

# プログラミング Java 3

## 第4回： サーブレットのHTTP Request入門

Ivan Tanev



# アウトライン

## 1. 第3回のまとめ

## 2. サーブレットのHTTP Request入門

2.1) User's inputコンポーネント:<INPUT>HTMLタグ

2.2) HTTP Requestの構造

2.3) HTTP Requestの処理

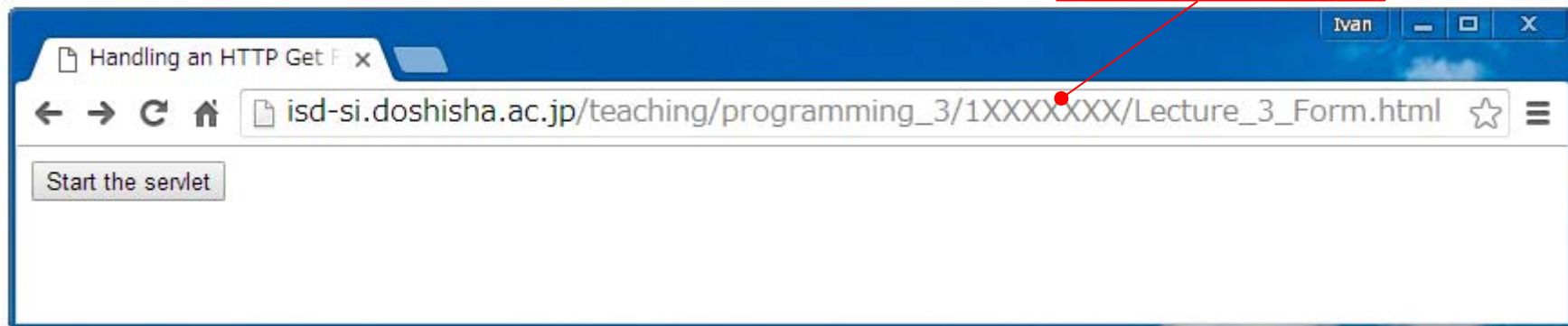
## 3. 演習

# 1. 第3回のまとめ



Web Browser: [http://isd-si.doshisha.ac.jp/teaching/Programming\\_3/1XXXXXXXX/Lecture\\_3\\_Form.html](http://isd-si.doshisha.ac.jp/teaching/Programming_3/1XXXXXXXX/Lecture_3_Form.html)

