

# ただ一足の青い猫のかげ

## —eclipseによるJavaサーブレットの作成—

岐阜経済大学 経営学部 経営情報学科 井戸 伸彦

来歴:

0.0版 2004年4月18日

0.1版 2005年2月14日 一部をeclipse3.01,lomboz3.01に修正

1.1版 2008年1月30日 一部をeclipse3.3,tomcat6に修正

### スライドの構成

はじめに

(1)環境設定

(1.1)なにがやりたいのか？

(1.2)プラグイン

(1.3)環境設定

(1.4)サーバ設定

(2)プロジェクト

(2.1)～(2.5)動的Webプロジェクトの作成

(2.6),(2.7)プロジェクトのサーバへの追加

(3)JSPの作成

(3.1)～(3.5)JSPの作成

(4)Tomcatの起動とJSPの閲覧

(4.1)Tomcatの起動

(4.2)JSPの閲覧

(5)Javaサーブレット

(5.1)～(5.6)Javaサーブレットの作成

(5.7)サーブレットの閲覧

(6)その他

(6.1)web.xmlファイルの中身

(6.2)web.xmlファイルの編集

(6.3)コンソールでのエラー表示

(6.4)デバッグモード、ログファイル

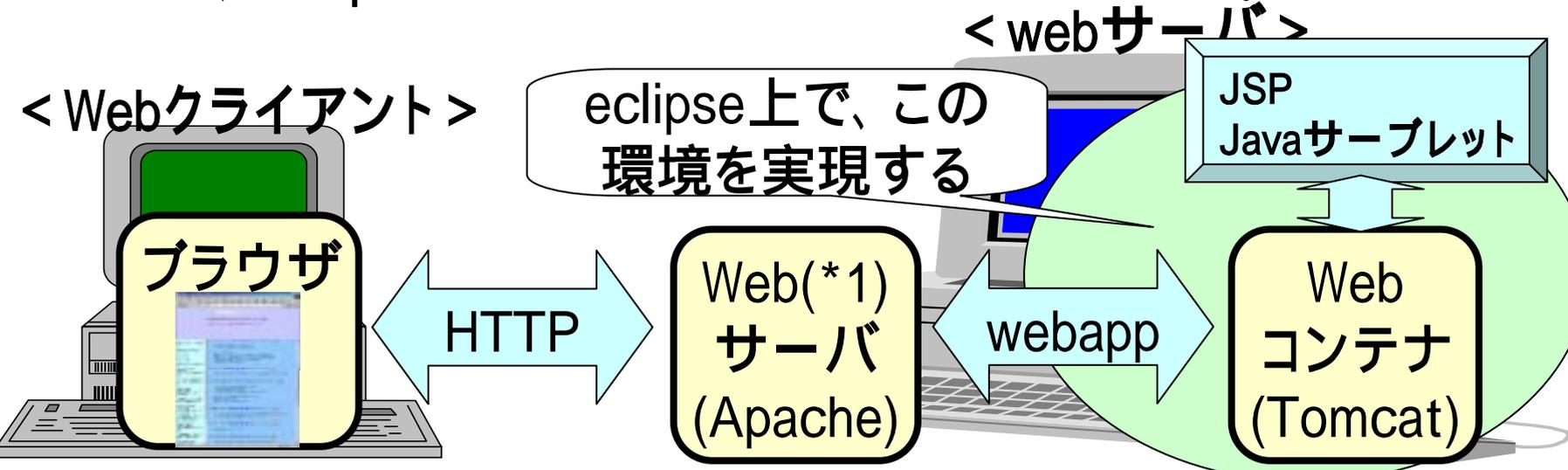
# はじめに

---

- 本スライドは、lombozがプラグインされたeclipse、および、Tomcatを用いてJavaサーブレットを作成する手順について記します。
- このスライドでは、手順のみを記していき、編集したファイルの意味については次の2つのスライドにて勉強することになります。
  - 「ウィークエンド・シャッフル - JSP入門 - 」
  - 「されど我らが日々 - サーブレット入門 - 」
- Jdk1.6 / Tomcat6.0 / eclipse3.3が、次の文書のとおりインストール済みであることを前提としています。
  - 「jdk1.6のインストール(Windows) 」
  - 「Tomcat6のインストール(Windows)」
  - 「eclipse3.3のインストール(Windows)」
- 岐阜経済大学では、全WindowsPCに、eclipse3.3、Tomcat6はインストールされています。本スライドでは、これらでの環境での操作を記します。

# (1.1) 何がやりたいのか？

- 下図のようなシステム構成で動作するサーブレットとJSPを、eclipse上で作成することが目的です。

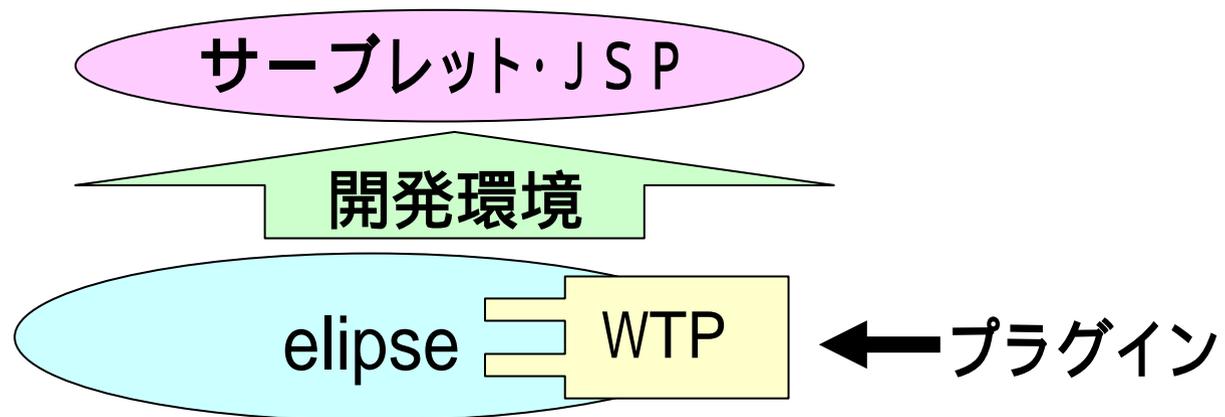


- (\*1) 大学のPCに設定した環境ではApacheは実装せず、Tomcatの持つWebサーバ機能を利用しています。

- Javaサーブレットは、コンパイルしておく必要がありますが、JSPの場合は、ファイルが変化すると自動でコンパイルが行われます。
- eclipse上の開発環境を前提としています。

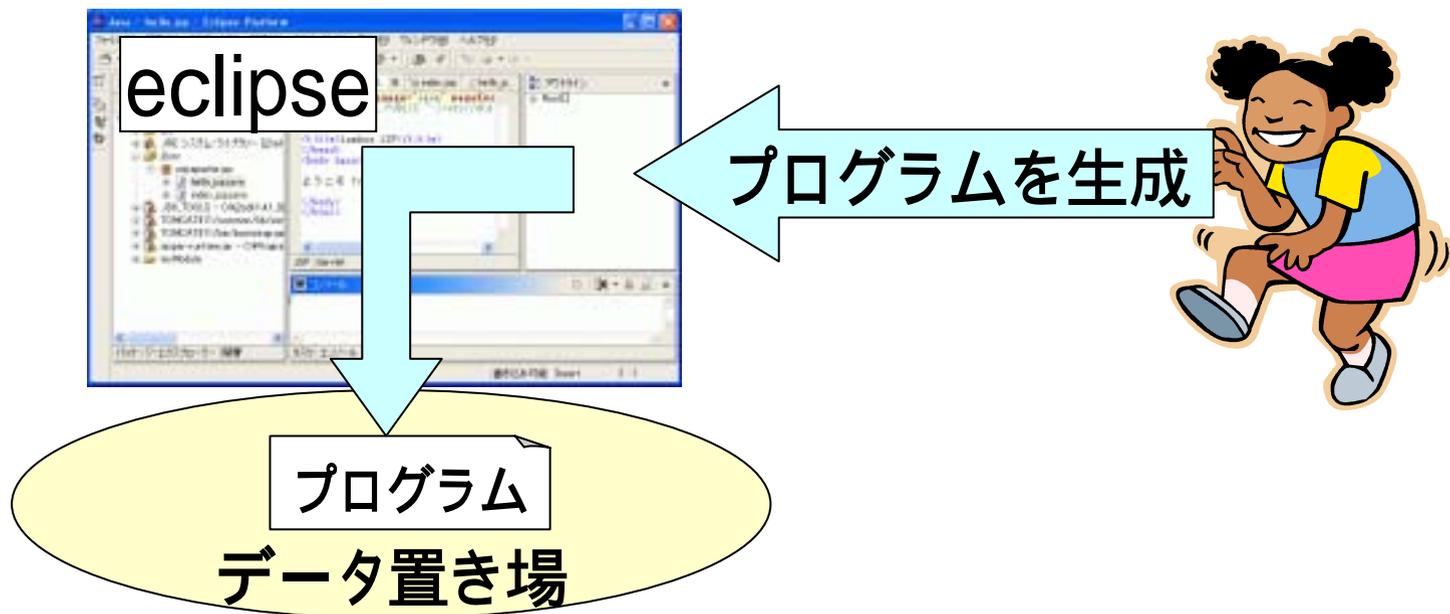
## (1.2) プラグイン

- 学内でLinux PC、Windows PCにインストールした eclipseには、WTPがプラグインされています。
- プラグインとは、eclipseに機能を追加することであると覚えておいてください。
- WTPをプラグインすることにより、eclipseでは、スライド(1.0)に示したようなJavaサーブレット(JSP)の作成を容易に行う環境を提供することが出来ます。WTPはJ2EEプロジェクト作成支援プラグインとして、標準の位置づけにあります。



# ( 1 . 3 ) 環境設定

- WTPを利用するには、環境設定が必要となります。
- 環境設定は、スライド「eclipseを用いたJavaアプリケーションの作成」で説明した、データの置き場所ごと、すなわち、各ユーザごとに必要となります。これを(1.4)で説明します。



# (1.4) サーバ設定 (1 / 4)

## ■ [ファイル]

- [新規]

