

Zaawansowane aplikacje internetowe

EJB 1

Rozróżniamy dwa rodzaje *beanów* sesyjnych:

- Stateless
- Statefull

Celem tego laboratorium jest zbadanie różnic funkcjonalnych tych dwóch rodzajów *beanów*.

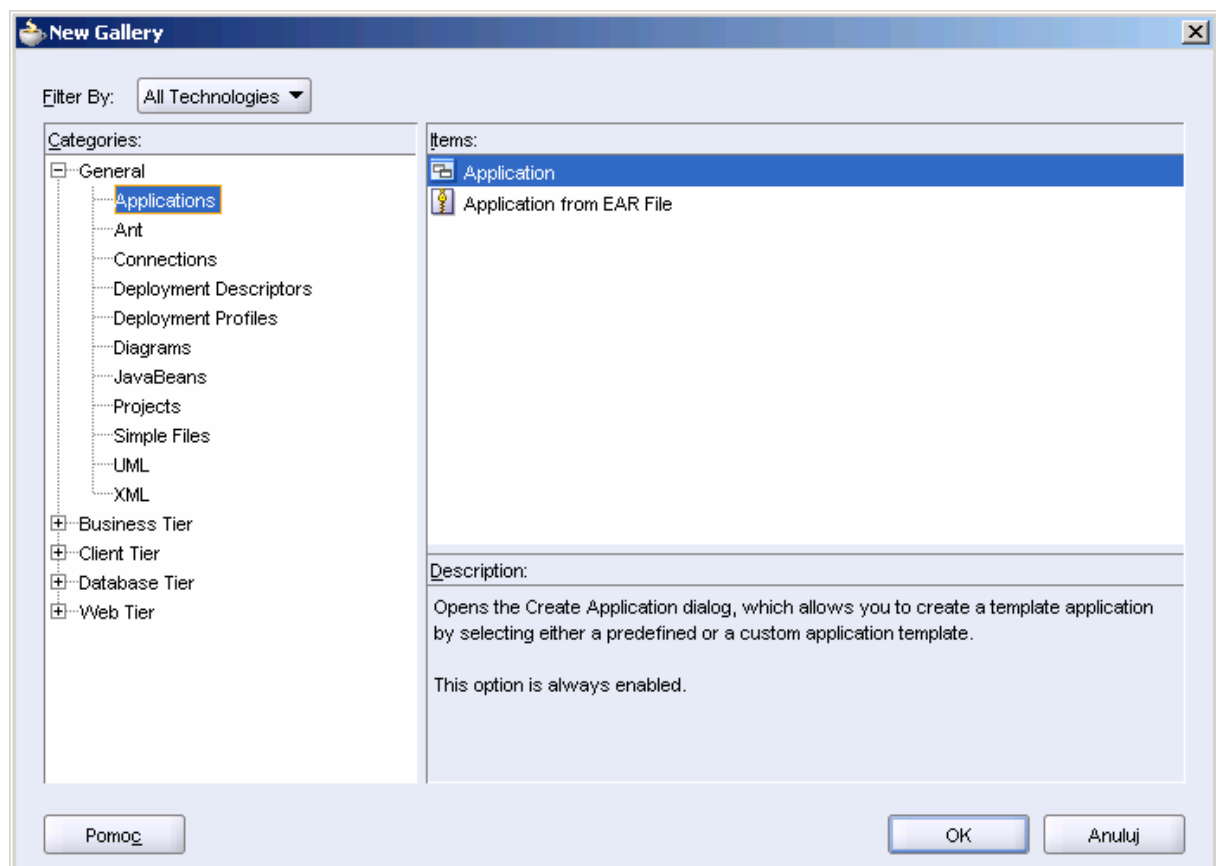
Poszczególne ćwiczenia prezentują sposób przygotowania środowiska, generowania i uruchamiania prostego *beana* oraz przykładowej aplikacji potrafiącej uruchamiać takiego *beana*.

