Złożone komponenty JSF

wg

https://docs.oracle.com/javaee/7/JEETT.pdf

podrozdział 8.5, rozdział 14

http://www.coreservlets.com

Technologie internetowe 9

Opis znaczników obsługiwanych przez Facelets (tutorial EE 7)

Przegląd znaczników JSF (UI) - composite

wykład3:

http://zofia.kruczkiewicz.staff.iiar.pwr.wroc.pl/wyklady/ti_/TINT_3.pdf

Znacznik	Funkcja znaczników szablonu
composite:interface	Definicja jednego komponentu jako połączenie cech wielu komponentów
composite:implementation	Definiuje implementację kompozytowego komponentu. W przypadku definicji **composite:interface** implementacja musi być zgodna z tą definicją
composite:attribute	Deklaracja atrybutu instancji komponentu, do którego ten znacznik jest przypisany
composite:insertChildren	Dowolny komponent lub tekst szablonu ze znacznikiem kompozytowym w używanej stronie jest powtarzany w punkcie umieszczenia tego znacznika w ramach znacznika **composite:implementation**

Przegląd znaczników JSF (UI) - composite (cd)

composite:valueHolder	Deklaracja znacznika wewnątrz znacznika **composite:interface**. Definicja implementacji **ValueHolder** właściwego dla obiektów używanych na stronie
composite:editableValueHolder	Deklaracja znacznika wewnątrz znacznika **composite:interface**. Definicja implementacji **EditableValueHolder** właściwego dla obiektów używanych na stronie
composite:actionSource	Deklaracja znacznika wewnątrz znacznika **composite:interface**. Definicja implementacji **actionSource** właściwego dla obiektów używanych na stronie

Biblioteki znaczników obsługiwanych przez Facelets

wykład3:

http://zofia.kruczkiewicz.staff.iiar.pwr.wroc.pl/wyklady/ti_/TINT_3.pdf

Biblioteki znaczników	URI	Prefiks	Przykład	Zawartość
Composite Component Tag Library	**http://xmlns.jcp.org/jsf/composite**	**cc:**	cc:interface	Znaczniki wspierające komponenty kompozytowe

Atrybuty komponentu złożonego

Nazwa atrybutu	Opis
name	Specyfikuje: • nazwę atrybutu komponentu złożonego, używanego przez stronę, wykorzystującej dany component. • alternatywnie, ten atrybut może reprezentować nazwę standardowego słuchacza zdarzeń: action lub actionListener, • nazwę ManagedBean
default	Specyfikuje domyślną nazwę atrybutu komponenetu złożonego
required	Określa, kiedy nazwa jest atrybutu jest obowiązkowa
method-signature **type**	**method-signature** specyfikuje podklasę klasy java.lang.Object, jako atrybut komponentu złożonego i deklaruje definicję metody. **type** specyfikuje w pełni kwalifikowaną nazwę klasy jako typ atrybutu. Atrybuty **type** i **method-signature** wzajemnie się wykluczają. Jeśli specyfikuje się te dwa atrybuty jednocześnie, atrybut **method-signature** jest ignorowany. Domyślny typ atrybutu **type i method-signature** jest java.lang.Object. **Uwaga:** Wyrażenie specyfikujące metodę jest podobne do specyfikacji atrybutu, ale zamiast definiować dynamikę właściwości metody, wspiera wywołanie metody podanego obiektu, podając specyfikację zbioru przekazywanych parametrów i zwracany wynik wywołania tej metody (jeśli ta metoda zwraca wynik).

Przykłady definicji atrybutów

default:

```
<composite:attribute name="username" default="admin"/>
```

method-signature:

```
<composite:attribute name="myaction"
                     method-signature="java.lang.String action()"/>
```

type:

```
<composite:attribute name="dateofjoining"
                              type="java.util.Date"/>
```

Odwołanie do komponentu typu Managed Bean

Aby komponent złożony mógł realizować przetwarzanie po stronie serwera, należy go powiązać z komponentem typu Managed Bean w następujący sposób:

- przekazanie referencji komponentu **Managed Bean** do złożonego komponentu
- bezpośrednie użycie składowych komponentu typu **Managed Bean** w komponencie złożonym

Walidacja wartości komponentu złożonego

Walidacja w technologii JSF jest używana w złożonych **komponentach wejściowych** zdefiniowanych za pomocą znaczników typu

- **composite:valueHolder**
- **composite:editableValueHolder.**

Znaczniki walidacji

Nazwa	Opis
f:validateBean	Deleguje walidację lokalnych wartości Bean Validation API
f:validateRegex	Używa wzorca atrybutu do walidacji komponentu. Wejściowy wzorzec odpowiada wartości łańcuchowej komponentu. Jeśli wartości są zgodne, walidacja przebiega poprawnie.
f:validateRequired	Sprawdza obecność wartości atrybutu – równoważny efekt to ustawienie elementu atrybutu komponenetu na wartość true.

Procedura tworzenia i wykorzystanie komponentu złożonego:

1. **Wykonanie folderu resource** w katalogu głównym **Web Pages** projektu i **zagnieżdżonego folderu**, gdzie będą umieszczane pliki z definicją komponentu złożonego.
2. **Wykonanie pliku xhtml** typu JSF Page w zagnieżdżonym w folderze resources i zadeklarowanie przestrzeni nazw komponentu złożonego
3. Wykorzystanie znaczników **composite:interface, composite:attribute** i **composite:implementation**, do zdefiniowania zawartości komponentu złożonego.
4. Znacznik **composite:interface** są używany do deklaracji wartości konfiguracyjnych. Znacznik **composite:implementation** jest używany do deklaracji znaczników XHTML. Wykorzystuje on atrybuty znacznika **composite:interface** za pomocą wyrażenia: **#{cc.attrs.attributeName}.**