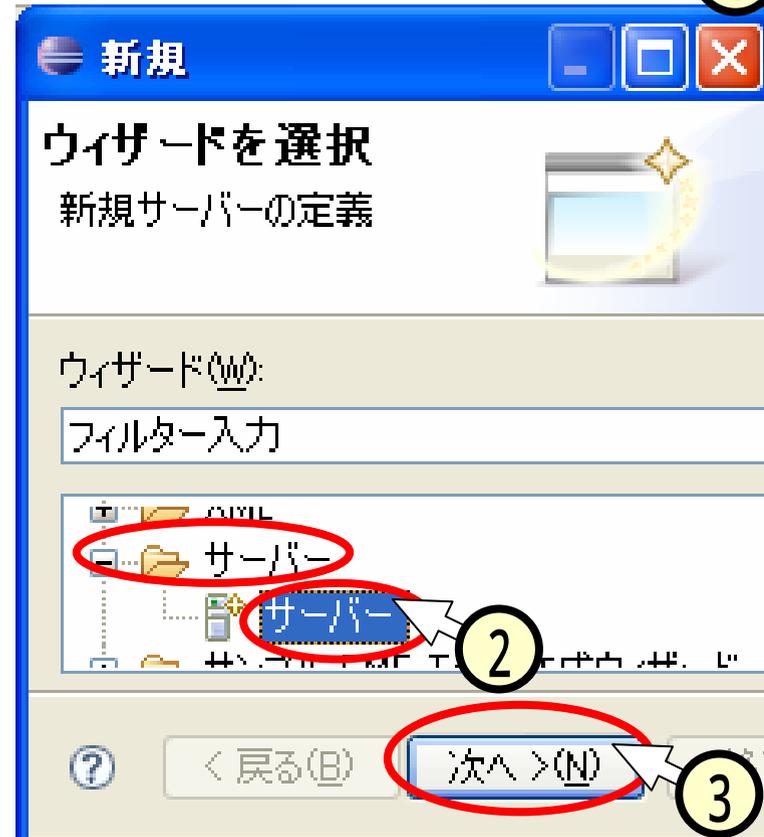
- [その他] をクリック ( )



## ■ 「新規:ウィザードを選択」にて、

[サーバー]-[サーバー] を  
クリック ( )。

## ■ [次へ] をクリック ( )。



# (1.5) サーバ設定 (2 / 4)

■「新規サーバー:新規サーバーの定義」にて、

[Apache]

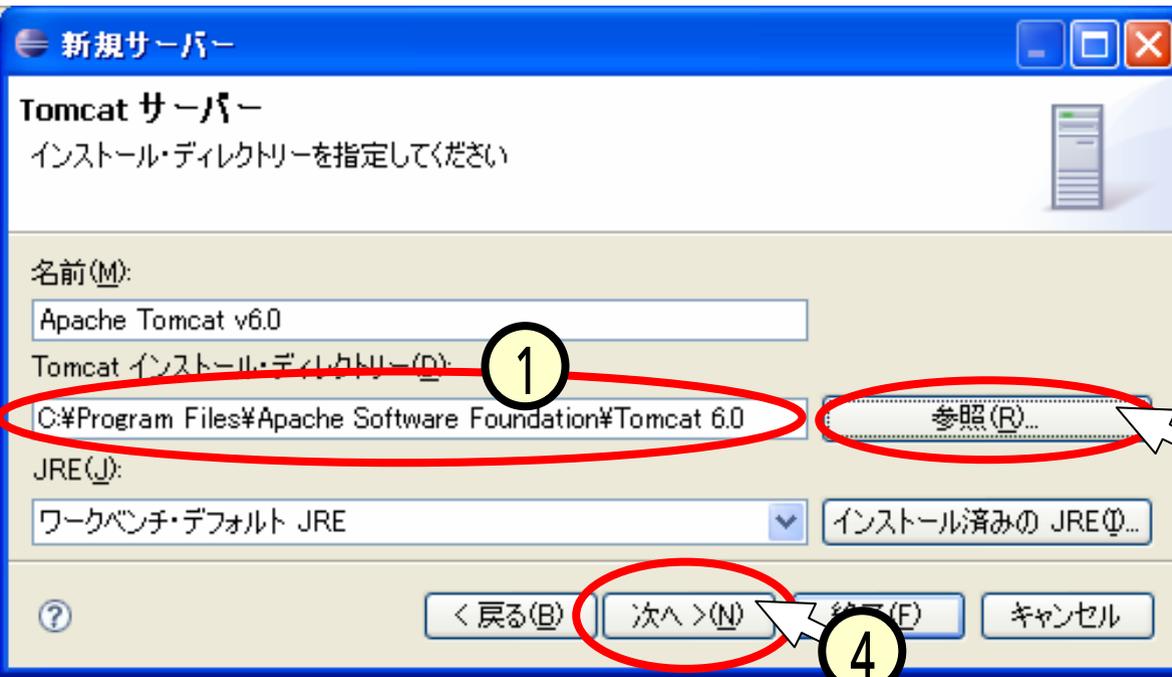
-[Tomcat v6.0 Server]をクリック( )。

■[次へ]をクリック( )。



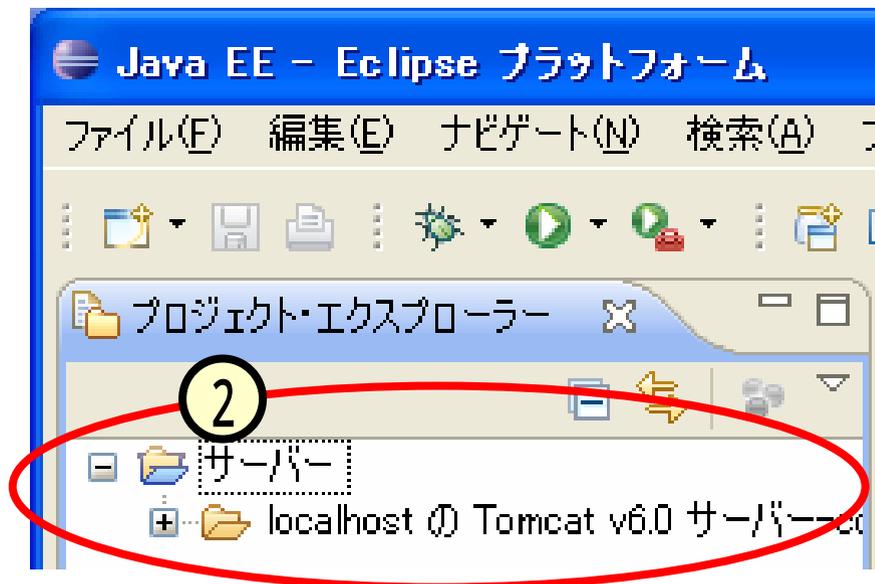
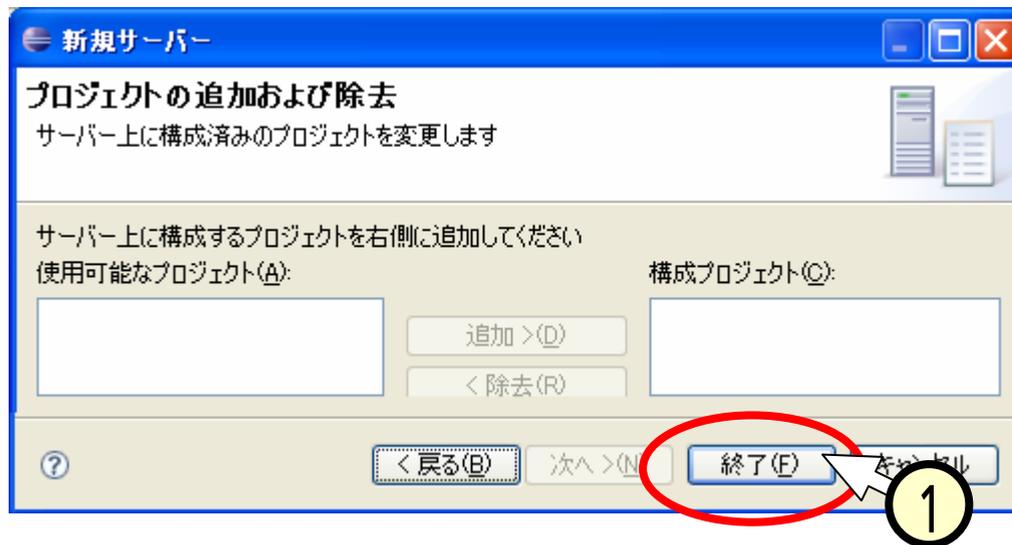
# ( 1 . 6 ) サーバ設定 ( 3 / 4 )

- 「新規サーバー: Tomcatサーバー」にて、「Tomcatインストールディレクトリー」の項に、次のとおり入力( )。
  - C:¥Program Files¥Apache Software Foundation¥Tomcat 6.0
- この際、[参照]をクリック( )して、「フォルダの参照( )」の画面で選択したほうが楽。
- [次へ]をクリック( )。



# (1.7) サーバ設定 (4 / 4)

- 「新規サーバー: プロジェクトの追加および除去」にて、[終了]をクリック( )。
- プロジェクトエクスプローラー中に、サーバーが表示( )されることを確認。



## (2) プロジェクト

---

- プロジェクトの作成は、スライド「eclipseによるJavaアプリケーションの作成」でも行いました。その時作成したのは、“Javaプロジェクト”でした。
- 今回は、“動的Webプロジェクト”を作成します。これは、スライド(1.1)のようなことが行えるプロジェクトであると理解しておいてください。

# (2.1) 動的Webプロジェクトの作成(1/5)

## ■[ファイル]

-[新規]

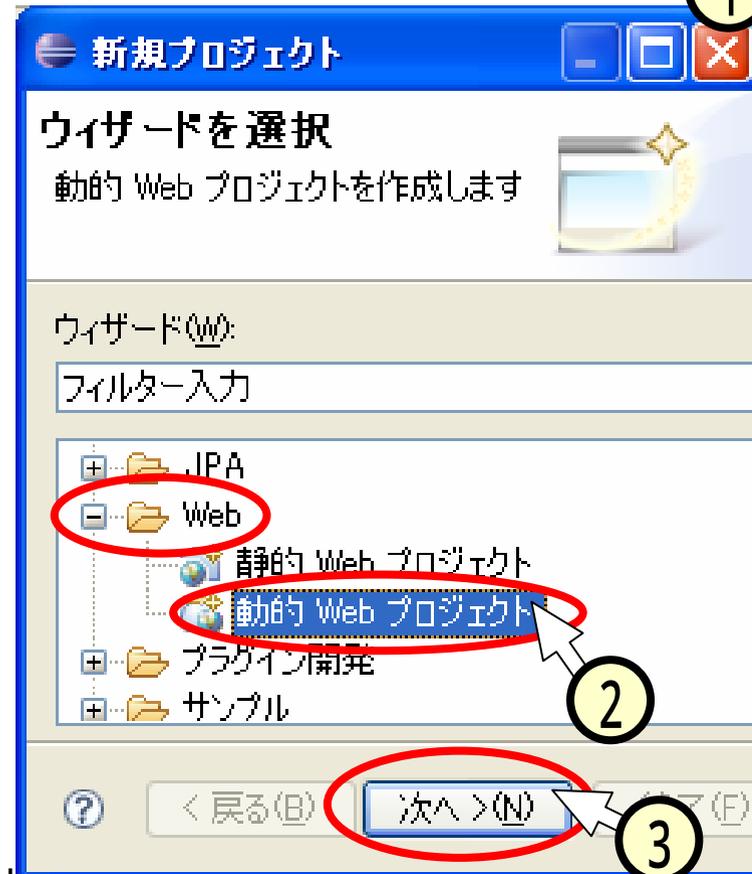
-[プロジェクト(R)]