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

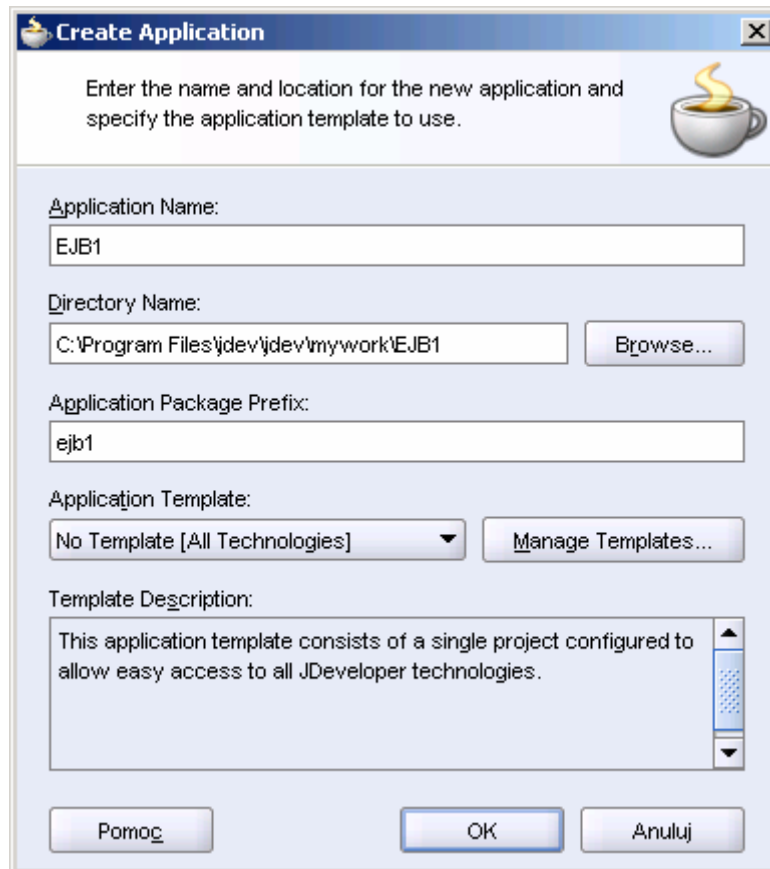
Ćwiczenie 1

W tym ćwiczeniu zostanie przygotowane środowisko pozwalające na wykonanie ćwiczeń – w repozytorium programu JDeveloper zostanie stworzona odpowiednia aplikacja i projekt.

- Po uruchomieniu programu Oracle JDeveloper należy utworzyć nową aplikację. W tym celu należy wybrać opcję: `File -> New`
- W wyświetlonym oknie należy zaznaczyć chęć utworzenia nowej aplikacji (wygląd okna może się różnić od poniższego):



- Następny ekran pozwala na ustawienie właściwości tworzonej aplikacji. Należy wprowadzić: nazwę aplikacji, nazwę katalogu, w którym znajdą się jej pliki, prefiks dla pakietów tworzonych w ramach tej aplikacji i wybrać aplikację bez wzorca



Create Application

Enter the name and location for the new application and specify the application template to use.

Application Name:
EJB1

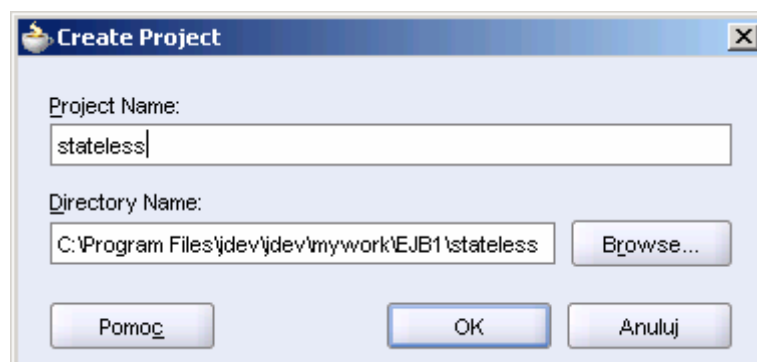
Directory Name:
C:\Program Files\jdev\jdev\mywork\EJB1

Application Package Prefix:
ejb1

Application Template:
No Template [All Technologies]

Template Description:
This application template consists of a single project configured to allow easy access to all JDeveloper technologies.