学籍番号

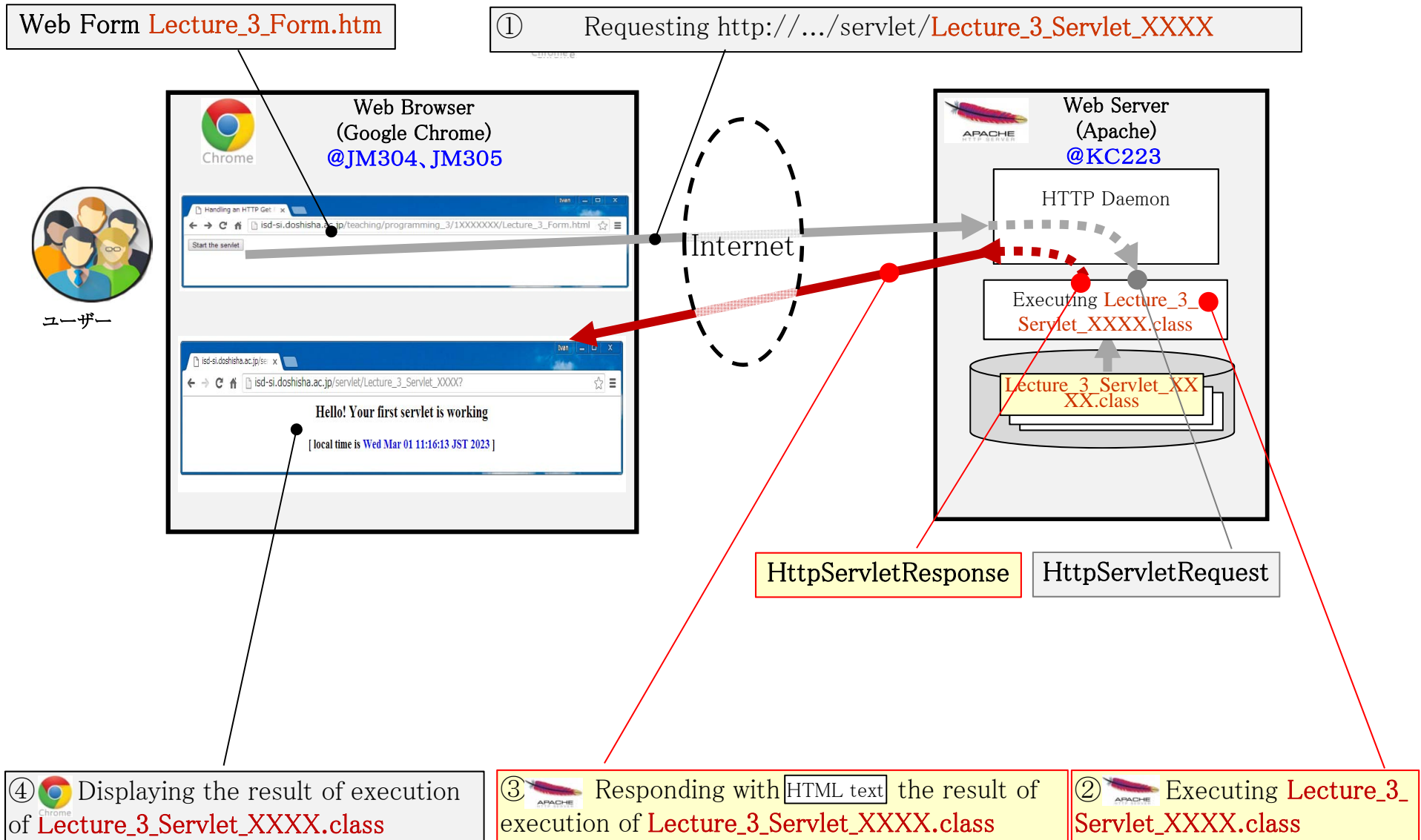


Lecture\_3\_Form.htmlの画面:サーブレットLecture\_3\_Servlet\_XXXX呼び出す前

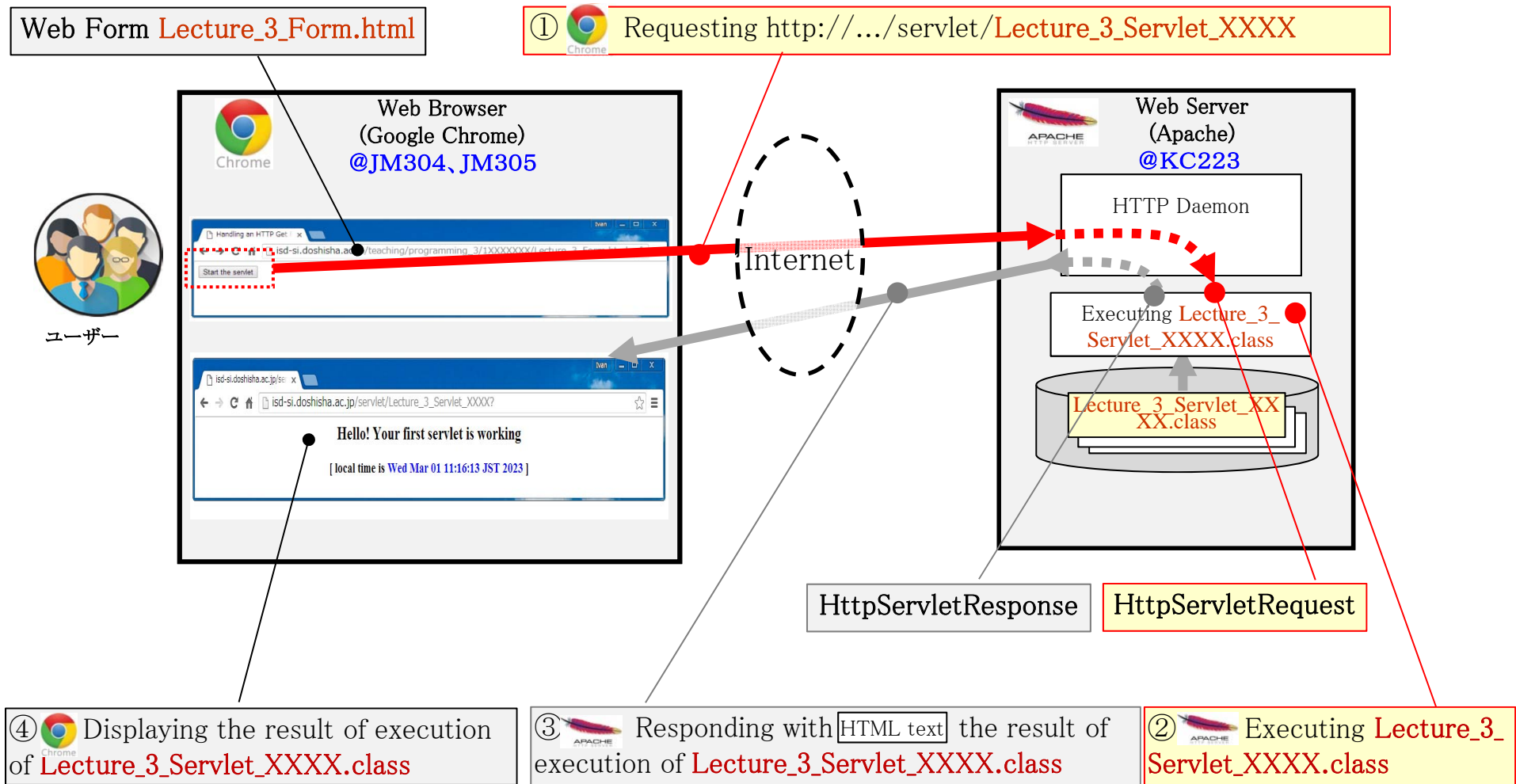


サーブレットLecture\_3\_Servlet\_XXXX呼び出す後

# 1. 第3回のまとめ



## 2. サーブレットのHTTP Request入門

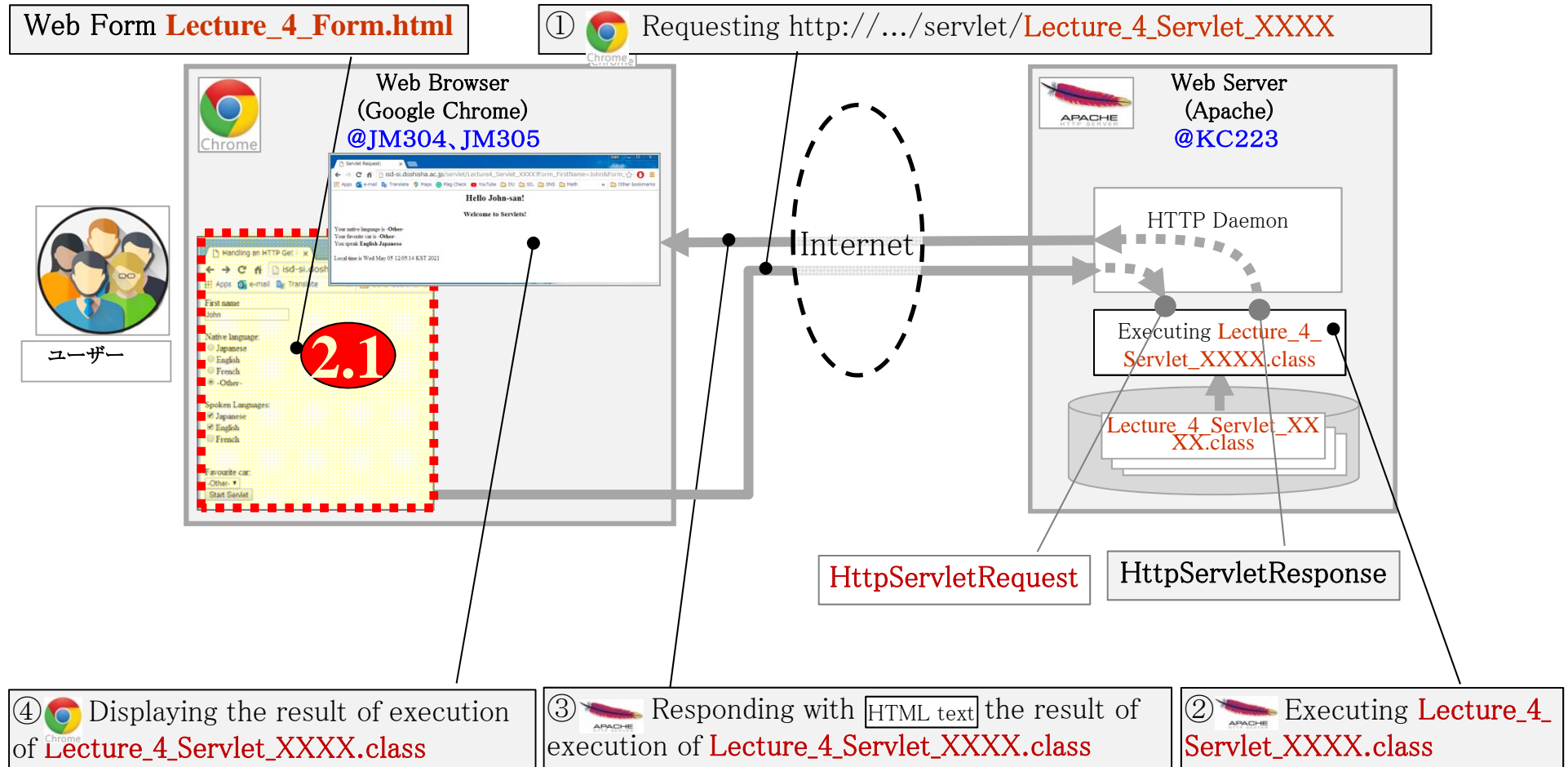


Q: サーブレットのHTTP Requestの処理の目的は？

A: ユーザーの入力 (User's input) を処理します。

# 2. サーブレットのHTTP Request入門

## 2.1) User's inputコンポーネント:<INPUT>HTMLタグ



# 2. サーブレットのHTTP Request入門

## 2.1) User's inputコンポーネント:<INPUT>HTMLタグ



Handling an HTTP Get F x

isd-si.d

Apps e-mail Translat

First name  
John

Native language:  
 Japanese  
 English  
 French  
 -Other-

Spoken Language  
 Japanese  
 English  
 French

Favourite car:  
-Other- ▼

Start Servlet

Lecture\_4\_Form.htmlのソース

```
1 <HTML>
2 <HEAD>
3 <TITLE>Handling an HTTP Get Request</TITLE>
4 <META HTTP-EQUIV="content-type" CONTENT="text/html">
5 </HEAD>
6 <BODY>
7 <FORM ACTION = "/servlet/Lecture_4_Servlet_XXXX" METHOD = "get">
8 First name<BR>
9 <INPUT TYPE = "text" NAME = "Form_FirstName" /><BR><BR>
10 Native language:<BR>
11 <INPUT TYPE="RADIO" NAME = "Form_Language" VALUE="Japanese" CHECKED="CHECKED">Japanese<BR>
12 <INPUT TYPE="RADIO" NAME = "Form_Language" VALUE="English">English<BR>
13 <INPUT TYPE="RADIO" NAME = "Form_Language" VALUE="French">French<BR>
14 <INPUT TYPE="RADIO" NAME = "Form_Language" VALUE="-Other-">-Other-<BR><BR>
15 Spoken languages: <BR>
16 <INPUT TYPE="CHECKBOX" NAME="Form_SpeakJapanese" CHECKED="CHECKED">Japanese<BR>
17 <INPUT TYPE="CHECKBOX" NAME="Form_SpeakEnglish">English<BR>
18 <INPUT TYPE="CHECKBOX" NAME="Form_SpeakFrench">French<BR>
19 <BR>
20 <BR>
21 Favourite car:<BR>
22 <SELECT NAME="Form_Cars">
23 <OPTION VALUE="Toyota">Toyota
24 <OPTION VALUE="Honda">Honda
25 <OPTION VALUE="Nissan">Nissan
26 <OPTION VALUE="-Other-">-Other-
27 </SELECT><BR>
28 <INPUT TYPE = "submit" VALUE = "Start Servlet"/>
29 </FORM>
30 </BODY>
31 </HTML>
```

2.1

# 2. サーブレットのHTTP Request入門

## 2.1) User's inputコンポーネント:<INPUT>HTMLタグ

 [https://www.w3schools.com/tags/tag\\_input.asp](https://www.w3schools.com/tags/tag_input.asp)

### Definition and Usage

The `<input>` tag specifies an input field where the user can enter data.

The `<input>` element is the most important form element.

The `<input>` element can be displayed in several ways, depending on the type attribute.