をクリック( )。

■「新規プロジェクト:ウィザード  
を選択」にて、

[Web]-[動的Webプロジェクト]を  
クリック( )。

■[次へ]をクリック( )。

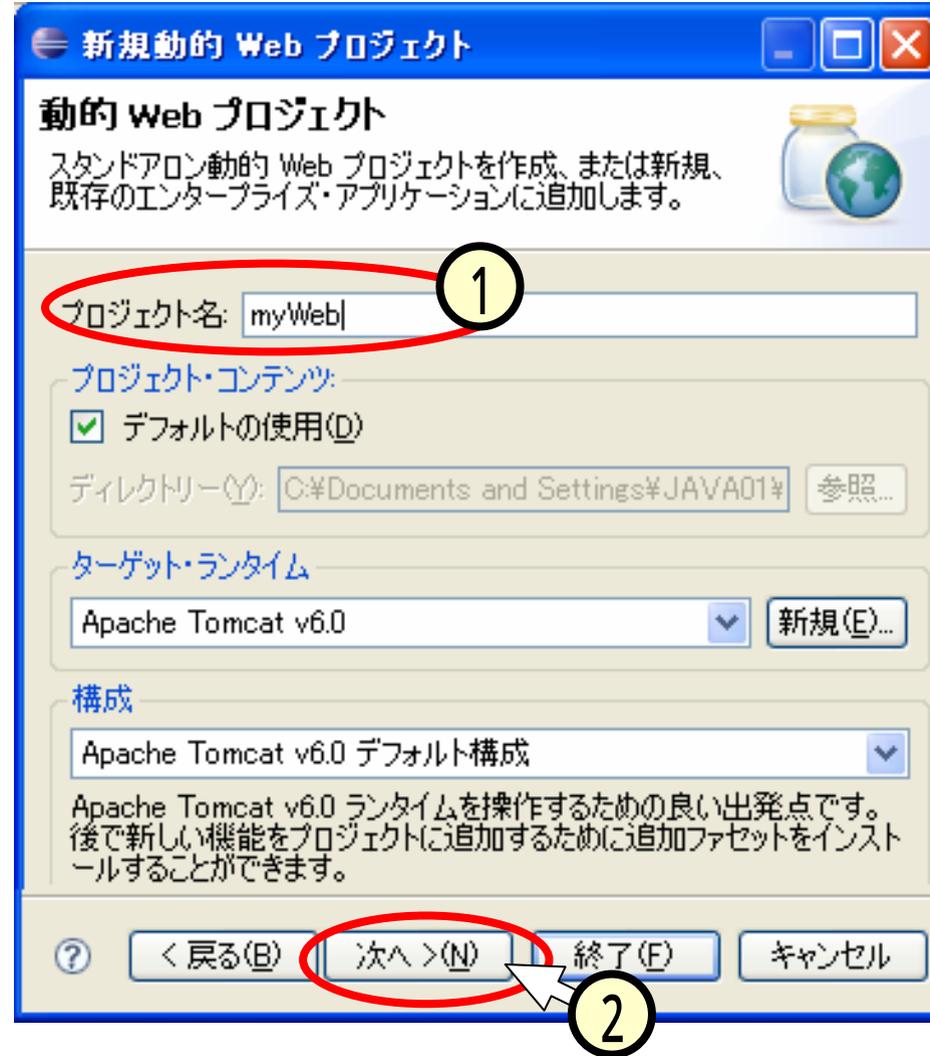


## (2.2) 動的Webプロジェクトの作成(2/5)

- 「新規動的Webプロジェクト」にて、[プロジェクト名]に適切な名前(右図では“myWeb”)と入力( )。

ここでのプロジェクト名は、後で作成したページにアクセスする際のURLの一部となる。スライド(4.2)参照。

- [次へ]をクリック( )。



## (2.3) 動的Webプロジェクトの作成(3/5)

■「新規動的Webプロジェクト:プロジェクト・ファセット」にて、次の2つがチェックされていることを確認( )。

- Java
- 動的Webモジュール

■[次へ]をクリック( )。

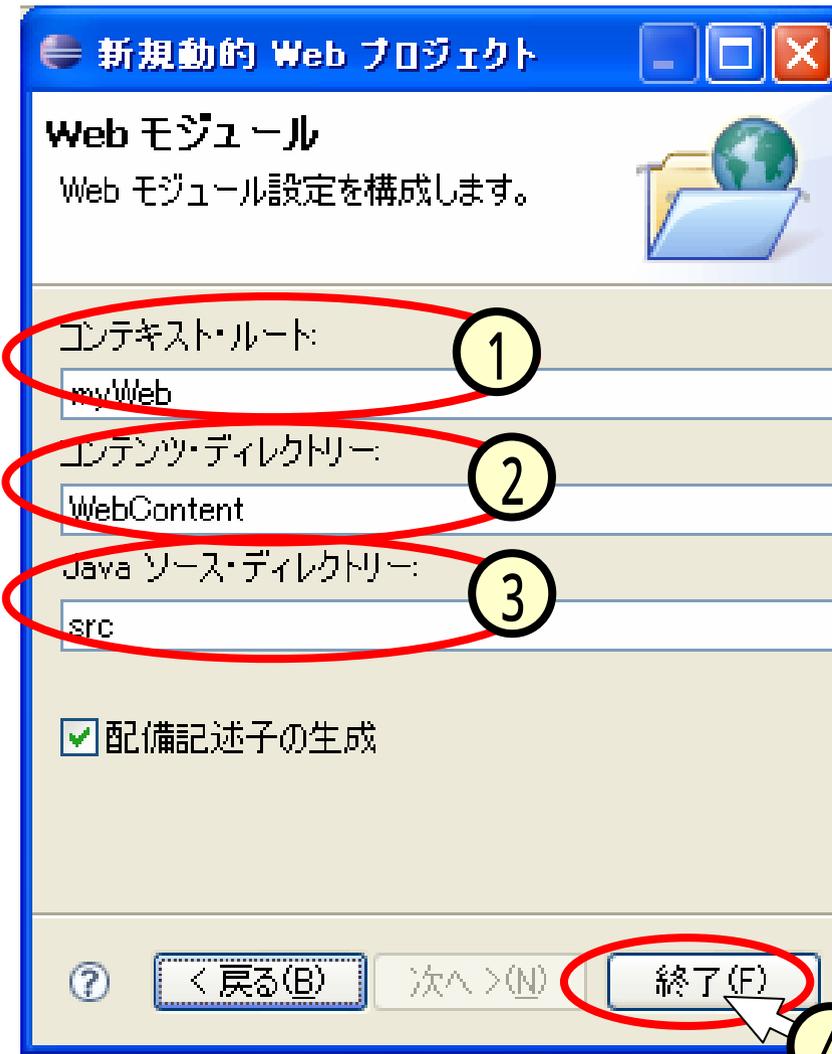


# (2.4) 動的Webプロジェクトの作成(4/5)

■「新規動的Webプロジェクト: Webモジュール」にて、次の3つの入力を確認。

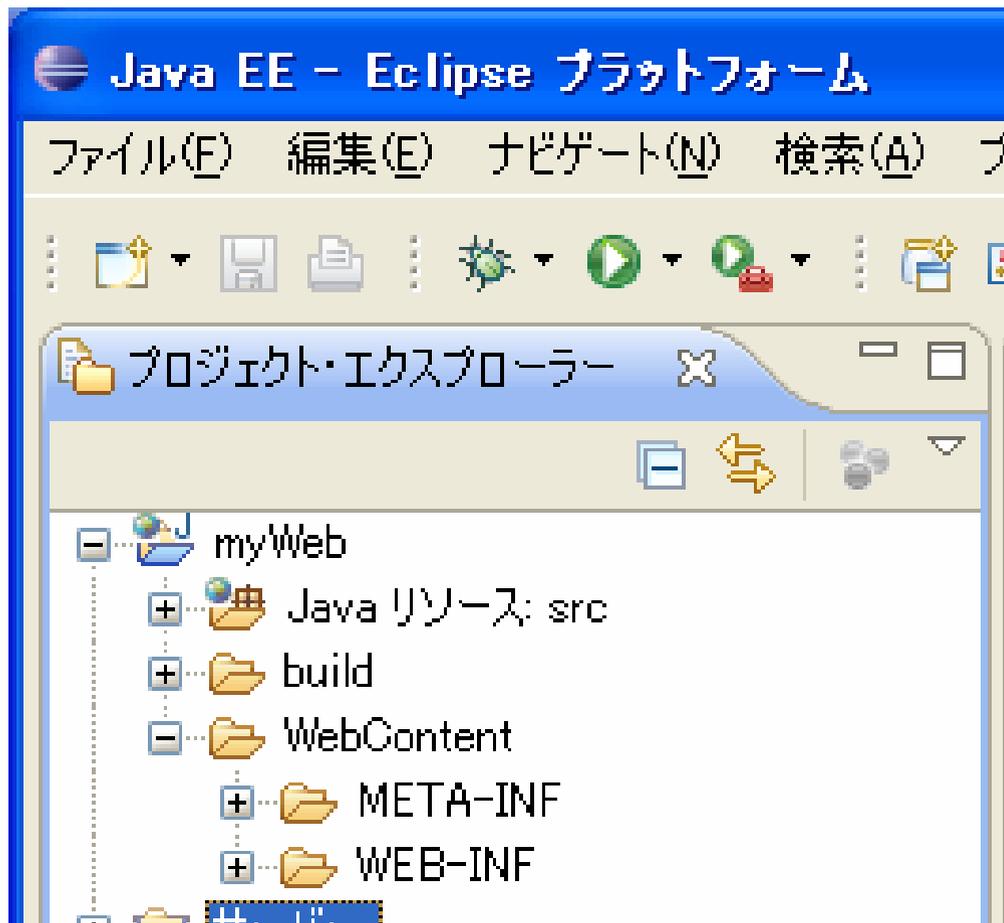
- コンテキストルート ( )
  - ◆ myWeb
- コンテンツディレクトリー ( )
  - ◆ WebContent
- Javaソースディレクトリー ( )
  - ◆ src

