

Aplikacje WWW - laboratorium

JavaServer Faces

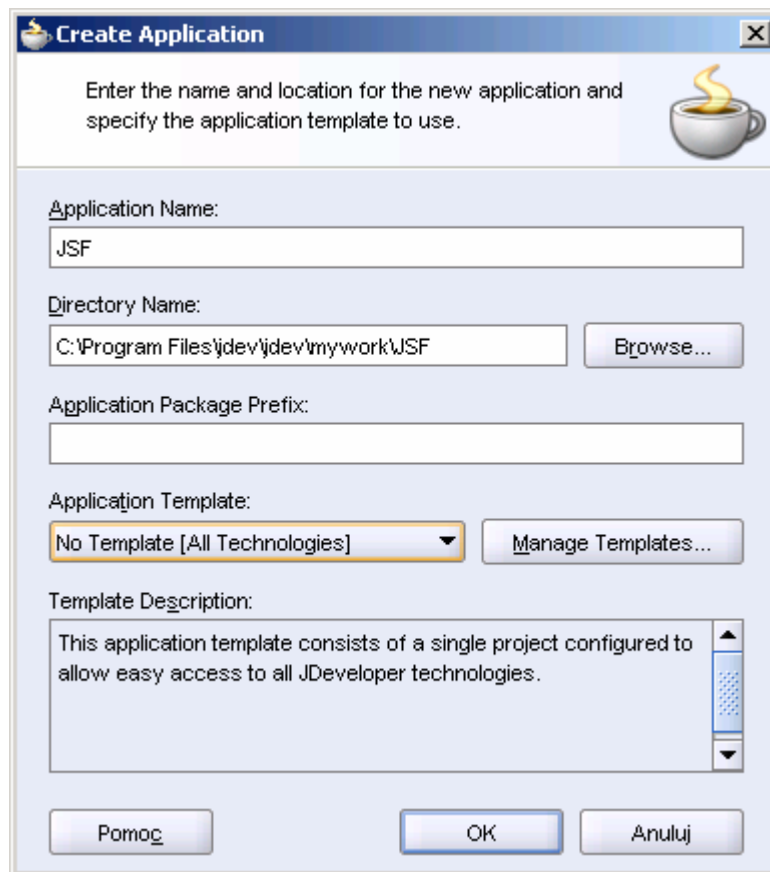
Celem ćwiczenia jest przygotowanie aplikacji internetowej z wykorzystaniem technologii JSF.

Prezentowane ćwiczenia zostały wykonane w środowisku Oracle JDeveloper 10.1.3.0.4 (do pobrania ze strony: <http://www.oracle.com/>).

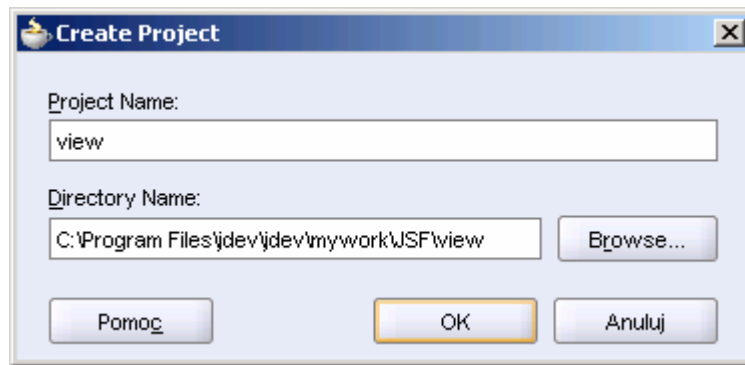
Ćwiczenie 1

W tym ćwiczeniu zostanie stworzony serwlet wyświetlający podstawowe informacje o pracownikach.

