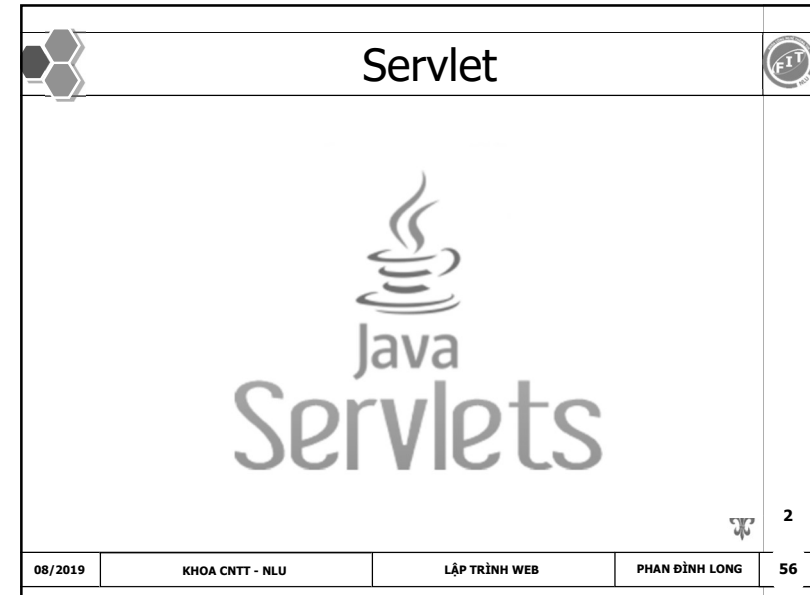
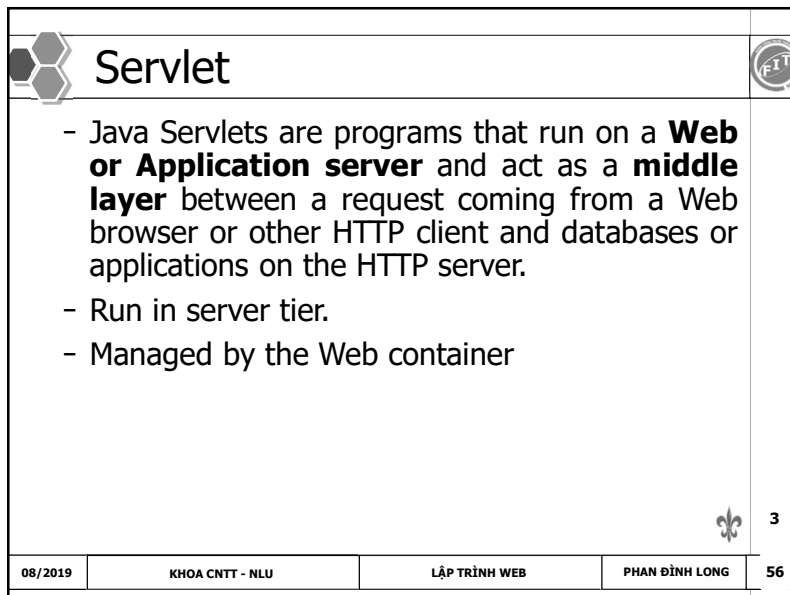