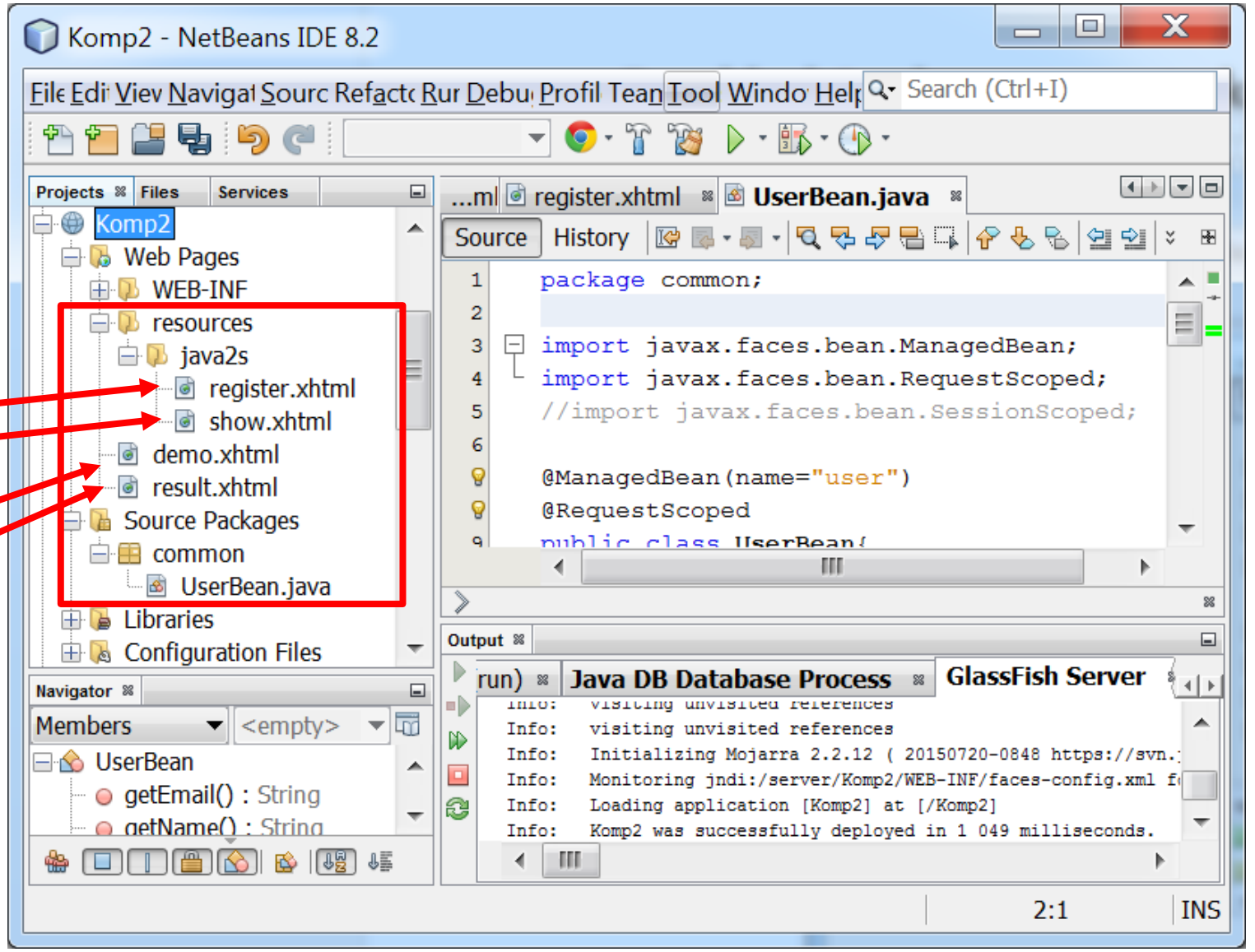
Procedura tworzenia i wykorzystanie komponentu złożonego (cd):

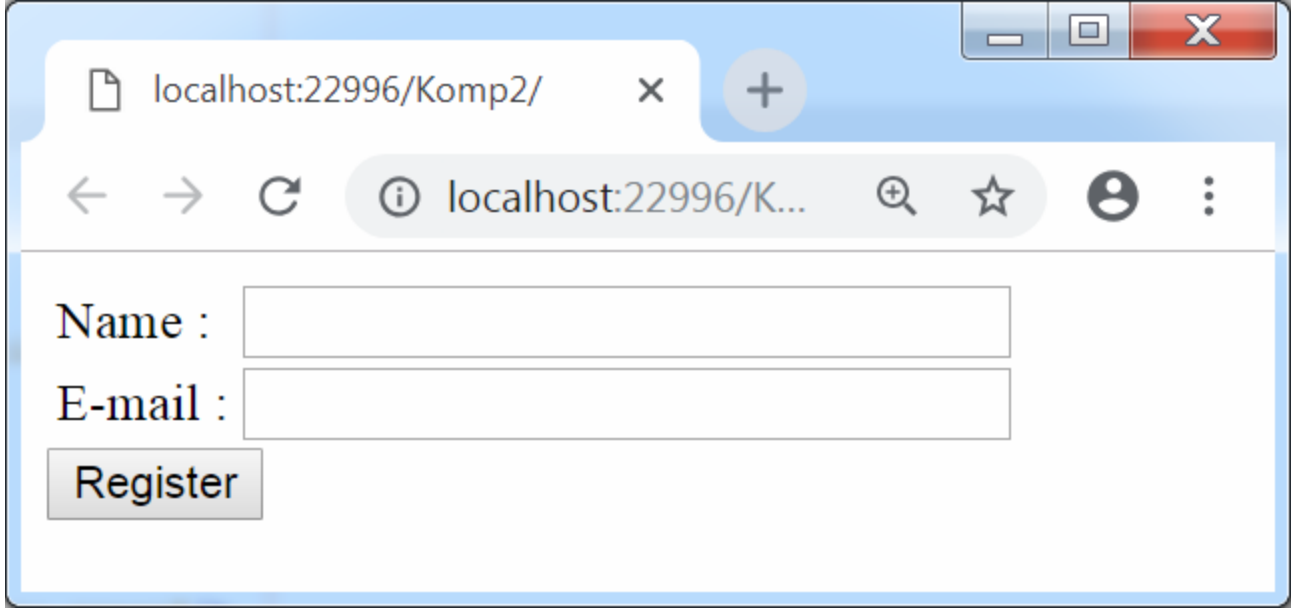
5. **W celu użycia komponentu kompozytowego** należy wykonać kolejny plik typu **xhtml** i w przestrzeni nazw strony umieścić wyrażenie reprezentujące miejsce definicji komponentu złożonego np **http://java.sun.com/jsf/nazwa_folderu,** gdzie **nazwa_folder** jest nazwą folderu wykonanego w folderze **resources** (p. 1),
6. Zastosowanie zdefiniowanego komponentu złożonego, podobnie jak komponenty JSF