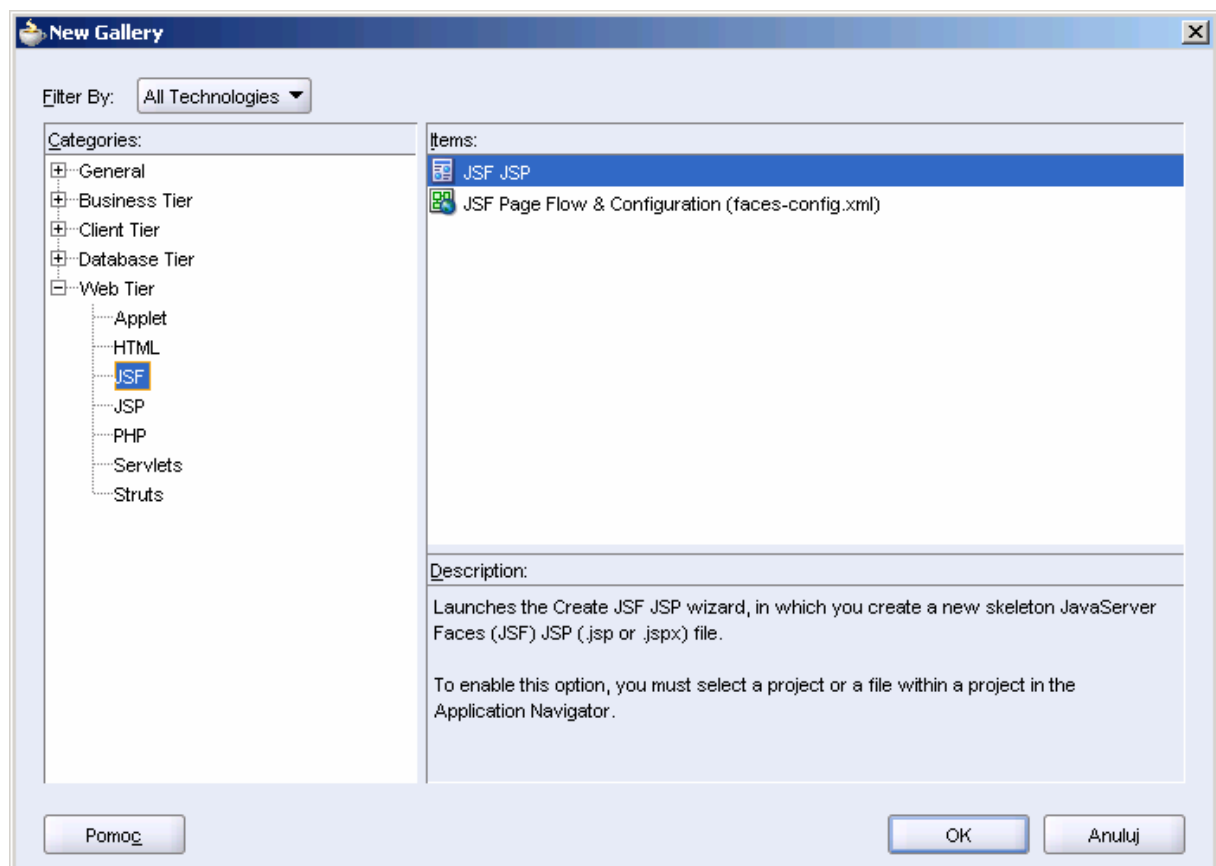
- Należy przygotować środowisko – nową aplikację (patrz opis do ćwiczenia „prosty serwlet”).



