

BỘ MÔN CNPM - KHOA CNTT - ĐH KHTN

J2EE PROGRAMMING TRAINING COURSE 2012 - 2.0.1

SERVLET BASIC

Nguyễn Hoàng Anh
nhanh@fit.hcmus.edu.vn

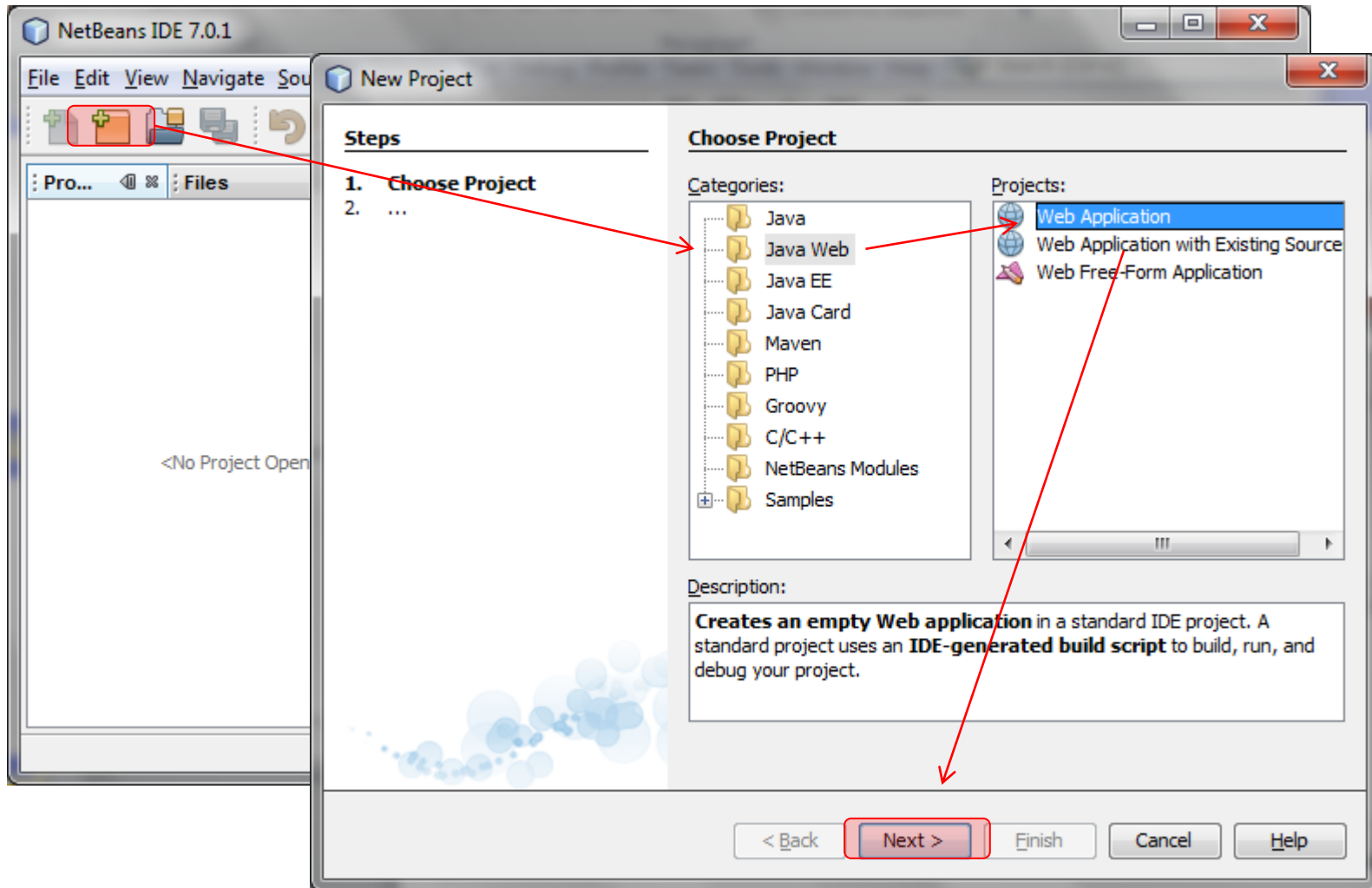
Nội dung trình bày

- Xây dựng Servlet đơn giản
- Nhiệm vụ của Servlet
- Các bước cơ bản xây dựng một trang Servlet
- Ví dụ minh họa
- Chu kỳ sống của Servlet
- Ví dụ minh họa