- Kolejnym krokiem jest stworzenie projektu w ramach aplikacji – należy wprowadzić nazwę projektu: stateless



Create Project

Project Name:
stateless

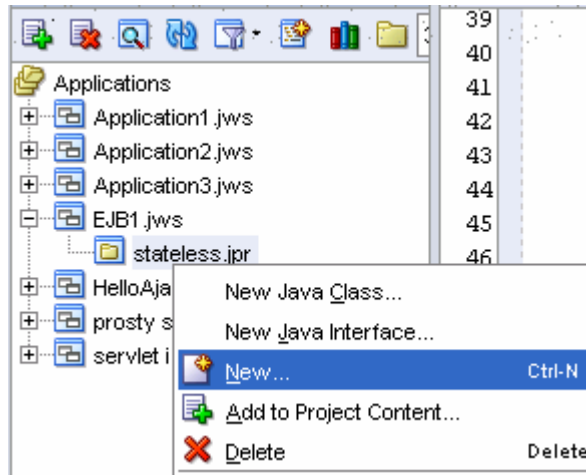
Directory Name:
C:\Program Files\jdev\jdev\mywork\EJB1\stateless

Ćwiczenie 2

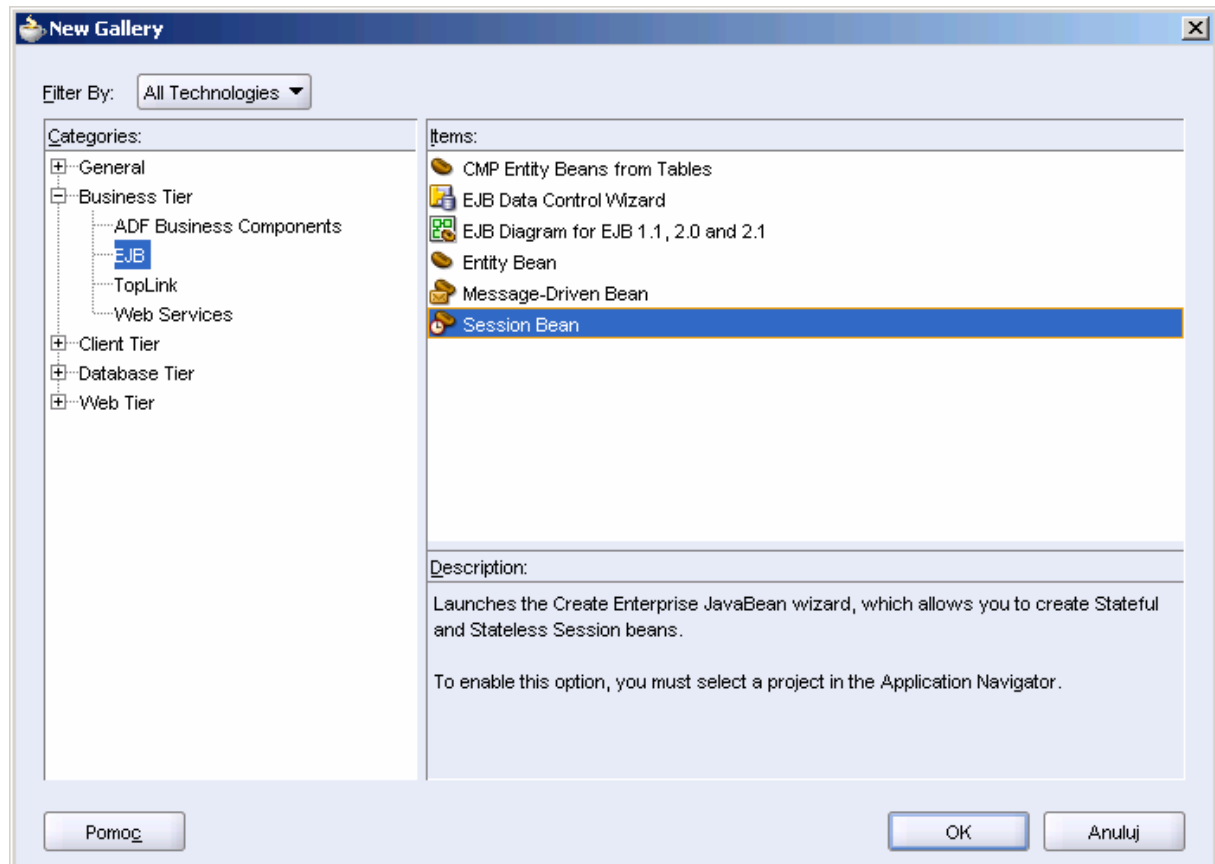
Ćwiczenie będzie polegało na wygenerowaniu szkieletu *beana* sesyjnego typu *stateless* i analizie jego kodu.