- W ramach aplikacji należy stworzyć projekt o nazwie view

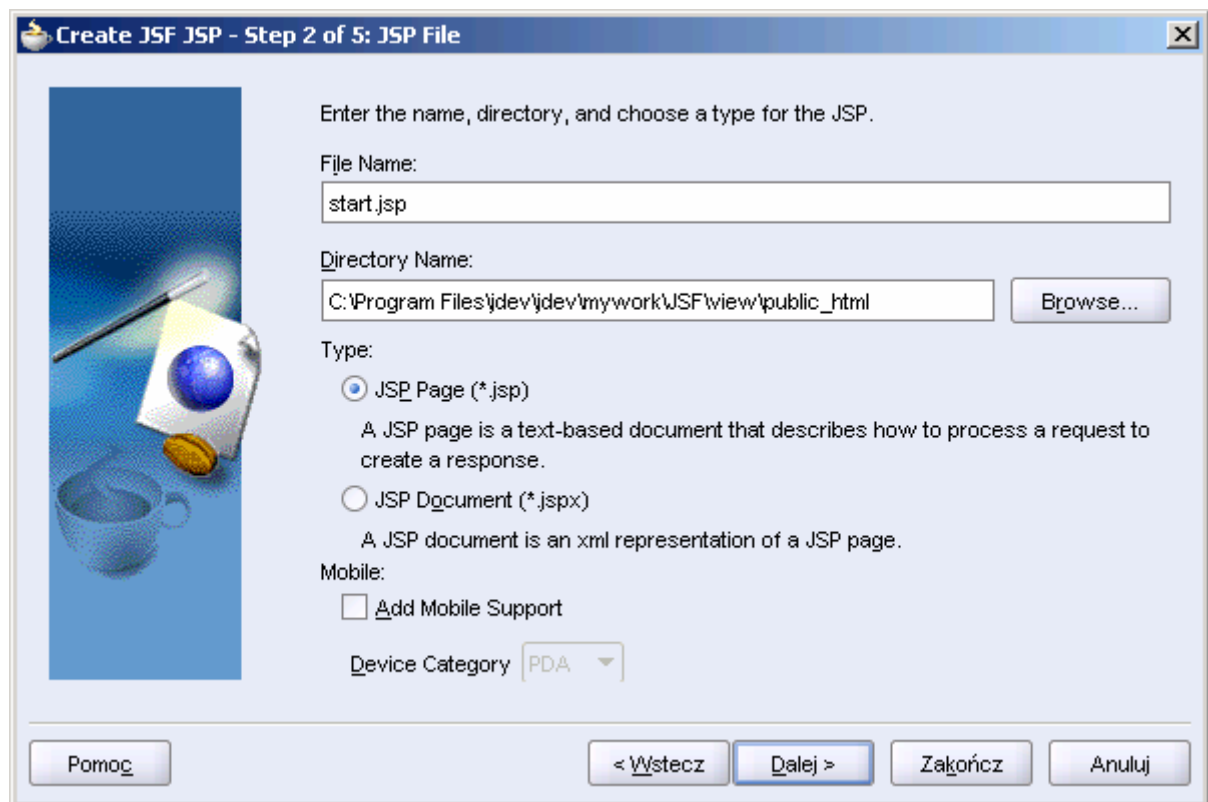


- W ramach utworzonego projektu należy utworzyć stronę JSF (File->New).

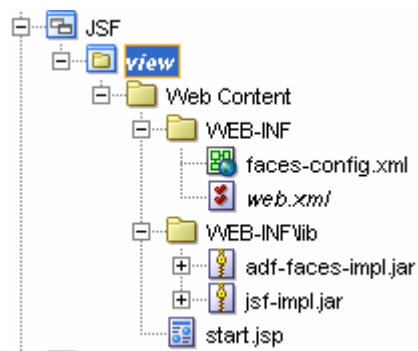


-

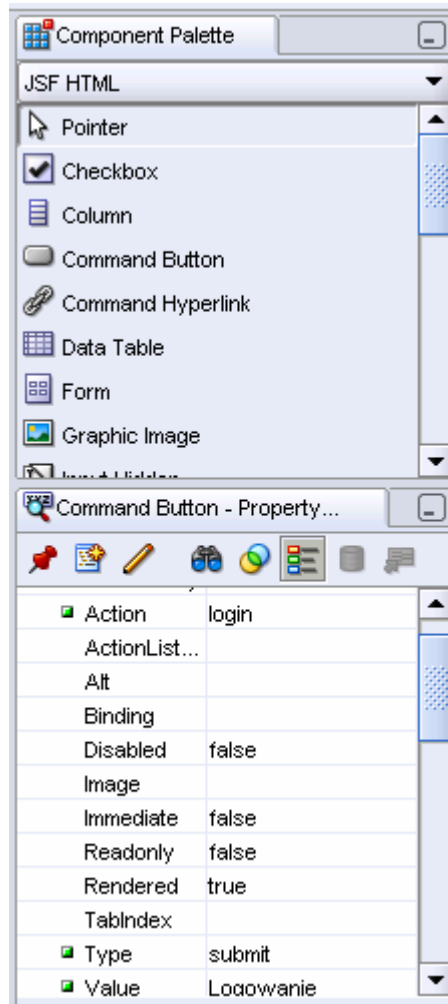
- Pierwsza strona będzie stroną startową, należy więc nazwać ją *start.jsp*.



- W następnych krokach należy zaakceptować proponowane ustawienia domyślne
- Po wygenerowaniu strony należy upewnić się, że została utworzona we właściwym kontekście



- Następnie należy na utworzonej stronie umieścić odpowiednie informacje powitalne oraz przycisk który będzie przenosił na użytkownika do strony logowania. Należy w tym celu wykorzystać element *Command Button* z palety JSF HTML. Dla tego przycisku ustaw następujące właściwości (wartości zmienione są zaznaczone zielonymi punktami):



- Ostatnim punktem tego ćwiczenia jest analiza wygenerowanego kodu.

