

## ※課題は「mvcKadai2」という 名前で新しくプロジェクトを作成して行うこと

### 【問題1】

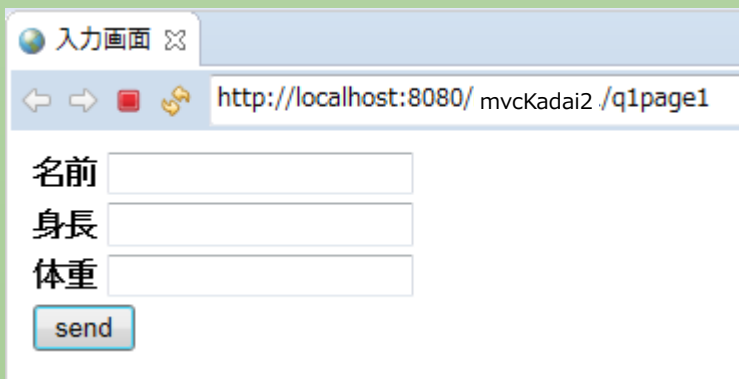
MVC モデルを使って、BMI を計算する Web アプリを作ろう！

次のように画面遷移する Web アプリケーションを作りなさい

○1 ページ目

URL: <http://localhost:8080/mvcKadai2/q1page1>

画面：



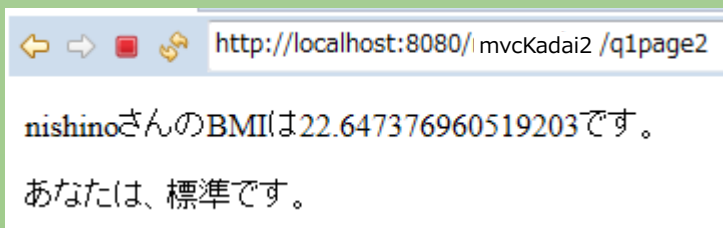
The screenshot shows a web browser window with the title '入力画面' (Input Screen). The address bar displays the URL 'http://localhost:8080/mvcKadai2/q1page1'. The main content area contains a form with three input fields labeled '名前' (Name), '身長' (Height), and '体重' (Weight). Below these fields is a button labeled 'send'.

仕様：名前と、身長、体重を入力して send ボタンをクリックすると 2 ページ目へ遷移する

○2 ページ目

URL: <http://localhost:8080/mvcKadai2/q1page2>

画面：



The screenshot shows a web browser window with the title '入力画面' (Input Screen). The address bar displays the URL 'http://localhost:8080/mvcKadai2/q1page2'. The main content area displays the text 'nishinoさんのBMIは22.647376960519203です。' (nishino's BMI is 22.647376960519203) and 'あなたは、標準です。' (You are standard).

仕様：BMI を計算して、BMI の値と測定結果に応じたメッセージを表示する。

### BMI の計算方法

BMI = 体重 ÷ (身長 (メートル) × 身長 (メートル))