W tym celu należy:

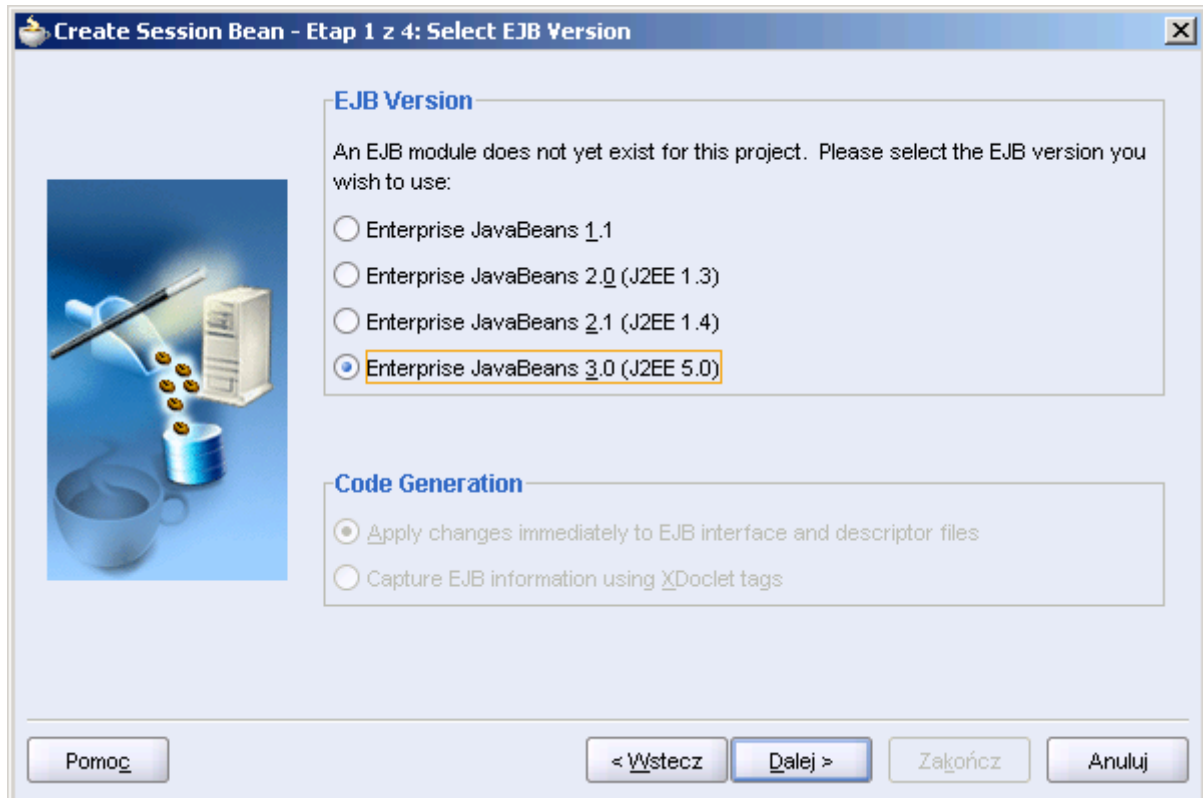
- W widocznym nawigatoryze aplikacji należy zaznaczyć utworzony w poprzednim ćwiczeniu projekt