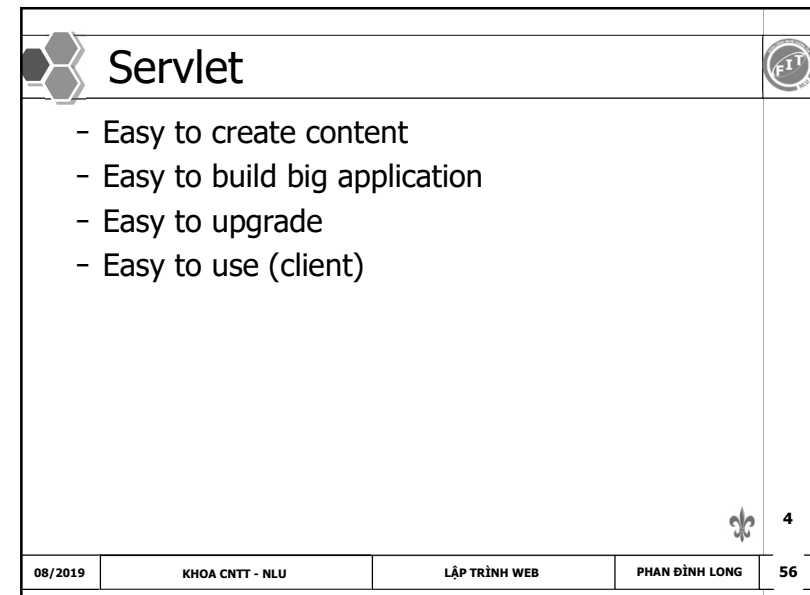
1



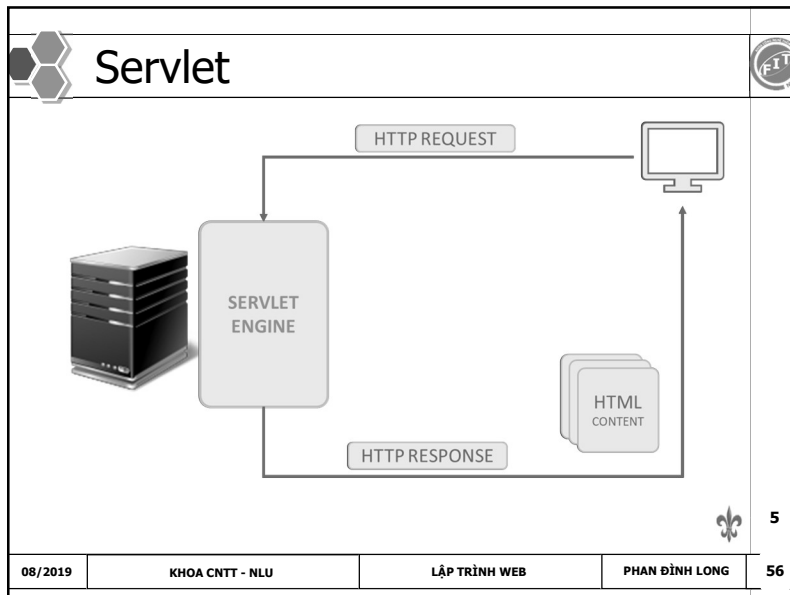
2



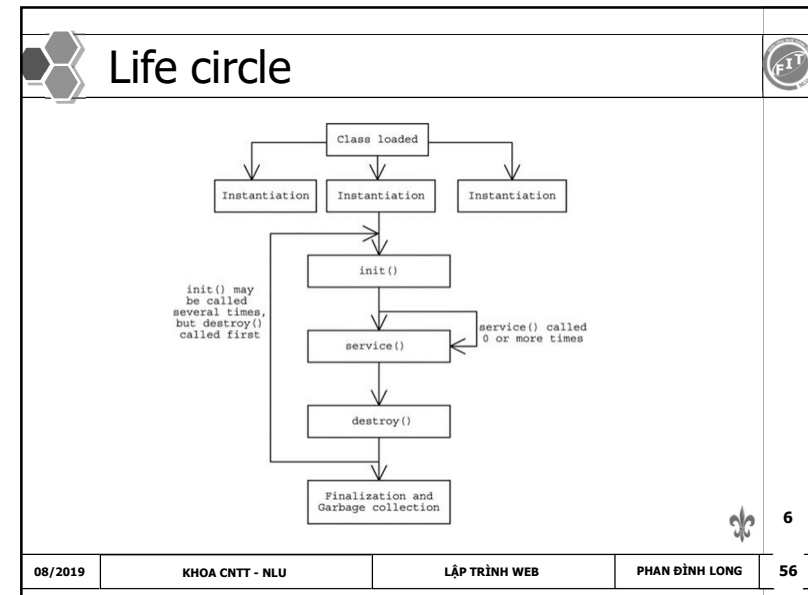
3



4



5



6

Servlet

- The lifecycle of a servlet is controlled by the container in which the servlet has been deployed. When a request is mapped to a servlet, the container performs the following steps.
- If an instance of the servlet does not exist, the web container:
 - Loads the servlet class
 - Creates an instance of the servlet class
 - Initializes the servlet instance by calling the `init` method (initialization is covered in Creating and Initializing a Servlet)