■ [終了]をクリック ( )。



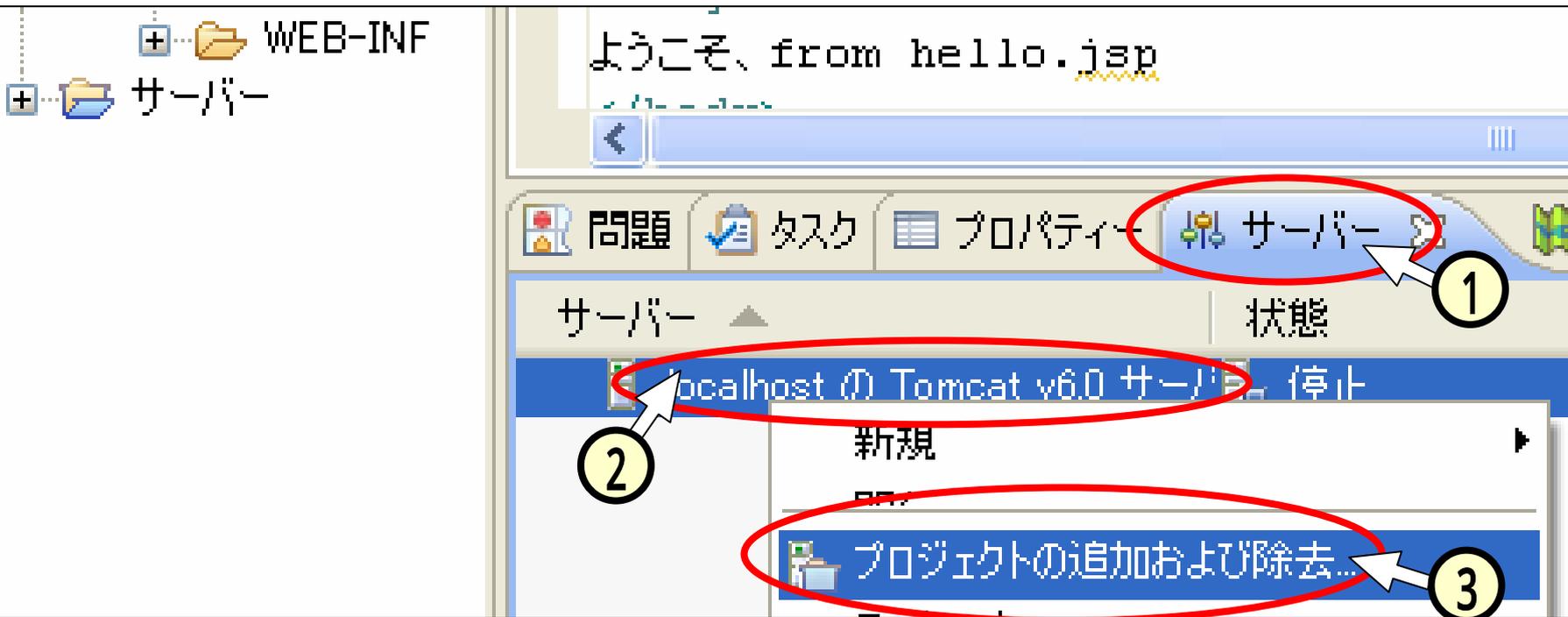
## (2.5) 動的Webプロジェクトの作成(5/5)

- 「プロジェクトエクスプローラー」に、作成したプロジェクトが表示されるのを確認。



## (2.6) プロジェクトのサーバへの追加(1 / 2)

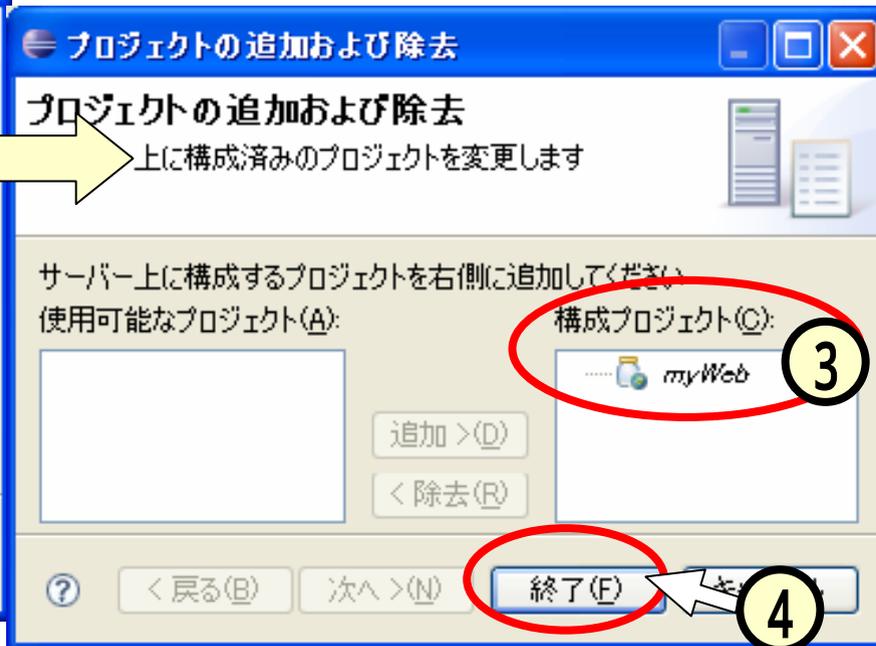
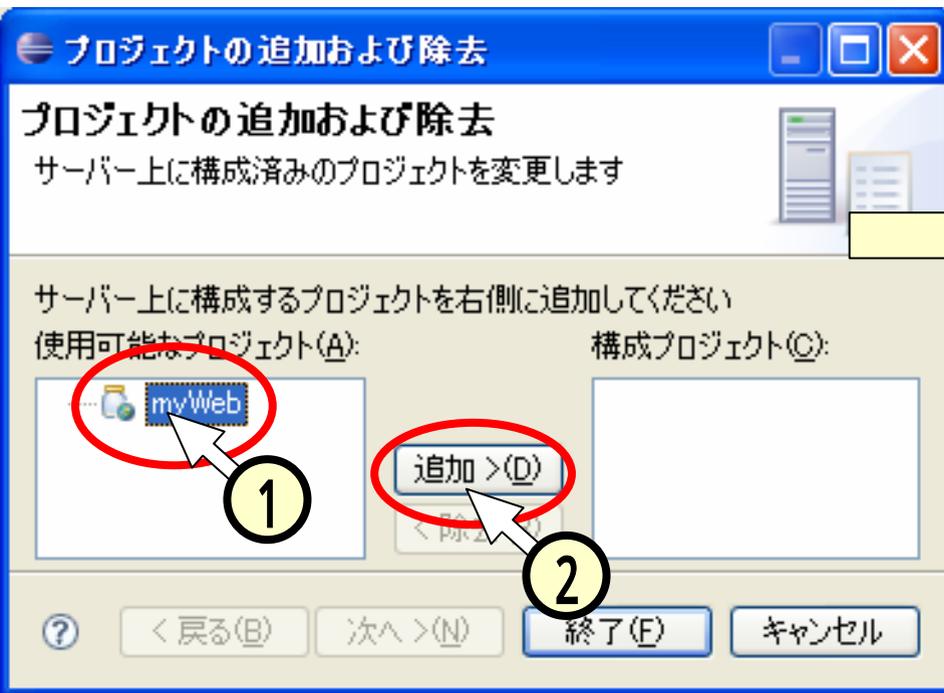
- ワークスペース下側の「サーバービュー」のタグをクリック( )。
- [localhostのTomcatv6.0サーバ]の欄を右クリック( )して、[プロジェクトの追加および削除]をクリック( )。



## (2.7) プロジェクトのサーバへの追加(2 / 2)

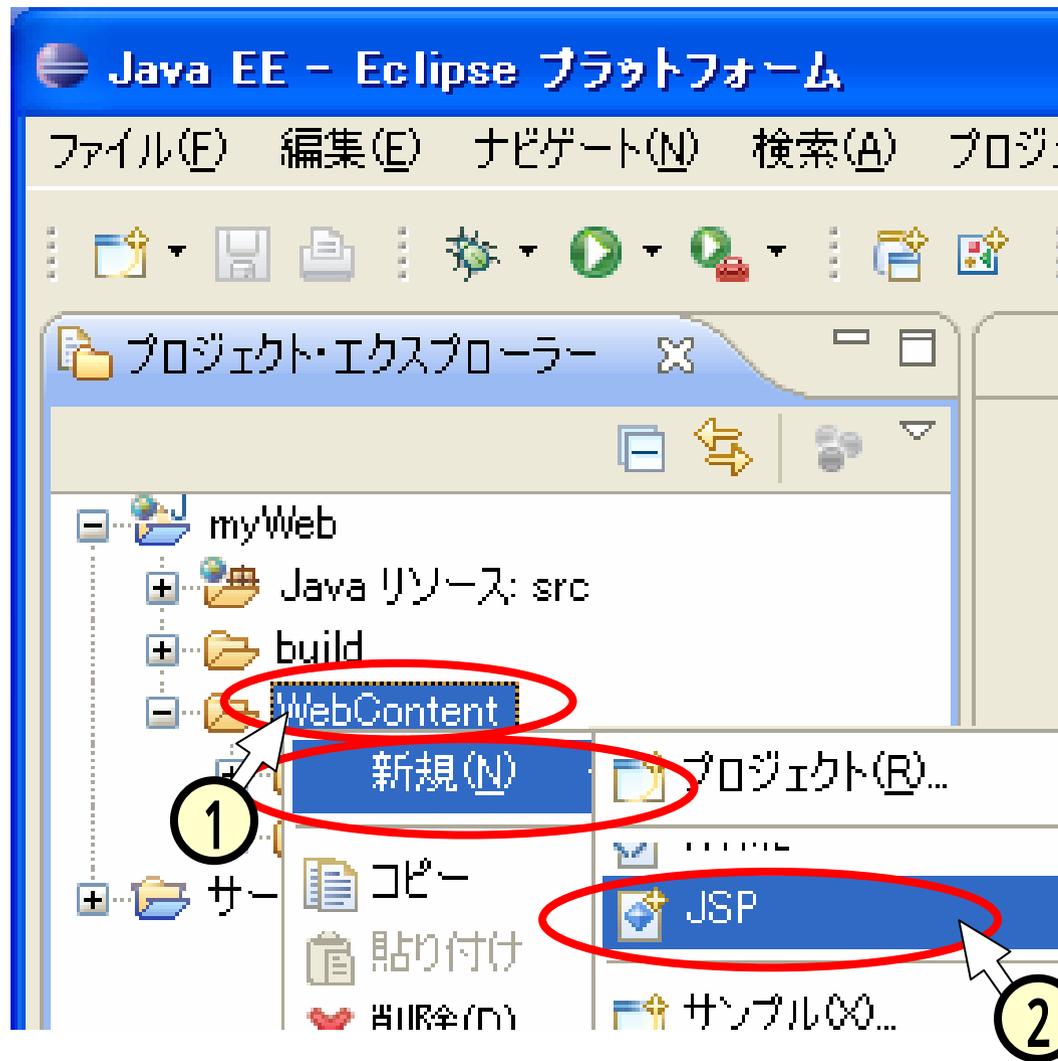
### ■「プロジェクトの追加および削除」にて、

- [使用可能なプロジェクト]中の“myWeb”を選択( 1 )
- [追加]をクリック( 2 )。 「構成プロジェクト」を確認( 3 )。
- [終了]をクリック( 4 )。



# (3.1) JSPの作成(1/4)

- “Hello!”と表示するJSPを作成します。
- 「パッケージ・エクスプローラー」中の“WebContent”( )を右クリックし、[新規]-[JSP]をクリック( )。