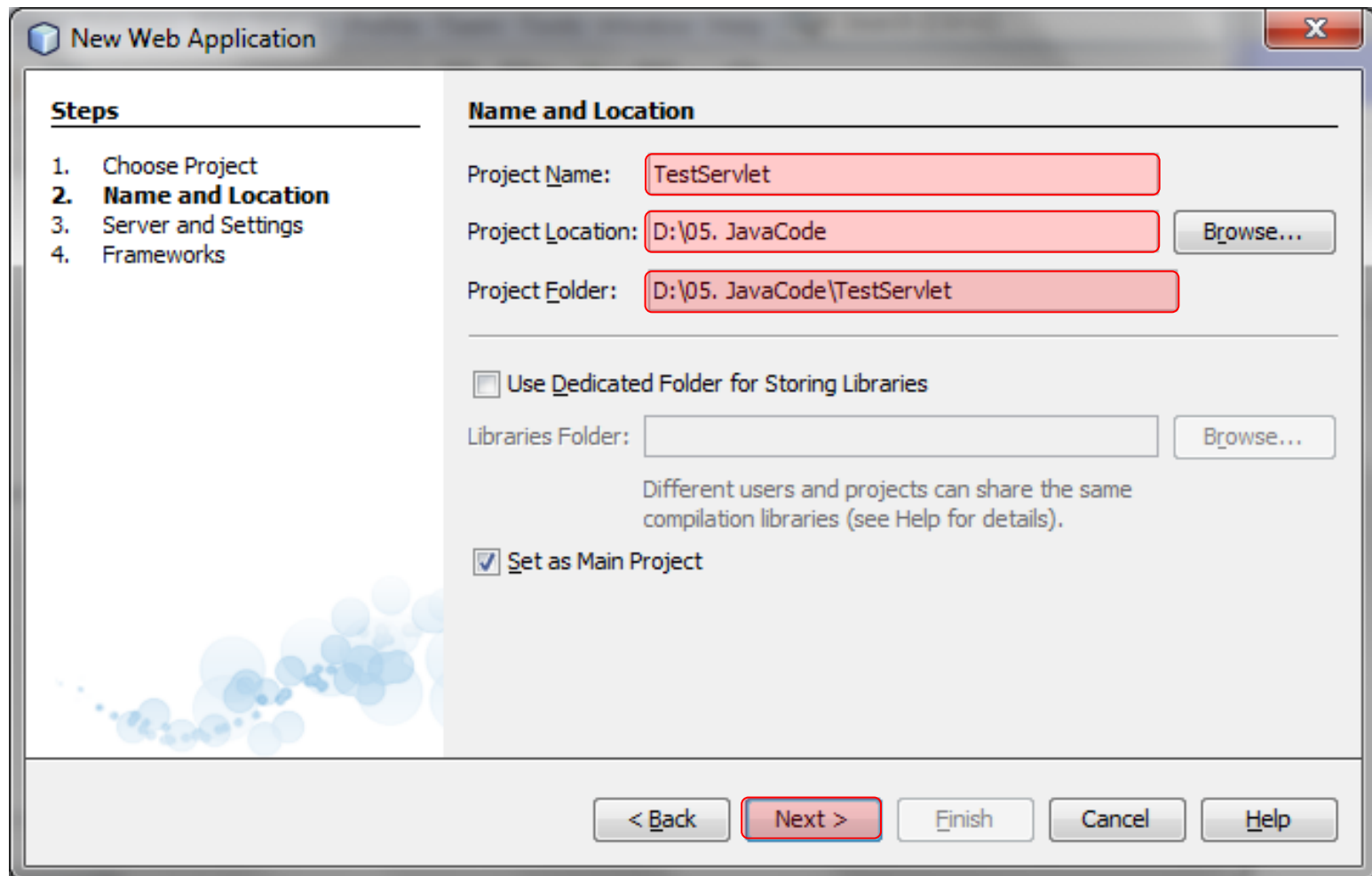
Xây dựng servlet đơn giản

- Xem nhanh các bước xây dựng một servlet đơn giản với Netbeans, tương tự với Eclipse.

Tạo mới Project



Đặt tên và lưu Project



New Web Application

Steps

1. Choose Project
- 2. Name and Location**
3. Server and Settings
4. Frameworks

Name and Location

Project Name:

Project Location:

Project Folder:

☐ Use Dedicated Folder for Storing Libraries

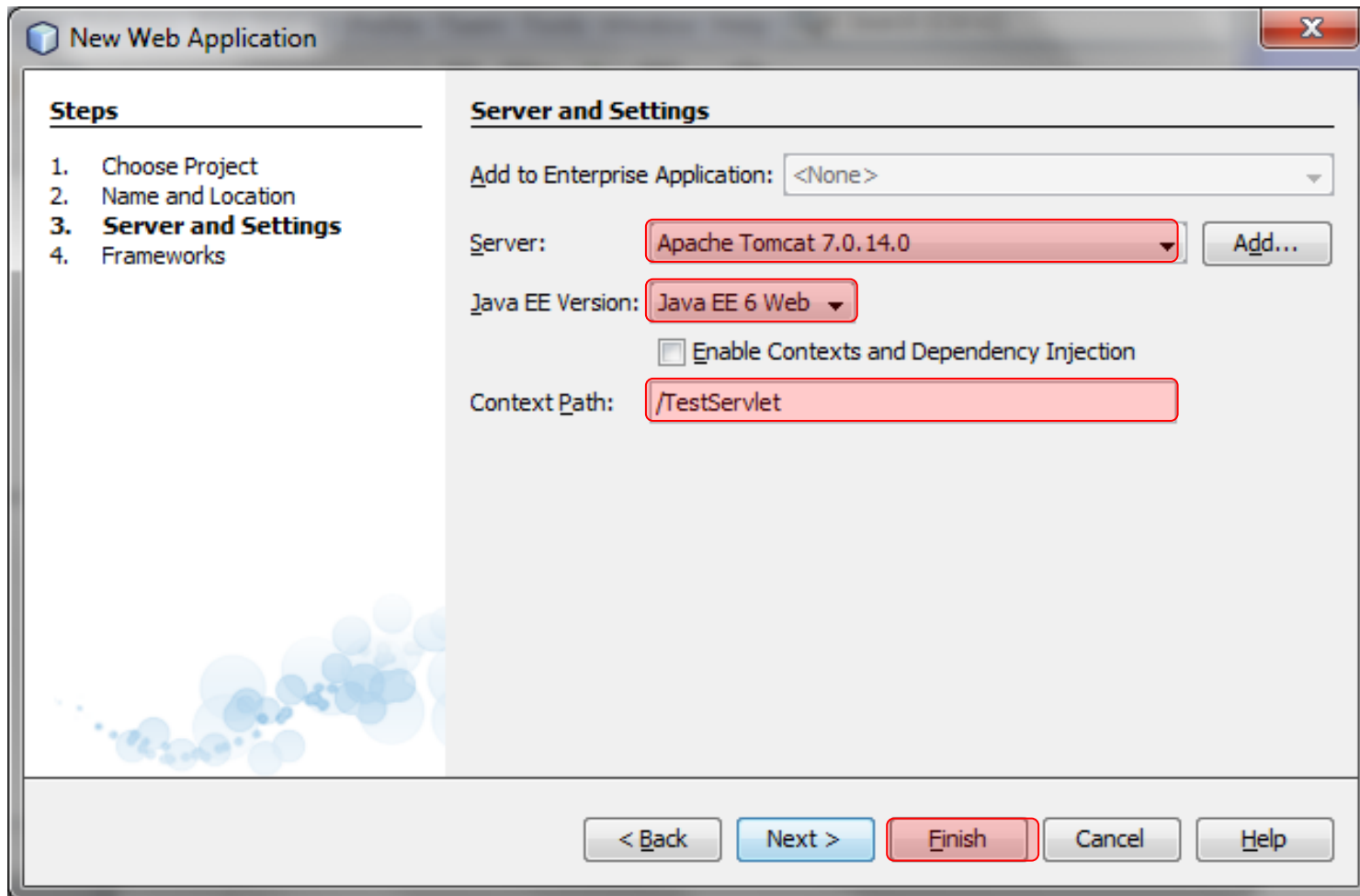
Libraries Folder:

Different users and projects can share the same compilation libraries (see Help for details).