08/2019 KHOA CNTT - NLU LẬP TRÌNH WEB PHAN ĐÌNH LONG 56

7

Servlet

- The container invokes the `service` method, passing request and response objects. Service methods are discussed in Writing Service Methods.
- If it needs to remove the servlet, the container finalizes the servlet by calling the servlet's `destroy` method. For more information, see Finalizing a Servlet.
- When a servlet is first started up, its `init(ServletConfig config)` method is called
 - `init` should perform any necessary initializations
 - `init` is called only once, and does not need to be thread-safe

08/2019 KHOA CNTT - NLU LẬP TRÌNH WEB PHAN ĐÌNH LONG 56

8

Servlet

- Every servlet request results in a call to service(ServletRequest **request**, ServletResponse **response**)
 - service calls another method depending on the type of service requested
 - Usually you would override the called methods of interest, not service itself
 - service handles multiple simultaneous requests, so it and the methods it calls *must be thread safe*
- When the servlet is shut down, destroy() is called
 - destroy is called only once, but must be thread safe (because other threads may still be running)

9

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

9

Servlet

10

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

10

Servlet – getting start

- Setup environments
 - JDK 1.8 or higher
 - Tomcat 8.5 or higher
 - IntelliJ IDEA Community or higher
 - MySQL Server (can use XAMPP)
 - Navicat

11

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

11

Servlet – First Demo

- Open IDE and create new project

12

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

12

Servlet – First Demo

13

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

Servlet – First Demo

14

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

Servlet – First Demo

15

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

Servlet – First Demo

– Add Lib Servlet.

16

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

Servlet – First Demo

The screenshot shows the 'Project Structure' dialog in IntelliJ IDEA. The 'Libraries' tab is selected, and the 'Select Library Files' dialog is open. The 'lib' directory is highlighted under the 'tomcat-8.5' library.

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

17

Servlet – First Demo

The screenshot shows the 'Project Structure' dialog in IntelliJ IDEA. The 'Artifacts' tab is selected, and the 'lib' directory is being added to the artifact.

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

18

Servlet – First Demo

– Add Servlet Plugin -> Restart IED.

The screenshot shows the 'Settings' dialog in IntelliJ IDEA. The 'Plugins' tab is selected, and the 'Servlet' plugin is being installed.

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

19

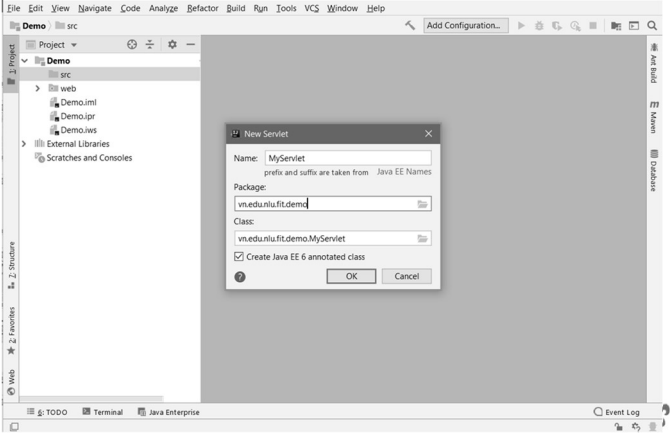
Servlet – First Demo

The screenshot shows the main window of IntelliJ IDEA. The 'New' menu is open, and the 'Create New Servlet' option is highlighted.