The different input types are as follows:

- `<input type="button">`
- `<input type="checkbox">`
- `<input type="color">`
- `<input type="date">`
- `<input type="datetime-local">`
- `<input type="email">`
- `<input type="file">`
- `<input type="hidden">`
- `<input type="image">`
- `<input type="month">`
- `<input type="number">`
- `<input type="password">`
- `<input type="radio">`
- `<input type="range">`
- `<input type="reset">`
- `<input type="search">`
- `<input type="submit">`
- `<input type="tel">`
- `<input type="text">` (default value)
- `<input type="time">`
- `<input type="url">`
- `<input type="week">`

### 例:Lecture\_4\_Form.htmlのソース

```
1 <HTML>
2 <HEAD>
3 <TITLE>Handling an HTTP Get Request
4 <META HTTP-EQUIV="content-type" (
5 </HEAD>
6 <BODY>
7 <FORM ACTION = "/servlet/Lecture_
8 First name<BR>
9 <INPUT TYPE = "text" NAME = "Form_
10 Native language:<BR>
11 <INPUT TYPE="RADIO" NAME = "Form_
12 <INPUT TYPE="RADIO" NAME = "Form_
13 <INPUT TYPE="RADIO" NAME = "Form_
14 <INPUT TYPE="RADIO" NAME = "Form_
15 Spoken Languages: <BR>
16 <INPUT TYPE="CHECKBOX" NAME="Form_
17 <INPUT TYPE="CHECKBOX" NAME="Form_
18 <INPUT TYPE="CHECKBOX" NAME="Form_
19 <BR>
20 <BR>
21 Favourite car:<BR>
22 <SELECT NAME="Form_Cars">
23 <OPTION VALUE="Toyota">Toyota
24 <OPTION VALUE="Honda">Honda
25 <OPTION VALUE="Nissan">Nissan
26 <OPTION VALUE="-Other-">-Other-
27 </SELECT><BR>
28 <INPUT TYPE = "submit" VALUE = "
29 </FORM>
30 </BODY>
31 </HTML>
```

# 2. サーブレットのHTTP Request入門

## 2.2) HTTP Requestの構造

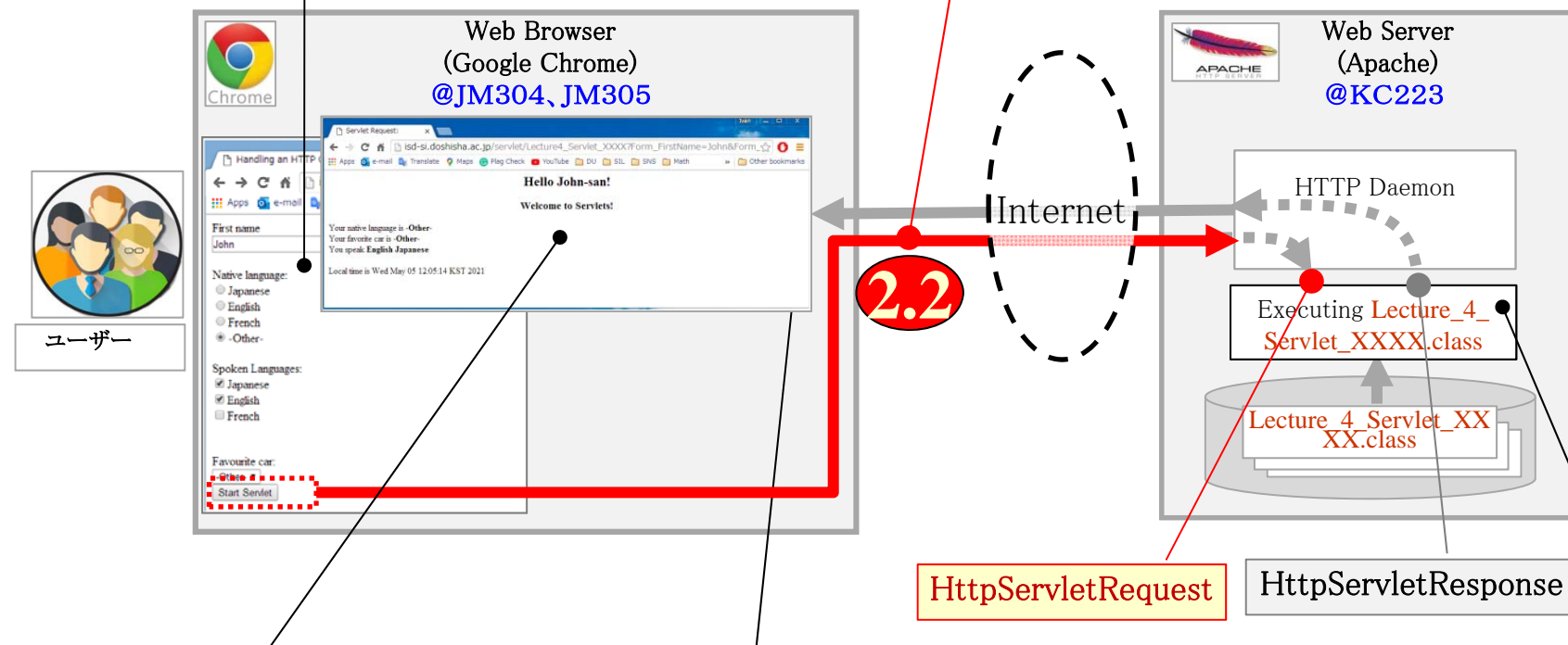
①  Requesting `http://.../servlet/Lecture_4_Servlet_XXXX`

### 2.2

### HTTP Requestの構造

`isd-si.doshisha.ac.jp/servlet/Lecture_4_Servlet_XXXX?Form_FirstName=John&Form_Language=-Other-&Form_SpeakJapanese=on&Form_SpeakEnglish=on&Form_Cars=-Other-`

### Web Form `Lecture_4_Form.html`



④  Displaying the result of execution of `Lecture_4_Servlet_XXXX.class`

③  Responding with `HTML text` the result of execution of `Lecture_4_Servlet_XXXX.class`

②  Executing `Lecture_4_Servlet_XXXX.class`

# 2. サーブレットのHTTP Request入門

## 2.2) HTTP Requestの構造

Web form Lecture\_4\_Form.html の ソース

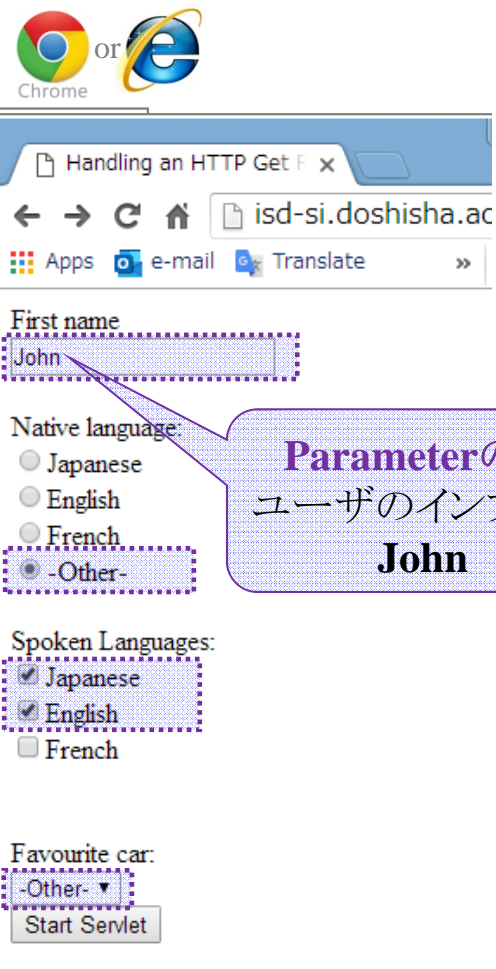
```

1 <HTML>
2 <HEAD>
3 <TITLE>Handling an HTTP Get Request</TITLE>
4 <META HTTP-EQUIV="content-type" CONTENT="text/html">
5 </HEAD>
6 <BODY>
7 <FORM ACTION = "/servlet/Lecture_4_Servlet_XXXX" METHOD = "get">
8 First name<BR>
9 <INPUT TYPE = "text" NAME = "Form_FirstName" /><BR><BR>
10 Native language:<BR>
11 <INPUT TYPE="RADIO" NAME = "Form_Language" VALUE="Japanese" CHECKED="CHECKED">Japanese<BR>
12 <INPUT TYPE="RADIO" NAME = "Form_Language" VALUE="English">English<BR>
13 <INPUT TYPE="RADIO" NAME = "Form_Language" VALUE="French">French<BR>
14 <INPUT TYPE="RADIO" NAME = "Form_Language" VALUE="-Other-">-Other-<BR><BR>
15 Spoken Languages: <BR>
16 <INPUT TYPE="CHECKBOX" NAME="Form_SpeakJapanese" CHECKED="CHECKED">Japanese<BR>
17 <INPUT TYPE="CHECKBOX" NAME="Form_SpeakEnglish">English<BR>
18 <INPUT TYPE="CHECKBOX" NAME="Form_SpeakFrench">French<BR>
19 <BR>
20 <BR>
21 Favourite car:<BR>
22 <SELECT NAME="Form_Cars">
23 <OPTION VALUE="Toyota">Toyota
24 <OPTION VALUE="Honda">Honda
25 <OPTION VALUE="Nissan">Nissan
26 <OPTION VALUE="-Other-">-Other-
27 </SELECT><BR>
28 <INPUT TYPE = "submit" VALUE = "Start Servlet"/>
29 </FORM>
30 </BODY>
31 </HTML>