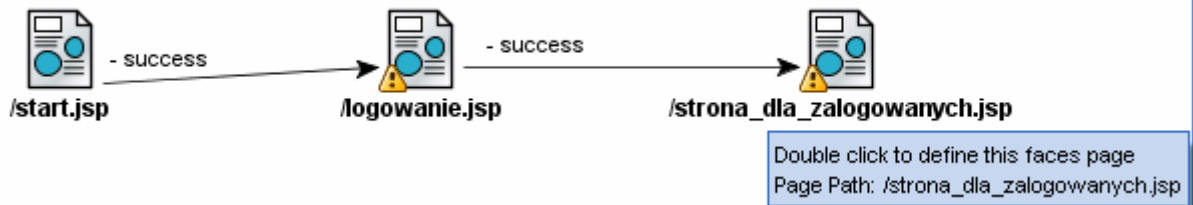
Ćwiczenie 2

Celem tego ćwiczenia jest zdefiniowanie przepływu sterowania pomiędzy stronami tworzonego serwisu internetowego

- Należy rozpocząć od zlokalizowania pliku *faces-config.xml* w gałęzi: Web Content | WEB-INF i następnie otwarcia jego zawartości
- Na otwartej stronie należy umieścić symbol stworzonego poprzednio pliku start.jsp a także wybierając z menu po prawej stronie dołożyć symbol kolejnego pliku. Nowemu plikowi nadać nazwę logowanie.jsp
- Następnie należy dodać nawigację pomiędzy tymi dwoma plikami