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

20

Servlet – First Demo



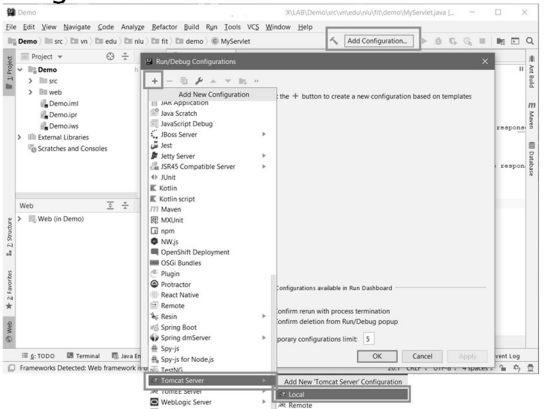
21

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

21

Servlet – First Demo

- Configure Tomcat server

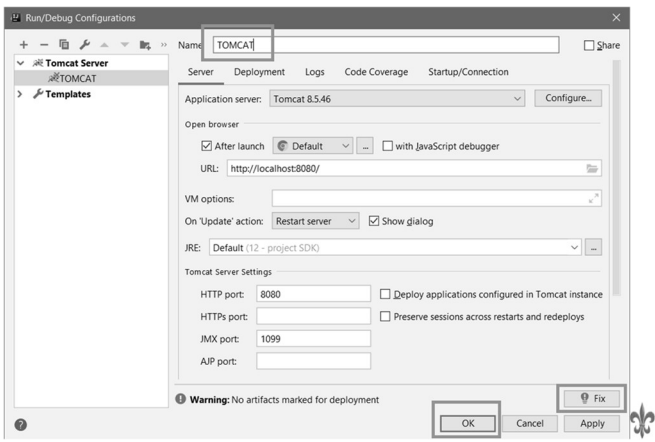


22

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

22

Servlet – First Demo



23

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

23

Requests, Responses and Headers

- HTTP is a simple, stateless protocol. A client, such as a web browser, makes a request, the web server responds, and the transaction is done.
- When the client sends a request, the first thing it specifies is an HTTP command, called a *method*, that tells the server the type of action it wants performed.
- After sending the request, the client can send optional header information to tell the server extra information about the request.

24

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

24

GET and POST

- When a client connects to a server and makes an HTTP request, the request can be of several different types, called methods.
- The most frequently used methods are GET and POST
 - HTTP **GET** requests include all required data in the URL.
 - HTTP **POST** requests supply additional data from the client (browser) to the server in the message body

25

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG
---------	-----------------	---------------	----------------

25

GET and POST

	GET	POST
BACK button /Reload	Harmless	Data will be re-submitted (the browser should alert the user that the data are about to be re-submitted)
Bookmarked	Can be bookmarked	Cannot be bookmarked
Cached	Can be cached	Not cached
Encoding type	application/x-www-form-urlencoded	application/x-www-form-urlencoded or multipart/form-data. Use multipart encoding for binary data
History	Parameters remain in browser history	Parameters are not saved in browser history
Restrictions on data length	Yes, when sending data, the GET method adds the data to the URL; and the length of a URL is limited (maximum URL length is 2048 characters)	No restrictions
Restrictions on data type	Only ASCII characters allowed	No restrictions. Binary data is also allowed
Security	GET is less secure compared to POST because data sent is part of the URL Never use GET when sending passwords or other sensitive information!	POST is a little safer than GET because the parameters are not stored in browser history or in web server logs
Visibility	Data is visible to everyone in the URL	Data is not displayed in the URL

26

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG
---------	-----------------	---------------	----------------

26

Servlet

- Every servlet must implement the javax.servlet.Servlet interface.
- HTTP servlet usually overrides doGet() to handle GET requests and doPost() to handle POST requests. An HTTP servlet can override either or both of these methods, depending on the type of requests it needs to handle.

27

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG
---------	-----------------	---------------	----------------




27

Servlet




28

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG
---------	-----------------	---------------	----------------




28

 <h2>Servlet</h2> 			
<ul style="list-style-type: none"> - doGet(HttpServletRequest req, HttpServletResponse resp) - Called by the server (via the service method) to allow a servlet to handle a GET request. - doPost(HttpServletRequest req, HttpServletResponse resp) - Called by the server (via the service method) to allow a servlet to handle a POST request. 			
 29			
08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG 56




29

 <h2>HttpServletRequest</h2> 			
<ul style="list-style-type: none"> - The HttpServletRequest represents the client's request. This object gives a servlet access to information about the client, the parameters for this request, the HTTP headers passed along with the request 			
 30			
08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG 56




30

 <h2>HttpServletRequest</h2> 			
<ul style="list-style-type: none"> - getAttribute(String name) - Returns the value of the named attribute as an Object, or null if no attribute of the given name exists. - getParameter(String name) - Returns the value of a request parameter as a String, or null if the parameter does not exist. - getParameterValues(String name) - Returns an array of String objects containing all of the values the given request parameter has, or null if the parameter does not exist. 			
 31			
08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG 56




31

 <h2>HttpServletRequest</h2> 			
<ul style="list-style-type: none"> - getRequestDispatcher(String path) - Returns a RequestDispatcher object that acts as a wrapper for the resource located at the given path. - setCharacterEncoding(String env) - Overrides the name of the character encoding used in the body of this request. - getSession() - Returns the current session associated with this request, or if the request does not have a session, creates one. 			
 32			
08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG 56