```

**Parameter名**  
(NAMEアトリビュート)

**Parameterの値**  
ユーザのインプット:  
**John**

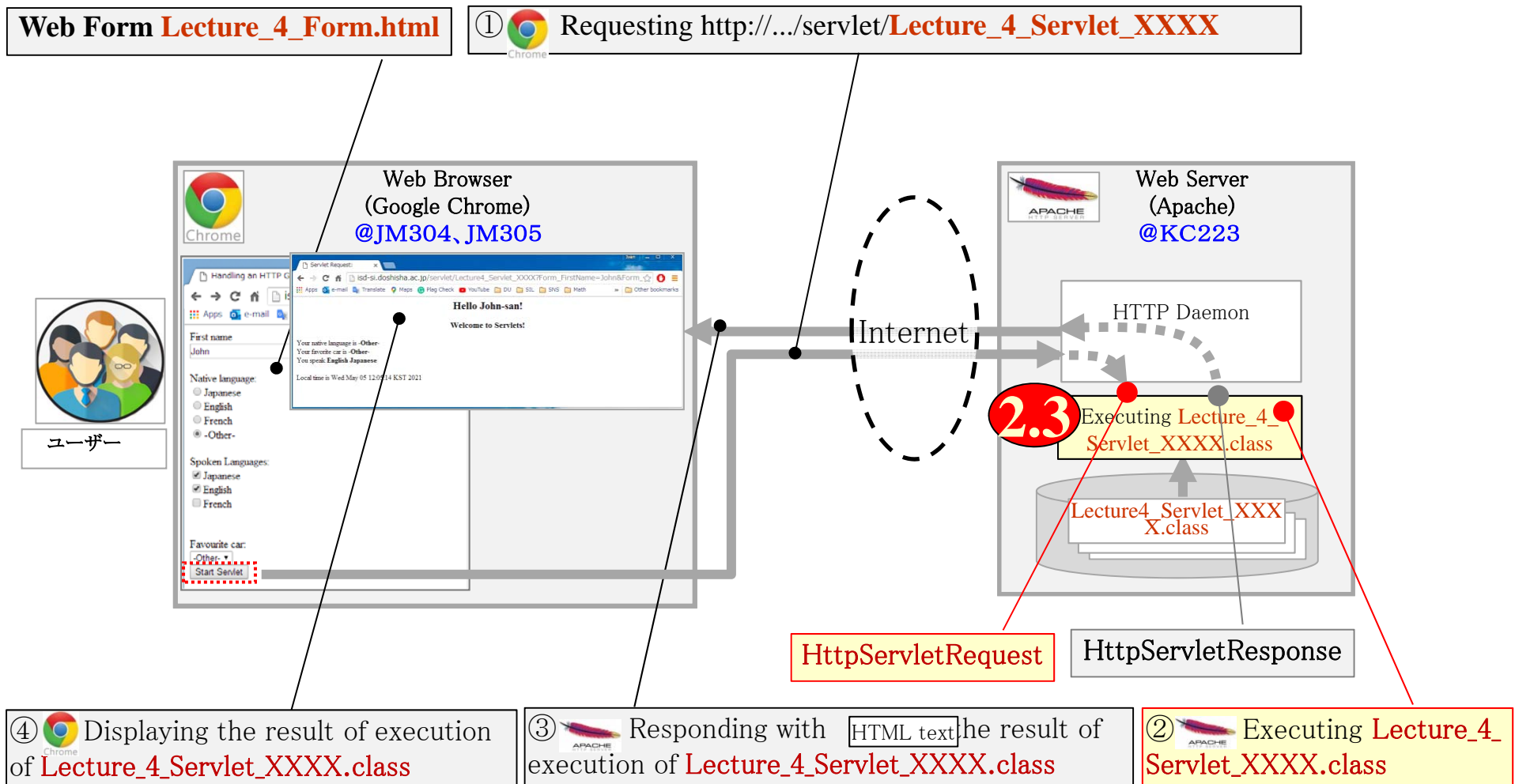


### 2.2 Requestの構造



# 2. サーブレットのHTTP Request入門

## 2.3) HTTP Requestの処理



# 2. サーブレットのHTTP Request入門

## 2.3) HTTP Requestの処理



Requestの内容:



HTTP Requestの処理:

Servlet Lecture\_4\_Servlet\_XXXX.java の ソース (1 of 2): HTTP Requestの処理

```
1 import java.io.*;
2 import javax.servlet.*;
3 import javax.servlet.http.*;
4 public class Lecture_4_Servlet_XXXX extends HttpServlet {
5     public void service (HttpServletRequest request, HttpServletResponse response)
6     throws ServletException {
7         response.setContentType("text/html");
8         // Getting the values of parameters of Web Form:
9         String firstName = request.getParameter("Form_FirstName");
10        String nativeLang = request.getParameter("Form_Language");
11        String car = request.getParameter("Form_Cars");
12        String spokenLang = "";
13        if (request.getParameter("Form_SpeakEnglish") != null)
14        if (request.getParameter("Form_SpeakJapanese") != null)
15        if (request.getParameter("Form_SpeakFrench") != null)
16        // Getting the communication channel with the request
17        PrintWriter out = response.getWriter();
```

2.3

firstName="John"


Parameter名

Servlet Lecture\_4\_Servlet\_XXXX.java の ソース (2 of 2): HTTP Responseの作成

```
19 // Writing the response to be sent to Web Browser
20 out.println("<HTML><HEAD><TITLE>Servlet Request: </TITLE>"
21 + " <META HTTP-EQUIV='Content-Type' Content='text/html';"
22 + " charset='SHIFT_JIS'</META></HEAD>");
23 out.println(" <CENTER>"
24 + "<H2>Hello "+firstName+"-san!</H2>"
25 + "<H3>Welcome to Servlets!</H3></CENTER>"
26 + " Your native language is <B>"+nativeLang+"</B><BR>"
27 + " You speak <B>"+ spokenLang + "</B><BR>"
28 + " Your favorite car is <B>"+car+"</B><BR><BR>");
29 out.println("Local time is " + new java.util.Date());
30 out.println("</BODY></HTML>");
31 }
32 }
```

## 2. サーブレットのHTTP Request入門

### 2.3) HTTP Requestの処理

 <https://docs.oracle.com/javaee/1.3/api/javax/servlet/ServletRequest.html>

javax.servlet

### Interface ServletRequest

All Known Subinterfaces:

[HttpServletRequest](#)

All Known Implementing Classes:

[ServletRequestWrapper](#)

---

public interface **ServletRequest**

Defines an object to provide client request information to a servlet. The servlet container creates a `ServletRequest` object and passes it as an argument to the servlet's `service` method.


A `ServletRequest` object provides data including parameter name and values, attributes, and an input stream. Interfaces that extend `ServletRequest` can provide additional protocol-specific data (for example, HTTP data is provided by [HttpServletRequest](#)).

See Also:

[HttpServletRequest](#)

# 2. サーブレットのHTTP Request入門

## 2.3) HTTP Requestの処理

 <https://docs.oracle.com/javaee/1.3/api/javax/servlet/ServletRequest.html>

javax.servlet

### Interface ServletRequest

Method Summary	
java.lang.Object	<b><a href="#">getAttribute</a></b> (java.lang.String name) Returns the value of the named attribute as an Object, or null if no attribute of the given name exists.
java.util.Enumeration	<b><a href="#">getAttributeNames</a></b> () Returns an Enumeration containing the names of the attributes available to this request.
java.lang.String	<b><a href="#">getCharacterEncoding</a></b> () Returns the name of the character encoding used in the body of this request.
int	<b><a href="#">getContentTypeLength</a></b> () Returns the length, in bytes, of the request body and made available by the input stream, or -1 if the length is not known.
java.lang.String	<b><a href="#">getContentType</a></b> () Returns the MIME type of the body of the request, or null if the type is not known.
<a href="#">ServletInputStream</a>	<b><a href="#">getInputStream</a></b> () Retrieves the body of the request as binary data using a <a href="#">ServletInputStream</a> .
java.util.Locale	<b><a href="#">getLocale</a></b> () Returns the preferred Locale that the client will accept content in, based on the Accept-Language header.
java.util.Enumeration	<b><a href="#">getLocales</a></b> () Returns an Enumeration of Locale objects indicating, in decreasing order starting with the preferred locale, the locales that are acceptable to the client based on the Accept-Language header.
java.lang.String	<b><a href="#">getParameter</a></b> (java.lang.String name) Returns the value of a request parameter as a String, or null if the parameter does not exist.
java.util.Map	<b><a href="#">getParameterMap</a></b> () Returns a java.util.Map of the parameters of this request.