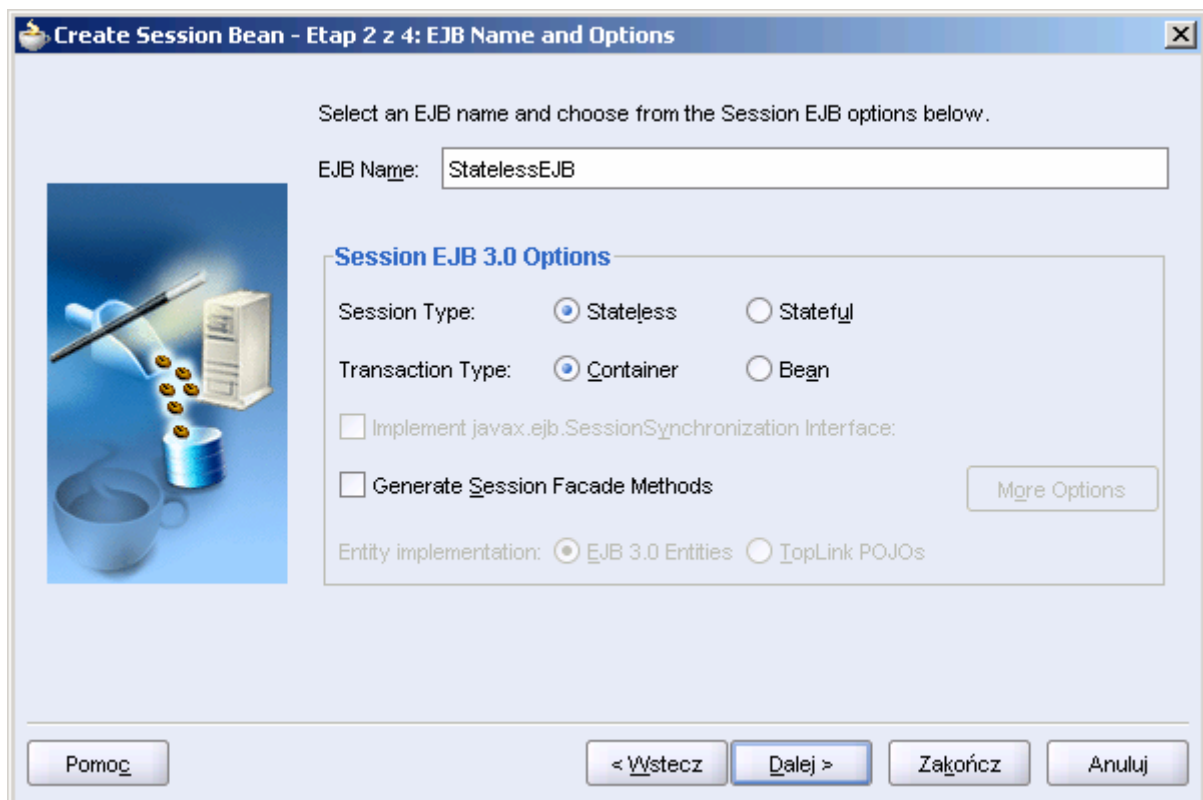
- Następnie należy wybrać opcję: File -> New, lub z menu kontekstowego wybrać opcję New
- W wyświetlonym oknie należy wybrać chęć utworzenia *beana* (sesyjnego)



- W wyświetlonym kreatorze ustawić odpowiednie parametry (wersja specyfikacji: EJB 3.0)



- Należy określić nazwę i typ dla tworzonego *beana*



- Następnie należy określić nazwę klasy beana

Create Session Bean - Etap 3 z 4: Class Definitions

Select the Bean class name and a source root directory for any newly created classes.

Bean Class:
ejb1.StatelessEJBBean Browse...

Source Directory:
C:\Program Files\jdev\jdev\mywork\EJB1\stateless\src Browse...

Pomoc < Wstecz Dalej > Zakończ Anuluj

- W ostatnim kroku należy określić typy i nazwy implementowanych interfejsów (dla potrzeb tego ćwiczenia interfejs zdalny jest zbędny, ale można go wyspecyfikować)

Create Session Bean - Etap 4 z 4: EJB Component Interfaces

Select the EJB interface names.

Implement a Remote Interface

Remote Interface:
ejb1.StatelessEJB Browse...

Implement a Local Interface

Local Interface:
ejb1.StatelessEJBLocal Browse...

