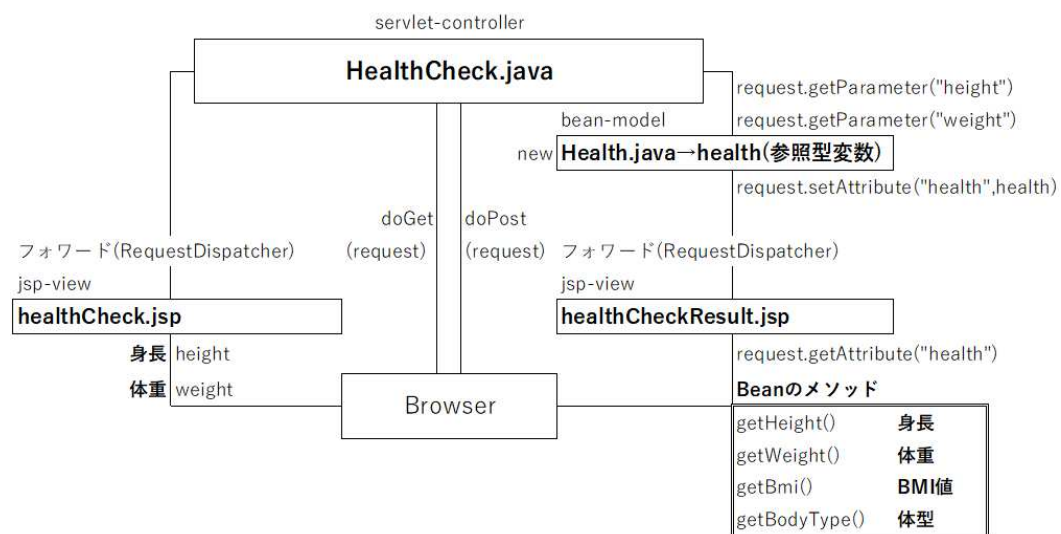


u第7章●MVCアーキテクチャの実装(健康診断)

7.3 リクエストスコープを使った「健康診断Webアプリケーション」をMVCモデルで作成

まずはサンプルから動くWebアプリを作成し、その後HealthCheckLogicをBean内に吸収したMVCモデルの標準的な形式に設計変更する。変更された内部設計は以下のとおりとする。

【内部設計】



【手順】

① サンプルプログラムをそのままの形で作成する

コード7-4、7-5、7-6、7-7、7-8を適切に配置し、必要に応じてコンパイルし実行する。

② BeanにLogicを組み込む

コンストラクタは明示的に実装する。

Healthクラス内にHealthCheckLogicの内容を実装した**execute()**メソッドを追加する。

Health.javaの内容

```
1 package model;
```

```
2
3 import java.io.Serializable;
4
5 public class Health implements Serializable {
6     private double height, weight, bmi;
7     private String bodyType;
8
9     public Health(){ //明示的にコンストラクタを実装
10 }
11 public void execute() {
12     // BMIを算出して設定
13     bmi = weight / (height / 100.0 * height / 100.0);
14
15     // BMI指数から体格を判定して設定
16     if (bmi < 18.5) {
17         bodyType = "痩せ型";
18     } else if (bmi < 25) {
19         bodyType = "普通";
20     } else {
21         bodyType = "肥満";
22     }
23 }
24 public double getHeight() {
25     return height;
26 }
27 public void setHeight(Double height) {
28     this.height = height;
29 }
30 public double getWeight() {
31     return weight;
32 }
33 public void setWeight(Double weight) {
34     this.weight = weight;
35 }
36 public void setBmi(Double bmi) {
37     this.bmi = bmi;
38 }
39 public double getBmi() {
40     return this.bmi;
41 }
42 public void setBodyType(String bodyType) {
43     this.bodyType = bodyType;
44 }
```

```
45 public String getBodyType() {
46     return this.bodyType;
47 }
48 }
49
```

③Controllerを変更する

HealthCheckLogicクラスはインポートしない。

urlパターン（アノテーション）は/Healthに変更する。

Healthクラスを実体化しパラメータをセット後にexecute()メソッドの実行を行う。

HealthCheck.javaの内容

```
1 package servlet;
2
3 import java.io.IOException;
4 import javax.servlet.RequestDispatcher;
5 import javax.servlet.ServletException;
6 import javax.servlet.annotation.WebServlet;
7 import javax.servlet.http.HttpServlet;
8 import javax.servlet.http.HttpServletRequest;
9 import javax.servlet.http.HttpServletResponse;
10 import model.Health;
11 //import model.HealthCheckLogic;
12
13 @WebServlet("/Health")
14 public class HealthCheck extends HttpServlet {
15     private static final long serialVersionUID = 1L;
16
17     protected void doGet(HttpServletRequest request,
18         HttpServletResponse response)
19         throws ServletException, IOException {
20         // フォワード
21         RequestDispatcher dispatcher =
22             request.getRequestDispatcher
23                 ("/WEB-INF/jsp/healthCheck.jsp");
24         dispatcher.forward(request, response);
25     }
26
27     protected void doPost(HttpServletRequest request,
28         HttpServletResponse response)
29         throws ServletException, IOException {
```

```

30
31 // リクエストパラメータを取得
32 String weight = request.getParameter("weight"); // 体
重
33 String height = request.getParameter("height"); // 身
長
34
35 // 入力値をプロパティに設定
36 Health health = new Health();
37 health.setHeight(Double.parseDouble(height));
38 health.setWeight(Double.parseDouble(weight));
39
40 // 健康診断を実行し結果を設定
41 //HealthCheckLogic healthCheckLogic = new
HealthCheckLogic();
42 //healthCheckLogic.execute(health);
43 health.execute();
44
45 // リクエストスコープに保存
46 request.setAttribute("health", health);
47
48 // フォワード
49 RequestDispatcher dispatcher =
50     request.getRequestDispatcher
51         ("/WEB-INF/jsp/healthCheckResult.jsp");
52 dispatcher.forward(request, response);
53 }
54 }
55

```

④結果画面でBMIの値を小数点以下2桁で表示する

healthCheckResult.jspでBMIの値を四捨五入して少数点以下2桁で出力
 するよう変更する。

healthCheckResult.jspの内容

```

1 <%@ page language="java" contentType="text/html;
charset=UTF-8"
2     pageEncoding="UTF-8" %>
3 <%@ page import="model.Health" %>
4 <%
5 // リクエストスコープに保存されたHealthを取得
6 Health health = (Health) request.getAttribute("health");

```

```

7 %>
8 <!DOCTYPE html>
9 <html>
10 <head>
11 <meta charset="UTF-8">
12 <title>スッカリ健康診断</title>
13 </head>
14 <body>
15 <h1>スッカリ健康診断の結果</h1>
16 <p>
17 身長 : <%= health.getHeight() %>
18 体重 : <%= health.getWeight() %>
19 BMI : <%=
○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○ %>
20 体格 : <%= health.getBodyType() %>
21 </p>
22 <a href="Health">戻る</a>
23 </body>
24 </html>

```

【演習】

○○○の部分を考えてみましょう。
 以下の実行例のようになりますか？どこか一つ修正箇所を忘れています
 のでうまくいかないはずです。
 問題箇所を修正して実行画面のスクショを提出してください。

【実行例】

↓urlパターンが見えていること

