

Zastosowanie komponentów EJB typu Session

na podstawie

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

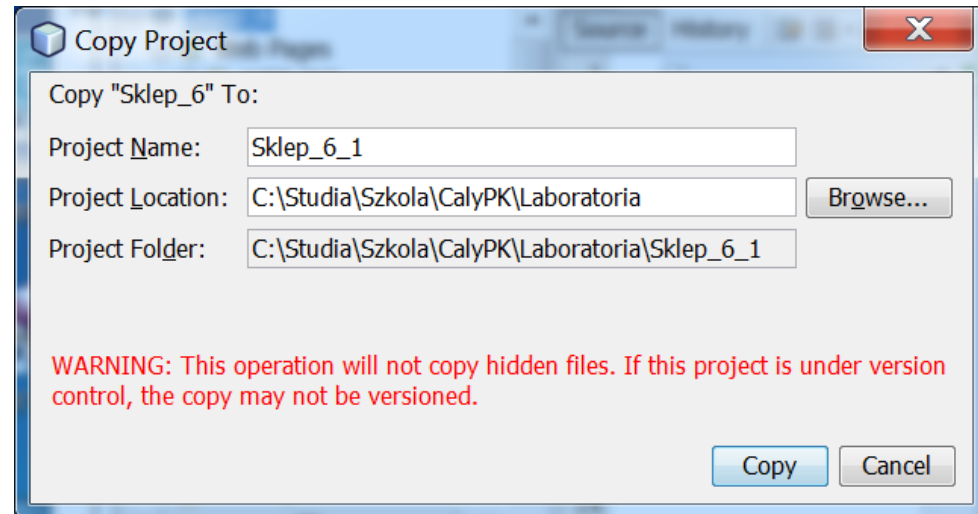
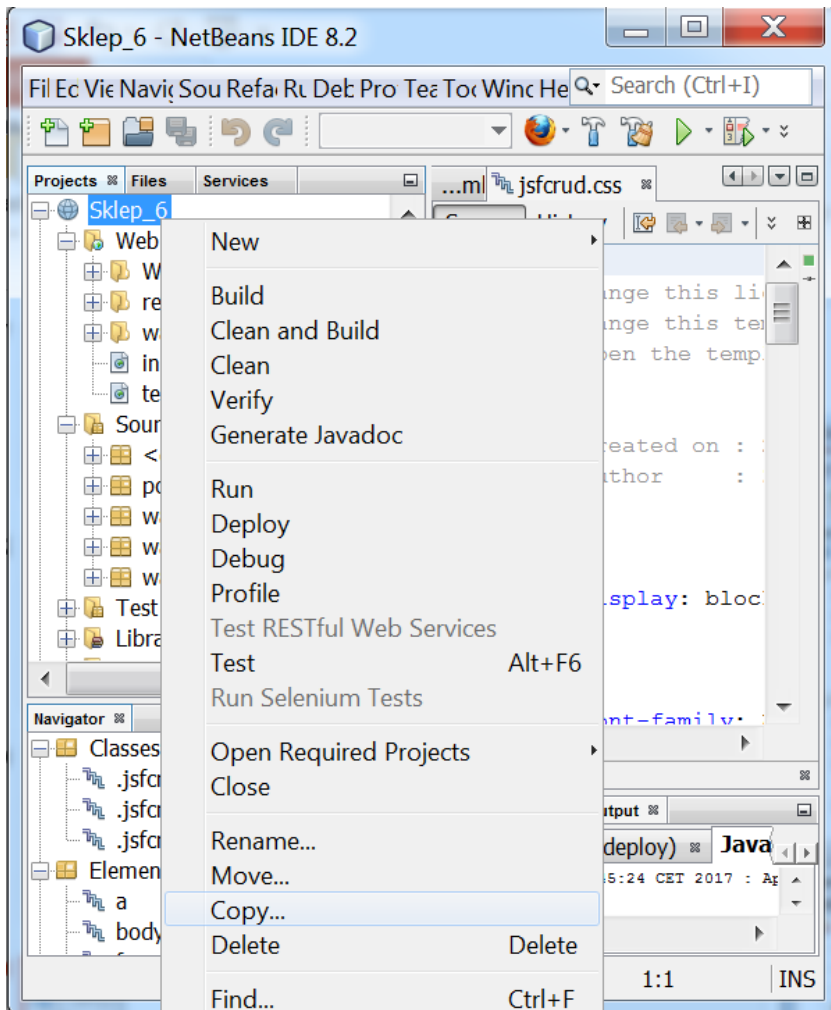
Programowanie komponentowe Lab1