Pomoc < Wstecz Dalej > Zakończ Anuluj

- Należy przeanalizować wygenerowany kod i zwróć uwagę na wykorzystane adnotacje

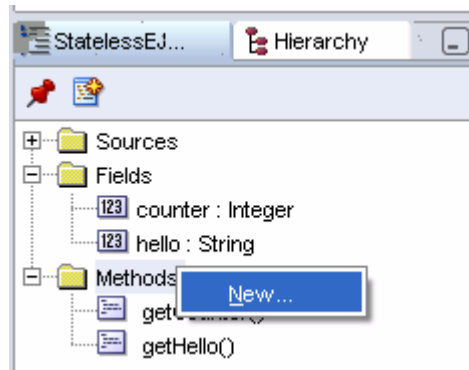
- W oknie struktury (View -> Structure) należy zdefiniować odpowiednią zmienną:
`Integer counter;`

- A następnie zdefiniować dwie metody o następującej specyfikacji:

```
public String getHello() - zwraca tekst:
```

```
"Hello from Stateless Java Bean "
```

```
public Integer getCounter() - zwraca wartość: counter++
```



```
package stateless;

import javax.ejb.Stateless;

@Stateless(name="StatelessEJB")
public class StatelessEJBBean implements StatelessEJB,
StatelessEJBLocal {

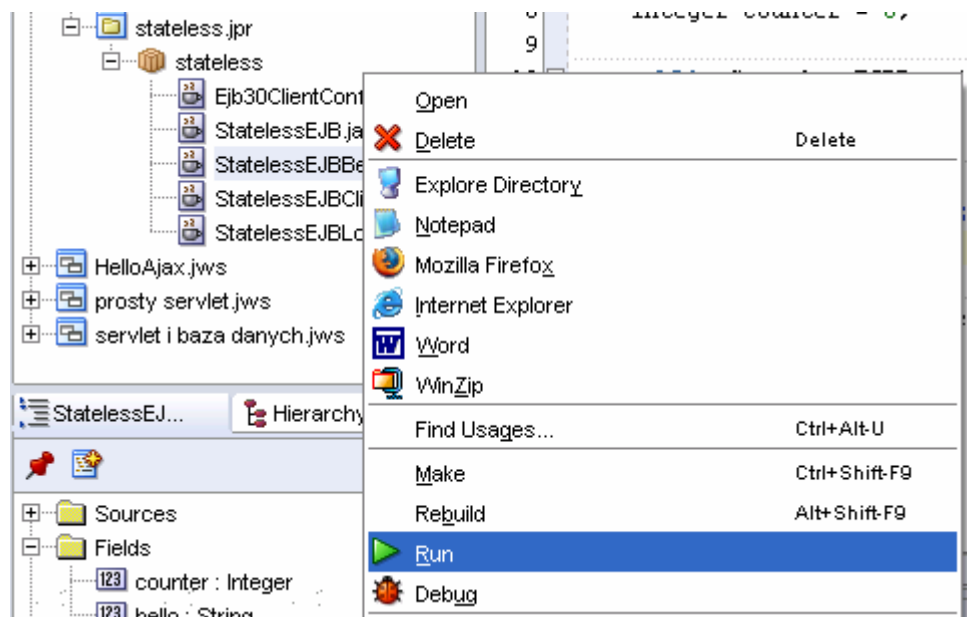
    Integer counter = 0;

    public StatelessEJBBean() {
    }

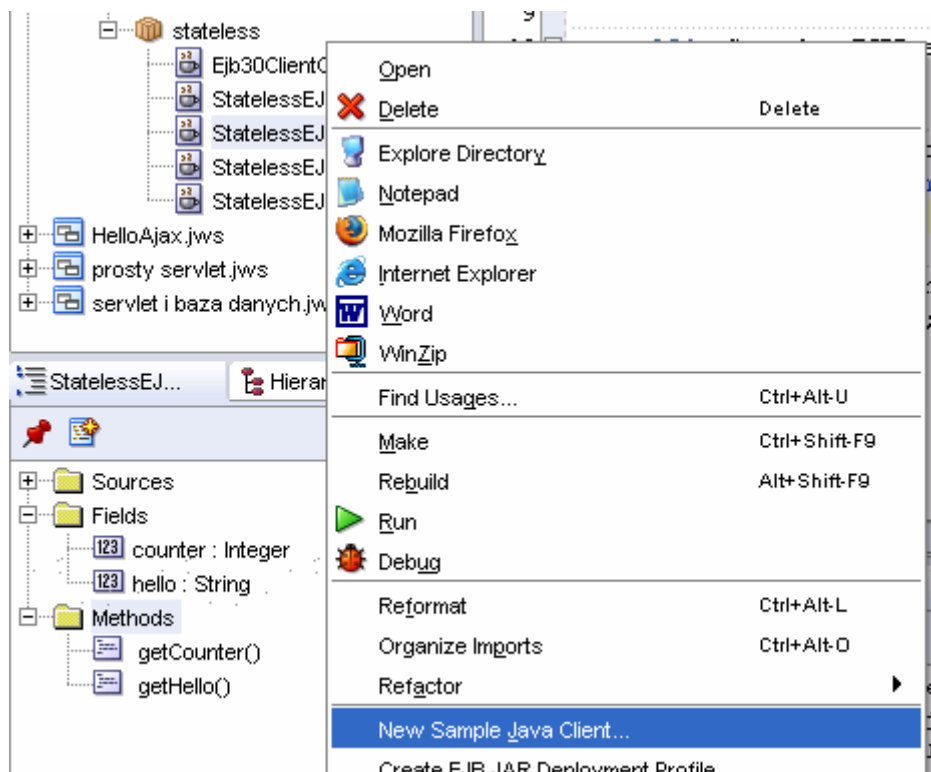
    public String getHello() {
        return "Hello from Stateless Java Bean";
    }

    public Integer getCounter() {
        return counter ++;
    }
}
```