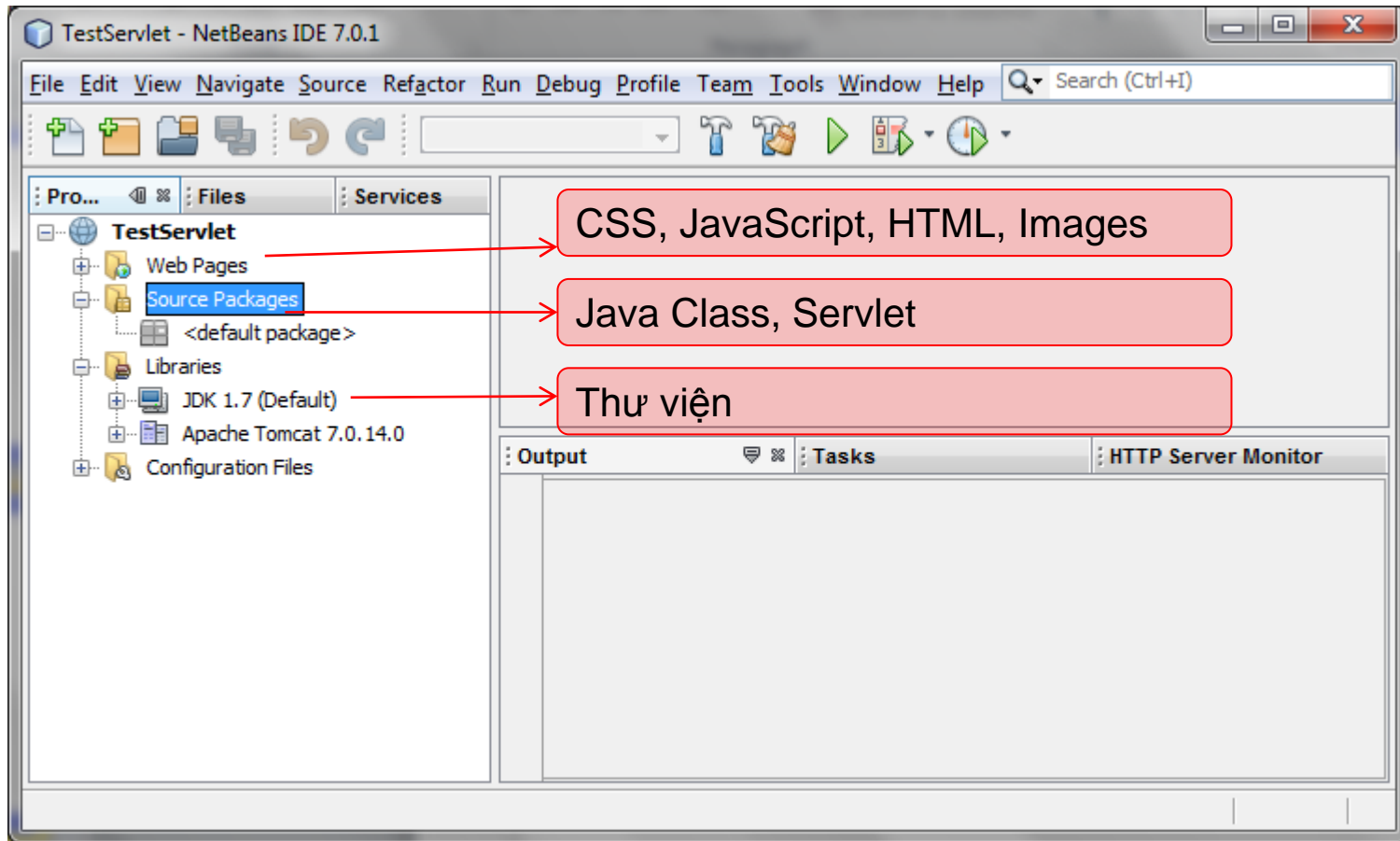
☒ Set as Main Project

< **Back** **Next** >

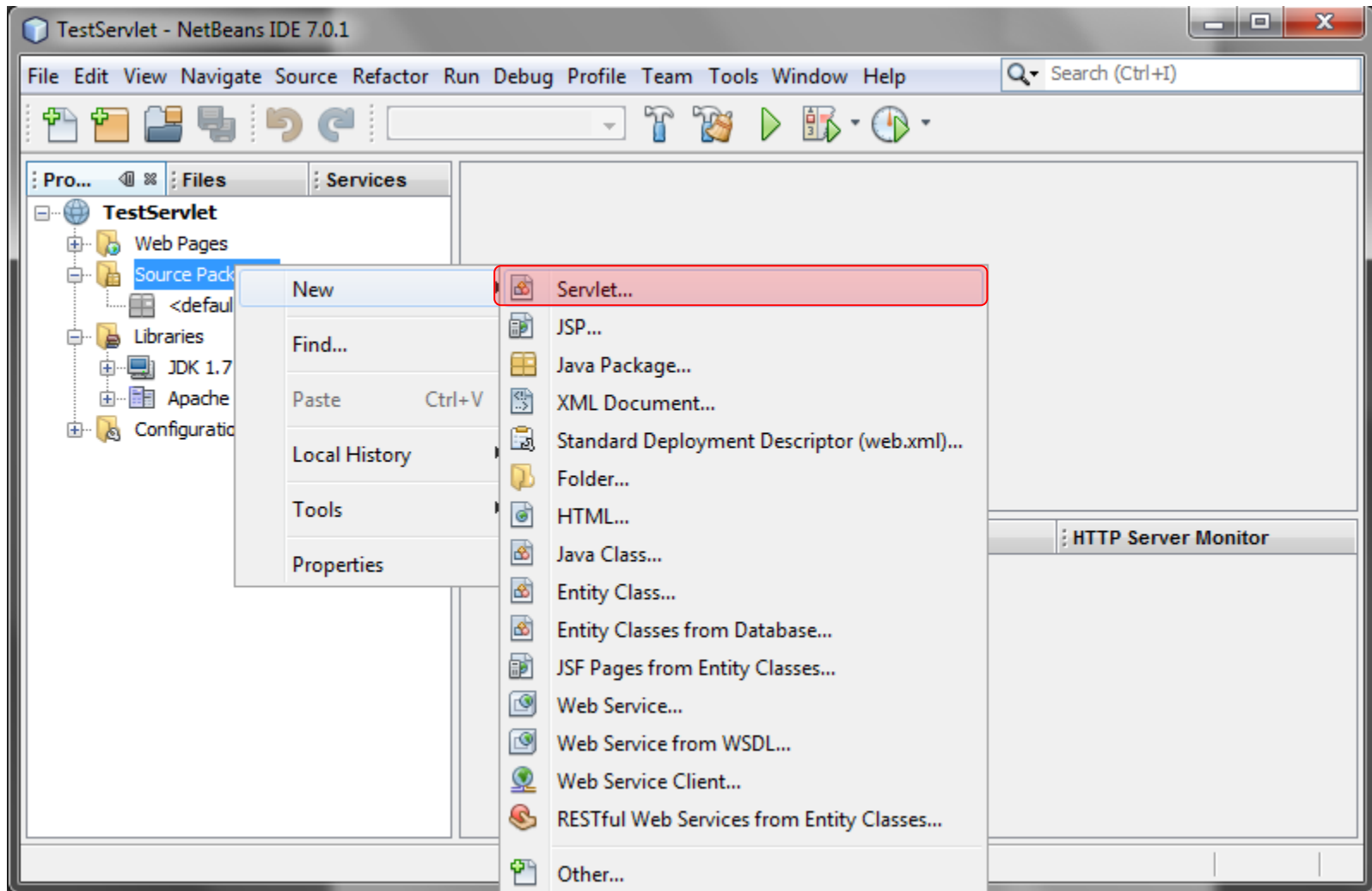
Chọn Server



Cấu trúc Project



Tạo Servlet



Tạo Servlet

New Servlet

Steps

1. Choose File Type
- 2. Name and Location**
3. Configure Servlet Deployment

Name and Location

Class Name:

Project:

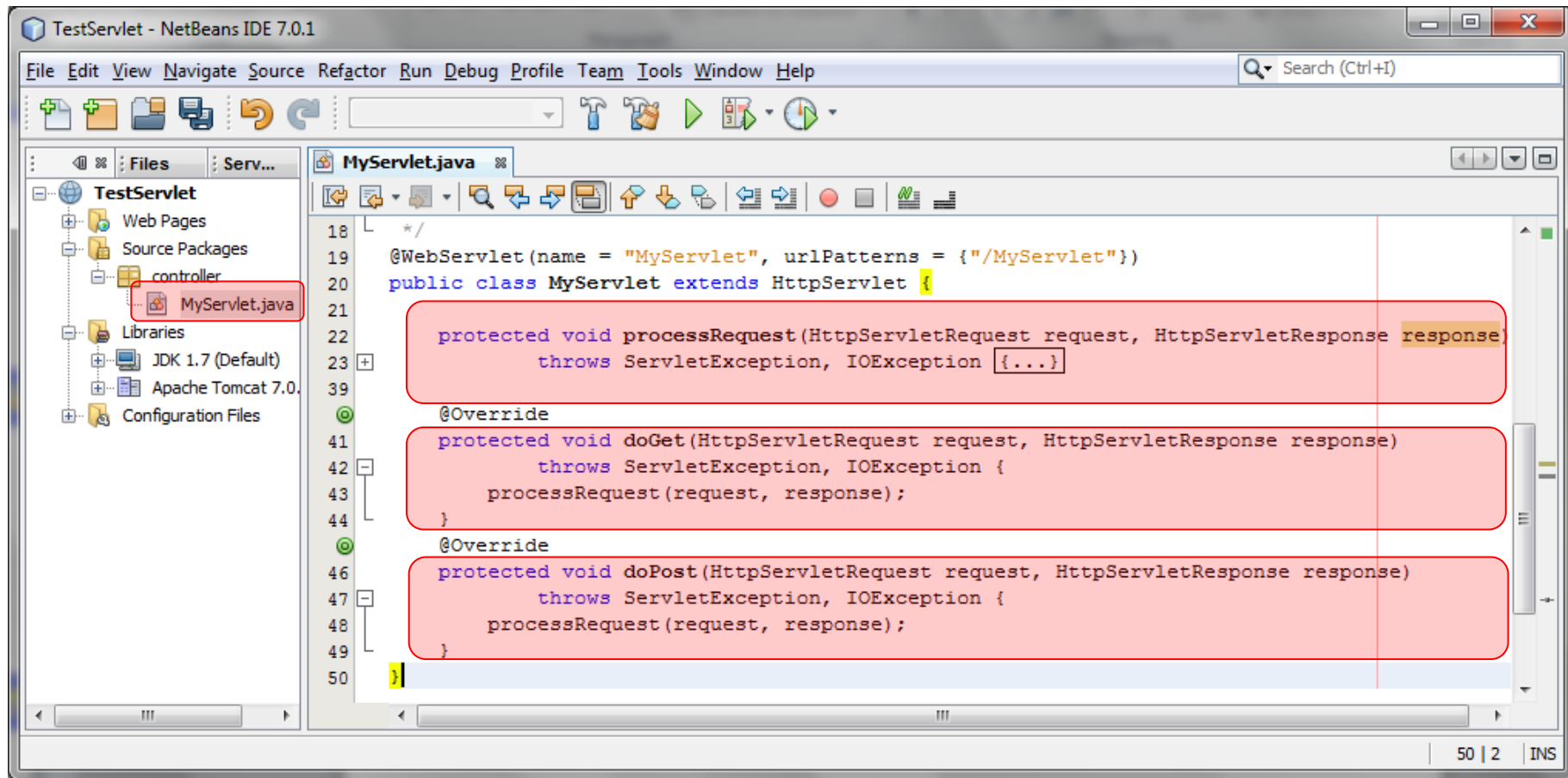
Location:

Package:

Created File:

< Back Next > Finish Cancel Help

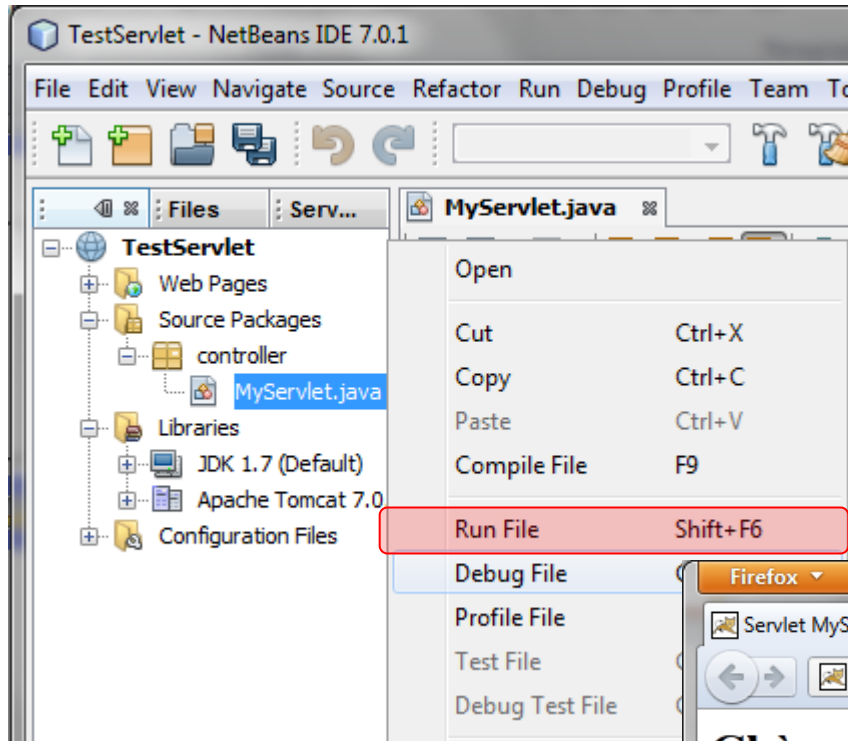
Tạo Servlet



Tạo giao diện web với HTML

```
1  protected void processRequest(HttpServletRequest request,
2  HttpServletResponse response)
3      throws ServletException, IOException {
4      response.setContentType("text/html;charset=UTF-8");
5      PrintWriter out = response.getWriter();
6      try {
7          out.println("<html>");
8          out.println("<head>");
9          out.println("<title>My Servlet</title>");
10         out.println("</head>");
11         out.println("<body>");
12         out.println("<h1>Chào mừng bạn đến với lập trình web
13                                     cơ bản/h1>");
14         out.println("</body>");
15         out.println("</html>");
16     } finally {
17         out.close();
18     }
19 }
20
21 }
```

Chạy thử nghiệm Servlet



Custom URL cho Servlet

```
1  @WebServlet(name="MyServlet", urlPatterns={"/YourServlet"})
2
3  public class MyServlet extends HttpServlet {
4
5      protected void processRequest (HttpServletRequest
6  request, HttpServletResponse response){
7          ...
8      }
9      @Override
10     protected void doGet(HttpServletRequest request,
11 HttpServletResponse response){
12         processRequest (request, response);
13     }
14     @Override
15     protected void doPost(HttpServletRequest request,
16 HttpServletResponse response) {
17         processRequest (request, response);
18     }
19 }
```