- Należy też dodać kolejną stronę o nazwie strona_dla_zalogowanych.jsp

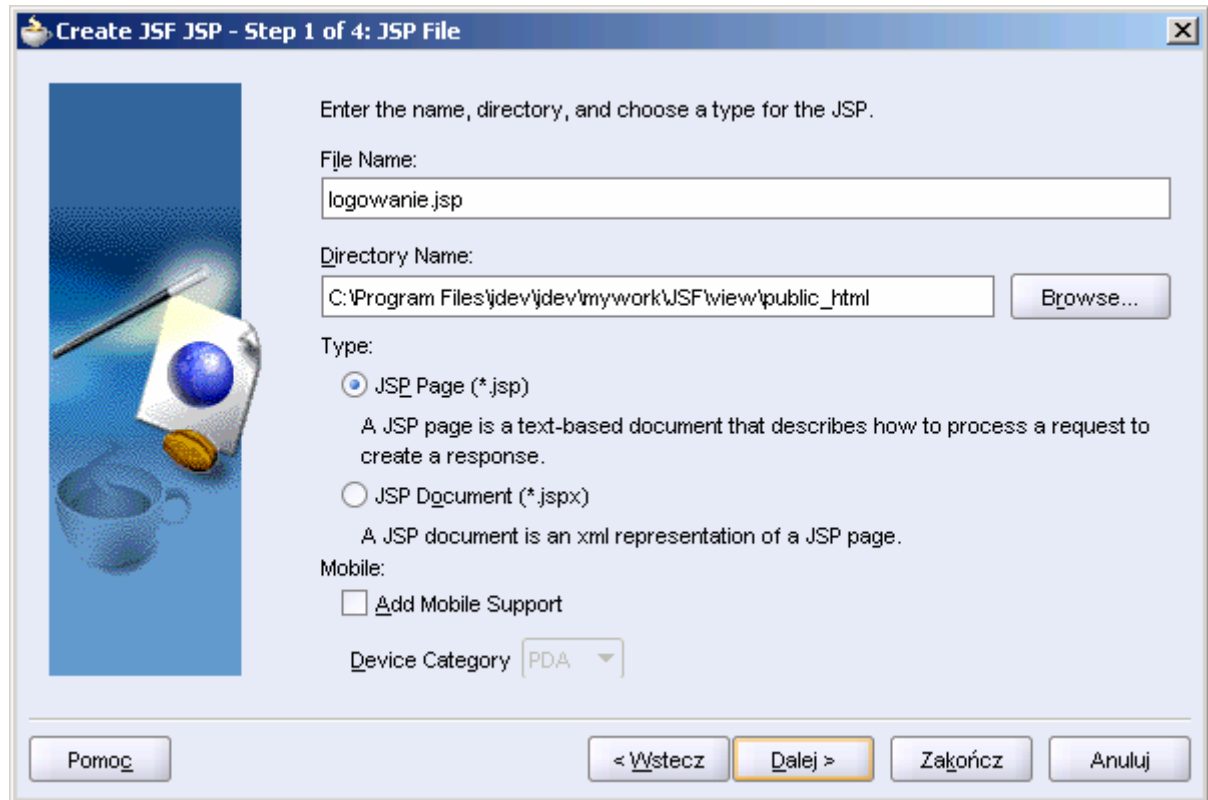


- Następnie należy przełączyć się na podgląd źródła stworzonego dokumentu XML i je przeanalizować

Ćwiczenie 3

W ramach tego ćwiczenia zostanie zaimplementowana strona logowania do serwisu.

- Należy dwukrotnie kliknąć ikonę pliku logowanie.jsp
- W uruchomionym kreatorze należy wybrać następujące opcje opisujące podstawowe parametry tworzonej strony



Enter the name, directory, and choose a type for the JSP.

File Name:
logowanie.jsp

Directory Name:
C:\Program Files\jdev\jdev\mywork\JSF\view\public_html Browse...

Type:
 JSP Page (*.jsp)
A JSP page is a text-based document that describes how to process a request to create a response.
 JSP Document (*.jspx)
A JSP document is an xml representation of a JSP page.

Mobile:
 Add Mobile Support

Device Category PDA

Pomoc < Wstecz Dalej > Zakończ Anuluj

- Następnie należy zdefiniować parametry serwletu, który będzie obsługiwał formularz logowania

Create JSF JSP - Step 2 of 4: Component Binding

If you need to programmatically modify this page's UI components you can choose to automatically expose them in a managed bean. This creates bean properties representing UI components added to the page, and binds each UI component to its corresponding property. If you choose not to automatically expose all UI components in code, you can expose individual components directly from the page later.

Do Not Automatically Expose UI Components in a Managed Bean

Automatically Expose UI Components in a New Managed Bean

Name:

Class:

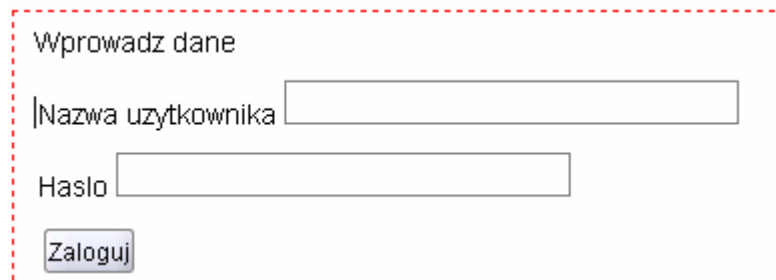
Package:

Automatically Expose UI Components in an Existing Managed Bean

Managed Bean:

- W kolejnych krokach można akceptować wartości podpowiadane przez kreatora

- Na wygenerowanej stronie należy stworzyć formularz logowania.
- Korzystając z komponentów *Output Label* i *Input Text* należy stworzyć odpowiednie rubryki i miejsca na wprowadzanie danych (pola mają być wymagane, należy zaznaczyć to w ich właściwościach)
- Z palety JSF CORE wybrać komponent *Validate Length* i powiązać go z polem przechowującym hasło użytkownika – minimalna długość hasła to 4 znaki
- Następnie należy dołożyć przycisk *Command Button*, który będzie miał ustawione następujące właściwości: `action = strona_dla_zalogowanych`, `value = Zaloguj`