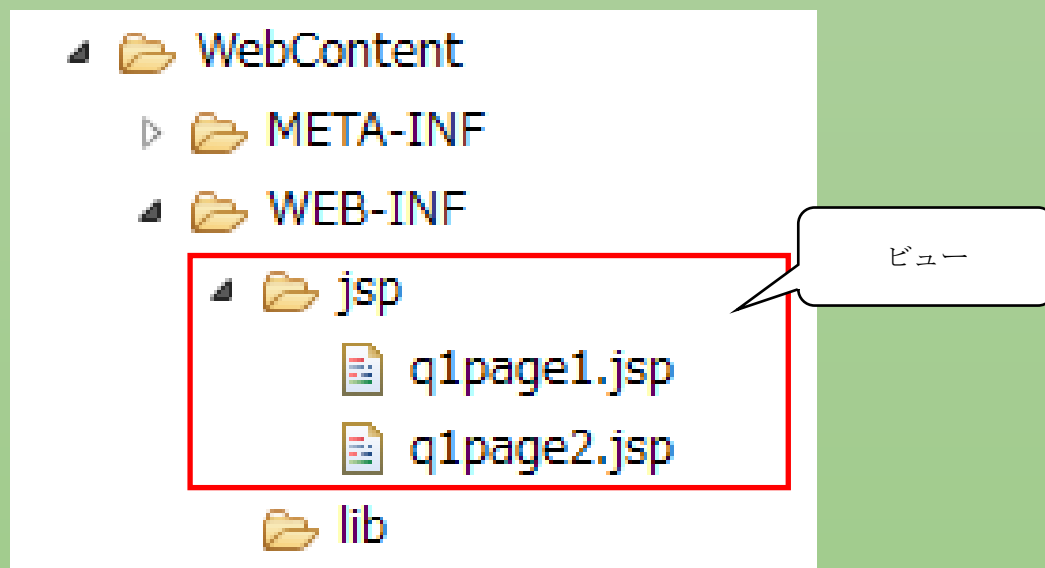
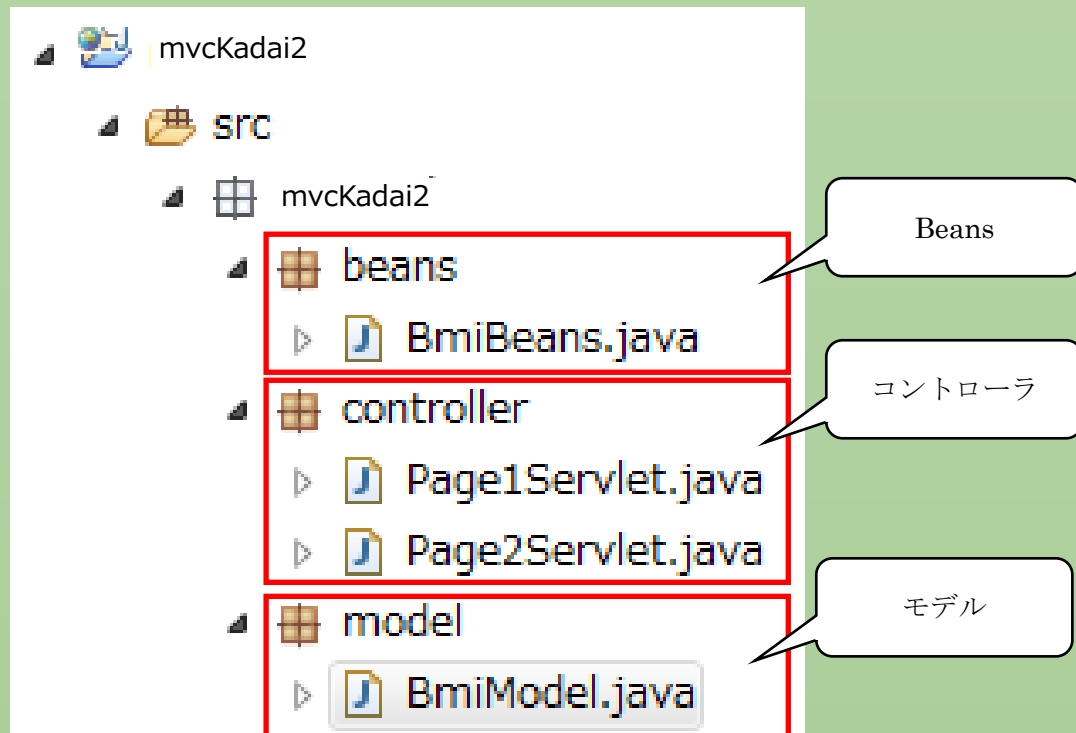
例) 身長が 172 センチで体重が 65 キロの場合

BMI = 65kg ÷ (1.72 × 1.72) となります。

BMI に順ずるメッセージは以下の通り (WHO の評価)

BMI	メッセージ
16.0 未満	痩せすぎ
17.0 未満	痩せ
18.5 未満	痩せぎみ
25.0 未満	標準
30.0 未満	過体重
35.0 未満	肥満 (1 度)
40.0 未満	肥満 (2 度)
40 以上	肥満 (3 度)

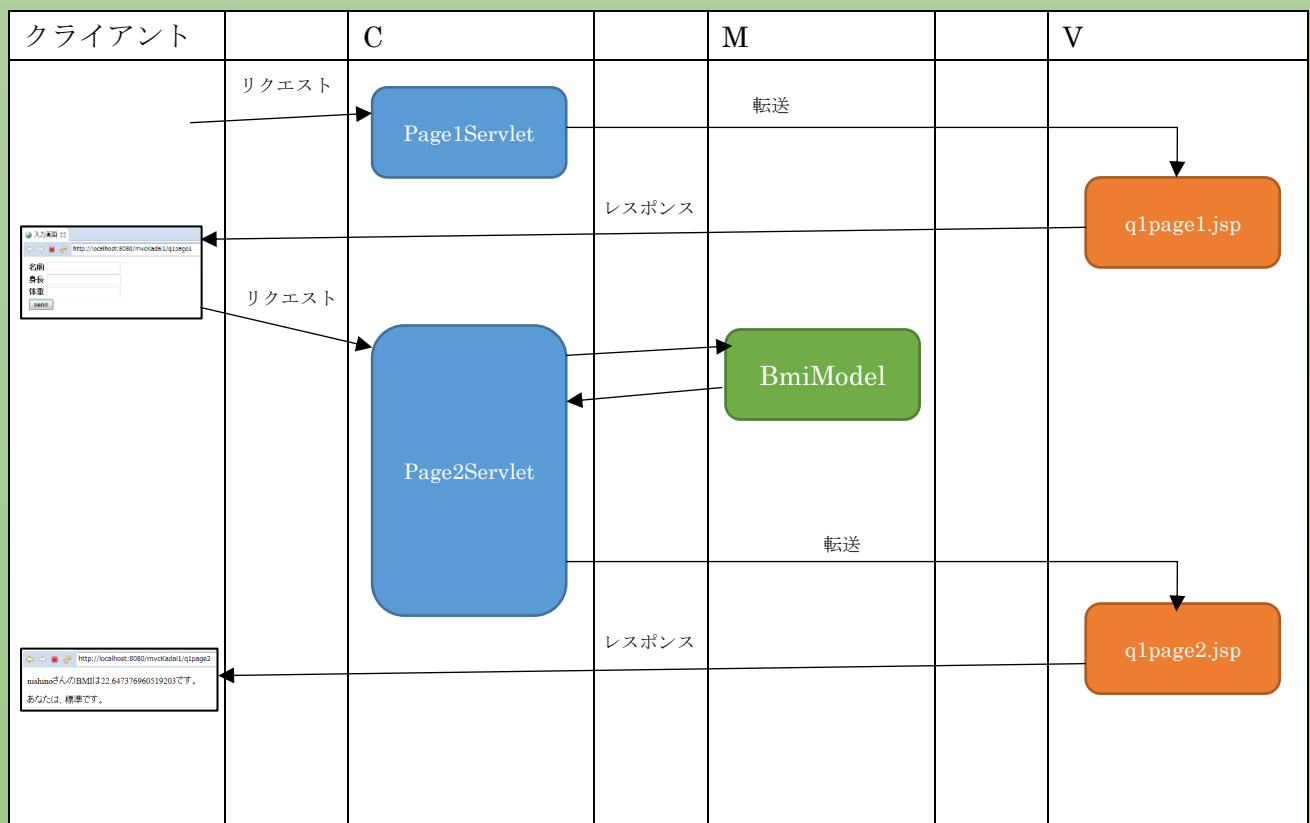
MVC モデル (+ビーンズ) で作成する為、クラス構成は以下のようにしましょう。