1. Refaktoryzacja kodu programu Sklep_6 z lab5, TINT

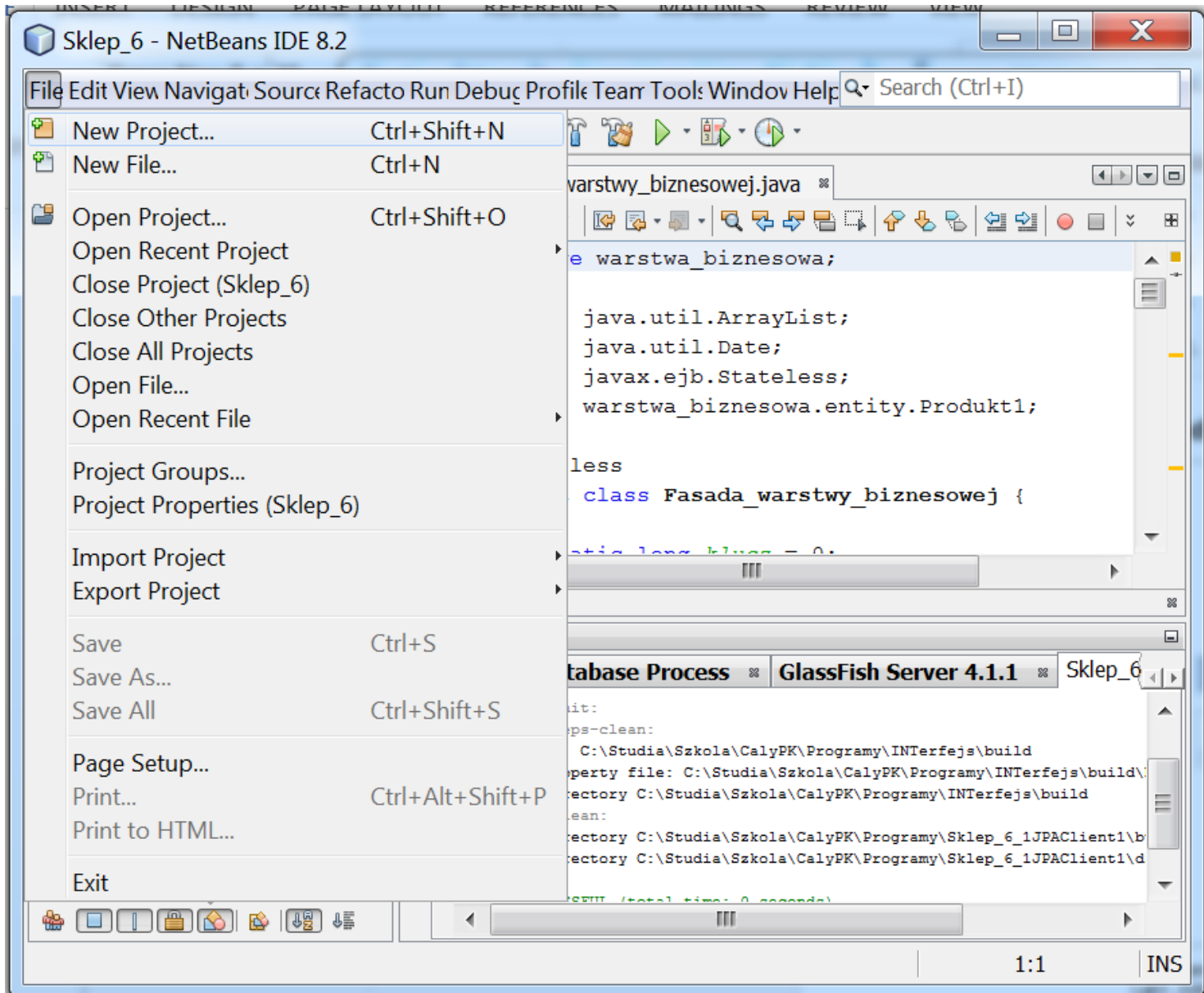
1.1. Należy wykonać kopię projektu Sklep_6 o nazwie Sklep_6_1.

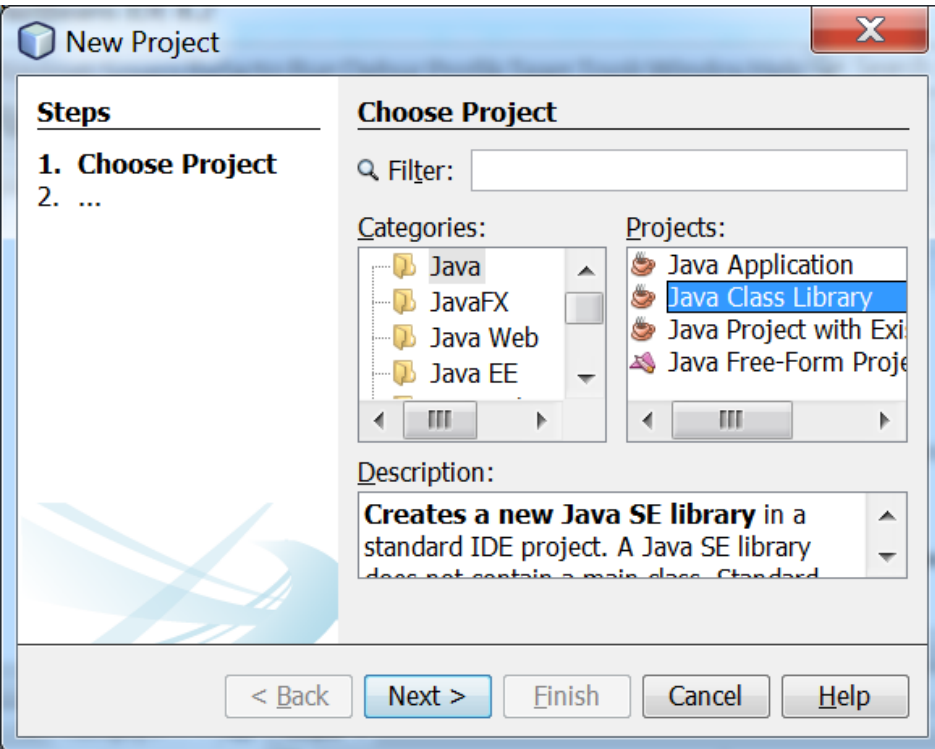
Plik do pobrania, jeśli brakuje programu wykonanego podczas lab4, TINT:

http://zofia.kruczkiewicz.staff.iar.pwr.wroc.pl/wyklady/javapk/Sklep_6.rar



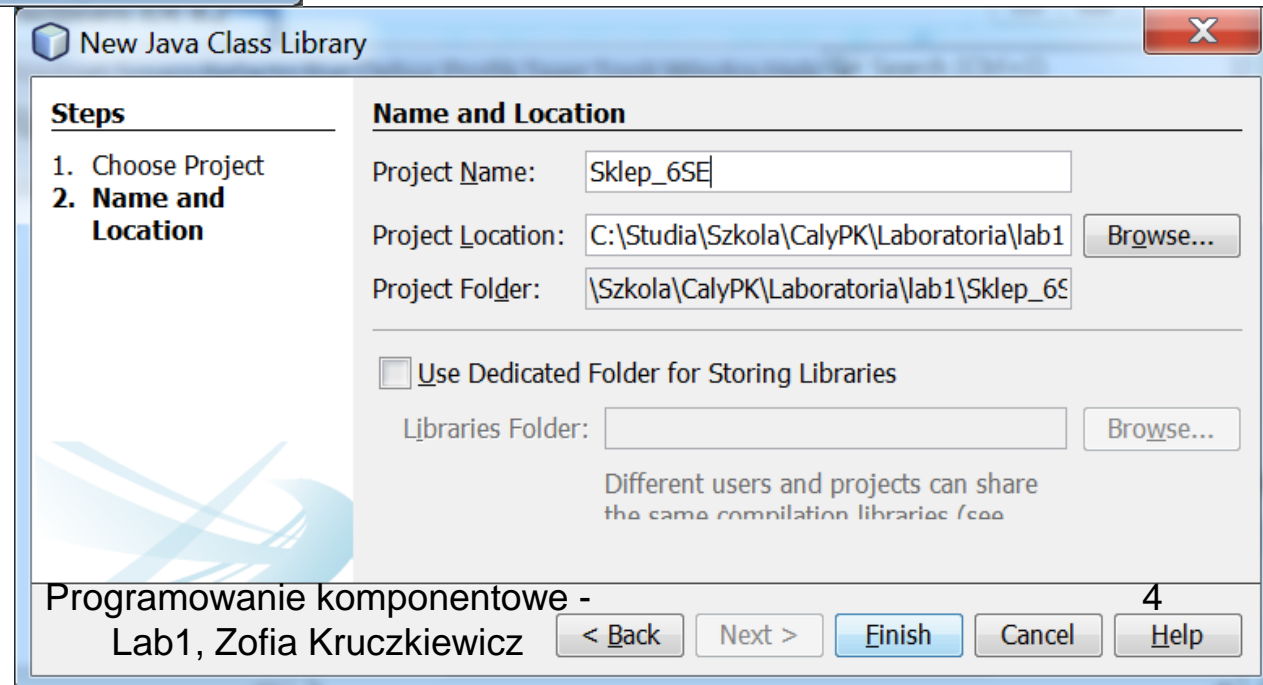
1.2. Wykonanie projektu Java SE typu Java Class Library: **File/New Project/**