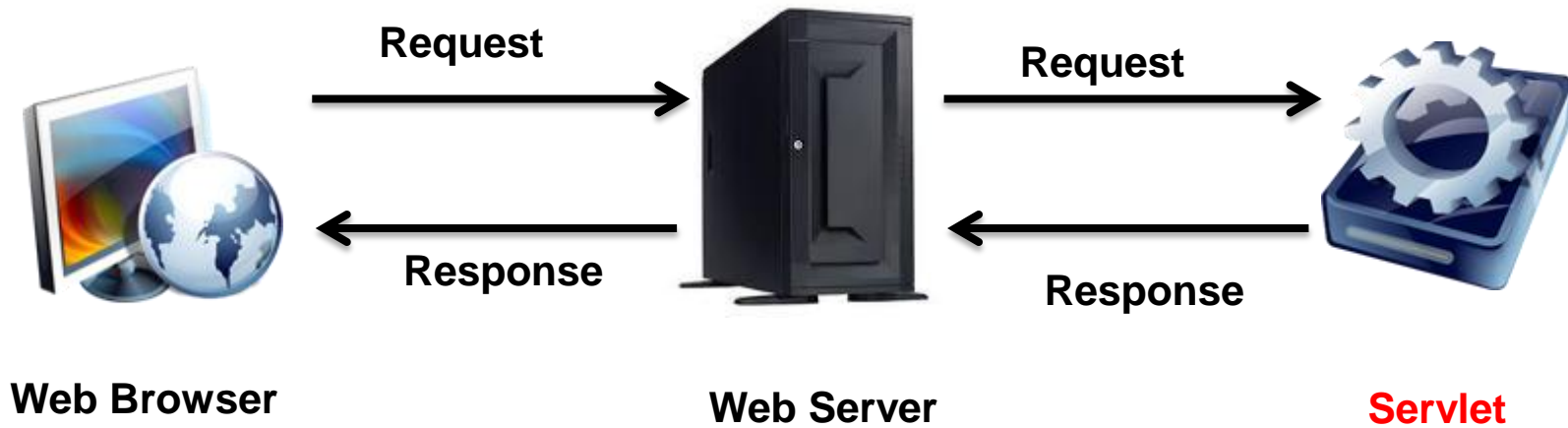
Custom URL cho Servlet



Bạn hãy thử đoán

- Servlet làm được gì và cho nhận xét?

Nhiệm vụ của Servlet



Nhiệm vụ của Servlet

- Đọc dữ liệu từ Client
 - Tường minh: Form Data
 - Ngầm định: Request Headers
- Xử lý đọc, thêm, xóa, sửa dữ liệu từ cơ sở dữ liệu
- Phát sinh kết quả
 - HTML, XML, Excel,...
- Trả kết quả về cho Client
 - Tường minh: HTML, XML, Excel,....
 - Ngầm định: Status Codes, Response Headers

HTTP GET & HTTP POST

- Hai cơ chế Request thông dụng nhất từ Client
 - HTTP GET và HTTP POST
- HTTP GET Request
 - Thông tin gửi đính kèm vào cuối URL
 - Dữ liệu gửi bị giới hạn
 - ../XemSanPham?id=S001&Trang=1
- HTTP POST Request
 - Thông tin gửi xem như dữ liệu (không đính kèm vào URL)
 - Không giới hạn dung lượng gửi

Cách đọc dữ liệu từ HTTP GET

- HTTP GET

- URL: `http://localhost/BookOnline/SachServlet`

- ? Action=Xem & MaSach = S001

- Cách đọc các tham số từ phương thức doGet

```
String action = request.getParameter("Action");
```

```
String maSach = request.getParameter("MaSach");
```

Cách đọc dữ liệu từ HTTP GET

- HTTP GET

- URL: `http://localhost/BookOnline/SachServlet`

- ? Action=Xem & MaSach = S001

- Cách đọc các tham số từ phương thức doGet

```
String action = request.getParameter("Action");
```

```
String maSach = request.getParameter("MaSach");
```

Cách đọc dữ liệu từ HTTP POST

- HTTP POST
 - Dữ liệu gửi từ form có method là POST
 - Cách đọc các tham số từ phương thức doPost

```
String action = request.getParameter("tenDangNhap");  
String maSach = request.getParameter("matKhau");
```

Các bước cơ bản xây dựng trang Servlet

- Bước 1: Sử dụng Dreamweaver để thiết kế giao diện.
- Bước 2: Sử dụng công cụ tiện ích Html2Servlet . exe để chuyển giao diện thiết kế ở bước 1 thành Servlet.
- Bước 3: Chép đoạn mã servlet từ bước 2 vào phương thức processRequest hoặc doGet, doPost
- Bước 4: Thực hiện phân tích Request từ client rồi thực hiện các yêu cầu tương ứng như đọc, ghi dữ liệu
- Bước 5: Thêm các câu lệnh if, do, while, for,... vào servlet tùy thuộc vào yêu cầu của trang để tạo một trang web động với Servlet một cách hoàn chỉnh.

Ví dụ minh họa

- Xây dựng website BookOnline với các trang web sau
 - Trang nhất
 - Xem danh sách sách theo danh mục có phân trang
 - Xem chi tiết sách

[TRANG NHẤT](#)[SÁCH BÁN CHẠY](#)[HƯỚNG DẪN](#)[TIN TỨC](#)[LIÊN HỆ](#)[GIỚI THIỆU](#)

TÌM KIẾM NHANH

Tên sách

Danh mục

--Chọn danh mục--

DANH MỤC



Java



C#

GIỚI THIỆU BOOKONLINE

BookOnline ra đời vào tháng 09 năm 2007

Sau 5 năm hoạt động, BookOnline hiện là trang bán sách trên mạng hàng đầu tại Việt Nam và top 3 trên thế giới

BookOline liên tục cập nhật nguồn sách mới nhất cho độc giả trong và ngoài nước.

NHAnh

[TrangNhat.do](#)

@ 2011 Nguyễn Hoàng Anh

[TRANG NHẤT](#)[SÁCH BÁN CHẠY](#)[HƯỚNG DẪN](#)[TIN TỨC](#)[LIÊN HỆ](#)[GIỚI THIỆU](#)

TÌM KIẾM NHANH

Tên sách

Danh mục

--Chọn danh mục--

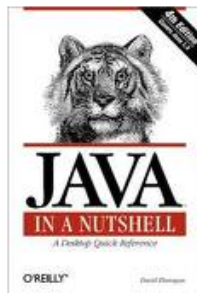
DANH MỤC

[Java](#)[C#](#)

DANH MỤC JAVA

Trang 1 | 2

Số lượng sách trên trang 5

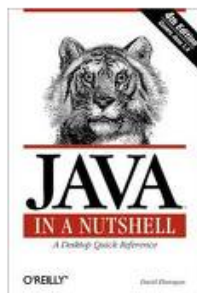


JAVA IN A NUTSHELL

Tác giả: Lý Hoàng Kỳ

Đơn giá: 500000 VNĐ

Số lượng online: 10 cuốn

[CHI TIẾT](#)[ĐẶT MUA](#)

JAVA IN A NUTSHELL

Tác giả: Lý Hoàng Kỳ

Đơn giá: 500000 VNĐ

Số lượng online: 10 cuốn

[CHI TIẾT](#)[ĐẶT MUA](#)[DanhSachSach.do](#)

[TRANG NHẤT](#)[SÁCH BÁN CHẠY](#)[HƯỚNG DẪN](#)[TIN TỨC](#)[LIÊN HỆ](#)[GIỚI THIỆU](#)

TÌM KIẾM NHANH

Tên sách

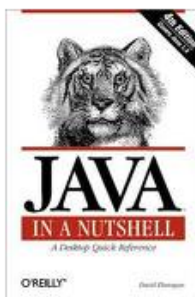
Danh mục

--Chọn danh mục--

DANH MỤC

[Java](#)[C#](#)

THÔNG TIN CHI TIẾT SÁCH JAVA IN NUTSHELL



JAVA IN A NUTSHELL

Tác giả: **Lý Hoàng Kỳ**Giá bìa: **500000 VNĐ**Giá bán: **500000 VNĐ**Giảm giá: **10%**Ngày xuất bản: **15/08/2011**Số trang: **320**Trọng lượng: **500g**Kích thước: **13 * 20 cm**Số lượng online: **10 cuốn**Số lần xem: **110**

NHẬN XÉT CỦA KHÁCH HÀNG

...