## 2. サーブレットのHTTP Request入門

### 2.3) HTTP Requestの処理



<https://docs.oracle.com/javaee/1.3/api/javax/servlet/ServletRequest.html>

#### **getParameter**

```
public java.lang.String getParameter(java.lang.String name)
```

Returns the value of a request parameter as a `String`, or `null` if the parameter does not exist. Request parameters are extra information sent with the request. For HTTP servlets, parameters are contained in the query string or posted form data.

You should only use this method when you are sure the parameter has only one value. If the parameter might have more than one value, use [getParameterValues\(java.lang.String\)](#).

If you use this method with a multivalued parameter, the value returned is equal to the first value in the array returned by `getParameterValues`.

If the parameter data was sent in the request body, such as occurs with an HTTP POST request, then reading the body directly via [getInputStream\(\)](#) or [getReader\(\)](#) can interfere with the execution of this method.

#### **Parameters:**

`name` - a `String` specifying the name of the parameter

#### **Returns:**

a `String` representing the single value of the parameter

#### **See Also:**

[getParameterValues\(java.lang.String\)](#)

## 2. サーブレットのHTTP Request入門

### 2.3) HTTP Requestの処理



The image shows two overlapping browser windows. The top window, titled "Handling an HTTP Get F x", displays a form with the following fields:

- First name: John
- Native language: Radio buttons for Japanese, English, French, and -Other- (selected).
- Spoken Languages: Checkboxes for Japanese, English (checked), and French.
- Favourite car: A dropdown menu showing "-Other-" and a "Start Servlet" button.

The bottom window, titled "Servlet Request:", shows the response from the servlet. The URL in the address bar is `isd-si.doshisha.ac.jp/servlet/Lecture_4_Servlet_XXXX?Form_FirstName=John&Form_La`. The response content is:









**Hello John-san!**

**Welcome to Servlets!**

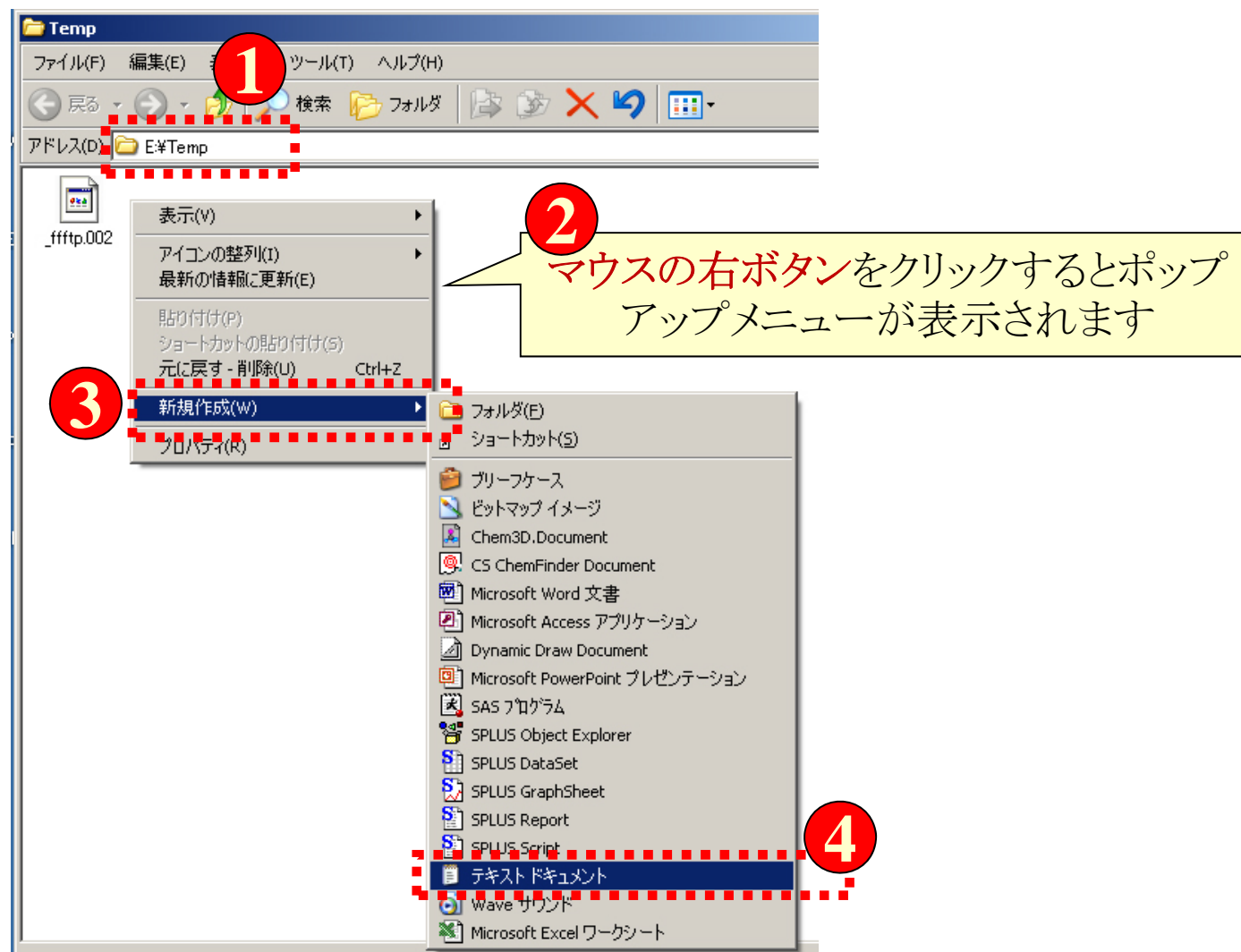
Your native language is **-Other-**  
You speak **English Japanese**  
Your favorite car is **-Other-**

Local time is Wed Mar 01 14:59:33 JST 2023

# 3. 演習

- A.  Webフォームの**開発** (**Notepad**)
- B.  WebフォームをWebサーバへの**Upload** (**FTP Client**)
- C.  Servletの**開発** (**Notepad**)
- D.  ServletをWebサーバへの**Upload** (**FTP Client**)
- E.  or  Servletの**コンパイル** (**CompileServlet.jsp via Web Browser**)
- F.  or  Webフォームの**Access** (**Web Browser**)

# A. Webフォームの開発



# A. Webフォームの開発



下記の内容を入力:

Lecture\_4\_Servlet\_XXXX:  
XXXX = 学籍番号  
例: Lecture\_4\_Servlet\_1116221001

```
1 <HTML>
2 <HEAD>
3 <TITLE>Handling an HTTP Get Request</TITLE>
4 <META HTTP-EQUIV="content-type" CONTENT="text/html">
5 </HEAD>
6 <BODY>
7 <FORM ACTION = "/servlet/Lecture_4_Servlet_XXXX" METHOD = "get">
8 First name<BR>
9 <INPUT TYPE = "text" NAME = "Form_FirstName" /><BR><BR>
10 Native language:<BR>
11 <INPUT TYPE="RADIO" NAME = "Form_Language" VALUE="Japanese" CHECKED="CHECKED">Japanese<BR>
12 <INPUT TYPE="RADIO" NAME = "Form_Language" VALUE="English">English<BR>
13 <INPUT TYPE="RADIO" NAME = "Form_Language" VALUE="French">French<BR>
14 <INPUT TYPE="RADIO" NAME = "Form_Language" VALUE="-Other-">-Other-<BR><BR>
15 Spoken Languages: <BR>
16 <INPUT TYPE="CHECKBOX" NAME="Form_SpeakJapanese" CHECKED="CHECKED">Japanese<BR>
17 <INPUT TYPE="CHECKBOX" NAME="Form_SpeakEnglish">English<BR>
18 <INPUT TYPE="CHECKBOX" NAME="Form_SpeakFrench">French<BR>
19 <BR>
20 <BR>
21 Favourite car:<BR>
22 <SELECT NAME="Form_Cars">
23 <OPTION VALUE="Toyota">Toyota
24 <OPTION VALUE="Honda">Honda
25 <OPTION VALUE="Nissan">Nissan
26 <OPTION VALUE="-Other-">-Other-
27 </SELECT><BR>
28 <INPUT TYPE = "submit" VALUE = "Start Servlet"/>
29 </FORM>
30 </BODY>
31 </HTML>
```