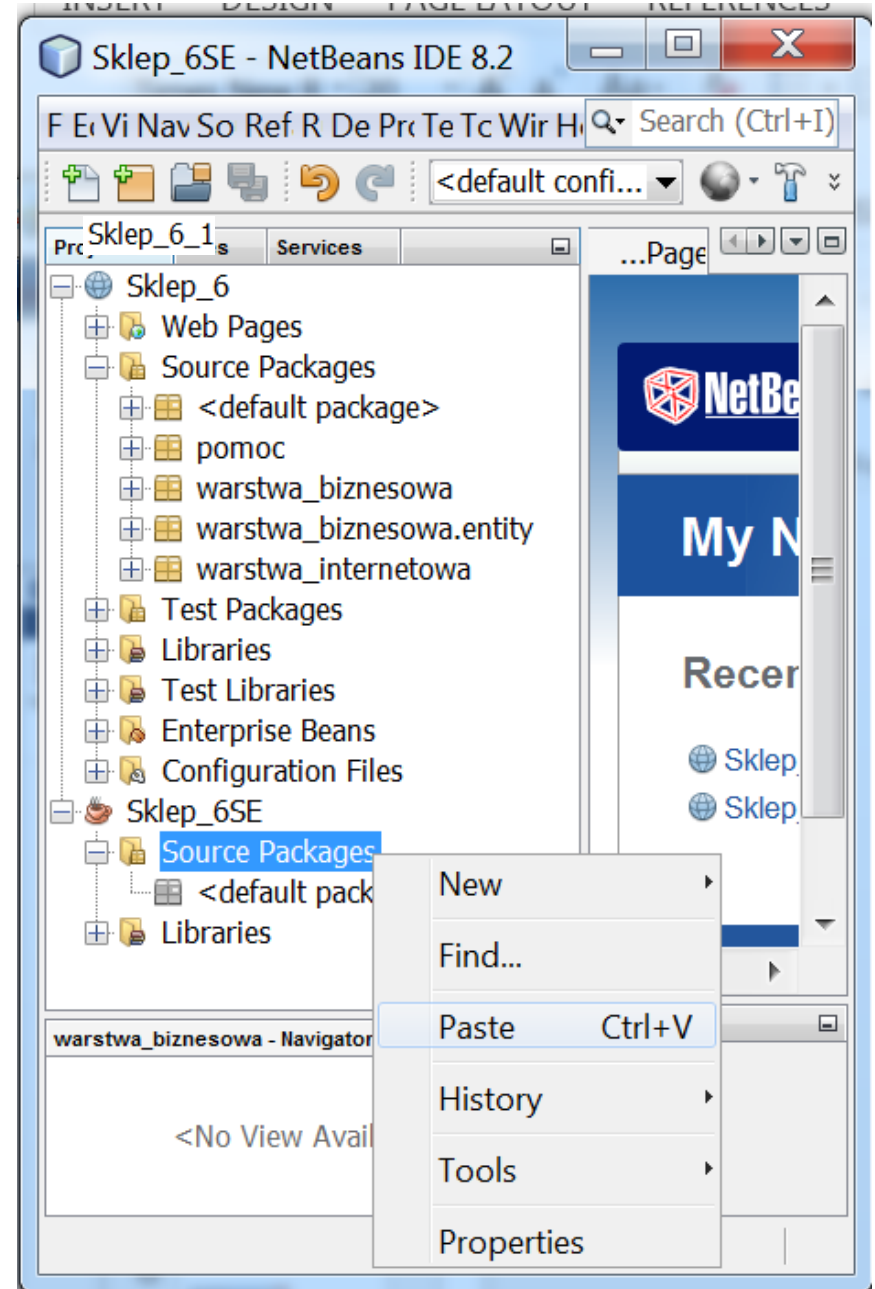
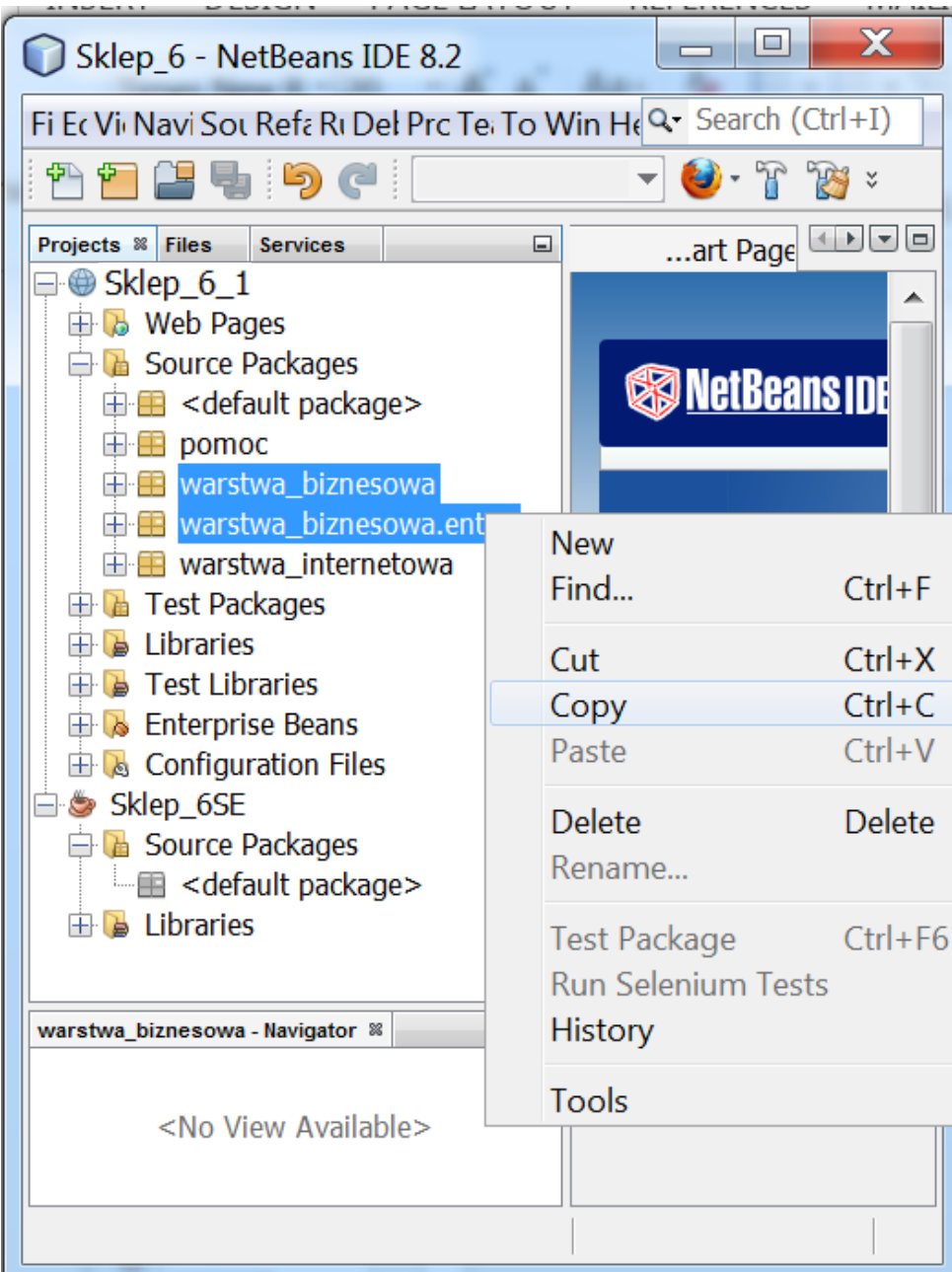