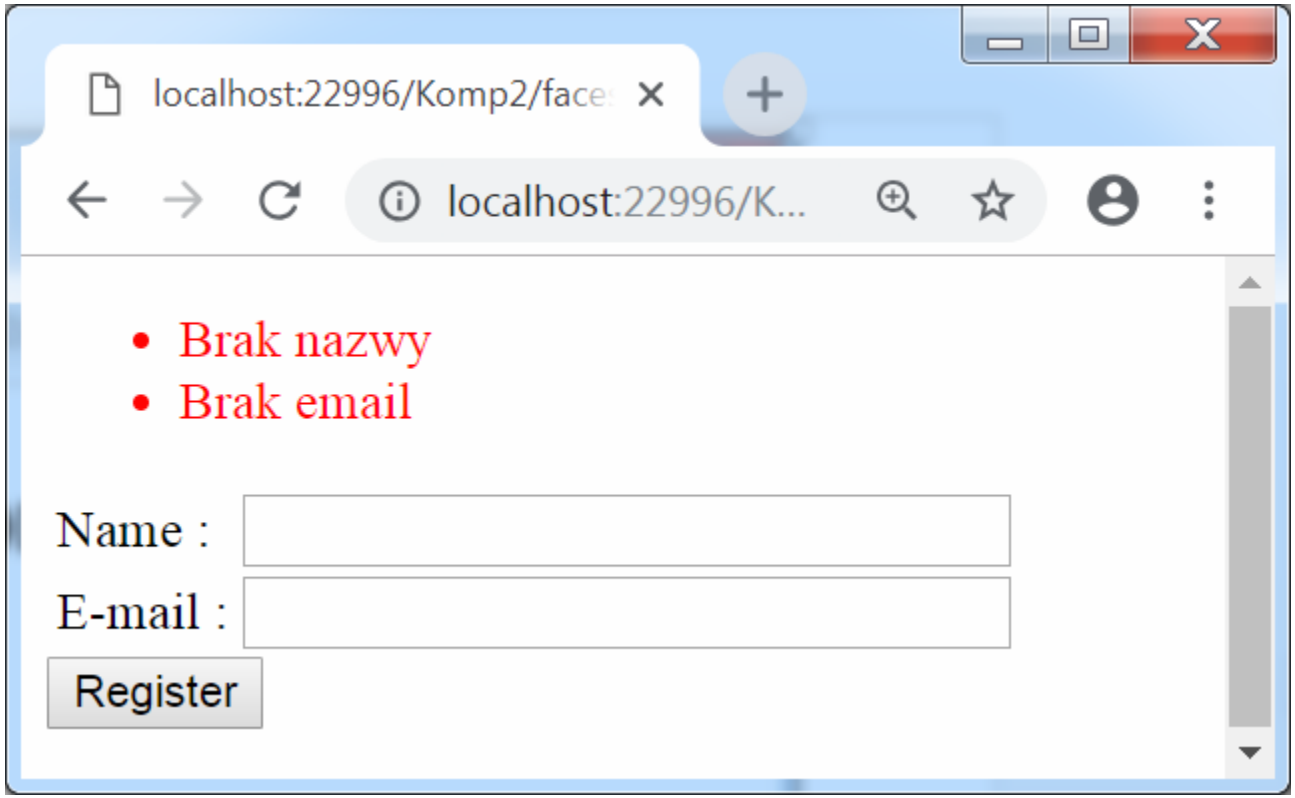
Przykład 1 – komponenty kompozytowe

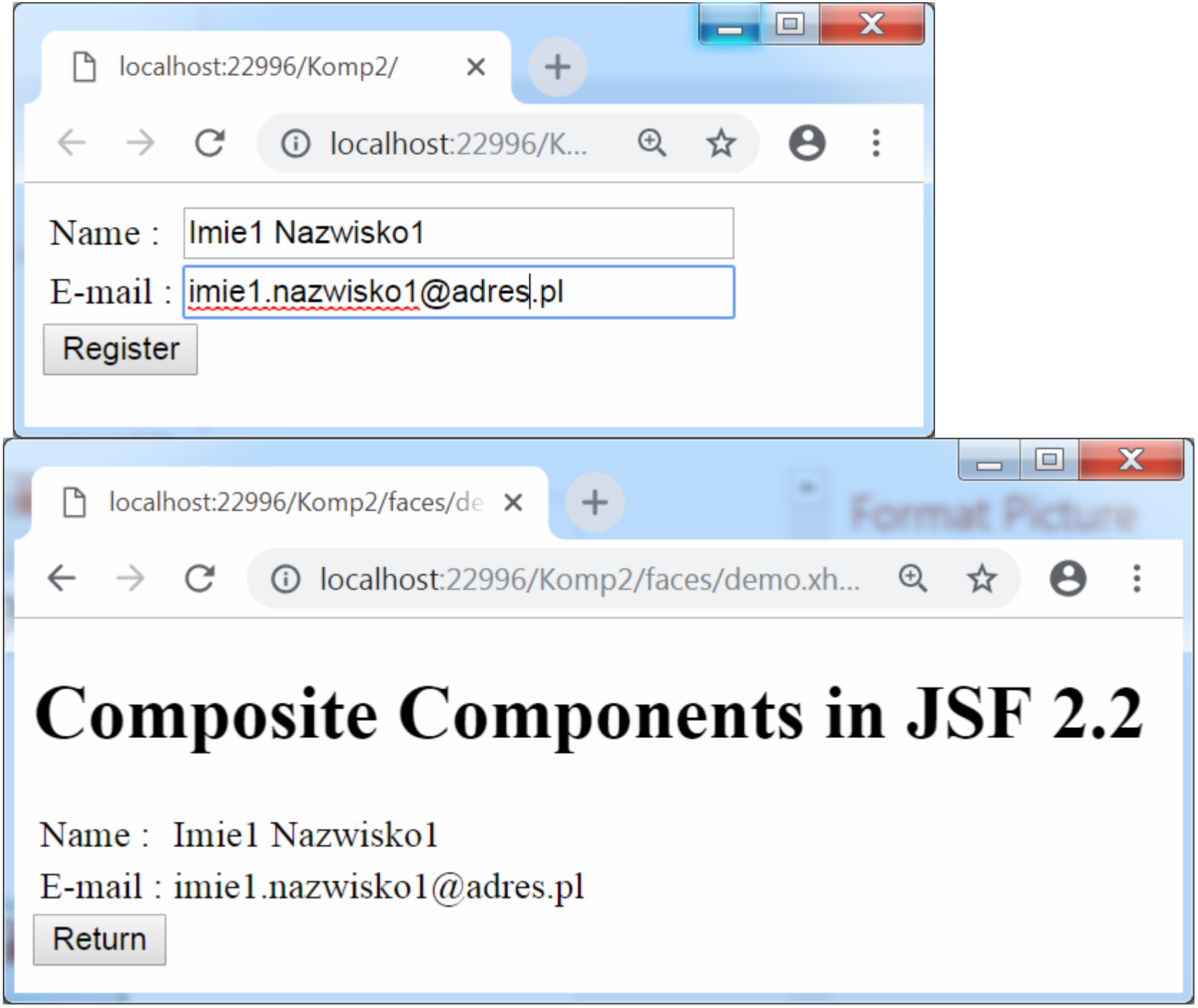
Definicje komponentów złożonych

Wykorzystanie komponentów złożonych do definicji stron




localhost:22996/Komp2/
localhost:22996/K...
Name :
E-mail :
Register


localhost:22996/Komp2/face
localhost:22996/K...
Brak nazwy
Brak email
Name :
E-mail :
Register


localhost:22996/Komp2/
localhost:22996/K...
Name :
Imie1 Nazwisko1
E-mail :
imie1.nazwisko1@adres.pl
Register
localhost:22996/Komp2/faces/de
localhost:22996/Komp2/faces/demo.xh...
Composite Components in JSF 2.2
Name : Imie1 Nazwisko1
E-mail : imie1.nazwisko1@adres.pl
Return

Definicja komponentu złożonego - register (1)

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
    xmlns:h="http://xmlns.jcp.org/jsf/html"   xmlns:f="http://xmlns.jcp.org/jsf/core"
    xmlns:composite="http://xmlns.jcp.org/jsf/composite">

 <composite:interface >
     <composite:attribute name="nameLable" />
     <composite:attribute name="nameValue" />
     <composite:attribute name="emailLable" />
     <composite:attribute name="emailValue" />
     <composite:attribute name="registerButtonText" />
     <composite:attribute name="showButtonText" />
     <composite:attribute name="messageName" />
     <composite:attribute name="messageEmail" />
     <composite:attribute name="registerButtonAction" method-
                                  signature="java.lang.String action()" />
  </composite:interface>
```

Definicja komponentu złożonego - register (2)

```
<composite:implementation >
  <h:form>
    <h:panelGroup id="messagePanel" layout="block" >
      <h:messages errorStyle="color: red" infoStyle="color: green"/>
    </h:panelGroup>
    <h:panelGrid columns="2" id="textPanel">
        #{cc.attrs.nameLable} :
        <h:inputText id="name" value="#{cc.attrs.nameValue}" size="30"
                required="true"  requiredMessage="#{cc.attrs.messageName}" />
       #{cc.attrs.emailLable} :
        <h:inputText id="email" value="#{cc.attrs.emailValue}"  size="30"
                required="true" requiredMessage="#{cc.attrs.messageEmail}"/>
    </h:panelGrid>
    <h:commandButton action="#{cc.attrs.registerButtonAction}"
                value="#{cc.attrs.registerButtonText}"/>
  </h:form>
</composite:implementation>