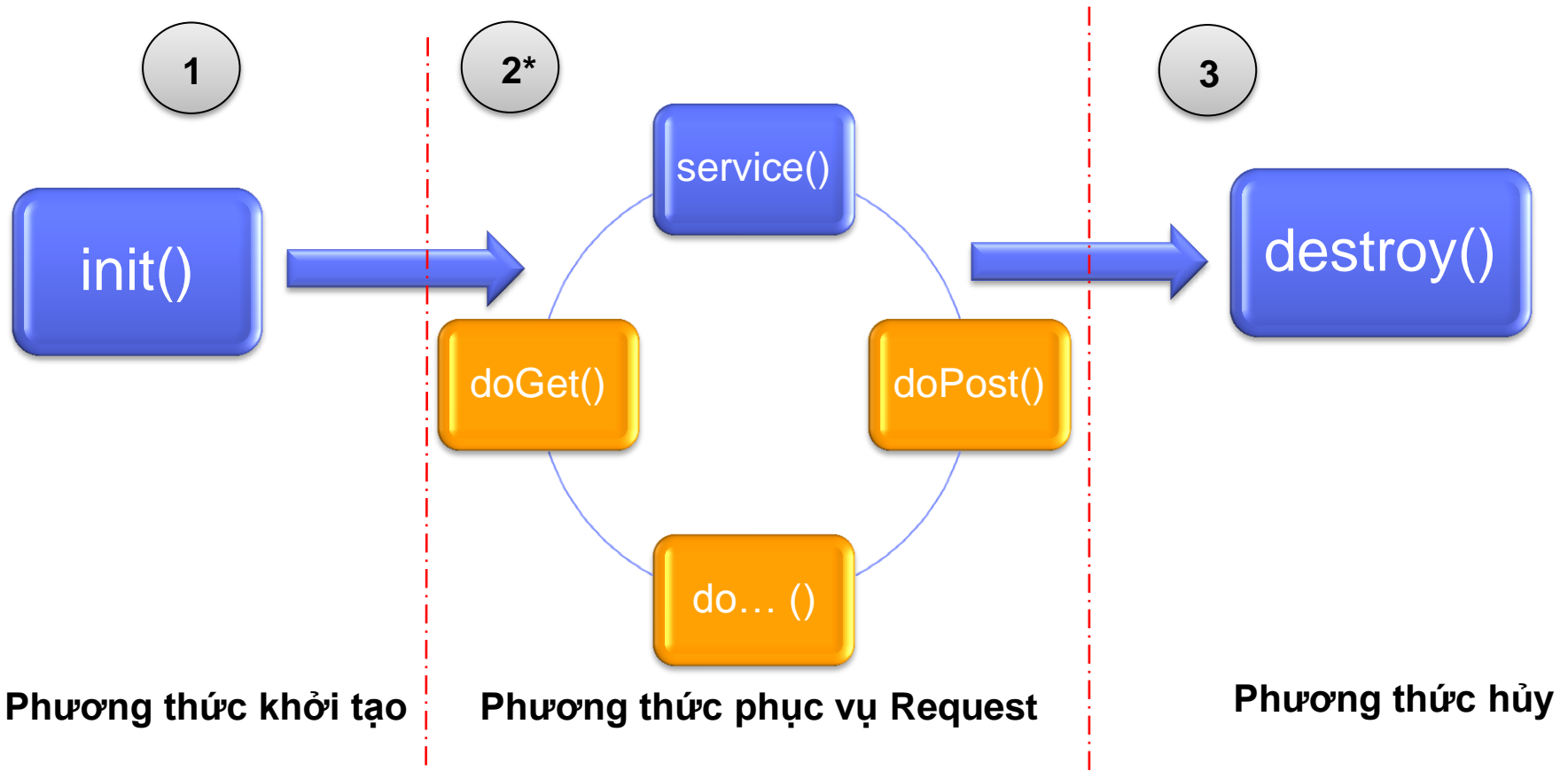
[ChiTietSach.do](#)

@ 2011 Nguyễn Hoàng Anh

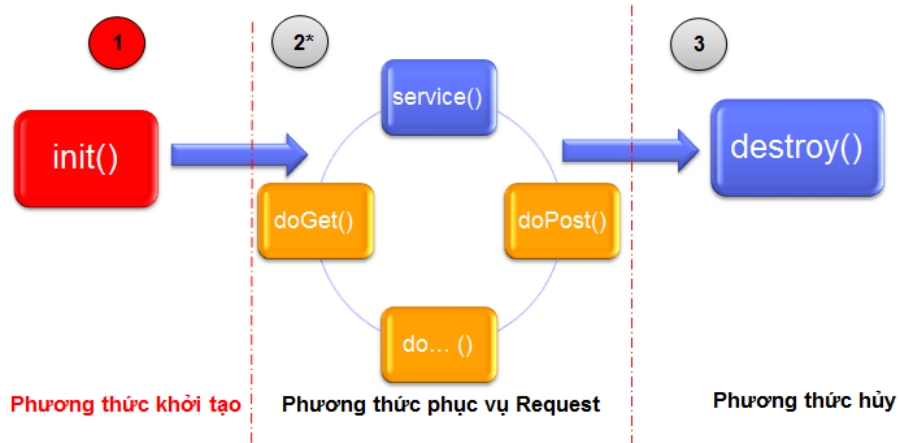
DEMO

Servlet Basic

Chu kỳ sống của servlet

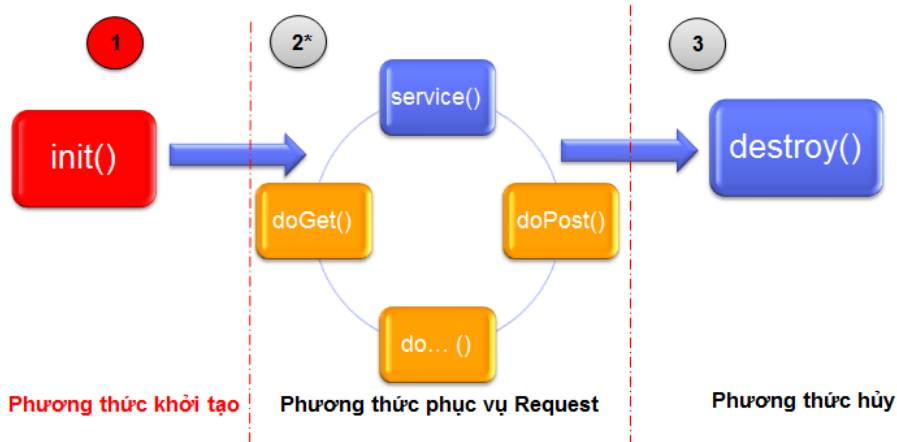


Chu kỳ sống của servlet



- Được tự động gọi thực hiện khi server tạo thể hiện Servlet
- Được xem như là phương thức khởi tạo