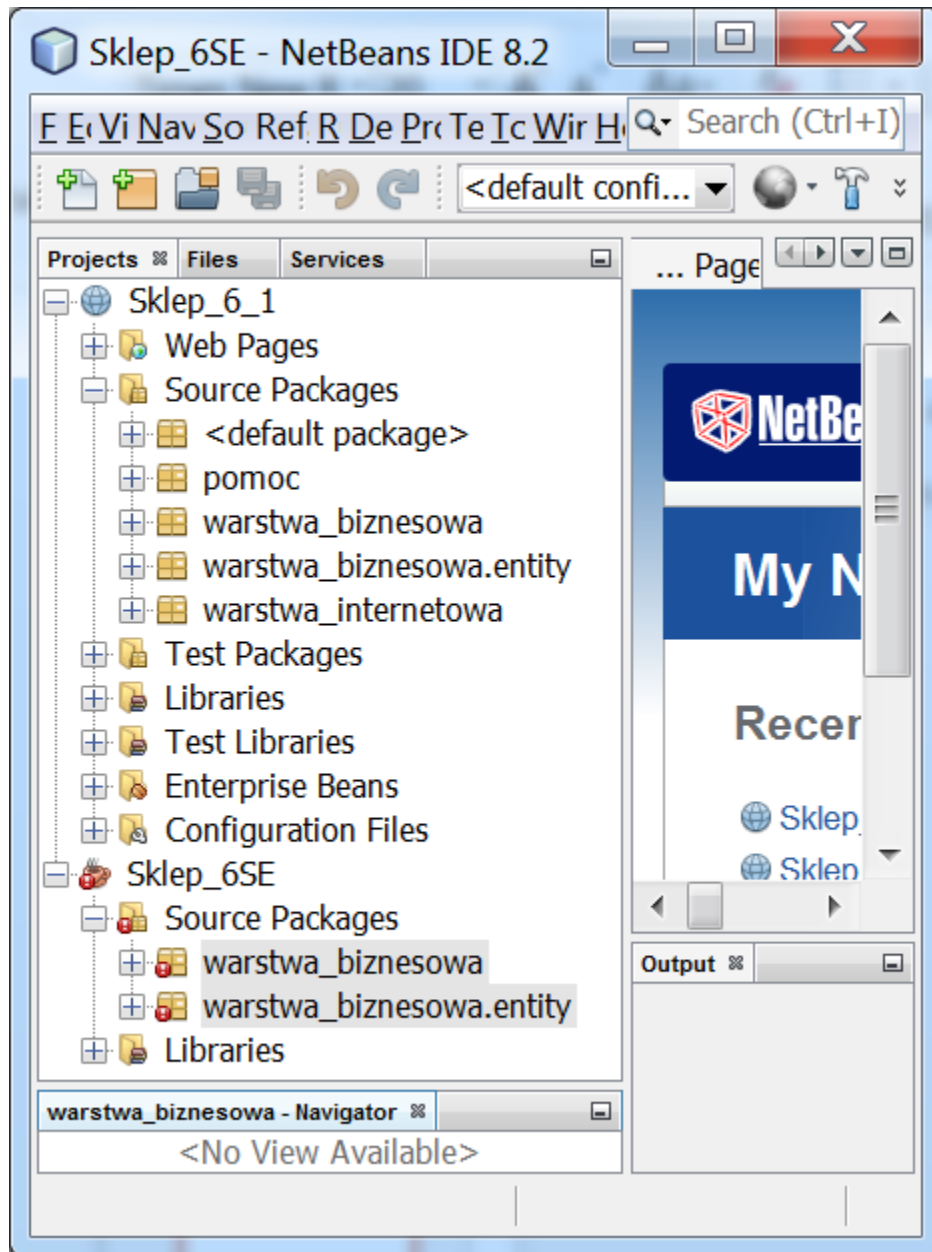
1.3. Wybór pozycji: **Java/Java ClassLibrary.**