</html>
```

Definicja komponentu złożonego - show (1)

```
<?xml version='1.0' encoding='UTF-8' ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
   <html xmlns="http://www.w3.org/1999/xhtml"
     xmlns:h="http://xmlns.jcp.org/jsf/html"
     xmlns:f="http://xmlns.jcp.org/jsf/core"
     xmlns:composite="http://xmlns.jcp.org/jsf/composite">

    <composite:interface>
      <composite:attribute name="nameLable" />
      <composite:attribute name="nameValue1" />
      <composite:attribute name="emailLable" />
      <composite:attribute name="emailValue1" />
      <composite:attribute name="showButtonText" />
      <composite:attribute name="showButtonAction"
                    method-signature="java.lang.String action()" />
  </composite:interface>
```

Definicja komponentu złożonego - show (2)

```
<composite:implementation >
      <h:form>
       <h:panelGrid columns="2" id="textPanel">
           #{cc.attrs.nameLable} :
           <h:outputText id="name" value="#{cc.attrs.nameValue1}" />
           #{cc.attrs.emailLable} :
           <h:outputText id="email" value="#{cc.attrs.emailValue1}" />
        </h:panelGrid>
        <h:commandButton action="#{cc.attrs.showButtonAction}"
                                 value="#{cc.attrs.showButtonText}"/>
      </h:form>
   </composite:implementation>
</html>
```

Definicja strony - demo

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
    xmlns:h="http://xmlns.jcp.org/jsf/html"
    xmlns:java2s="http://xmlns.jcp.org/jsf/composite/java2s" >
  <h:body>
    <java2s:register
       nameLable="Name"
       nameValue="#{user.name}"
       emailLable="E-mail"
       emailValue="#{user.email}"
       messageName="Brak nazwy"
       messageEmail="Brak email"
       registerButtonText="Register"
       registerButtonAction="#{user.registerAction}" />
  </h:body>
</html>
```

Definicja strony - result

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
     xmlns:h="http://xmlns.jcp.org/jsf/html"
     xmlns:java2s="http://xmlns.jcp.org/jsf/composite/java2s">

   <h:body>
      <h1>Composite Components in JSF 2.2</h1>
      <java2s:show
         nameLable="Name"
         nameValue1="#{user.name}"
         emailLable="E-mail"
         emailValue1="#{user.email}"
         showButtonText="Return"
         showButtonAction="#{user.registerAction1}"/>
   </h:body>
</html>
```

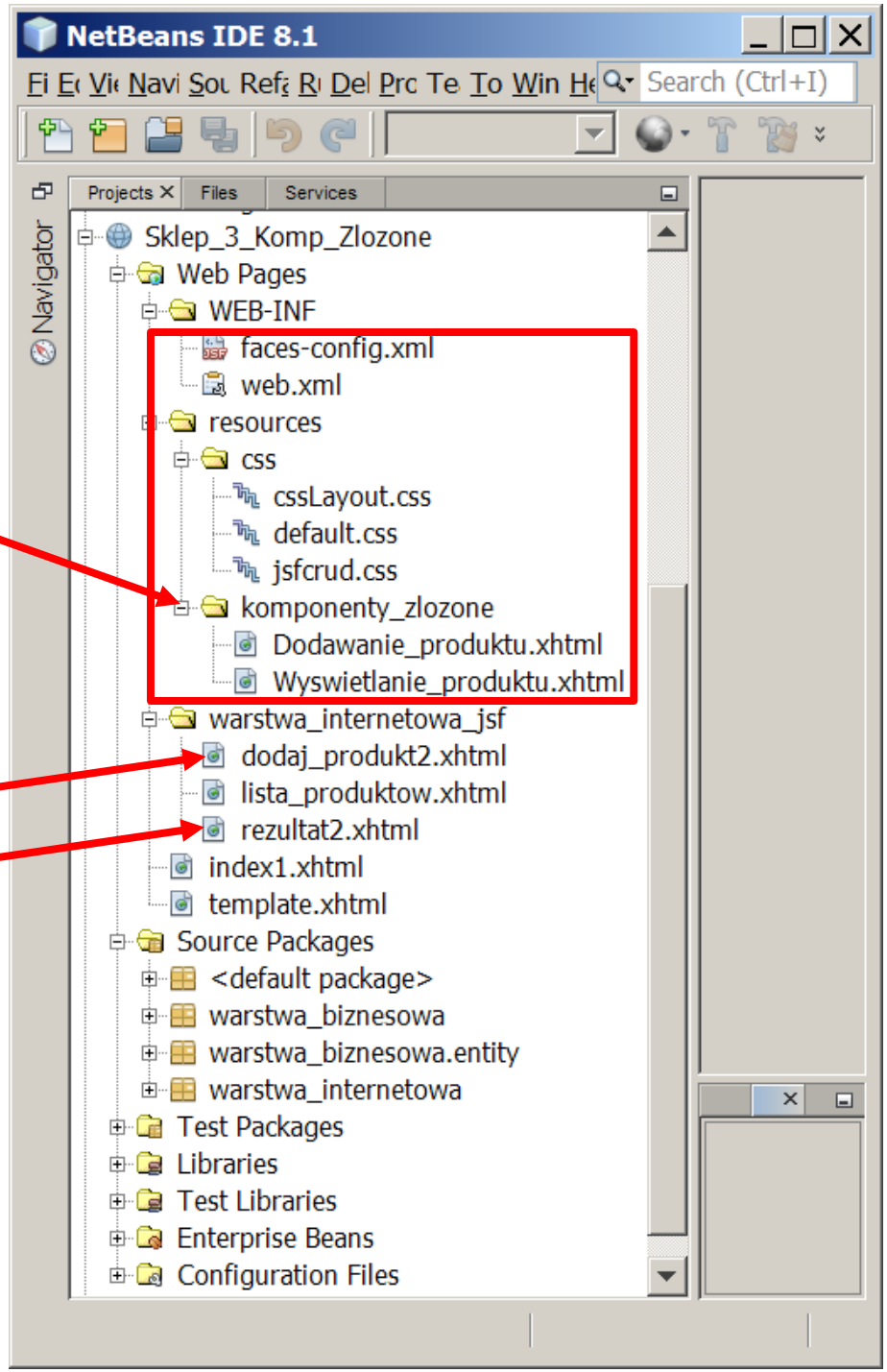
Komponent typu Managed Bean

```
package common;
import javax.faces.bean.ManagedBean;
import javax.faces.bean.SessionScoped;

@ManagedBean(name="user")
@SessionScoped
public class UserBean{
        public String name;
        public String email;

        public String getName()              {   return name;          }
        public void setName(String name)     {   this.name = name;     }
        public String getEmail()             {   return email;         }
        public void setEmail(String email)   {   this.email = email;   }
        public String registerAction()       {   return "result";      }
        public String registerAction1()      {   return "demo";        }
}
```

Przykład 2 – Sklep3

Definicje komponentów złożonych

Wykorzystanie komponentów złożonych do definicji stron



Dodawanie_produktu