- Po wykonaniu poprzednich czynności należy uruchomić utworzonego *beana* wewnątrz wbudowanego serwera aplikacji (OC4J) – opcja: Run



- Oraz przeanalizować wyświetlane komunikaty
- Następnie należy stworzyć przykładową aplikację korzystającą z metod udostępnianych przez *beana*



- Analiza wygenerowanego kod powinna pomóc zrozumieć sposób wykorzystywania komponentów EJB przez aplikacje (w tym wypadku została wygenerowana aplikacja konsolowa, ale sposób dostępu i wywoływania metod *beana* jest identyczny zarówno w aplikacjach typu servlet, czy też *beany* wywołujące się na wzajem). Należy wskazać w wygenerowanym kodzie fragmenty odpowiedzialne za utworzenie instancji beana, oraz wywołania jego metod.

- W przykładowej aplikacji należy kilkakrotnie powtórzyć linię kodu odpowiedzialną za wywołanie metody zwracającej stan licznika.

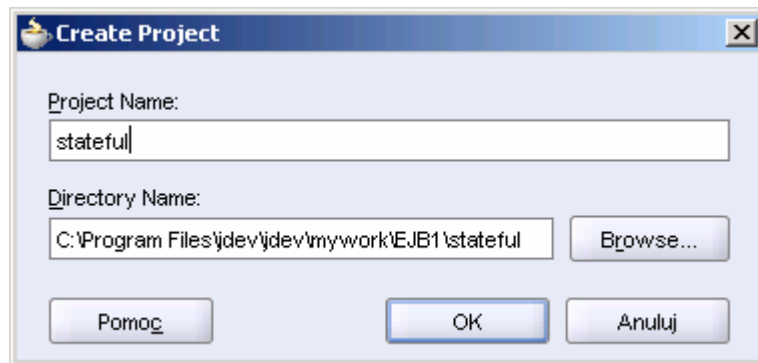
```
System.out.println( statelessEJB.getCounter( ) );
```

- Następnie należy uruchomić przykładową aplikację i przeanalizować jej wyniki.
- Na koniec należy uruchomić kilka procesów przykładowej aplikacji, aby w ten sposób zasymulować zachowanie *beana* w środowisku rzeczywistego serwera aplikacji, gdzie bardzo często wielu użytkowników równocześnie korzysta z tych samych komponentów.

