.    if (tentbao != null && tentbao.length() != 0) {
13.        newSQL = newSQL + " where tentbao like '%" + tentbao + "%'";
14.    }
15.    String conStr = "jdbc:odbc:DanhBaDT";
16.    Statement stmt = null;    ResultSet rs = null;
17.    .....
```

Viết servlet SearchAccount

```
1. try { Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
2.     con = DriverManager.getConnection(conStr, "", "");
3.     stmt = con.createStatement();
4.     rs = stmt.executeQuery(newSQL);
5.
6.     out.println("<tr><th>So thu tu</th><th>Ten thue bao</th><th>So dien" +
7.         " thoai</th><th>Dia chi</th></tr>\n");
8.     if (rs != null) {
9.         for ( int i = 1; rs.next(); ) {
10.            out.println("<tr><td>" + i + "</td><td>" + rs.getString(2) +
11.                "</td><td>" + rs.getString(3) + "</td><td>" +
12.                rs.getString(4) + "</td></tr>\n");
13.        } // end for
14.    } // end if
15.    out.println("</table>");
16.    rs.close();    stmt.close();    con.close();
17. } catch (Exception e) {    System.out.println("Error : " + e); }
18. ...
19. } // End doPost()
```

Kết quả chạy thử



Trang Web demo viec tra danh ba dien thoai.

Nhap ten thue bao :

Nhap so dien thoai :

Nhap dia chi :

Ket qua tra dien thoai theo yeu cau cua ban :

So thu tu	Ten thue bao	So dien thoai	Dia chi
1	Xi Muoi	08-8481234	123 Ly Thuong Kiet
2	Nguyen Quang Hung	84-8481235	234 Lac Long Quan

[Quay ve trang chu](#)