```
<?xml version='1.0' encoding='UTF-8' ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://xmlns.jcp.org/jsf/html"
      xmlns:composite="http://xmlns.jcp.org/jsf/composite">
   <composite:interface >
      <composite:attribute name="nameLable" />
      <composite:attribute name="nameValue" />
      <composite:attribute name="titleName" />
      <composite:attribute name="nameMessage" />
      <composite:attribute name="cenaLable" />
      <composite:attribute name="cenaValue" />
      <composite:attribute name="titleCena" />
      <composite:attribute name="cenaMessage" />
      <composite:attribute name="promocjaLable" />
      <composite:attribute name="promocjaValue" />
      <composite:attribute name="titlePromocja" />
      <composite:attribute name="promocjaMessage" />
      <composite:attribute name="dodajButtonText" />
      <composite:attribute name="dodajButtonAction" method-signature="java.lang.String action()" />
   </composite:interface>
```

Dodawanie_produktu.xhtml

```
<composite:implementation >
    <h:form>
      <h:panelGrid columns="2" id="textPanel">
        #{cc.attrs.nameLable} :
        <h:inputText id="name" value="#{cc.attrs.nameValue}"
                title="#{cc.attrs.titleName}"
                required="true"  requiredMessage="#{cc.attrs.nameMessage}"/>
        #{cc.attrs.cenaLable} :
        <h:inputText id="cena" value="#{cc.attrs.cenaValue}"
                title="#{cc.attrs.titleCena}"
                required="true"  requiredMessage="#{cc.attrs.cenaMessage}"/>
        #{cc.attrs.promocjaLable} :
        <h:inputText id="promocja" value="#{cc.attrs.promocjaValue}"
                title="#{cc.attrs.titlePromocja}"
                required="true"  requiredMessage="#{cc.attrs.promocjaMessage}"/>
      </h:panelGrid>
      <h:commandButton action="#{cc.attrs.dodajButtonAction}"
                       value="#{cc.attrs.dodajButtonText}"/>
    </h:form>
  </composite:implementation>
</html>
```

Dodaj_produkt2.xhtml

```
<?xml version='1.0' encoding='UTF-8' ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
     xmlns:ui="http://xmlns.jcp.org/jsf/facelets"
     xmlns:h="http://xmlns.jcp.org/jsf/html"
    xmlns:komponenty_zlozone="http://xmlns.jcp.org/jsf/composite/
                                                komponenty_zlozone">
   <body>
     <ui:composition template="./../template.xhtml">
        <ui:define name="title">    Dodaj produkt      </ui:define>
        <ui:define name="content">
          <komponenty_zlozone:Dodawanie_produktu
             nameLable="Podaj nazwe produktu"
             nameValue="#{managed_produkt.nazwa}"
            titleName="Podaj nazwe:"
            nameMessage="Blad: Podaj nazwe."
```

Dodaj_produkt2.xhtml

```
            cenaLable="Podaj cene netto produktu"
            cenaValue="#{managed_produkt.cena}"
            titleCena="Podaj cene:"
            cenaMessage="Blad: Podaj cene."
            promocjaLable="Podaj promocje produktu"
            promocjaValue="#{managed_produkt.promocja}"
            titlePromocja="Podaj promocje:"
            promocjaMessage="Blad: Podaj promocje."
            dodajButtonText="OK"
            dodajButtonAction="#{managed_produkt.dodaj_produkt}"
        />
    </ui:define>
  </ui:composition>
 </body>
</html>
```

Wyswietlanie_produktu.xhtml

```
<?xml version='1.0' encoding='UTF-8' ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://xmlns.jcp.org/jsf/html"
      xmlns:composite="http://xmlns.jcp.org/jsf/composite">
  <composite:interface>
    <composite:attribute name="nameLable" />
    <composite:attribute name="nameValue" />
    <composite:attribute name="cenaLable" />
    <composite:attribute name="cenaValue" />
    <composite:attribute name="promocjaLable" />
    <composite:attribute name="promocjaValue" />
    <composite:attribute name="cena_bruttoLable" />
    <composite:attribute name="cena_bruttoValue" />
    <composite:attribute name="pokazButtonText" />
    <composite:attribute name="pokazButtonAction"   method-signature="java.lang.String  action()"/>
    <composite:attribute name="brak"/>
    <composite:attribute name="brakdanych"/>
    <composite:attribute name="standanych1"/>
    <composite:attribute name="standanych2"/>
  </composite:interface>
```

Wyswietlanie_produktu.xhtml