それぞれのクラスでやることは以下のとおり

種類	ファイル名	処理概要
ビーンズ	BmiBeans	画面から受け取った、名前、身長、体重 計算結果の BMI 値、メッセージを保持するクラス
モデル (M)	BmiModel	BMI の計算を行い、表示するメッセージを決定する
コントローラ (C)	Page1Servlet	1 ページ目を表示する (VIEW (jsp) に転送のみ)
	Page2Servlet	JSP から身長、体重を取得して、ビーンへ格納。 BmiModel の計算処理を呼び出し計算結果を取得して、 BmiBeans を 2 ページ目へ転送する
ビュー (V)	q1page1.jsp	名前、身長、体重の入力画面
	q1page2.jsp	結果の表示画面

シーケンスは以下のようになる



さらなるヒント

ビーンズは、4 ページ目にある項目を保持すればよい。

↓のような感じになります

```
5 public class BmiBeans implements Serializable {  
6     private String name; //名前  
7     private Integer weight; //体重  
8     private Integer tall; //身長  
9     private Double bmi; //BMI  
10    private String Message; //メッセージ  
11  
12  
13    ここには  
14    getterr / setter  
15    を書く  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43
```

Model は、BMI を計算するメソッドを持てばよいので、以下のようになります。

```
12 public class BmiModel {  
13  
14     /**  
15      * BMIを計算する  
16      *  
17      * @param bmiBeans BMIの入力情報が入っているビーンズ  
18      * @return 計算結果が入っているビーンズ  
19      */  
20     public BmiBeans calculate(BmiBeans bmiBeans){  
21  
22         //BMIを計算  
23  
24         //メッセージを取得  
25  
26         if 文を使って BMI の計算結果に基づいたメッセージを決定する  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47         //ビーンにセット  
48  
49         結果をビーンズに setter を使ってセットし、戻り値として return する  
50  
51  
52     }  
53 }
```

戻り値は  
メソッドの出  
力情報  
この場合  
入力情報に  
BMI 情報とメ  
ッセージを追  
加したもの

引数はこのメソ  
ッドの入力情報

計算式つかって BMI を計算する

結果をビーンズに setter を使ってセットし、戻り値として return する

2 ページ目のサーブレット側の処理では、

- JSP からの値の取得
- M（モデル）の呼び出しを行えばよいので下のようになります

```
18  @Override
19  protected void doPost(HttpServletRequest request, HttpServletResponse response)
20      throws ServletException, IOException {
21
22      //////////////////////////////////////
23      //パラメータの取得
24
25      JSP から名前、身長、体重を取得する
26
27
28      //////////////////////////////////////
29      //文字列→数値変換
30
31      ビーンズのインスタンスを作成し（new する）、setter を使って取得した名前、身長、体重を
32      ビーンズに保存する。
33
34      その際、数値型（int）は文字列→数値変換を行う
35
36
37
38
39
40
41
42
43      //////////////////////////////////////
44      //モデル呼び出し
45
46      モデルクラスを new でインスタンスを作成し、モデルに作ったメソッドを呼び出す
47
48
49      //////////////////////////////////////
50      //リクエストにセット
51
52      モデルを呼び出した結果をリクエストスコープにセット
53
54      //////////////////////////////////////
55      //転送
56
57      処理を JSP へ転送する
58  }
59 }
```