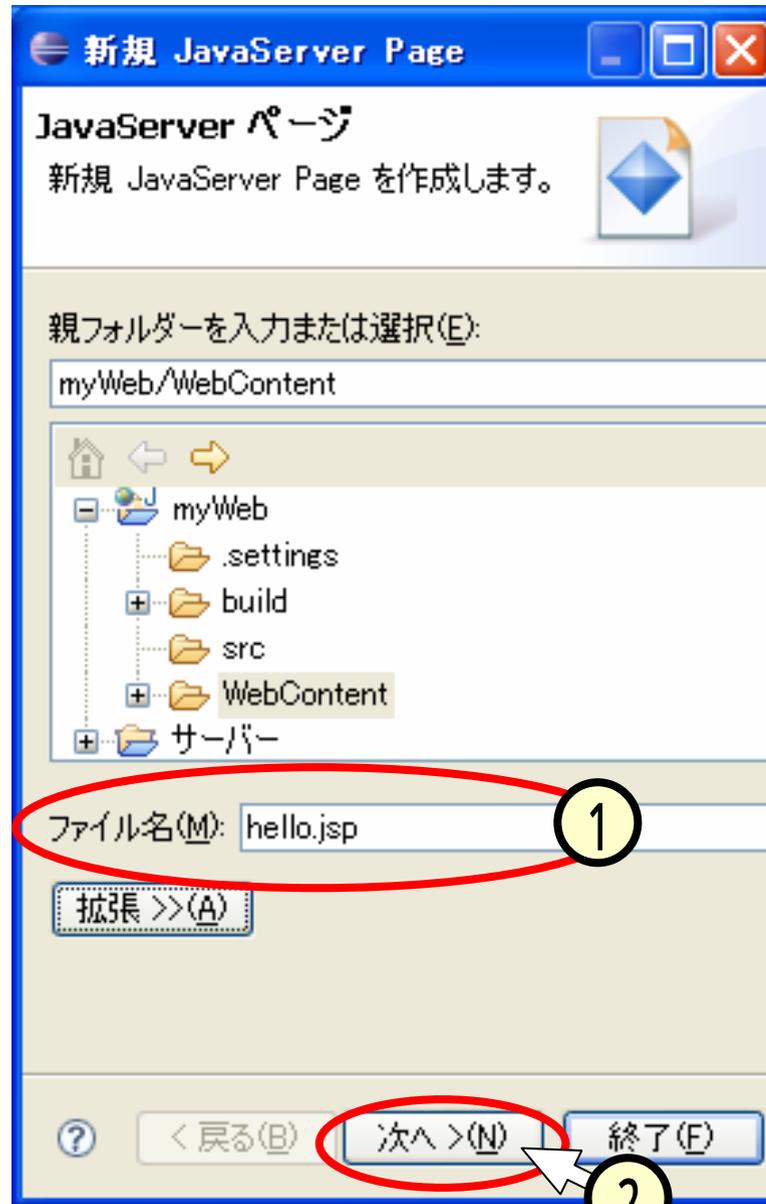


## ( 3 . 2 ) J S P の作成 ( 2 / 4 )

- 「新規JavaServer Page」にて、[ファイル名]の欄にファイル名を入力( 、右図では“hello.jsp” )。

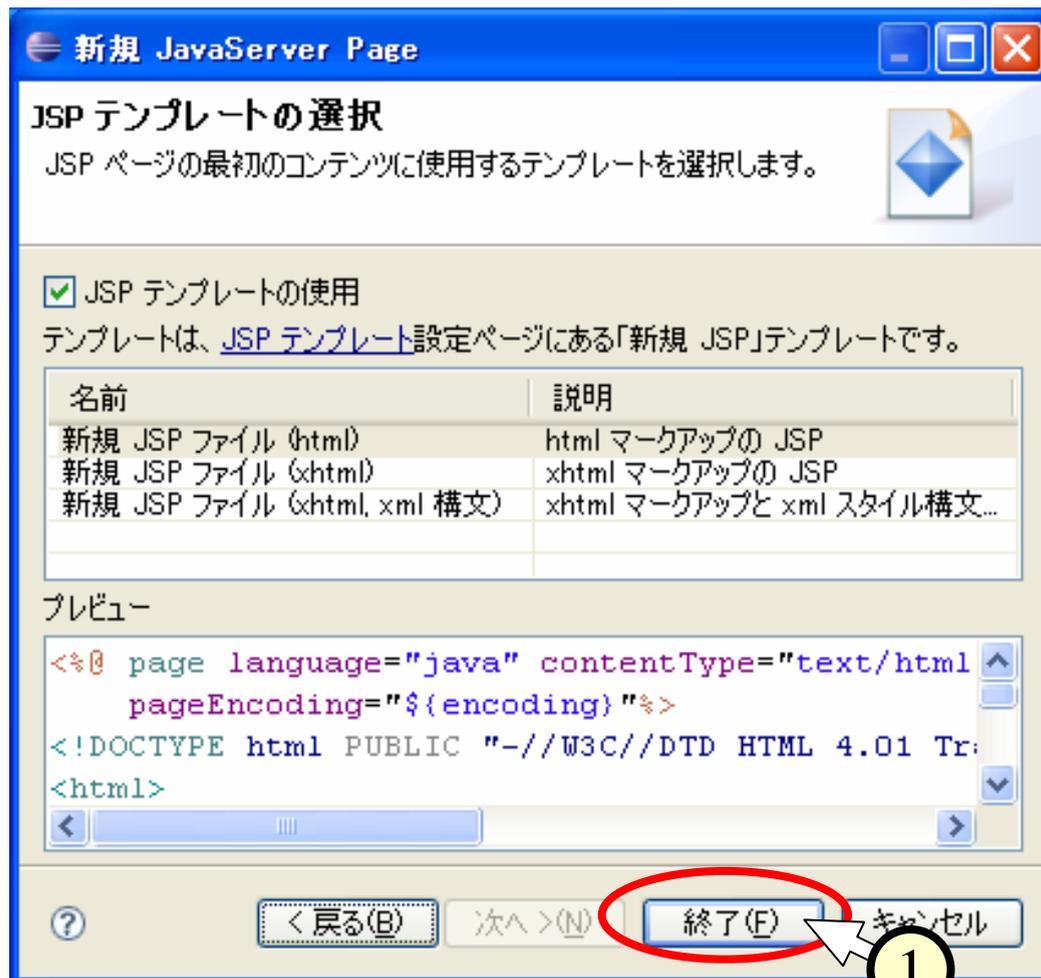
ここでのJSPのファイル名は、後で作成したページにアクセスする際のURLの一部となる。スライド(4.2)参照。

- [次へ]をクリック( )。



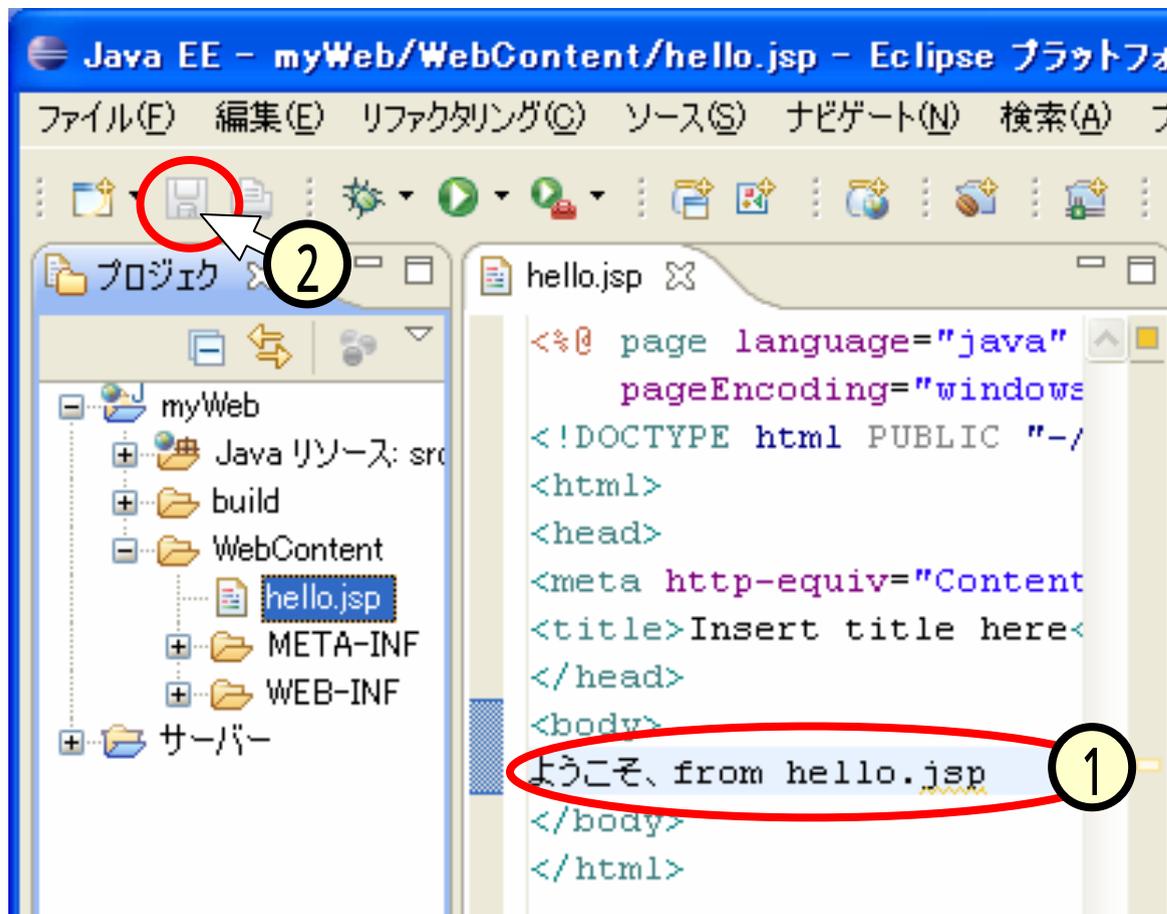
## ( 3 . 3 ) J S P の作成 ( 3 / 4 )