```
<composite:implementation >
    <h:form>
      <h:outputText escape="#{cc.attrs.brak}" value="#{cc.attrs.brakdanych}"
                    rendered="#{cc.attrs.standanych1}"/>
      <h:panelGrid columns="2" rendered="#{cc.attrs.standanych2}">
        #{cc.attrs.nameLable} :
        <h:outputText id="name" value="#{cc.attrs.nameValue}" />
        #{cc.attrs.cenaLable} :
        <h:outputText id="cena" value="#{cc.attrs.cenaValue}" />
        #{cc.attrs.promocjaLable} :
        <h:outputText id="promocja" value="#{cc.attrs.promocjaValue}" />
        #{cc.attrs.cena_bruttoLable} :
        <h:outputText id="cena_brutto" value="#{cc.attrs.cena_bruttoValue}" />
      </h:panelGrid>
      <h:commandButton action="#{cc.attrs.pokazButtonAction}"
                       value="#{cc.attrs.pokazButtonText}"/>
    </h:form>
  </composite:implementation>
</html>
```

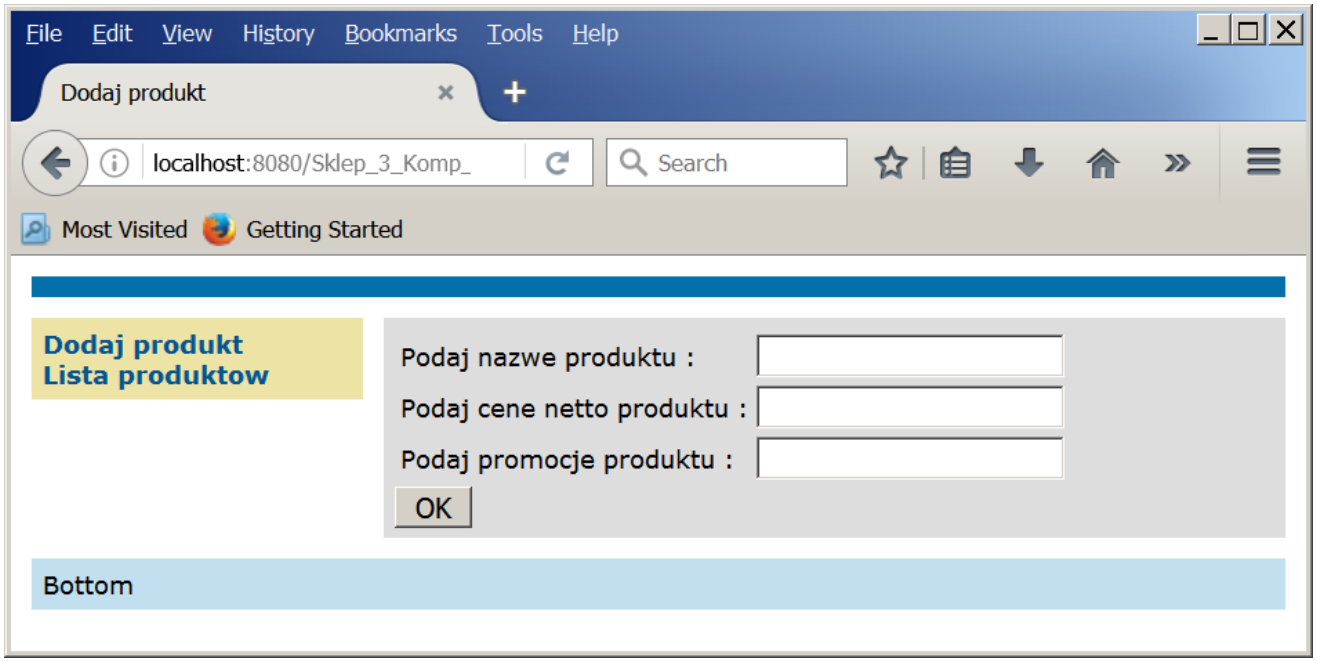
Rezultat2.xhtml

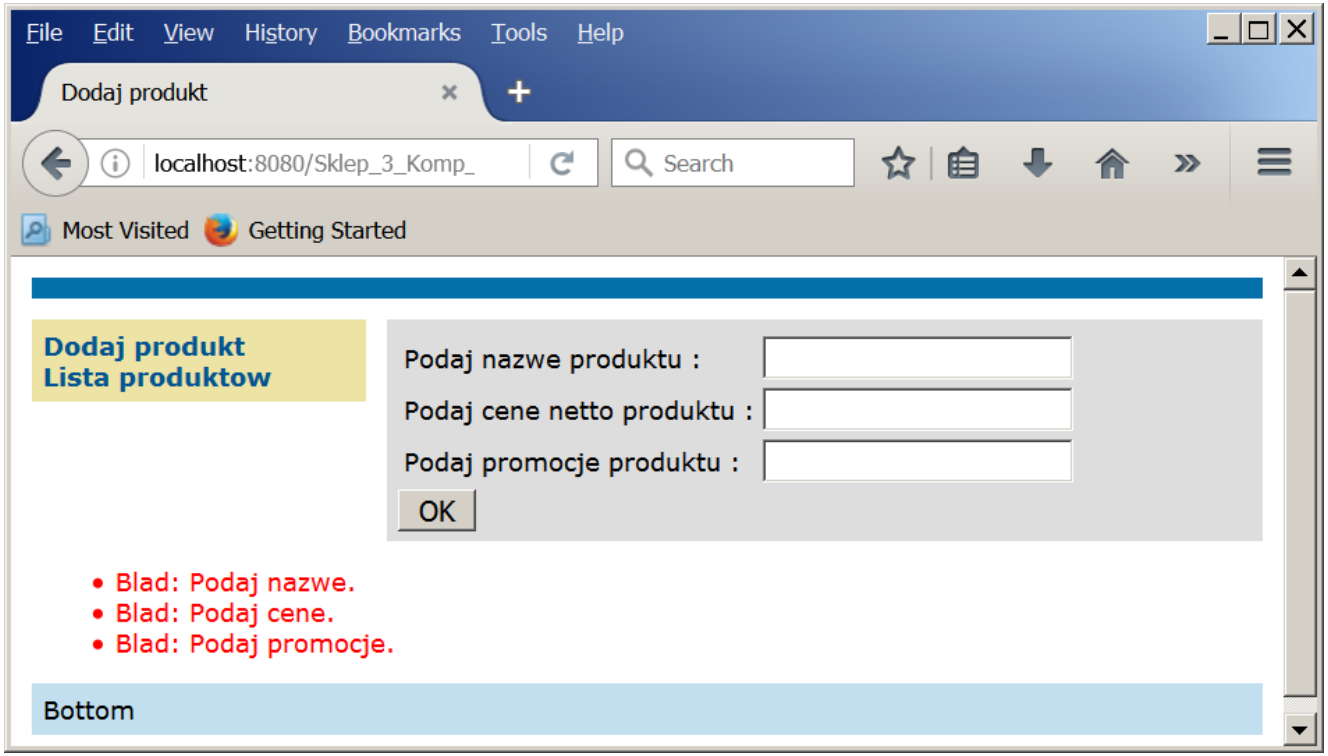
```
<?xml version='1.0' encoding='UTF-8' ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:ui="http://xmlns.jcp.org/jsf/facelets"
      xmlns:komponenty_zlozone="http://xmlns.jcp.org/jsf/
                                composite/komponenty_zlozone">
  <body>
    <ui:composition template="./../template.xhtml">
      <ui:define name="title">  Rezultat  </ui:define>
      <ui:define name="content">
        <komponenty_zlozone:Wyswietlanie_produktu