参考: Web Browser画面

isd-si.do

First name  
John

Native language:  
 Japanese  
 English  
 French  
 -Other-

Spoken Languages:  
 Japanese  
 English  
 French

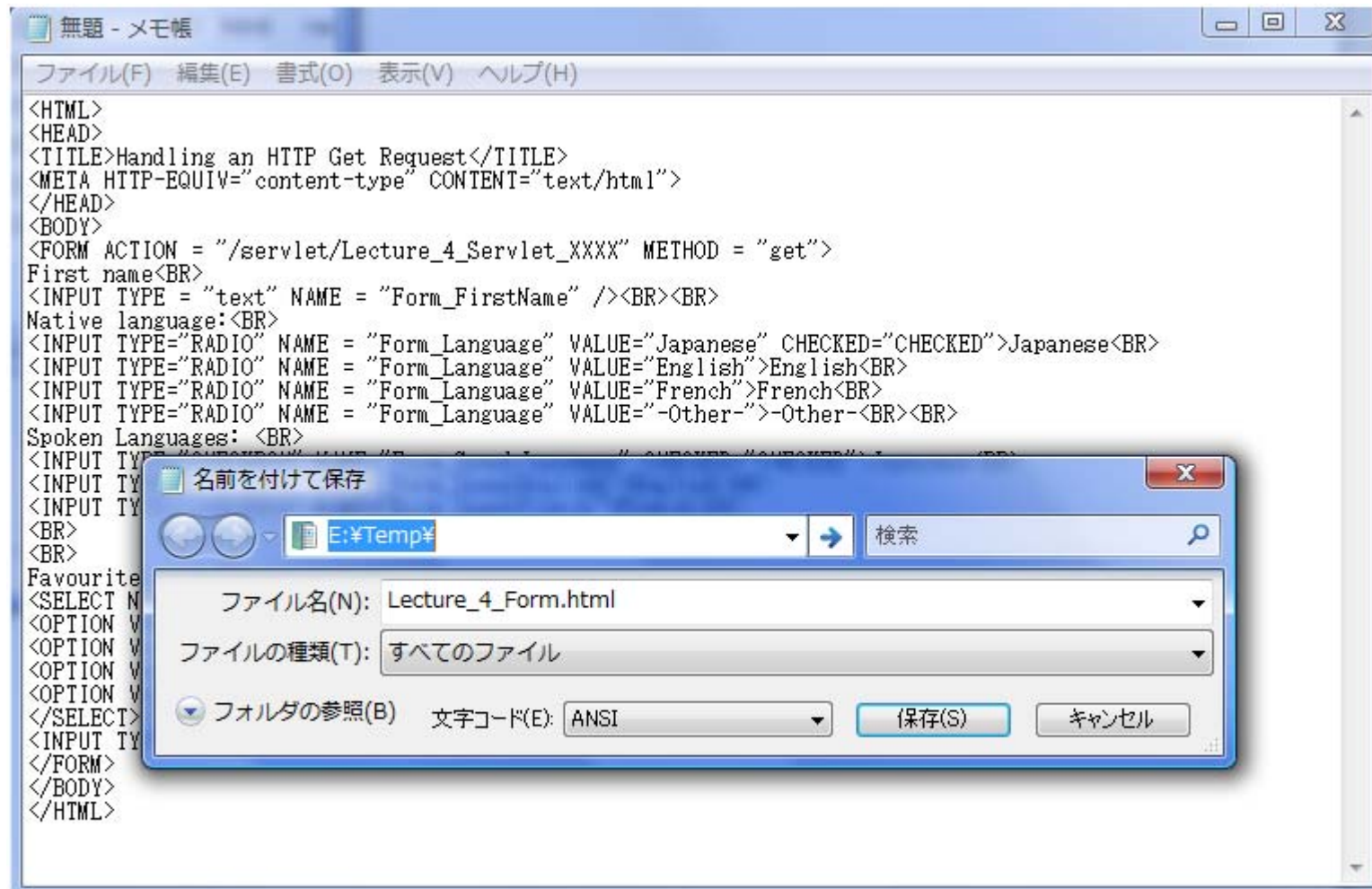
Favourite car:  
-Other- ▾  
Start Servlet

# A. Webフォームの開発

入力された内容をE:/Temp/に保存する: ファイル名Lecture\_4\_Form.html



notepad



# B. WebフォームをWebサーバへのUpload



FTP Client (FFFTP)を実行し、「ホストの設定」「基本」のページでFTPサーバーのパラメーターを記入。

ホスト名④は「isd-si.doshisha.ac.jp」で、⑤ユーザ名と⑥パスワードは「guest」です。

The screenshot shows the FFTP application interface. A red circle labeled '1' highlights the main window. A 'ホスト一覧' (Host List) dialog box is open, with a red circle '2' highlighting the '新規ホスト(N)...' (New Host) button. The 'ホストの設定' (Host Settings) dialog box is also open, showing the '基本' (Basic) tab. Red circles '3' through '7' highlight specific fields and buttons: '3' points to the 'ホストの設定名(T)' (Host Name) field containing 'guest'; '4' points to the 'ホスト名(アドレス)(N)' (Host Name (Address)) field containing 'isd-si.doshisha.ac.jp'; '5' points to the 'ユーザー名(U)' (Username) field containing 'guest'; '6' points to the 'パスワード/パスフレーズ(P)' (Password/Phrase) field containing masked characters; and '7' points to the 'OK' button.

# B. WebフォームをWebサーバへのUpload



自分のフォルダーを選択してから、Lecture\_4\_Form.htmlをUploadします

The screenshot shows the FFTP interface with two file lists. The local file list on the left includes 'Lecture\_4\_Form.html' which is highlighted with a red dashed circle. The remote file list on the right shows the contents of the selected folder on the server.

名前	日付	サイズ	種類
init.cfg	2023/03/01 1...	131	cfg
JSP_Editor.exe	2022/08/23 0...	1,12...	exe
Lecture_1.html	2023/02/28 1...	143	html
Lecture_2_Form.html	2023/03/01 1...	324	html
Lecture_3_Form.html	2023/03/01 1...	323	html
Lecture_3_Servlet_XXXX.java	2023/03/01 1...	880	java
Lecture_4_Form.html	2023/03/01 1...	1,102	html

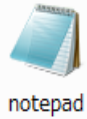
  

名前	日付	サイズ	種類
Compile_Servlet.jsp	2022/08/23	1,807	jsp
HelloFromServlet.java	2023/03/01 9:11	1,163	java
Lecture_1.html	2023/02/28 10:...	143	html
Lecture_2_Form.html	2023/03/01 8:41	324	html
Lecture_3_Form.html	2023/03/01 16:...	323	html
logo.htm	2023/02/28 10:...	45	htm
logo_013.png	2023/02/28 10:...	13,010	png

>LIST  
150 Opening ASCII mode data connection for /bin/lis [448 bytes].  
226 Transfer successful.  
ファイル一覧の取得は正常終了しました。 [448 Bytes]

ホスト 選択0個 (0 Bytes) ローカル空 343503.62M Bytes 転送待ちファイル0個

# C. Servletの開発



- 1 Notepadで、下記のServletのソースコード(Java)を入力する
- 2 Servletは **E:\Temp\Lecture\_4\_Servlet\_XXXX.java** に保存する

```
1 import java.io.*;
2 import javax.servlet.*;
3 import javax.servlet.http.*;
4 public class Lecture_4_Servlet_XXXX extends HttpServlet {
5     public void service (HttpServletRequest request, HttpServletResponse response)
6     throws ServletException, IOException
7     {
8         response.setContentType("text/html");
9         // Getting the values of parameters of Web Form:
10        String firstName = request.getParameter("Form_FirstName");
11        String nativeLang = request.getParameter("Form_Language");
12        String car = request.getParameter("Form_Cars");
13        String spokenLang = "";
14        if (request.getParameter("Form_SpeakEnglish") !=null) spokenLang="English";
15        if (request.getParameter("Form_SpeakJapanese") !=null) spokenLang=spokenLang + " Japanese";
16        if (request.getParameter("Form_SpeakFrench") !=null) spokenLang=spokenLang + " French";
17        // Getting the communication channel with the requesting client
18        PrintWriter out = response.getWriter();
19        // Writing the response to be sent to Web Browser
20        out.println("<HTML><HEAD><TITLE>Servlet Request: </TITLE>"
21            + " <META HTTP-EQUIV='Content-Type' Content='text/html';"
22            + " charset='SHIFT_JIS'></META></HEAD>");
23        out.println(" <CENTER>"
24            + "<H2>Hello "+firstName+"-san!</H2>"
25            + "<H3>Welcome to Servlets!</H3></CENTER>"
26            + " Your native language is <B>"+nativeLang+"</B><BR>"
27            + " You speak <B>"+ spokenLang + "</B><BR>"
28            + " Your favorite car is <B>"+car+"</B><BR><BR>");
29        out.println("Local time is " + new java.util.Date());
30        out.println("</BODY></HTML>");
31    }
32 }
```

注意: class名 = ファイル名 (.java)  
例: class名: **Lecture\_4\_Servlet\_1116221001**  
ファイル名: **Lecture\_4\_Servlet\_1116221001.java**