Nadanie nazwy programowi np Sklep_6SE (**Project Name**) w wybranym katalogu (**Project Location**)



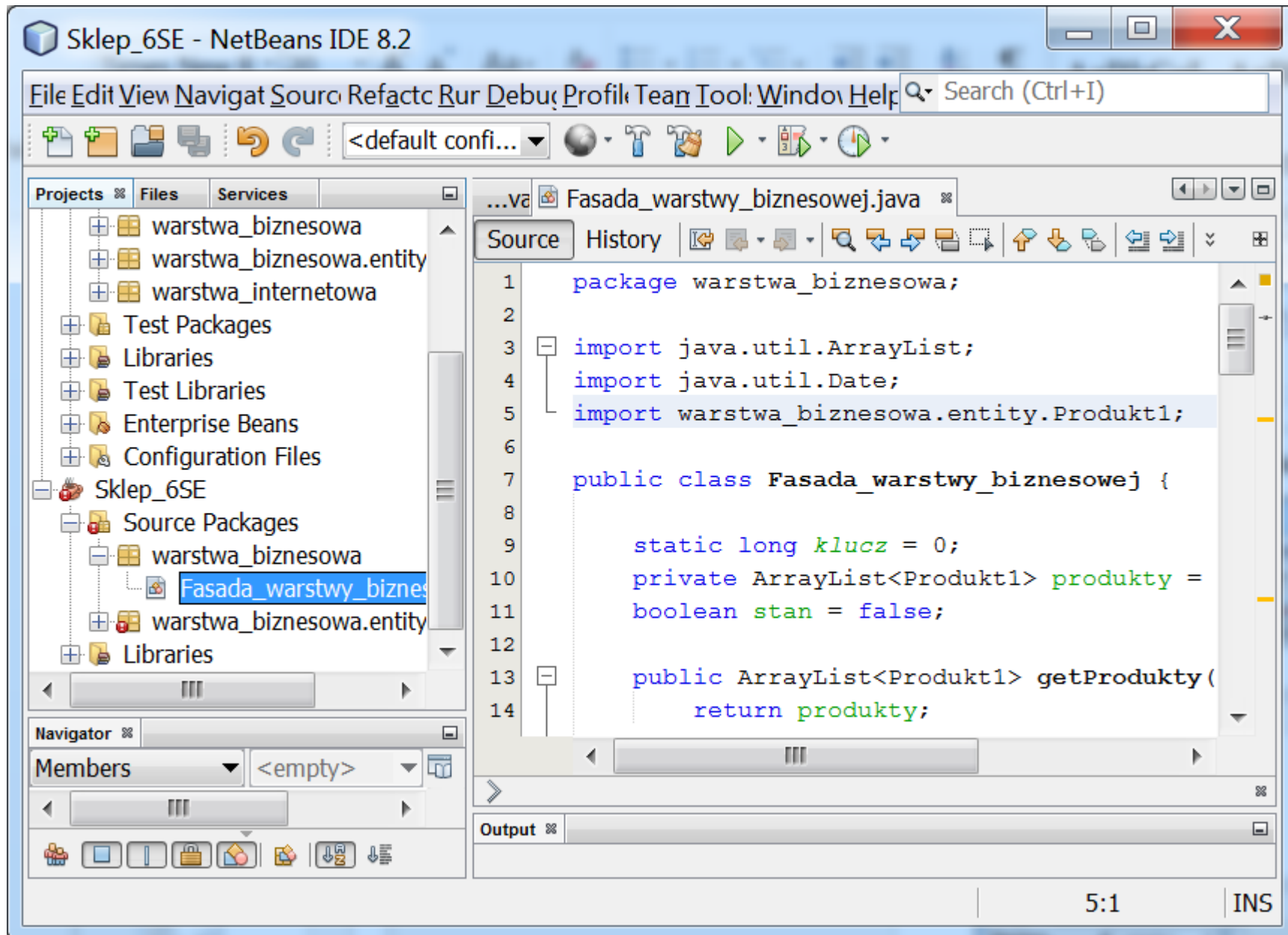
1.4. Wykonanie kopii dwóch pakietów: warstwa_biznesowa i warstwa_biznesowa.entity w projekcie Sklep_6_1 i wklejenie ich do nowego projektu z p.1.3.