          brak="false"
          brakdanych="Taki produkt juz istnieje"
          standanych1="#{managed_produkt.stan==0}"
          standanych2="#{managed_produkt.stan!=0}"
```

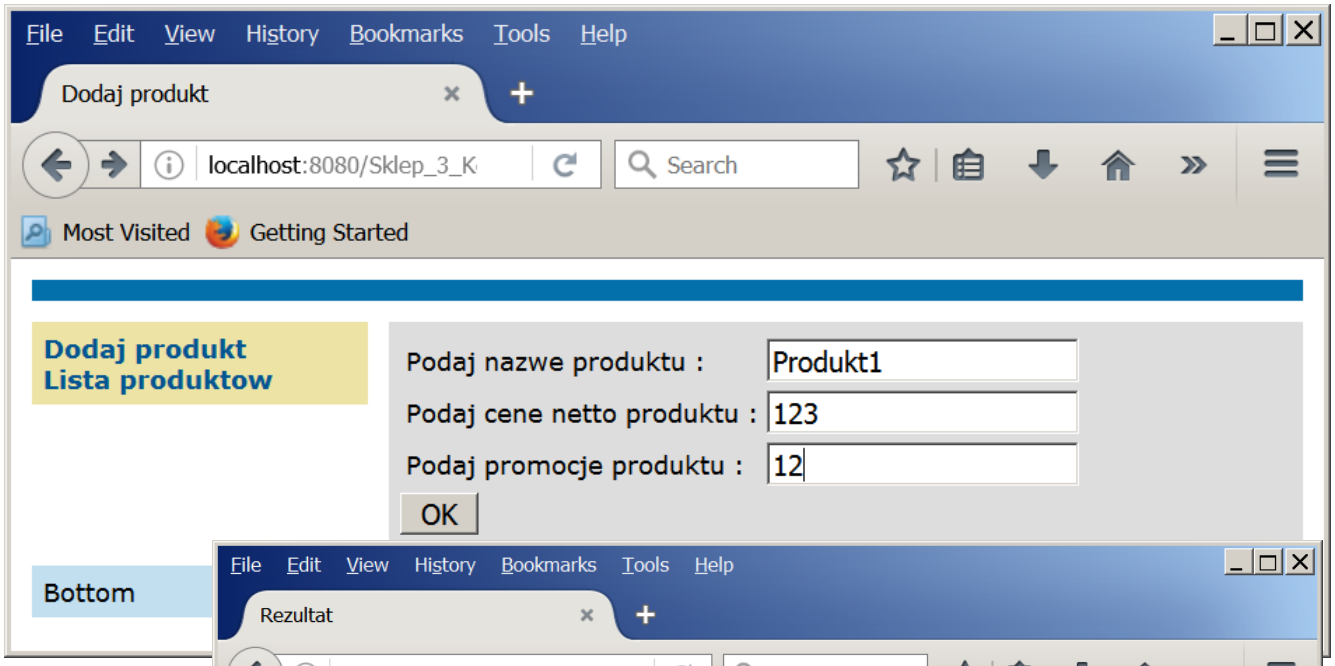
Rezultat2.xhtml

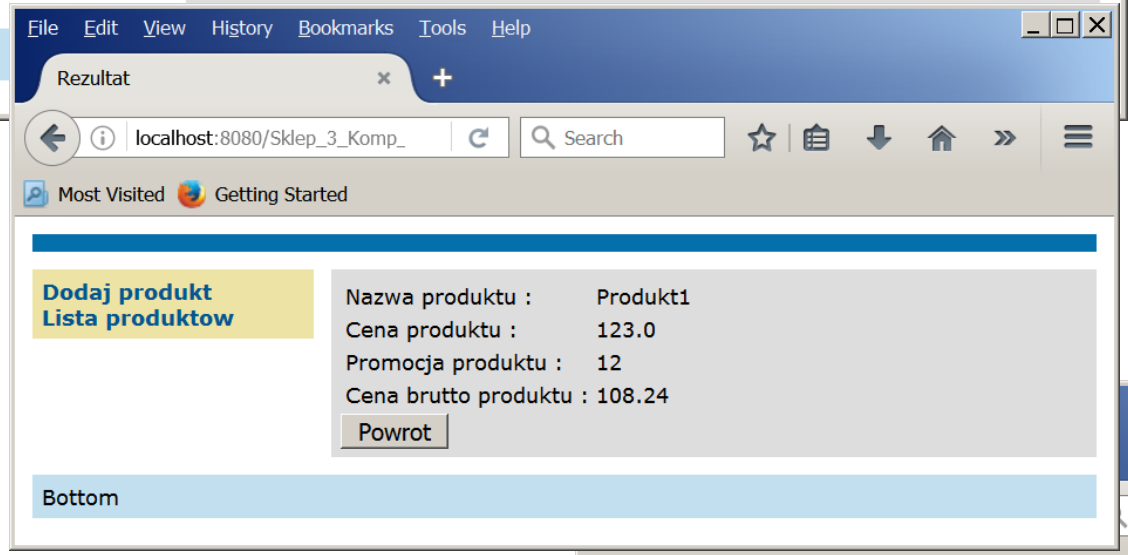
```
nameLable="Nazwa produktu"
nameValue="#{managed_produkt.nazwa}"
cenaLable="Cena produktu"
cenaValue="#{managed_produkt.cena}"
promocjaLable="Promocja produktu"
promocjaValue="#{managed_produkt.promocja}"
cena_bruttoLable="Cena brutto produktu"
cena_bruttoValue="#{managed_produkt.cena_brutto}"
pokazButtonText="Powrot"
pokazButtonAction="#{managed_produkt.powrot}"
/>
</ui:define>
</ui:composition>
</body>
</html>
```

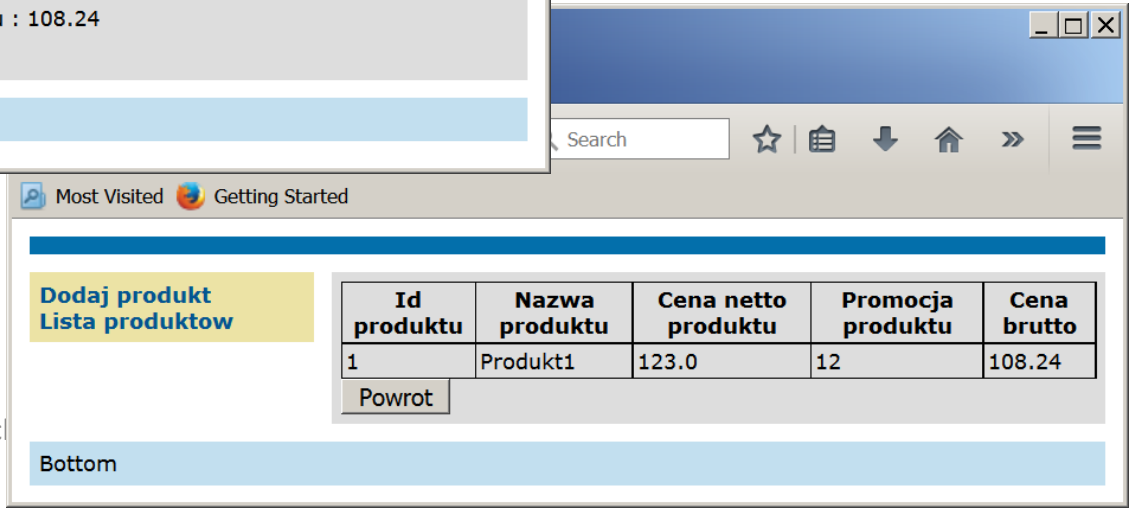
```
public String powrot()
  {
     return "/faces/index1";
  }
```

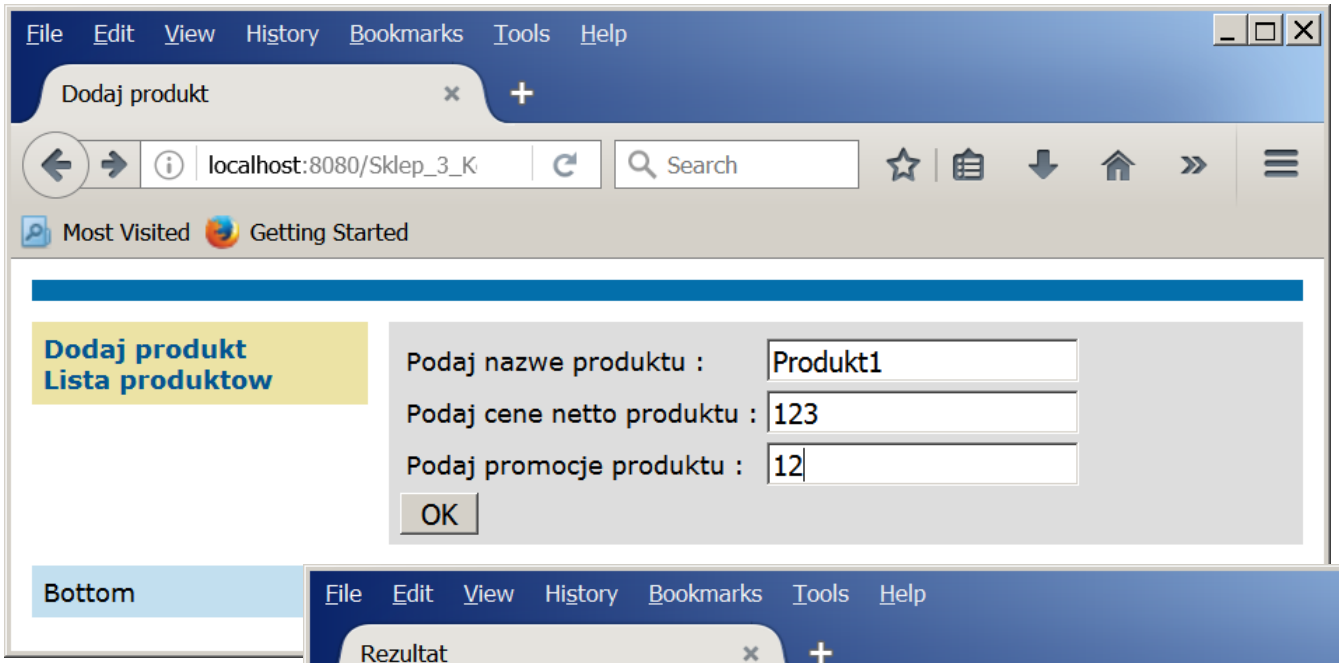

File Edit View History Bookmarks Tools Help
Dodaj produkt
localhost:8080/Sklep_3_Komp_
Search
Most Visited
Getting Started
Dodaj produkt
Lista produktow
Podaj nazwe produktu :
Podaj cene netto produktu :
Podaj promocje produktu :
OK
Bottom

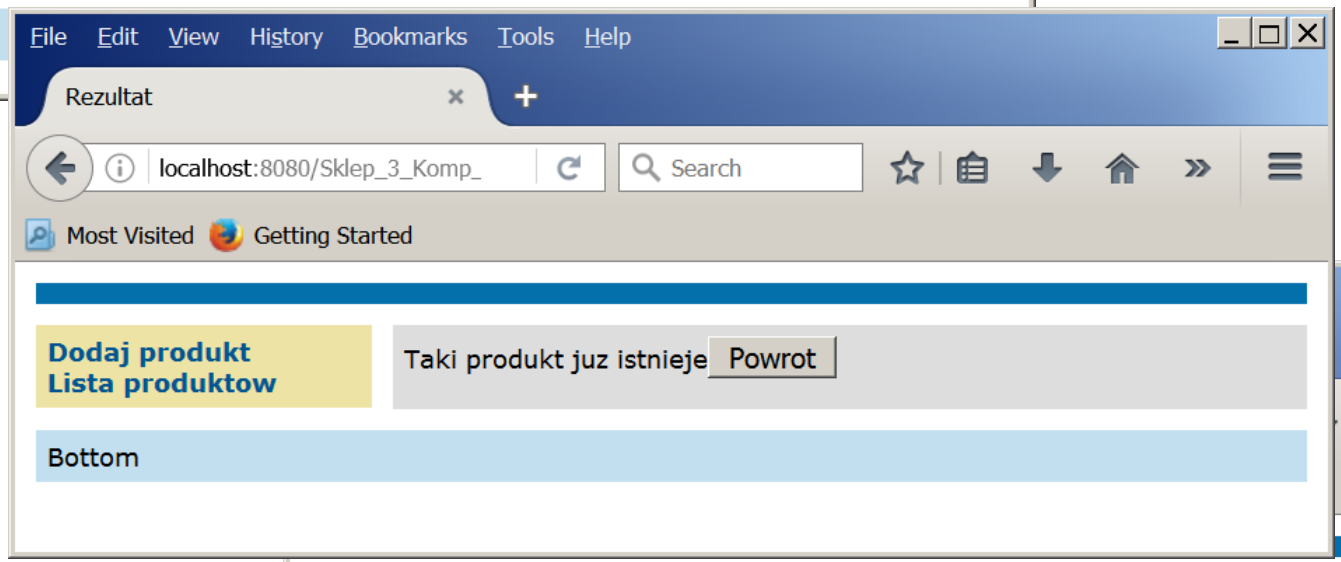

File Edit View History Bookmarks Tools Help
Dodaj produkt
localhost:8080/Sklep_3_Komp_
Search
Most Visited
Getting Started
Dodaj produkt
Lista produktow
Podaj nazwe produktu :
Podaj cene netto produktu :
Podaj promocje produktu :
OK