HTTP Request  
の処理

HTTP Response  
の作成

# D. ServletをWebサーバへのUpload



- 1 FFFTPを実行する。
- 2 新規ホスト(N)... ボタンを押し、
- 3 ホストを設定し、

接続します。ホストの設定:

ホスト名: isd-si.doshisha.ac.jp

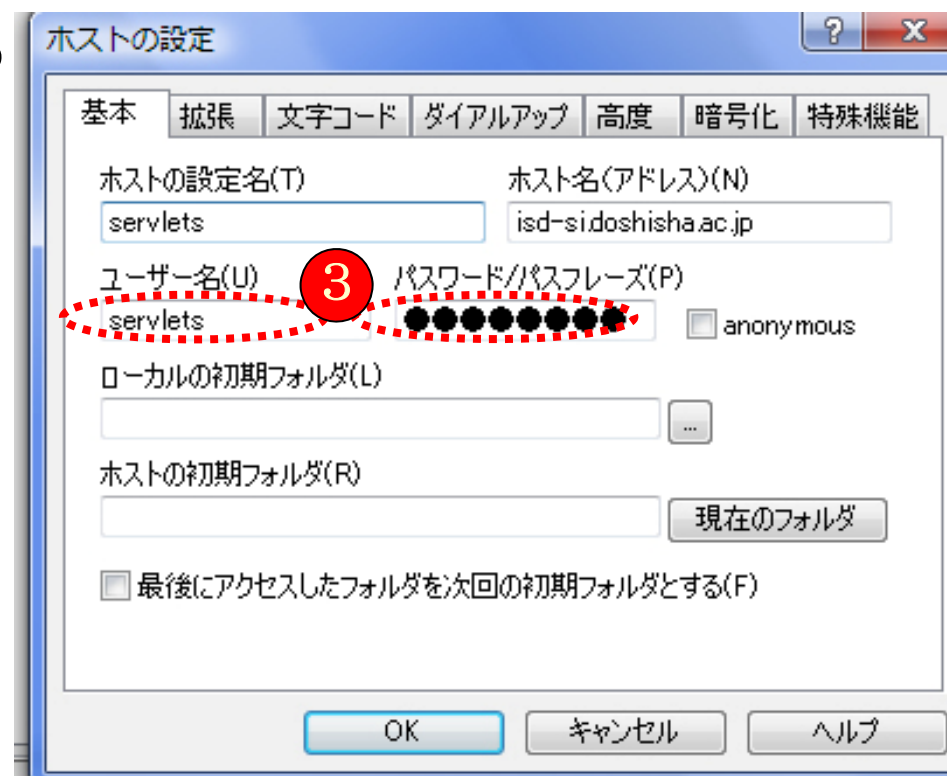
ユーザ名: **servlets**

パスワード: **servlets**

## 注意:

ServletのUploadの時に  
ユーザ名: **servlets**  
パスワード: **servlets**  
です。

Web FormのUpload時に  
ユーザ名: **guest**  
パスワード: **guest**  
でした。



# D. ServletをWebサーバへのUpload



The screenshot shows the FFTP application window with two panes. The left pane displays a local directory 'E:\Temp' with a table of files. The file 'Lecture\_4\_Servlet\_XXXX.java' is selected. A yellow arrow points from this file to the right pane, which displays a remote directory with a table of files. The status bar at the bottom indicates 'ローカル 選択1個 (1.50K Bytes)'.

名前	日付	サイズ	種類
init.cfg	2023/03/01 1...	131	cfg
JSP_Editor.exe	2022/08/23 0...	1,12...	exe
Lecture_1.html	2023/02/28 1...	143	html
Lecture_2_Form.html	2023/03/01 1...	324	html
Lecture_3_Form.html	2023/03/01 1...	323	html
Lecture_3_Servlet_XXXX.java	2023/03/01 1...	880	java
Lecture_4_Form.html	2023/03/01 1...	1,102	html
Lecture_4_Servlet_XXXX.java	2023/03/01 1...	1,531	java

名前	日付	サイズ	種類
AQDemoServlet.java	2022/08/23	3,064	java
AQDemoServlet22.java	2022/08/23	2,628	java
AQPropServlet.java	2022/08/23	2,759	java
AQPropServlet22.java	2022/08/23	2,634	java
HelloFromServlet.class	2022/08/23	1,302	class
IsItWorking.class	2023/02/26 15:...	2,028	class
IsItWorking.java	2002/08/20	4,476	java
Lecture_3_Servlet_XXXX.class	2023/03/01 16:...	1,231	class
Lecture_3_Servlet_XXXX.java	2023/03/01 16:...	880	java
Bean.class	2022/08/23	479	class
GameBean.java	2022/08/23	233	java
zone.properties	2022/08/22	7,012	prop

ローカル 選択1個 (1.50K Bytes) ローカル空 343499.74M Bytes 転送待ちファイル0個

## E. Servletのコンパイル



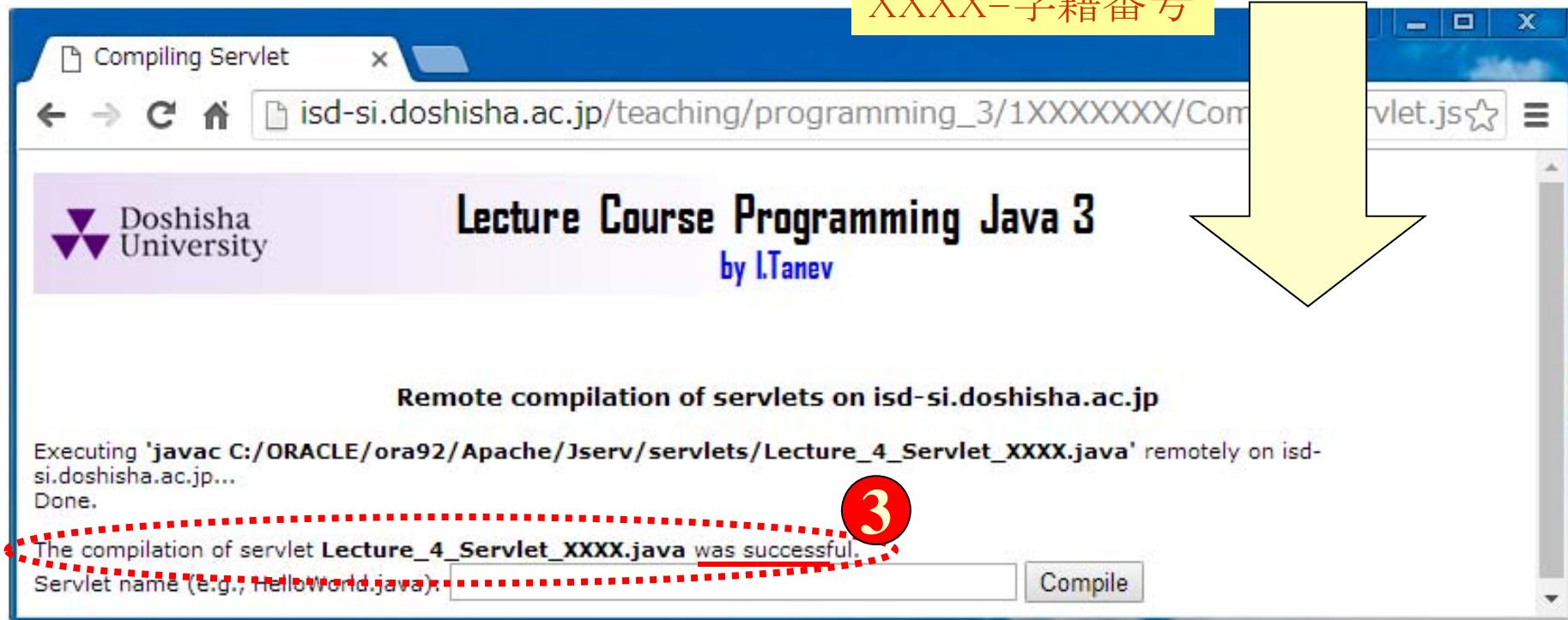
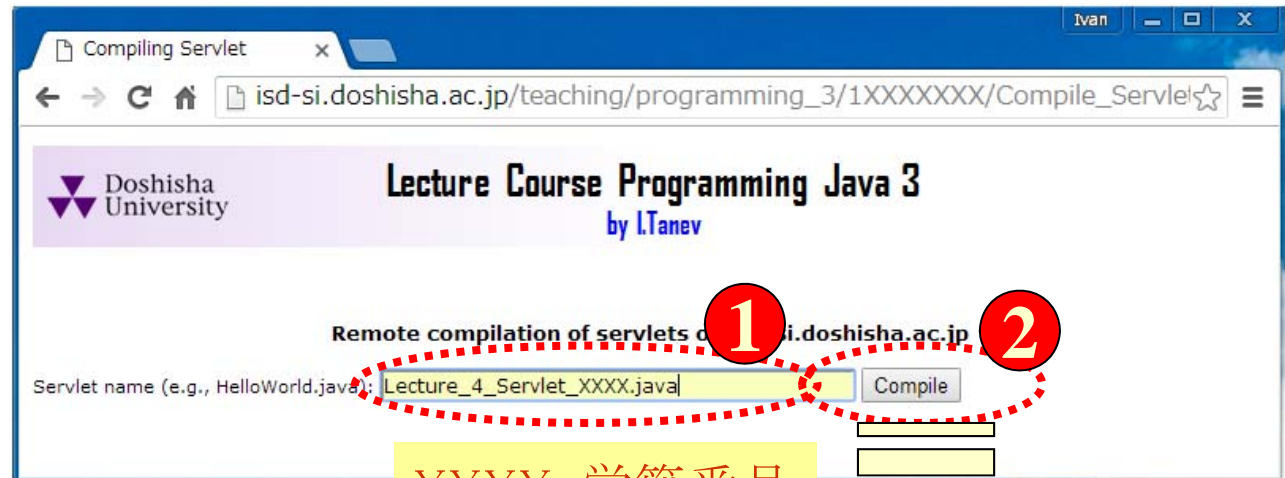
- 1 Web Browser: [http://isd-si.doshisha.ac.jp/teaching/Programming\\_3/1XXXXXXXX/](http://isd-si.doshisha.ac.jp/teaching/Programming_3/1XXXXXXXX/)  
(1XXXXXXXX= 学籍番号)
- 2 Compile\_Servlet.jspにAccessする:



# E. Servletのコンパイル



- 1 Servletのネームの入力
- 2 Compile ボタンの押す
- 3 コンパイル結果の確認



# E. Servletのコンパイル



Servletはコンパイルができましたか？

FFFTP ClientでLecture\_4\_Servlet\_XXXX.classファイルの確認

The screenshot shows the FFTP Client interface with a file listing and a terminal window. A red circle '1' is around the menu bar, and a red circle '2' is around the file 'Lecture\_4\_Servlet\_XXXX.class' in the file list.

名前	日付	サイズ	種類
init.cfg	2023/03/01 1...	131	cfg
JSP_Editor.exe	2022/08/23 0...	1,12...	exe
Lecture_1.html	2023/02/28 1...	143	html
Lecture_2_Form.html	2023/03/01 1...	324	html
Lecture_3_Form.html	2023/03/01 1...	323	html
Lecture_3_Servlet_XXXX.java	2023/03/01 1...	880	java
Lecture_4_Form.html	2023/03/01 1...	1,102	html
Lecture_4_Servlet_XXXX.java	2023/03/01 1...	1,531	java

名前	日付	サイズ	種類
AQDemoServlet.java	2022/08/23	3,064	java
AQDemoServlet22.java	2022/08/23	2,628	java
AQPropServlet.java	2022/08/23	2,759	java
AQPropServlet22.java	2022/08/23	2,634	java
HelloFromServlet.class	2022/08/23	1,302	class
IsItWorking.class	2023/02/26 15:...	2,028	class
IsItWorking.java	2002/08/20	4,476	java
Lecture_3_Servlet_XXXX.class	2023/03/01 16:...	1,231	class
Lecture_3_Servlet_XXXX.java	2023/03/01 16:...	880	java
Lecture_4_Servlet_XXXX.class	2023/03/01 16:...	1,826	class
Lecture_4_Servlet_XXXX.java	2023/03/01 16:...	1,531	java
NameBean.class	2022/08/23	470	class

>LIST  
150 Opening ASCII mode data connection for 1093 bytes.  
226 Transfer successful.  
ファイル一覧の取得は正常終了しました。 [1093 Bytes]

選択1個 (1.50K Bytes) ローカル空 343500.07M Bytes 転送待ちファイル0個

Lecture\_4\_Servlet\_XXXX.class ?

XXXX=学籍番号

# F. WebフォームのAccess



Web Browser: [http://isd-si.doshisha.ac.jp/teaching/Programming\\_3/1XXXXXXX/Lecture\\_4\\_Form.html](http://isd-si.doshisha.ac.jp/teaching/Programming_3/1XXXXXXX/Lecture_4_Form.html)

1XXXXXXXは学籍番号です。

Handling an HTTP Get F x

isd-si.doshisha.ac.jp/teaching/programming\_3/1XXXXXXX/Lecture\_4\_Form.html ☆

First name  
John

Native language:  
 Japanese  
 English  
 French  
 -Other-

Spoken Languages:  
 Japanese  
 English  
 French

Favourite car:  
-Other- ▾  
Start Servlet

Servlet Request: x

isd-si.doshisha.ac.jp/servlet/Lecture\_4\_Servlet\_XXXX?Form\_FirstName=John& ☆

**Hello John-san!**

**Welcome to Servlets!**

Your native language is **-Other-**  
You speak **English Japanese**  
Your favorite car is **-Other-**

Local time is Wed Mar 01 16:37:46 JST 2023

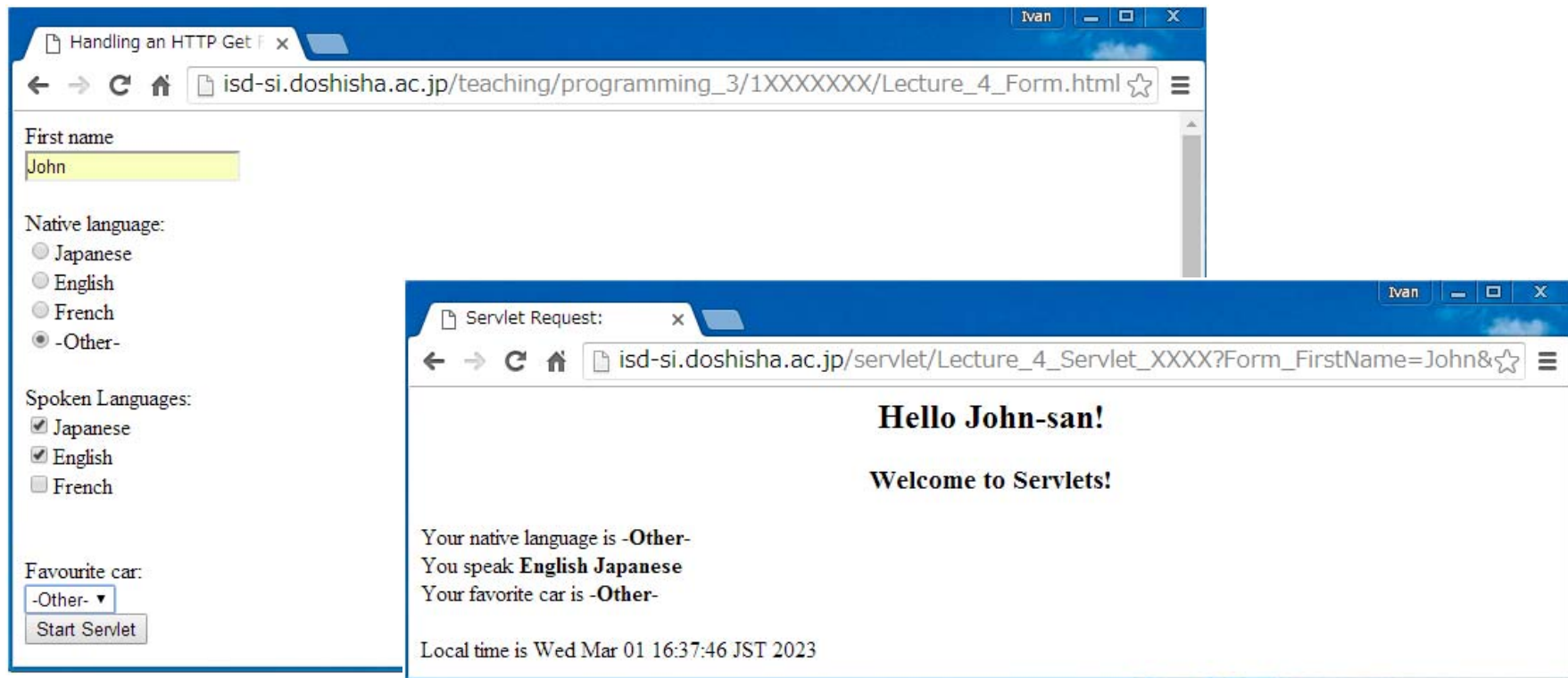
# 課題

Lecture\_4\_Form.htmlの開発

Lecture\_4\_Servlet\_XXXX.javaの開発

Lecture\_4\_Servlet\_XXXX.classのコンパイル

Lecture\_4\_Form.htmlのAccess結果:



締め切り:5月14日(木曜日)、23:59

The End