Chu kỳ sống của servlet

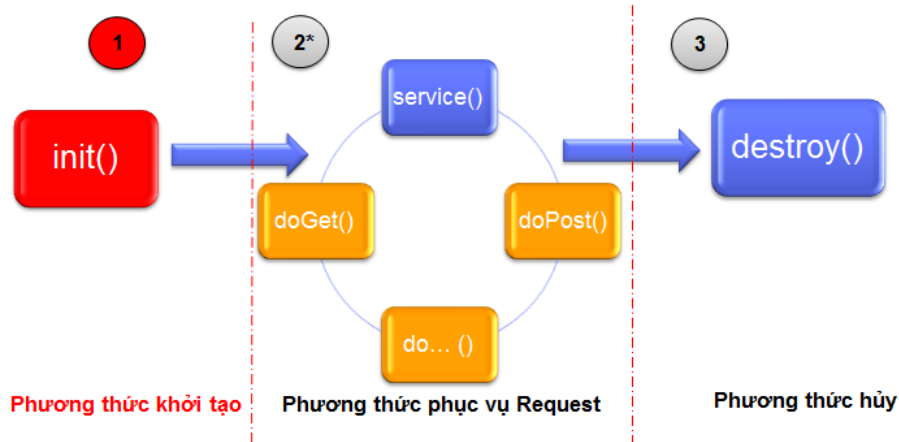


```
@WebServlet(name="ServletSample",
urlPatterns={"/ServletSample"})
public class ServletSample
    extends HttpServlet {
    private KieuDuLieu thuocTinh1;
    private KieuDuLieu thuocTinh2;
    ...
    private KieuDuLieu thuocTinhN;

    @Override
    public void init(){
        super.init();
        //Khởi tạo thuocTinh1
        //Khởi tạo thuocTinh2
        ...
        //Khởi tạo thuocTinhn
    }

    protected void processRequest(...) {...}
    @Override
    protected void doGet(...) {...}
    @Override
    protected void doPost(...) {...}
```

Chu kỳ sống của servlet



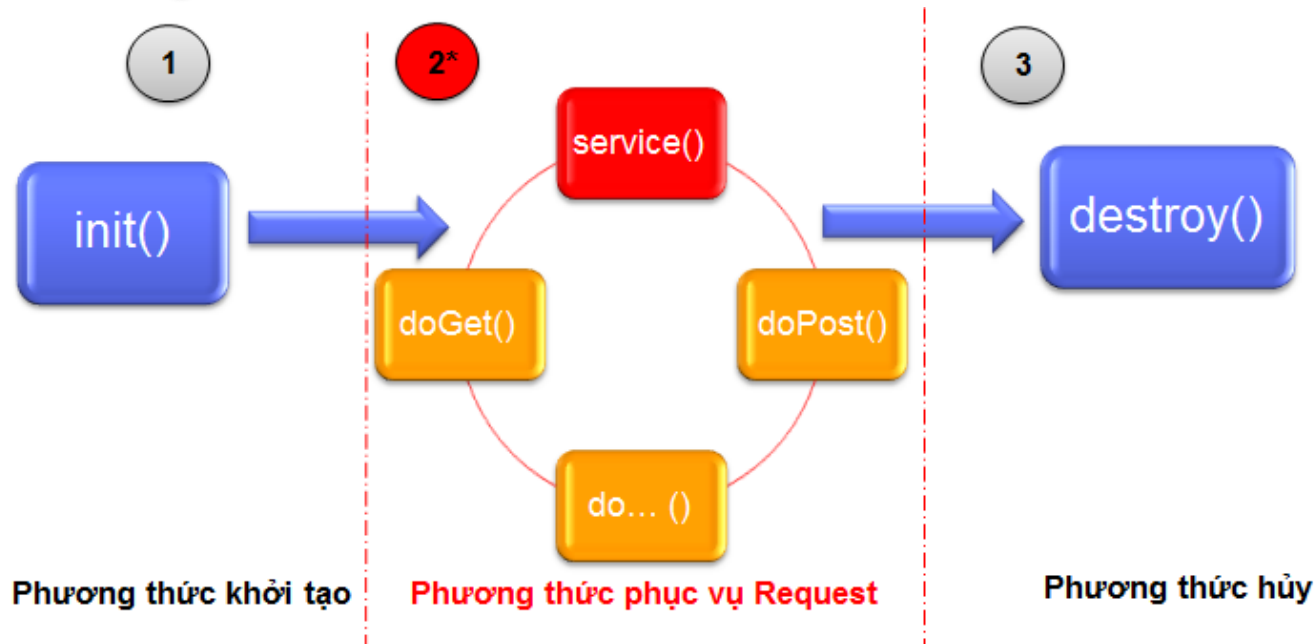
```
@WebServlet(name="ServletSample",
urlPatterns={"/ServletSample"})
public class ServletSample
    extends HttpServlet {
    private ArrayList<DanhMuc> ds;

    @Override
    public void init(){
        super.init();
        ds=DanhMucDAO.layDanhSachDanhMuc();
    }

    protected void processRequest(...) {
        Sử dụng ds
    }

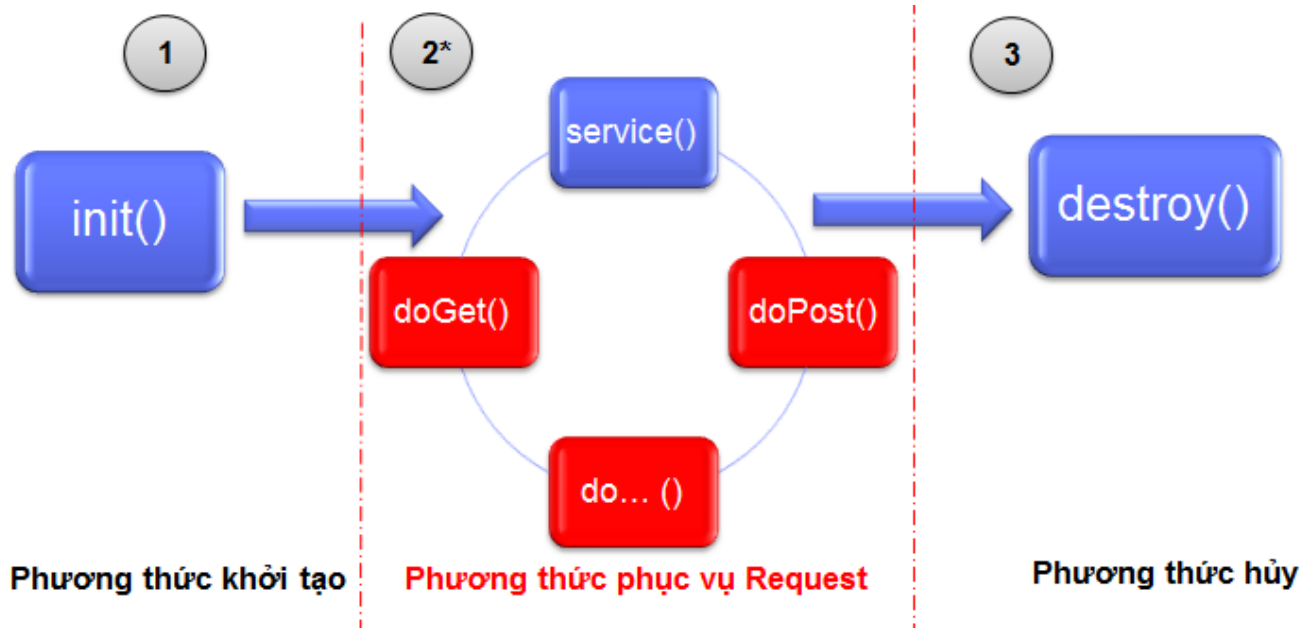
    @Override
    protected void doGet(...) {...}
    @Override
    protected void doPost(...) {...}
}
```