- 「新規JavaServer Page: J S P テンプレートの選択」にて、  
[終了]をクリック( )。



## ( 3 . 4 ) J S P の作成 ( 4 / 4 )

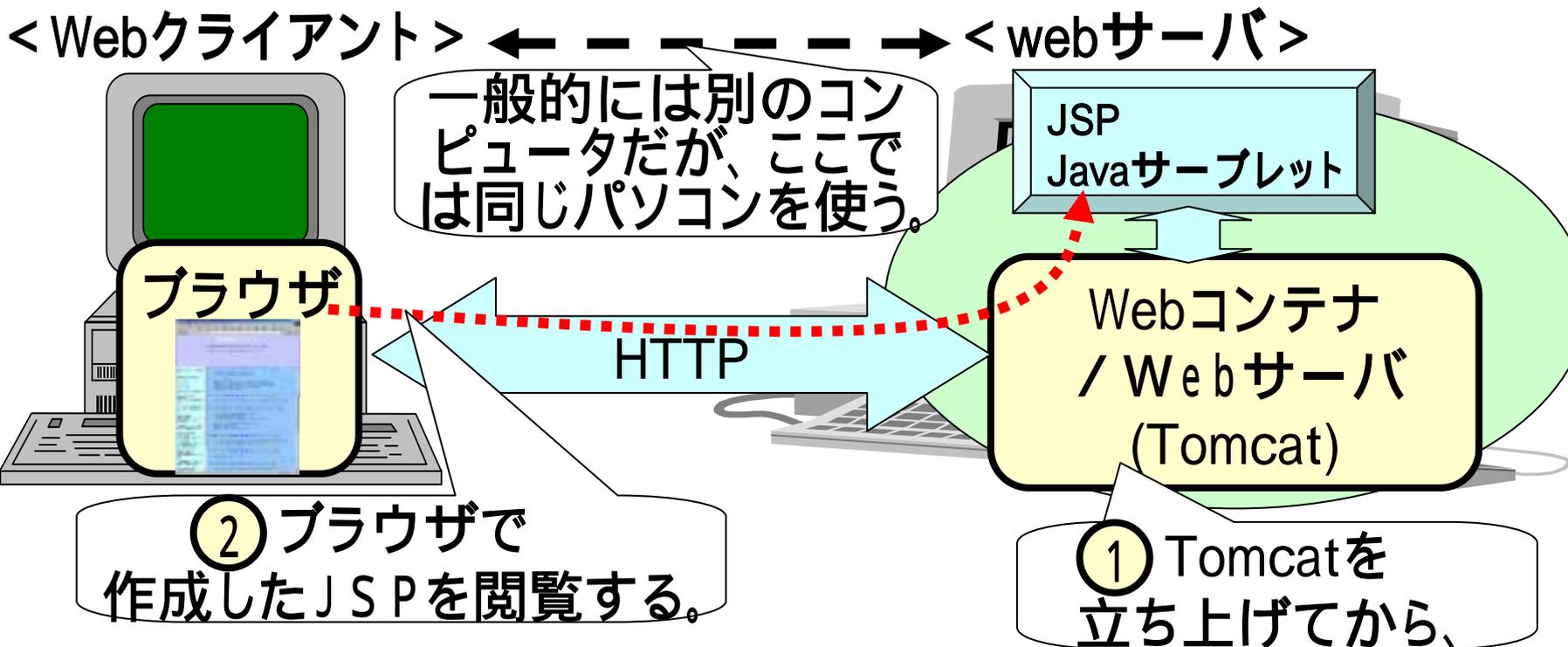
- エディターにて、何か適当なメッセージ( 、下図では“ようこそ from hello.jsp”)を編集。
- 保管ボタンをクリック( )。



# (4) Tomcatの起動とJSPの閲覧

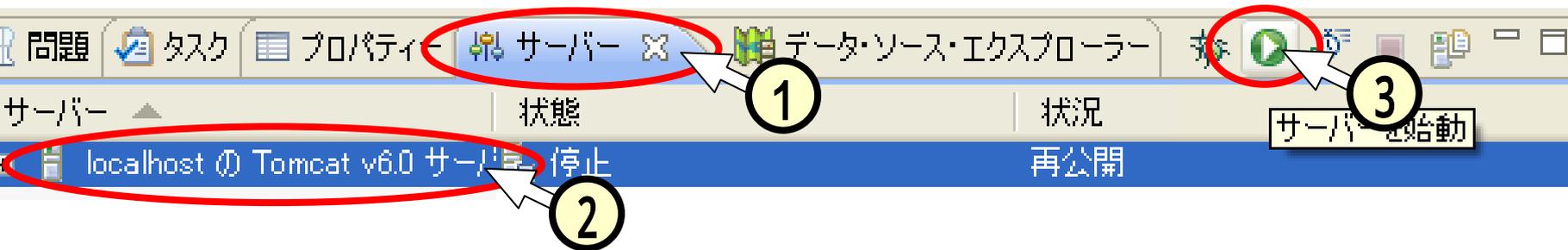
■スライド(1.1)に記したように、作成したJSPはTomcat上で動作します。したがって、次の手順で作成したJSPを閲覧します。

- Tomcatの起動( )
- WebブラウザからのJSPの閲覧( )

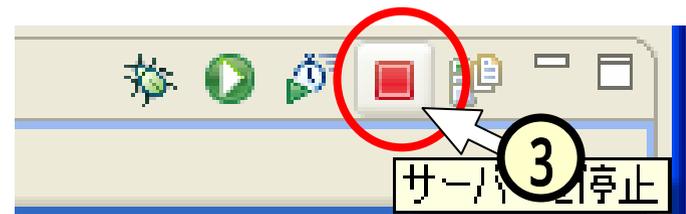


# ( 4 . 1 ) Tomcatの起動

- 「サーバービュー」のタブをクリック( )。
- [localhostのTomcatv6.0サーバ]の欄を選択( )。
- 右側の、[サーバーを始動]ボタンをクリック( )。