1.5. W wyniku wklejenia dwóch pakietów pojawią się błędy składni. W kolejnych krokach błędy zostaną usunięte.

1.6. Należy usunąć adnotację @Stateless i niepotrzebny import javax.ejb.Stateless.

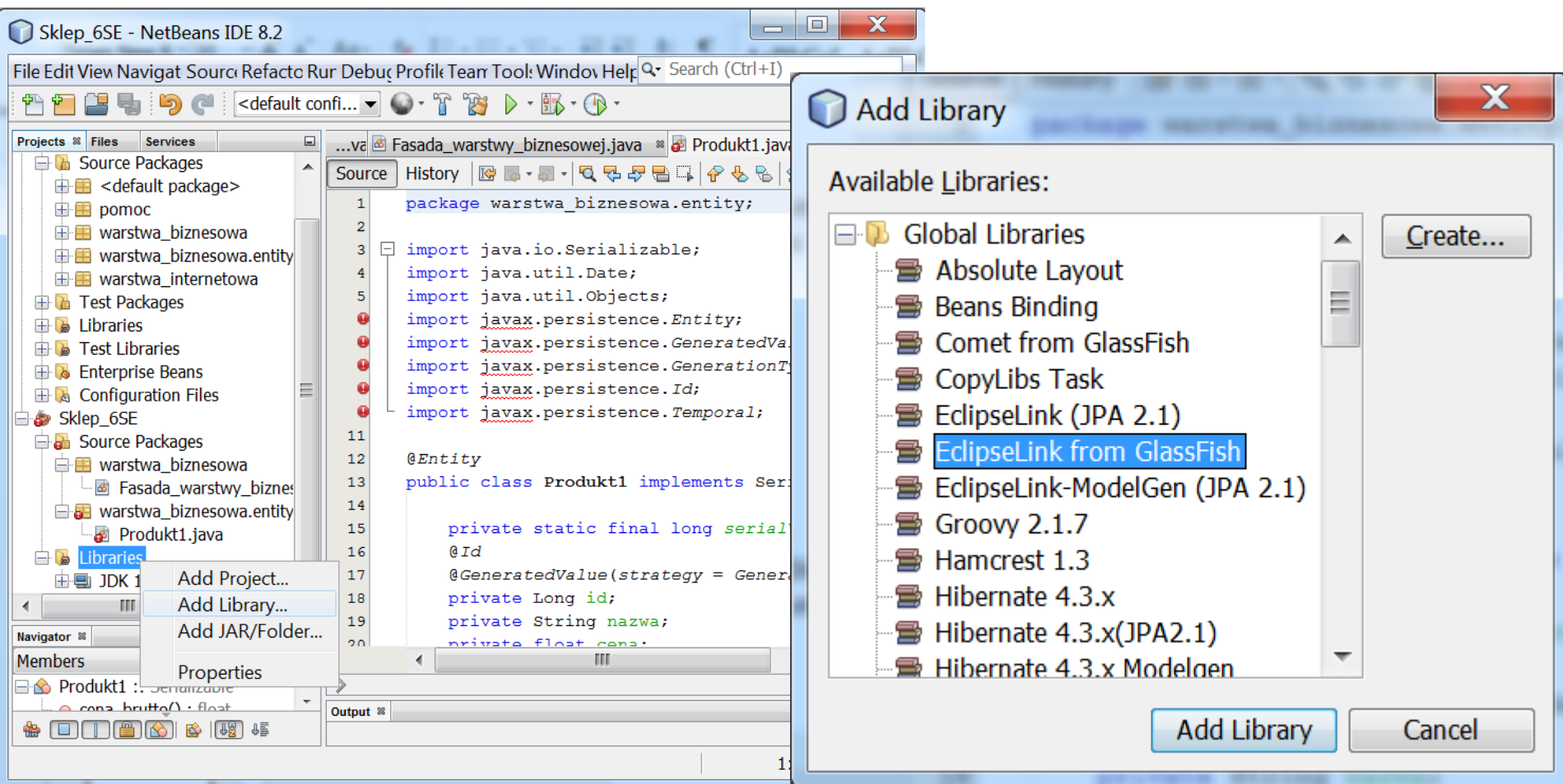


The screenshot shows the NetBeans IDE 8.2 interface. The main editor window displays the source code for the file `Fasada_warstwy_biznesowej.java`. The code is as follows:

```
1 package warstwa_biznesowa;
2
3 import java.util.ArrayList;
4 import java.util.Date;
5 import warstwa_biznesowa.entity.Produkt1;
6
7 public class Fasada_warstwy_biznesowej {
8
9     static long klucz = 0;
10    private ArrayList<Produkt1> produkty =
11    boolean stan = false;
12
13    public ArrayList<Produkt1> getProdukty(
14        return produkty;
```

The left sidebar shows the project structure for `Sklep_6SE`, with the `Fasada_warstwy_biznesowej` class selected under the `warstwa_biznesowa` package. The bottom status bar indicates a zoom level of 5:1 and the cursor is in Insert (INS) mode.

1.7. Należy uzupełnić biblioteki projektu: w zakładce **Projects** po kliknięciu prawym klawiszem na katalog **Libraries**, należy wybrać pozycję **Add Library...** i wybrać bibliotekę **EclipseLink from GlassFish** – w celu pobrania elementów definiujących klasę **Produkt1** typu **Entity**



1.8. Pobrane biblioteki

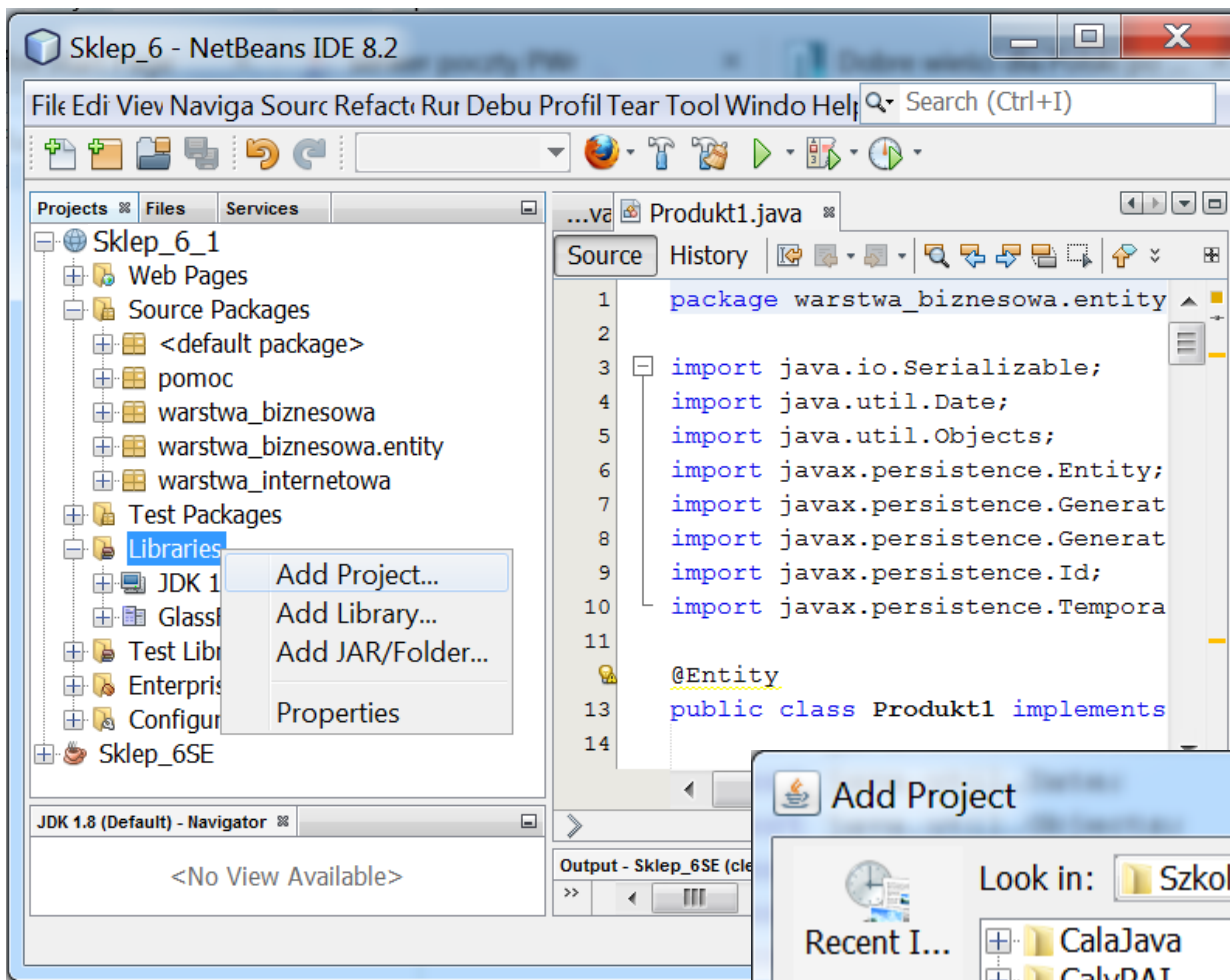
The screenshot displays the NetBeans IDE 8.2 interface for a project named 'Sklep_6SE'. The 'Libraries' section in the left-hand pane lists the following dependencies:

- EclipseLink from GlassFish - org.eclipse.persistence.antlr.jar
- EclipseLink from GlassFish - org.eclipse.persistence.asm.jar
- EclipseLink from GlassFish - org.eclipse.persistence.core.jar
- EclipseLink from GlassFish - org.eclipse.persistence.dbws.jar
- EclipseLink from GlassFish - org.eclipse.persistence.jpa.jar
- EclipseLink from GlassFish - org.eclipse.persistence.jpa.jpql.jar
- EclipseLink from GlassFish - org.eclipse.persistence.jpa.modelgen
- EclipseLink from GlassFish - org.eclipse.persistence.moxy.jar
- EclipseLink from GlassFish - org.eclipse.persistence.oracle.jar
- EclipseLink from GlassFish - javax.persistence.jar
- JDK 1.8 (Default)

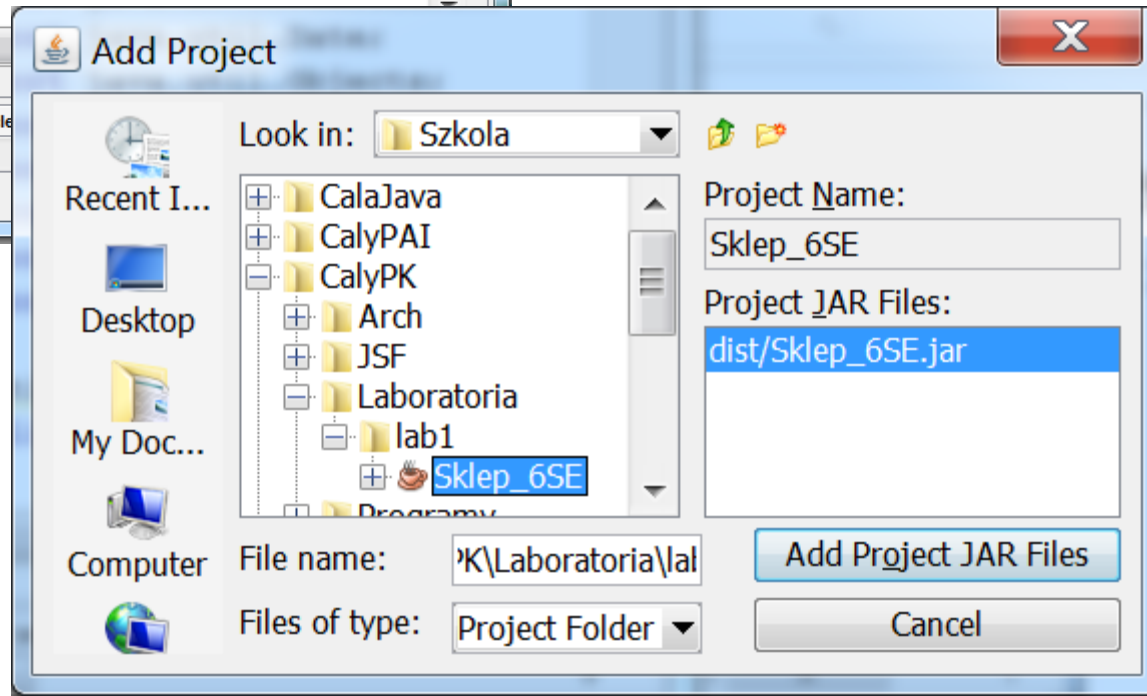
The main editor window shows the source code for 'Produkt1.java'. The code includes the following imports and class definition:

```
1 package warstwa_b
2
3 import java.io.Se
4 import java.util.
5 import java.util.
6 import javax.pers
7 import javax.pers
8 import javax.pers
9 import javax.pers
10 import javax.pers
11
12 @Entity
13 public class Prod
14
15     private stati
16     @Id
17     @GeneratedValue
18     private Long
19     private Strin
20     private float
```

