

```

1  package org.wicket_sapporo.handson.validation;
2
3  import java.util.Arrays;
4  import java.util.List;
5
6  import org.apache.wicket.markup.html.WebPage;
7  import org.apache.wicket.markup.html.form.Form;
8  import org.apache.wicket.markup.html.form.RadioChoice;
9  import org.apache.wicket.markup.html.form.TextField;
10 import org.apache.wicket.markup.html.panel.FeedbackPanel;
11 import org.apache.wicket.model.CompoundPropertyModel;
12 import org.apache.wicket.model.IModel;
13 import org.apache.wicket.model.util.ListModel;
14 import org.apache.wicket.validation.validator.StringValidator;
15 import org.wicket_sapporo.handson.beans.UserLunch;
16
17 /**
18  * 入力チェックを加えた CPModelFormPage.
19  */
20 public class ValidationFormPage extends WebPage {
21     private static final long serialVersionUID = 1L;
22
23     /**
24      * コンストラクタ.
25      */
26     public ValidationFormPage() {
27         IModel<UserLunch> userLunchModel = new CompoundPropertyModel<>(new UserLunch());
28
29         Form<UserLunch> form = new Form<UserLunch>("form", userLunchModel) {
30             private static final long serialVersionUID = 1L;
31
32             @Override
33             protected void onSubmit() {
34                 super.onSubmit();
35                 System.out.println("name : " + getModelObject().getName());
36                 System.out.println("lunch : " + getModelObject().getLunch());
37                 if (getModelObject().getLunch().equals("鳥ガーリック定食")) {
38                     error(getModelObject().getName() + "さん、鳥ガーリック定食は売り切れです...");
39                 } else {
40                     info(getModelObject().getName() + "さんの注文が完了しました！");
41                 }
42             }
43         };
44         add(form);
45
46         // エラーメッセージを表示するためのFeedbackPanelを用意
47
48         FeedbackPanel feedbackPanel = new FeedbackPanel("feedback");
49         form.add(feedbackPanel);
50
51         TextField<String> nameField = new TextField<String>("name") {
52             private static final long serialVersionUID = 1L;
53
54             @Override
55             protected void onInitialize() {
56                 super.onInitialize();
57                 // テキストフィールドを必須にして、最小3文字最大8文字に設定する。
58                 setRequired(true);
59                 add(StringValidator.minimumLength(3));
60                 add(StringValidator.maximumLength(8));
61             }
62         };
63         form.add(nameField);
64
65         List<String> lunches = Arrays.asList("鶏唐揚げ定食", "鳥かつ定食", "鳥ガーリック定食");
66         IModel<List<String>> lunchesModel = new ListModel<>(lunches);
67
68         RadioChoice<String> radioChoice = new RadioChoice<String>("lunch", lunchesModel) {
69             private static final long serialVersionUID = 1L;

```

```
70     @Override
71     protected void onInitialize() {
72         super.onInitialize();
73         // ラジオボタンの選択を必須にする。
74         setRequired(true);
75     }
76 };
77 form.add(radioChoice);
78
79 }
80
81 }
```