- 停止の際は、  
[サーバーを停止]ボタンを  
クリック( )。



## (4.2) JSPの閲覧

■ Webブラウザを立ち上げ、次のURLにアクセス。

- `http://localhost:8080/myWeb/hello.jsp`

スライド(2.2)で設定した  
プロジェクト名

スライド(3.2)で設定した  
JSPのファイル名

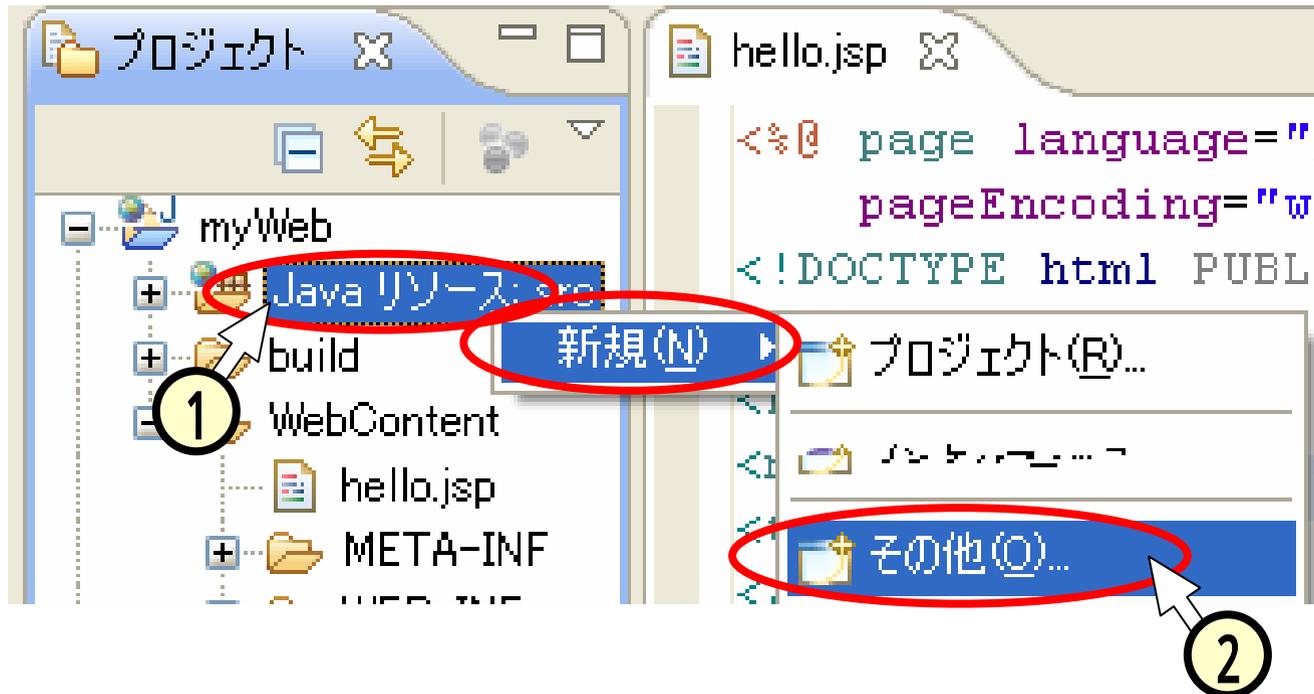


JSPを変更したり新規に作っても、一度立ち上げたTomcatを、立ち上げなおす必要はありません。

# ( 5 . 1 ) J a v a サ ー ブ レ ッ ト の 作 成 ( 1 / 6 )

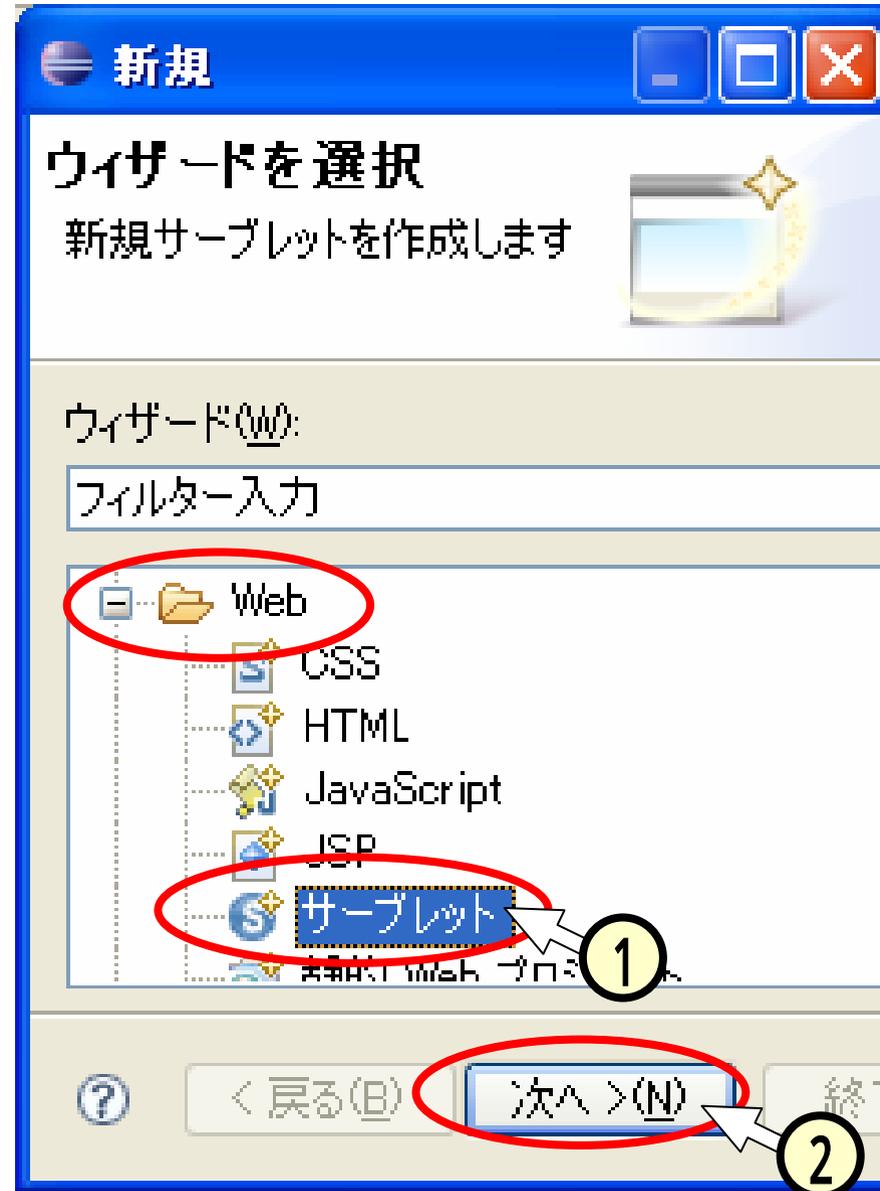
■「プロジェクトエクスプローラー」にて、

- ・作成した[myWeb]中の[Javaリソース:src]を右クリック( )。
- ・[新規]-[その他]をクリック( )。



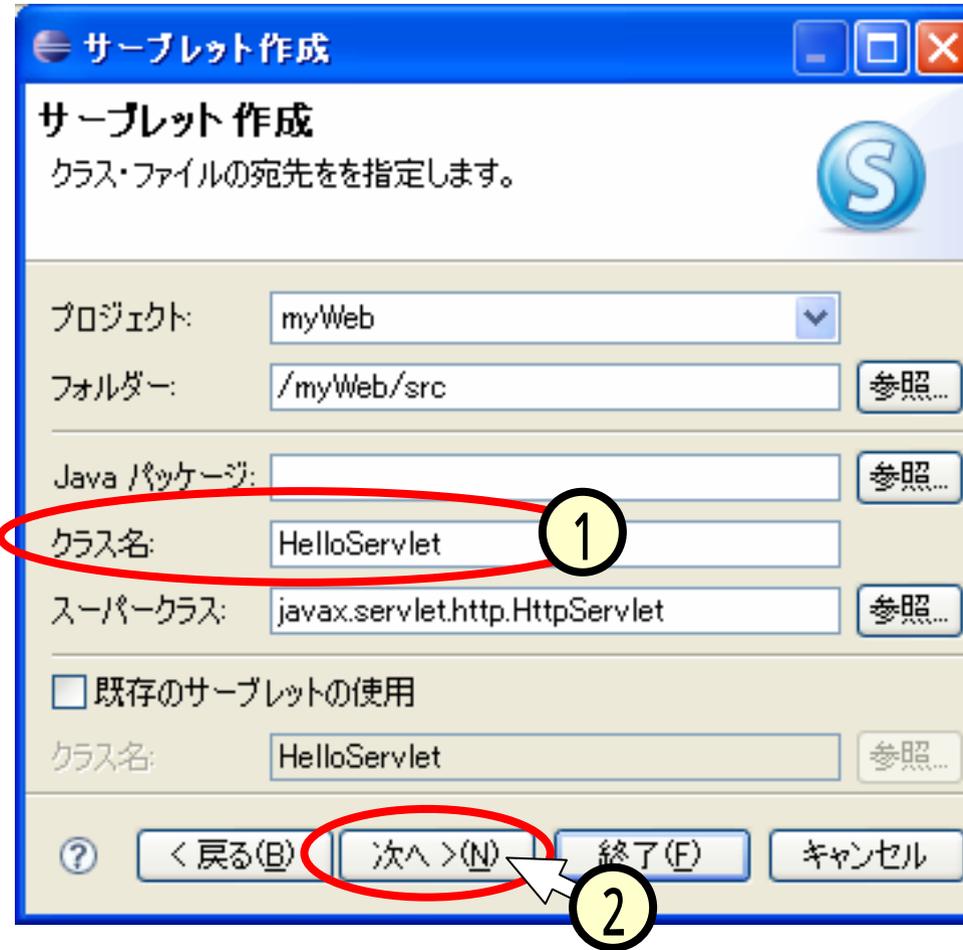
# (5.2) Javaサーブレットの作成(2 / 6)

- 「新規:ウィザードを選択」にて、
  - ・[Web]-[サーブレット]を選択( )。
  - ・[次へ]をクリック( )。



# (5.3) Javaサーブレットの作成(3 / 6)

- 「サーブレット作成」にて、[クラス名]の欄にクラス名を入力(、右図では“HelloServlet”)。
- [次へ]をクリック( )。  
パッケージの利用方法を知っている方は、パッケージを指定してもOK。



# (5.4) Javaサーブレットの作成(4/6)

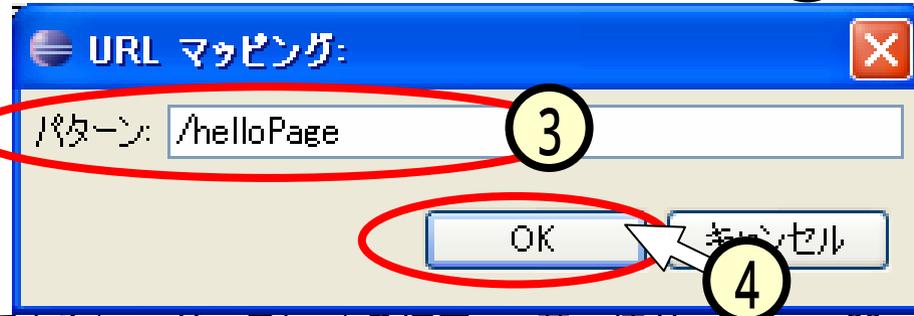
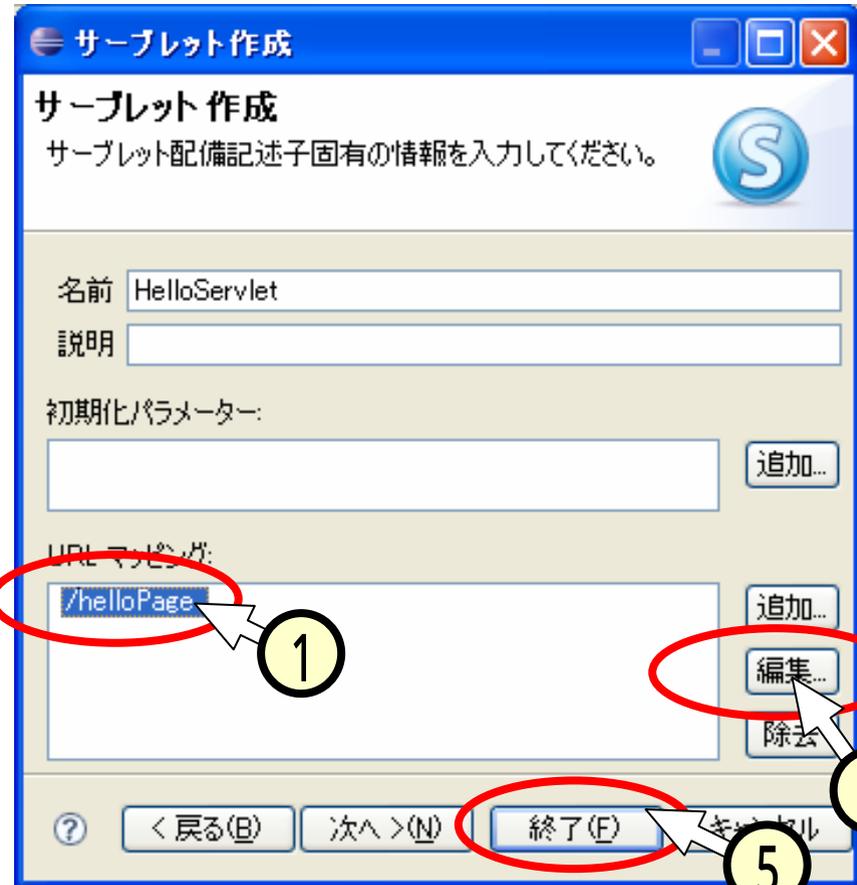
■「サーブレット作成」にて、

・[URLマッピング]欄内の項目を選択( )し、[編集]をクリック( )。

■「URLマッピング」にて、適当な名前を入力( 、右図では、“/helloPage”)して、[OK]をクリック( )。

■[終了]をクリック( )。

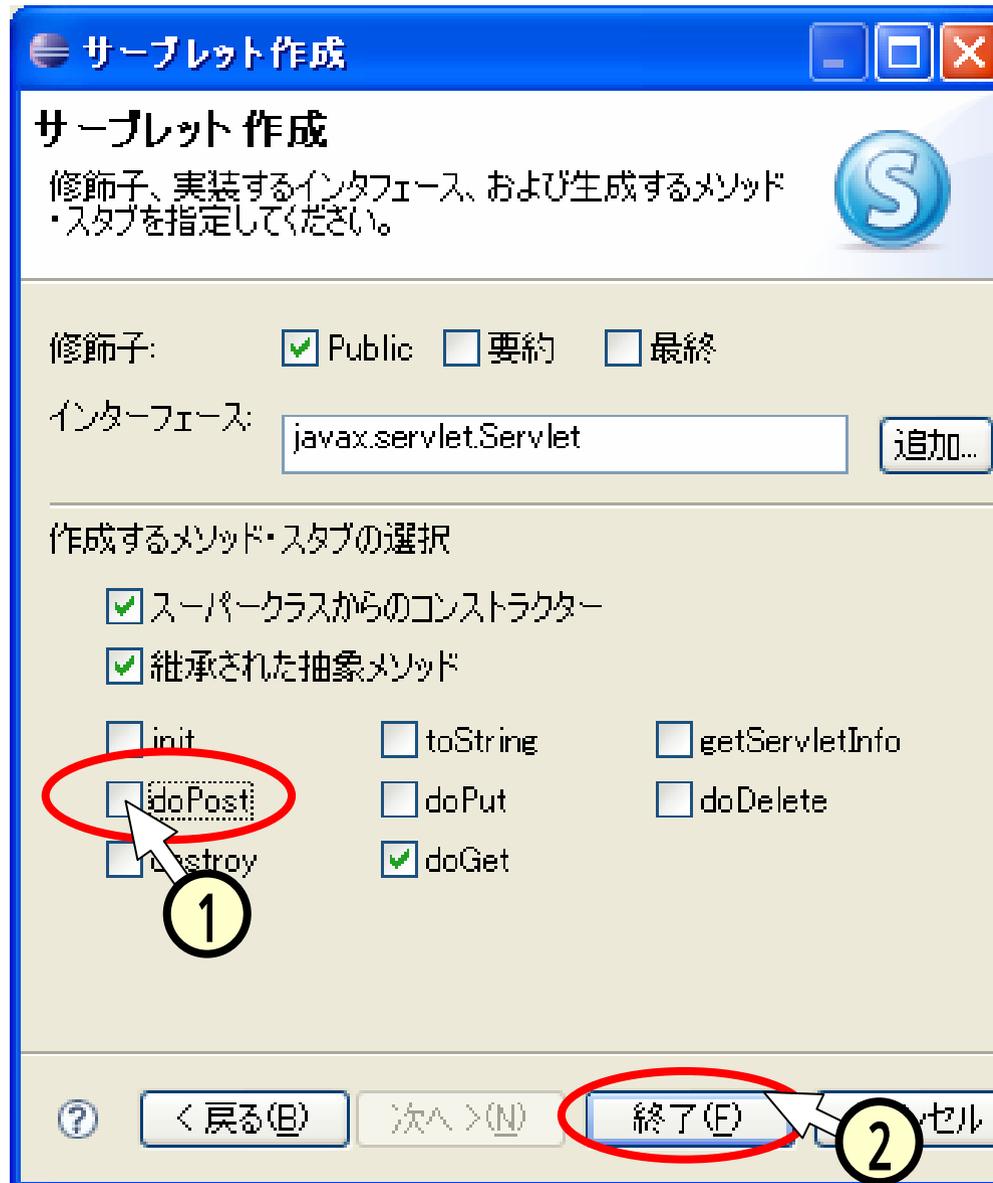
ここで指定したURLマッピングは、後で作成したサーブレットにアクセスする際のURLの一部となる。スライド( )参照。