32

 <h2>HttpServletRequest</h2> 			
<ul style="list-style-type: none"> - setAttribute(String name, Object o) - Stores an attribute in this request. - removeAttribute(String name) - Removes an attribute from this request. - setCharacterEncoding(String env) - Overrides the name of the character encoding used in the body of this request. 			
 33			
08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG 56




33

 <h2>HTTP Request</h2> 			
<ul style="list-style-type: none"> - setAttribute(String name, Object o) - Stores an attribute in this request. - removeAttribute(String name) - Removes an attribute from this request. - setCharacterEncoding(String env) - Overrides the name of the character encoding used in the body of this request. 			
 34			
08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG 56

34

 <h2>HTTP Response</h2> 			
<ul style="list-style-type: none"> - The HttpServletResponse represents the servlet's response. A servlet can use this object to return data to the client. This data can be of any content type, though the type should be specified as part of the response. A servlet can also use this object to set HTTP response headers. 			
 35			
08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG 56

35

 <h2>HttpServletResponse</h2> 			
<ul style="list-style-type: none"> - sendRedirect(String location) - Sends a temporary redirect response to the client using the specified redirect location URL. - setCharacterEncoding(String charset) - Sets the character encoding (MIME charset) of the response being sent to the client, for example, to UTF-8. - getWriter() - Returns a PrintWriter object that can send character text to the client. 			
 36			
08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG 56

36

Forward page

- Redirect page and attack data, keep old data.

Page A
Add data to request

Page B
Keep old data from page A, add new data

Page C
Receive data from page A and B

08/2019 KHOA CNTT - NLU LẬP TRÌNH WEB PHAN ĐÌNH LONG 56

37

Redirect page.

- Redirect page to another web page with new data.

Page A
Add data to request

Page B
Processing of data from A, after redirect to C

Page C
Only receive data from page B

08/2019 KHOA CNTT - NLU LẬP TRÌNH WEB PHAN ĐÌNH LONG 56

38

Example

```
protected void doGet(HttpServletRequest request,
    HttpServletResponse response)
    throws ServletException, IOException {

    String uname = request.getParameter("uname");
    String pass = request.getParameter("pass");
    if(uname ??? && pass ???){
        response.sendRedirect("index.jsp");
    }else{
        request.getRequestDispatcher("login.jsp")
            .forward(request, response);
    }
}
```

08/2019 KHOA CNTT - NLU LẬP TRÌNH WEB PHAN ĐÌNH LONG 56

39

Response Status Codes

- Status code is a code of server sent to client.
- Status code is used to notify status of server

08/2019 KHOA CNTT - NLU LẬP TRÌNH WEB PHAN ĐÌNH LONG 56

40

Response Status Codes

- Status codes for HTTP 1.1:
 - **1xx** Informational: Request received, continuing process (the client needs to respond in some way)
 - **2xx** Successful
 - **3xx** Redirection: Further action must be taken in order to complete the request
 - **4xx** Incomplete: The request contains bad syntax or some other client error
 - **5xx** Server Error: The request is valid, but the server failed to fulfill the request
 - Default status code is 200 (OK)
- Status code constants are defined in **HttpServletResponse** beginning with **SC_**

41

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

41

Response Status Codes

- Status code constants are defined in **HttpServletResponse** beginning with **SC_**
 - 200 (SC_OK)
 - Success and document follows
 - Default for servlets
 - 302 (SC_MOVED_TEMPORARILY)
 - Requested document temporarily moved elsewhere
 - Browsers go to new location automatically
 - Servlets should use **sendRedirect**, not **setStatus**, when setting this header
 - 401 (SC_UNAUTHORIZED)
 - Browser tried to access password-protected page without proper Authorization header
 - 404 (SC_NOT_FOUND)
 - No such page

42

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

42

Response Status Codes

How to set status code?

- Method
 - **setStatus(int statusCode)**
- Can use **sendError** method if status code from form 400 - 599
 - **void sendError(int sc)**
 - The server may give the error special treatment
 - **void sendError(int code, String message)**
 - Wraps message inside small HTML document

43

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

43

web.xml File

- Web.xml is called as deployment descriptor file and its is is an XML file that contains information on the configuration of the web application, including the configuration of servlets.