Chu kỳ sống của servlet



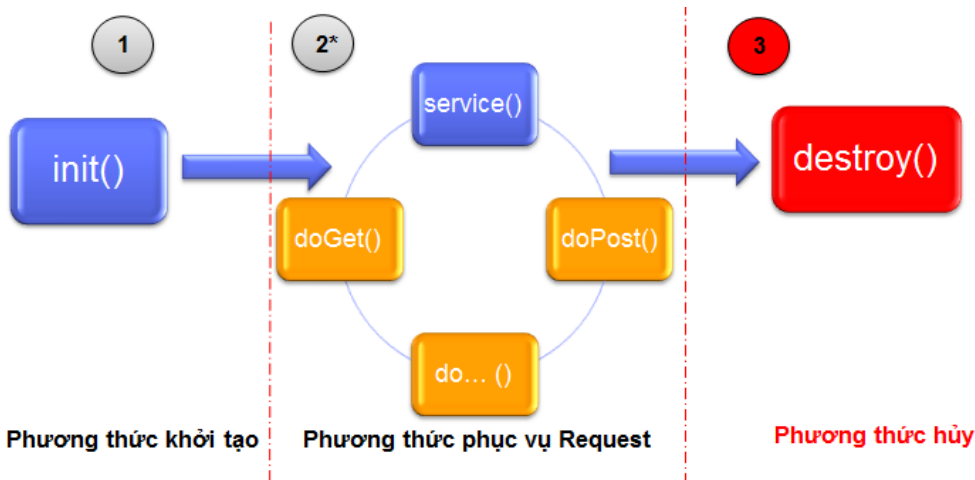
- Nhận xử lý tất cả các request từ client
- Phương thức **service** tự động được gọi thực hiện khi request từ client gửi đến
- Chuyển request của client đến **doGet()**, **doPost()**, **do... ()** tương ứng từ client
- Điều kiện: phương thức **init ()** đã được gọi
- **Không cài đặt lại phương thức này**

Chu kỳ sống của servlet



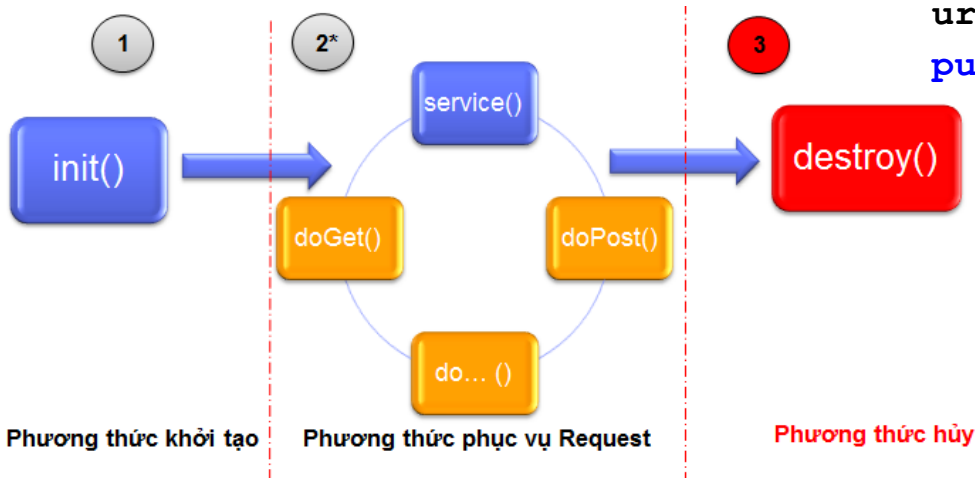
- Request được `service()` chuyển đến `doGet()`, `doPost()`, `do... ()` tương ứng phụ thuộc từ client
- Xử lý GET, POST, ... từ request
- Cần cài đặt lại phương thức `doGet()`, `doPost()`, `do... ()` để xử lý request tương ứng từ client

Chu kỳ sống của servlet



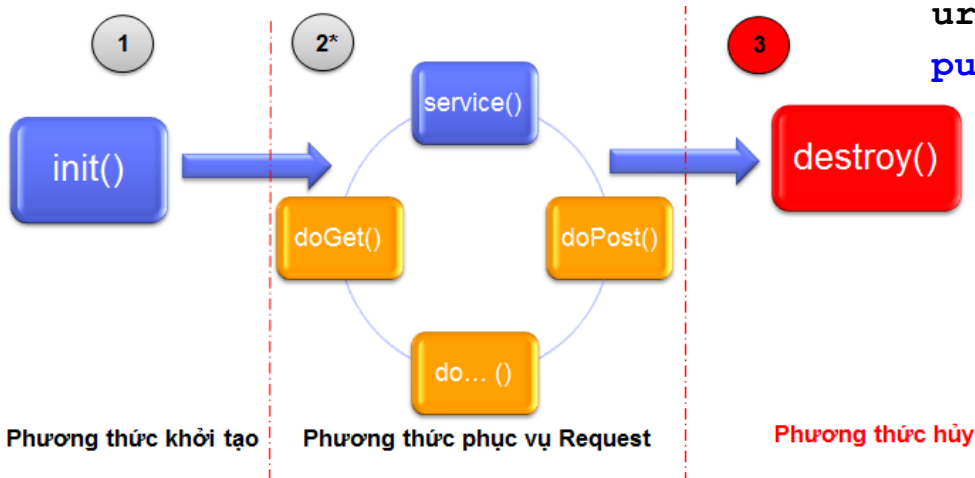
- Được tự động gọi thực hiện khi server hủy thể hiện của Servlet
- Được xem như là phương thức hủy

Chu kỳ sống của servlet



```
@WebServlet(name="ServletSample",
urlPatterns={"/ServletSample"})
public class ServletSample
    extends HttpServlet {
    private KieuDuLieu thuocTinh1;
    private KieuDuLieu thuocTinh2;
    ...
    private KieuDuLieu thuocTinhN;
    @Override
    public void init(){...}
    @Override
    public void destroy(){
        Giải phóng thuocTinh1, 2, ...,n
    }
    protected void processRequest(...){...}
    @Override
    protected void doGet(...){...}
    @Override
    protected void doPost(...){...}
}
```

Chu kỳ sống của servlet



```
@WebServlet(name="ServletSample",
urlPatterns={"/ServletSample"})
public class ServletSample
    extends HttpServlet {
    private ArrayList<DanhMucPOJO> ds;
    @Override
    public void init(){
        super.init();
        ds=DanhMucDAO.layDanhSachDanhMuc();
    }
    @Override
    public void destroy(){
        ds = null;
    }
    protected void processRequest(...) {...}
    @Override
    protected void doGet(...) {...}
    @Override
    protected void doPost(...) {...}
}
```

DEMO

Servlet Basic

Tài liệu tham khảo

- Nguyễn Hoàng Anh, Tập bài giảng môn chuyên đề Java, 2011
- Marty Hall, Tập bài giảng J2EE, 2010



HỎI VÀ ĐÁP