# ( 5 . 5 ) Javaサーブレットの作成 ( 5 / 6 )

■「サーブレット作成」にて、

- ・ [doPost]のチェックを外す( )。
- ・ [終了]をクリック( )。



# (5.6) Javaサーブレットの作成(6 / 6)

■エディターにて、“doGet”のメソッド中( )に、次の文を追加。

- `response.getWriter().println("hello, from HelloSevlet.");`

■保管ボタンをクリック。



```
protected void doGet(HttpServletRequestRequest request, HttpServletResponse response) {
    // TODO Auto-generated method stub
    response.getWriter().println("hello, from HelloSevlet.");
}
```

## (5.7) サーブレットの閲覧

■ Webブラウザを立ち上げ、次のURLにアクセス。

- `http://localhost:8080/myWeb/helloPage`

スライド(2.2)で設定した  
プロジェクト名

スライド(5.4)で設定した  
URLマッピング



Javaサーブレットを変更したり新規に作ると、Tomcatを立ち上げなおす必要がありますが、eclipseではこれを自動的に行ってくれます。

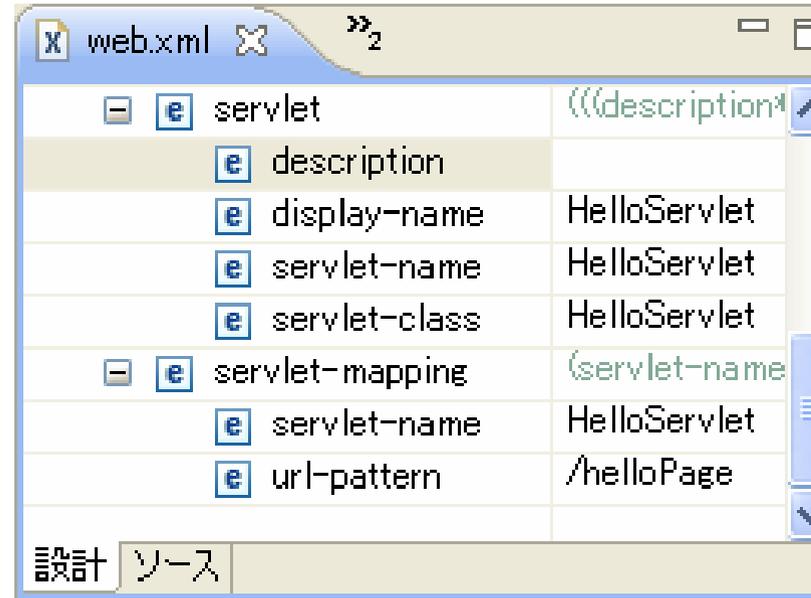


## ( 6 . 2 ) web.xmlファイルの編集

■ 次のように“web.xml”をXMLエディターで開くと、右図のような形で編集を行うことができます。

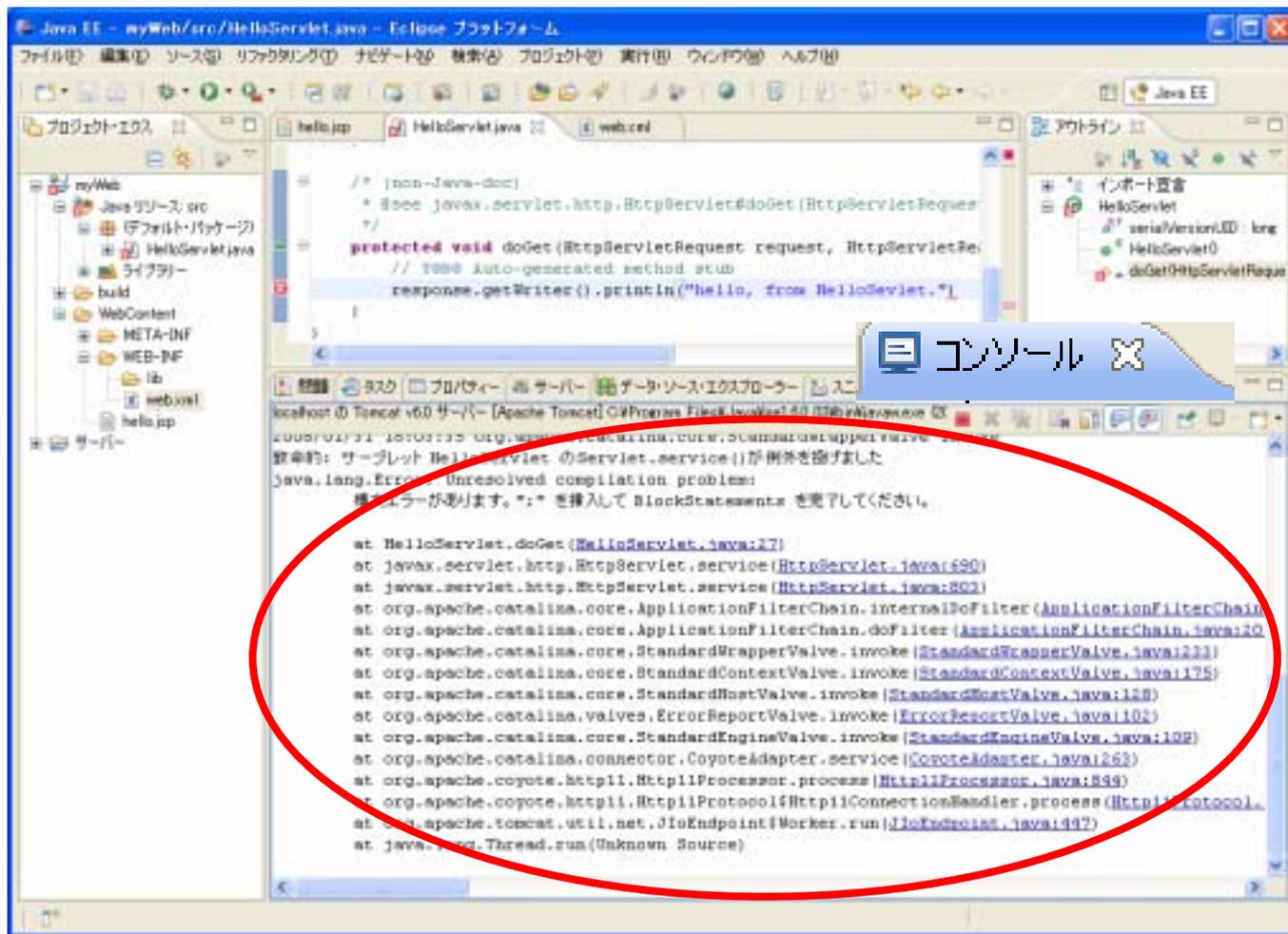
■ 上記のようにXMLエディタでファイルを開きたいとき：

・ ファイルを右クリック ( ) して、[アプリケーションから開く]-[XML Editor]をクリック ( )。



# (6.3) コンソールでのエラー表示

- 誤りのあるJSPやサーブレットが実行された場合、「コンソール・ビュー」でこれに関する情報が表示されます。

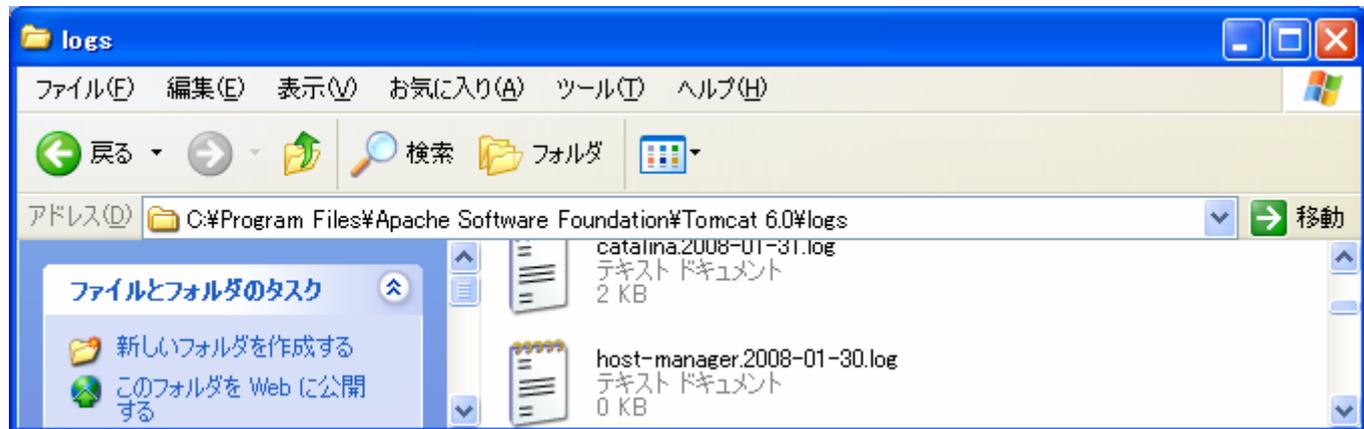


## ( 6 . 4 ) デバッグモード、ログファイル

- 下図のように[デバッグモードで始動]をクリック( )すると、Javaアプリケーションと同様な方法で、JSPやサーブレットにブレークポイントを設定することができます。



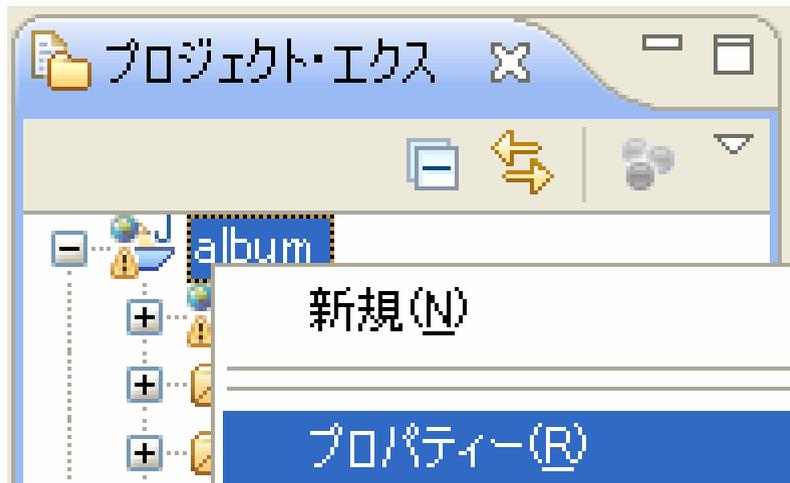
- Tomcatのログファイルは、次のフォルダにあります。  
C:\Program Files\Apache Software Foundation\Tomcat 6.0\logs



# (7.1) MySQLの利用 (1 / 3)

■MySQLを利用するには、JDBCドライバが必要です。  
これが次のフォルダにコピー済みであるとします。

- フォルダ C:\Program Files\Java\jdk1.5.0\_13\jre\lib\ext
- ファイル mysql-connector-java-5.1.5-bin.jar



# (7.2) MySQLの利用 (2 / 3)