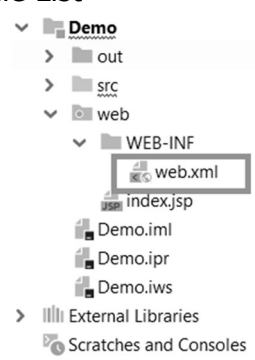
44

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

44

web.xml File

- Servlet Mapping
- Default Welcome File List
- Error page
- Fitter
- Context-param
- Permission



45

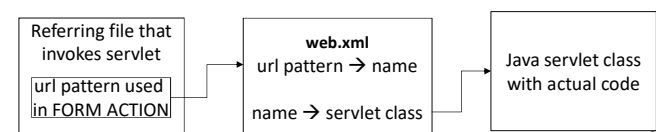
08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

45

Servlet Mapping

Change Url servlet /login → /dangnhap

```
<servlet>
  <servlet-name>login</servlet-name>
  <servlet-class>login</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>login</servlet-name>
  <url-pattern>/dangnhap</url-pattern>
</servlet-mapping>
```



46

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

46

Error page

```
<error-page>
  <error-code>401</error-code>
  <location>/error401.html</location>
</error-page>
<error-page>
  <error-code>404</error-code>
  <location>/error404.html</location>
</error-page>
<error-page>
  <error-code>500</error-code>
  <location>/error500.html</location>
</error-page>
<error-page>
  <error-code>503</error-code>
  <location>/error503.html</location>
</error-page>
```

47

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

47

Initial Parameters



- Using to config some value using in application. You can config value using in a single page or using in all page.
- EX:
 - Using in sign page:


```
<servlet>
  <servlet-name>login</servlet-name>
  <servlet-class>login</servlet-class>
  <init-param>
    <param-name>title</param-name>
    <param-value>Baby Shop</param-value>
  </init-param>
</servlet>
```
 - Using value: `String title= getServletConfig().getInitParameter("title");`
 - Use only in login page

48

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

48

- Value using in all page


```
<context-param>
  <param-name>title</param-name>
  <param-value>Shop đồ trẻ em</param-value>
</context-param>
```
- Using value
 - Servlet:


```
ServletConfig config = getServletConfig();
ServletContext context = config.getServletContext();
String value = context.getInitParameter("title");
```
 - In jsp page


```
String value = application.getInitParameter("title");
```

08/2019



KHOA CNTT - NLU

LẬP TRÌNH WEB

PHAN ĐÌNH LONG

56

49

Servlet Fitter

- A **filter** is an object that is invoked at the preprocessing and postprocessing of a request.
- It can also intercept the response and do post-processing before sending to the client in web application.
- Servlet Filters are **pluggable** java components that we can use to intercept and process requests *before* they are sent to servlets and response *after* servlet code is finished and before container sends the response back to the client.

08/2019



KHOA CNTT - NLU

LẬP TRÌNH WEB

PHAN ĐÌNH LONG

56

50

Servlet Filter

- **void init(FilterConfig paramFilterConfig)** method is called only once in the lifecycle of filter and we should initialize any resources in this method
- **doFilter(ServletRequest request, ServletResponse response, FilterChain fe)** the method invoked every time by container when it has to apply filter to a resource
- **void destroy()**

08/2019



KHOA CNTT - NLU

LẬP TRÌNH WEB

PHAN ĐÌNH LONG

56

51

Fitter

```
<filter>
  <filter-name>utf8</filter-name>
  <filter-class>utf8</filter-class>
</filter>
<filter-mapping>
  <filter-name>utf8</filter-name>
  <url-pattern>/*</url-pattern>
</filter-mapping>
```

08/2019



KHOA CNTT - NLU

LẬP TRÌNH WEB

PHAN ĐÌNH LONG

56

52





```

public class utf8 implements Filter {
    @Override
    public void destroy() {}
    @Override
    public void doFilter(ServletRequest request,
        ServletResponse response, FilterChain arg2)
        throws IOException, ServletException {



        request.setCharacterEncoding("utf8");
        response.setCharacterEncoding("utf8");
        arg2.doFilter(request, response);
    }
    @Override
    public void init(FilterConfig filterConfig) throws
        ServletException {}
}

```



53

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

53






- All of page in application (jsp and servlet) will run file utf8.java first and request and response will encode to utf8.



54

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

54

DEMO


55

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

55


Q & A





56

08/2019	KHOA CNTT - NLU	LẬP TRÌNH WEB	PHAN ĐÌNH LONG	56
---------	-----------------	---------------	----------------	----

56