• Blad: Podaj nazwe.
• Blad: Podaj cene.
• Blad: Podaj promocje.
Bottom

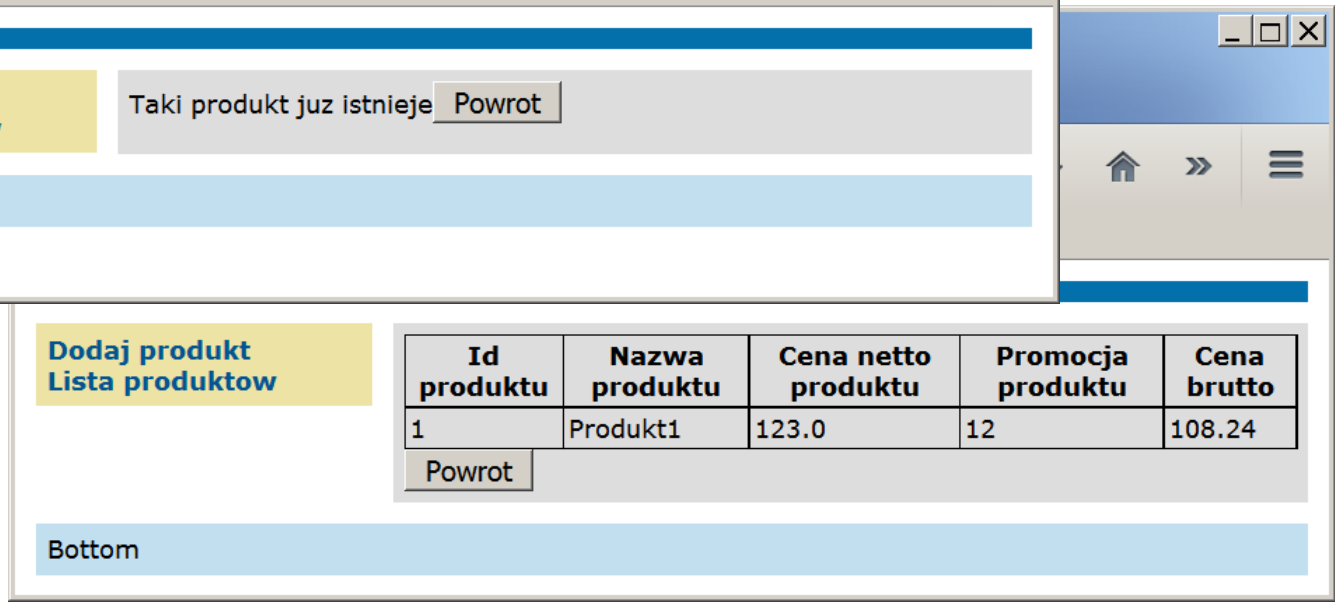

File Edit View History Bookmarks Tools Help
Dodaj produkt
localhost:8080/Sklep_3_K
Search
Most Visited
Getting Started
Dodaj produkt
Lista produktow
Podaj nazwe produktu :
Produkt1
Podaj cene netto produktu :
123
Podaj promocje produktu :
12
OK
Bottom


File Edit View History Bookmarks Tools Help
Rezultat
localhost:8080/Sklep_3_Komp_
Search
Most Visited
Getting Started
Dodaj produkt
Lista produktow
Nazwa produktu : Produkt1
Cena produktu : 123.0
Promocja produktu : 12
Cena brutto produktu : 108.24
Powrot
Bottom


Search
Most Visited
Getting Started
Dodaj produkt
Lista produktow
Id produktu
Nazwa produktu
Cena netto produktu
Promocja produktu
Cena brutto
1
Produkt1
123.0
12
108.24
Powrot
Bottom


File Edit View History Bookmarks Tools Help
Dodaj produkt
localhost:8080/Sklep_3_K
Search
Most Visited
Getting Started
Dodaj produkt
Lista produktow
Podaj nazwe produktu :
Produkt1
Podaj cene netto produktu :
123
Podaj promocje produktu :
12
OK
Bottom


File Edit View History Bookmarks Tools Help
Rezultat
localhost:8080/Sklep_3_Komp_
Search
Most Visited
Getting Started
Dodaj produkt
Lista produktow
Taki produkt juz istnieje
Powrot
Bottom


Dodaj produkt
Lista produktow
Id produktu
Nazwa produktu
Cena netto produktu
Promocja produktu
Cena brutto
1
Produkt1
123.0
12
108.24
Powrot
Bottom