Wprowadz dane

Nazwa użytkownika

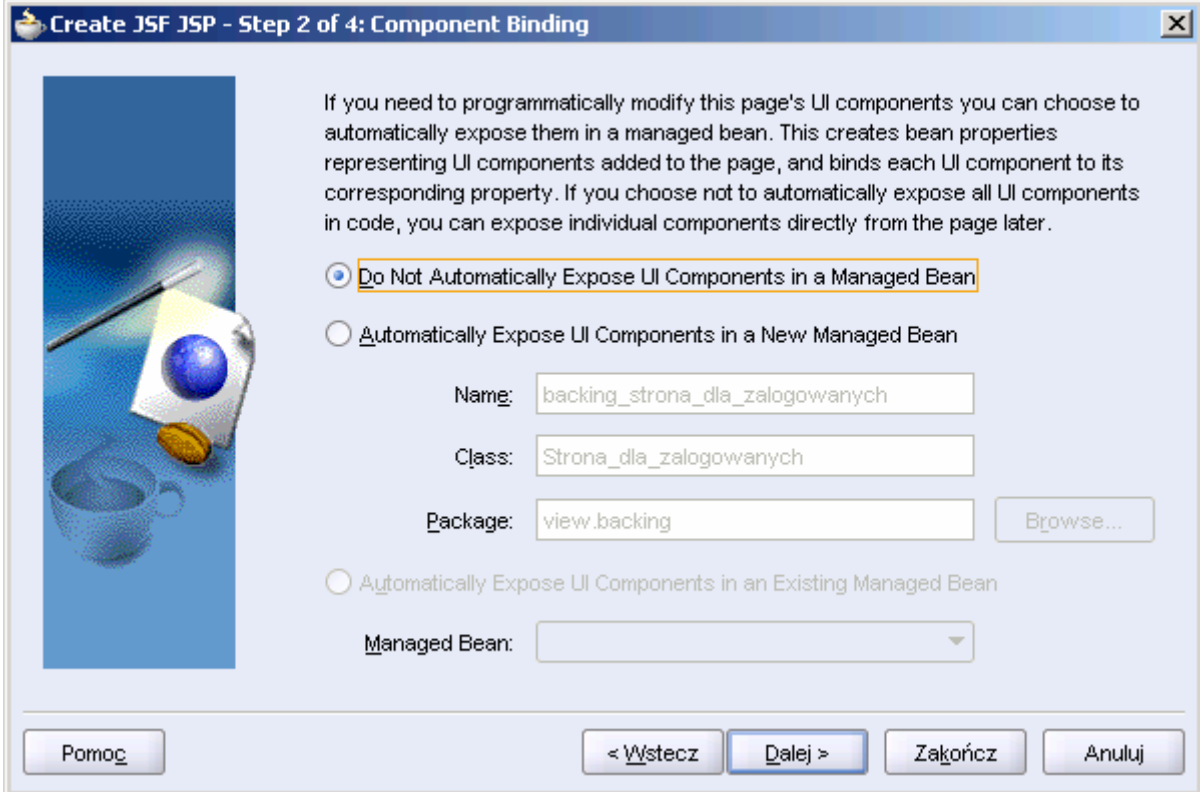
Hasło

Zaloguj

Ćwiczenie 4

Celem tego ćwiczenia jest przygotowanie strony wyświetlającej informacje powitalne dla zalogowanego użytkownika

- Na początku tego ćwiczenia należy na diagramie sterowania kliknąć dwukrotnie na ikonie strony dla zalogowanego użytkownika i korzystając z uruchomionego kreatora stworzyć taką stronę.



Create JSF JSP - Step 2 of 4: Component Binding

If you need to programmatically modify this page's UI components you can choose to automatically expose them in a managed bean. This creates bean properties representing UI components added to the page, and binds each UI component to its corresponding property. If you choose not to automatically expose all UI components in code, you can expose individual components directly from the page later.

Do Not Automatically Expose UI Components in a Managed Bean

Automatically Expose UI Components in a New Managed Bean

Name:

Class:

Package:

Automatically Expose UI Components in an Existing Managed Bean

Managed Bean:

- Umieść na stronie element Output Text. W polu value skasuj istniejące dane i korzystając z kreatora bindowania danych wprowadź następującą wartość:

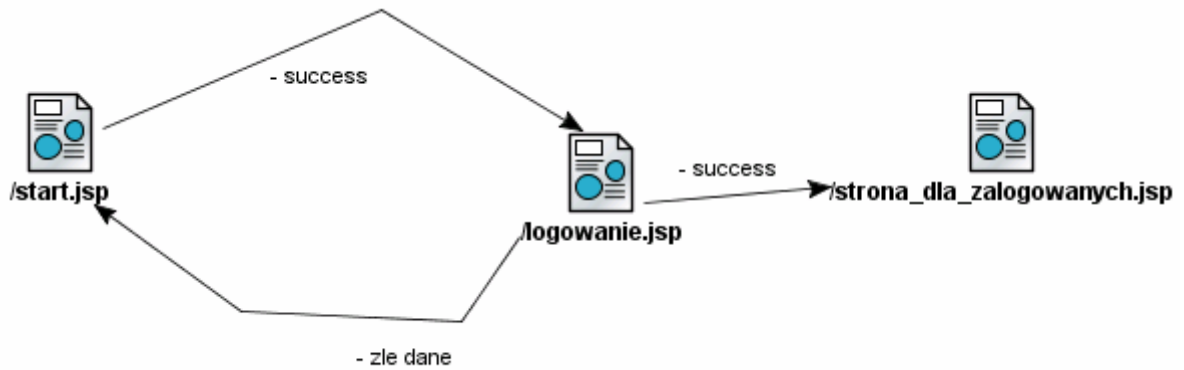
`#{backing_logowanie.inputText1.value}`

- Aplikacja jest gotowa do uruchomienia

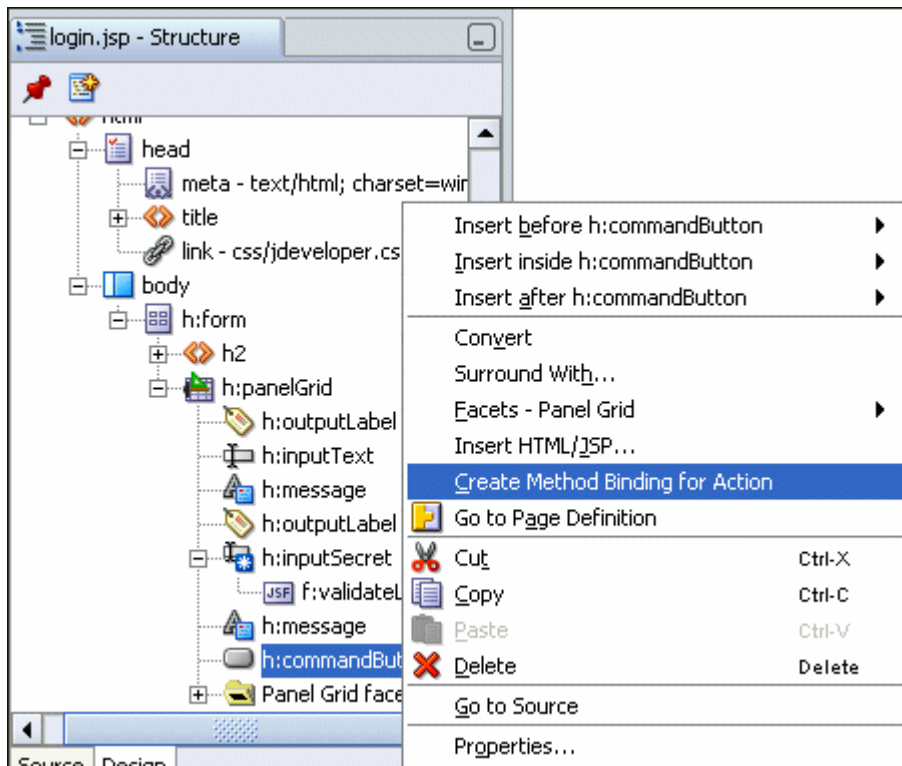
Ćwiczenie 5

Celem tego ćwiczenia jest zaprojektowanie akcji wykonywanej w sytuacji, kiedy użytkownik wprowadzi błędne dane do formularza logowania.

- Wykonywanie ćwiczeni należy rozpocząć od zdefiniowania kolejnego przepływu sterowania na diagramie JSF. Tym przepływem będzie przepływ między stroną z formularzem logowania a stroną startową.



- Na stronie z formularzem logowania należy zmienić akcję dla przycisku na *DEFAULT*
- W znajdującym się po prawej stronie oknie struktury dokumentu należy wybrać z menu kontekstowego dla przycisku tworzenie bindowania metody (*Create Method Binding for Action*)



- Kod metody powinien wyglądać następująco

```
public String commandButton1_action()
{
    // Add event code here...
    if (inputSecret1.getValue().toString().equals("welcome"))
    {
        return "success";
    }
    else
    {
        String message = "Invalid login";
        FacesContext.getCurrentInstance().addMessage(null,
            new FacesMessage(message));
        return "zle dane";
    }
}
```

- Następnie można uruchomić i przetestować aplikację