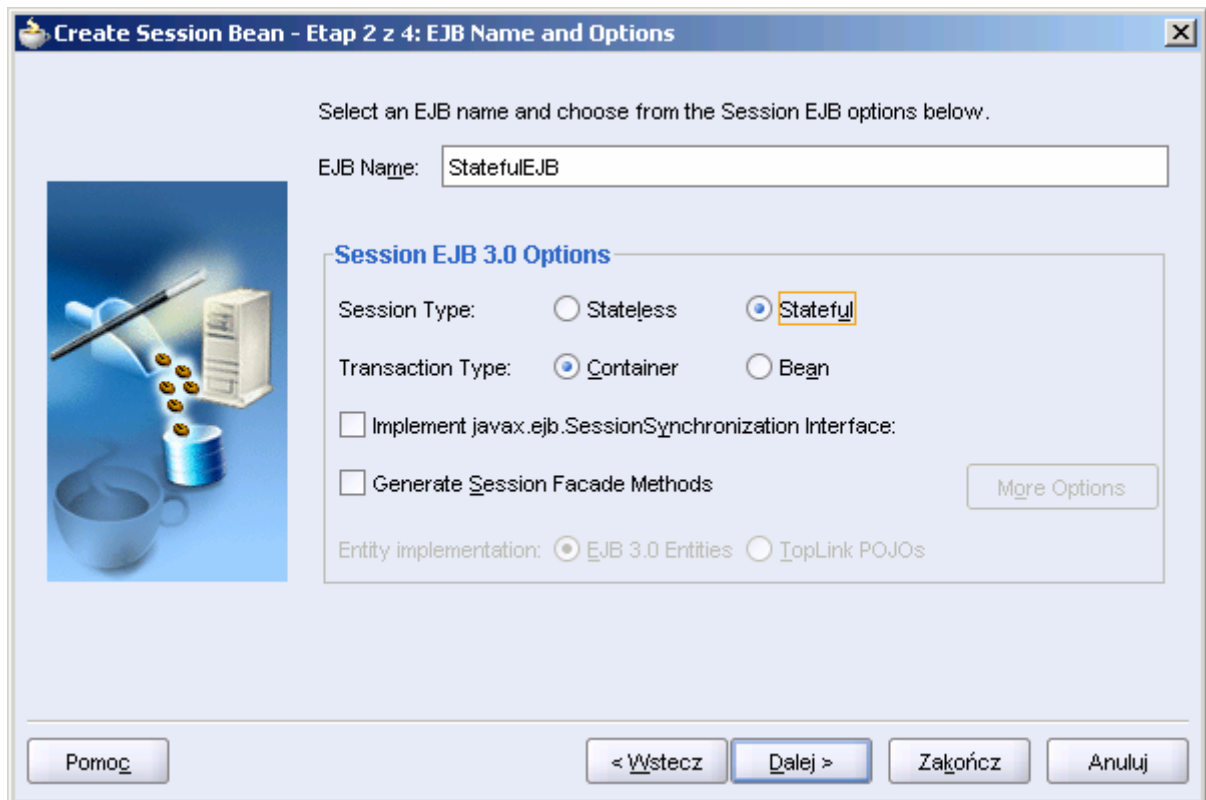
Ćwiczenie 3

W tym ćwiczeniu należy utworzyć *beana* sesyjnego typu *stateful*, zaimplementować w nim funkcjonalność identyczną, jak w bezstanowym *beanie* z poprzedniego ćwiczenia, wygenerować przykładową aplikację i porównać wyniki jej działania z wynikami aplikacji z poprzedniego ćwiczenia.

- Pracę należy rozpocząć od utworzenia nowego projektu w ramach tworzonej w tym ćwiczeniu aplikacji. Projektowi należy nadać nazwę *stateful*



- W ramach projektu należy utworzyć *beana* sesyjnego typu stanowego



Create Session Bean - Etap 2 z 4: EJB Name and Options

Select an EJB name and choose from the Session EJB options below.

EJB Name: StatefulEJB

Session EJB 3.0 Options

Session Type: Stateless Stateful

Transaction Type: Container Bean

Implement javax.ejb.SessionSynchronization Interface:

Generate Session Facade Methods More Options

Entity implementation: EJB 3.0 Entities TopLink POJOs

Pomoc < Wstecz **Dalej >** Zakończ Anuluj

- Dalej wykonywać dokładnie te same kroki, co w poprzednim ćwiczeniu.
- Po kilkukrotnym uruchomieniu przykładowej aplikacji należy porównać jej wyniki z wynikami zwracanymi przez aplikacje z poprzedniego ćwiczenia
- Należy też zasymulować równoległą pracę kilku użytkowników przykładowej aplikacji.