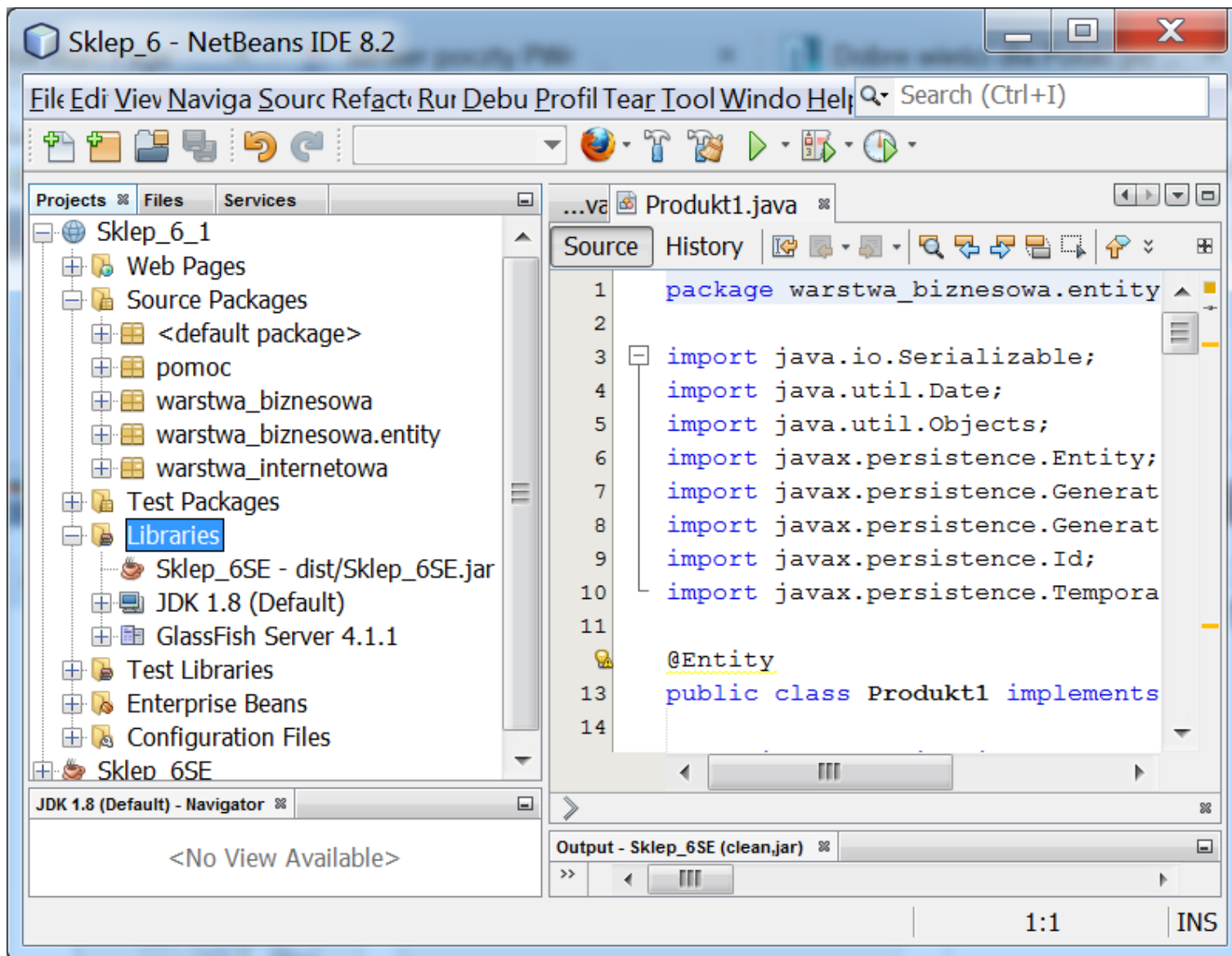
The 'Members' pane at the bottom shows 'Produkt1 :: Serializable'. The status bar at the bottom right indicates '1:1' and 'INS'.



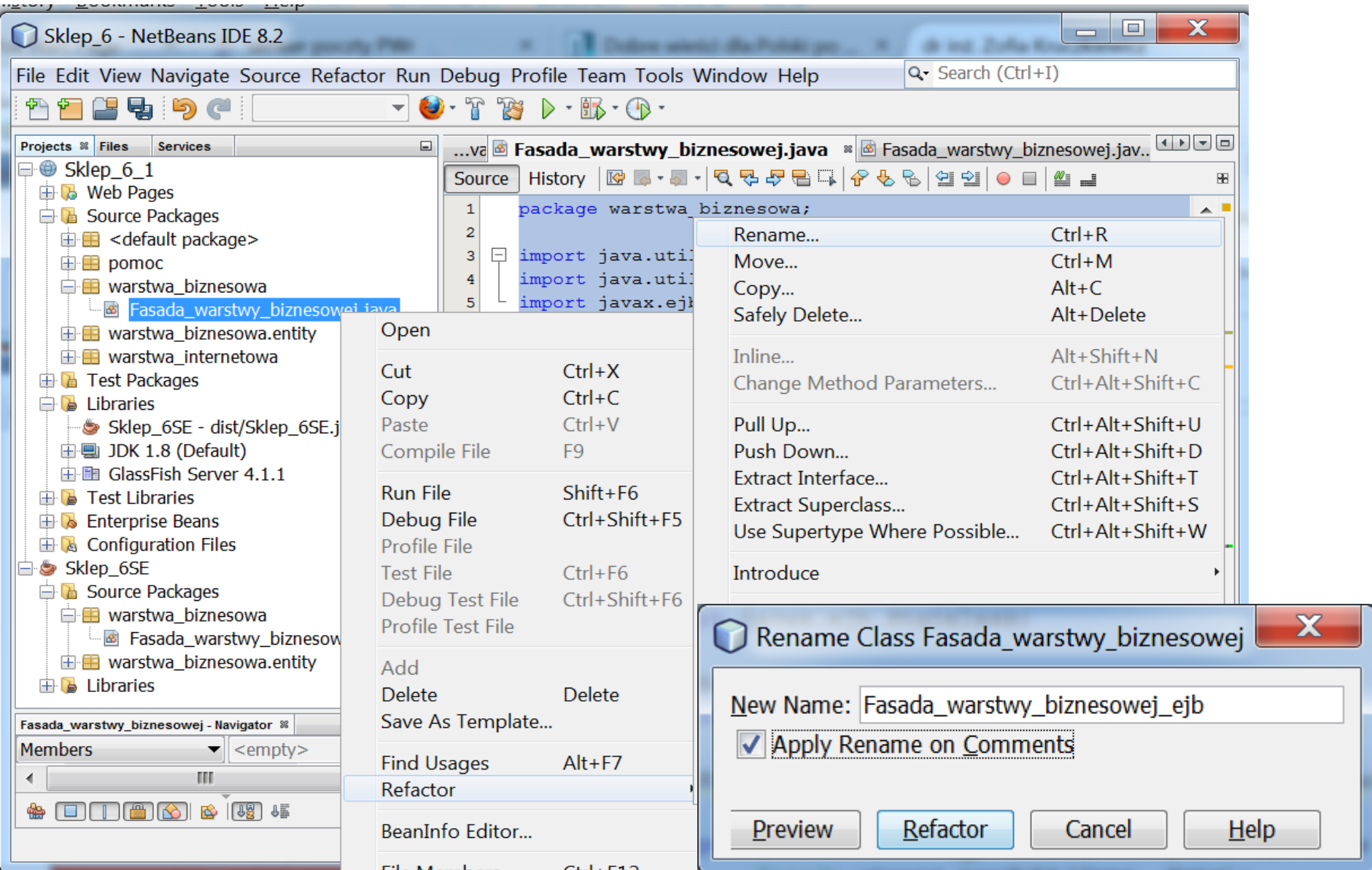
1.9. Połączenie wykonanego projektu Java SE z projektem Java Web: po wyborze **Libraries/Add Project...** w projekcie Java Web należy wybrać z listy projekt Java SE i kliknąć na klawisz **Add Project JAR Files**



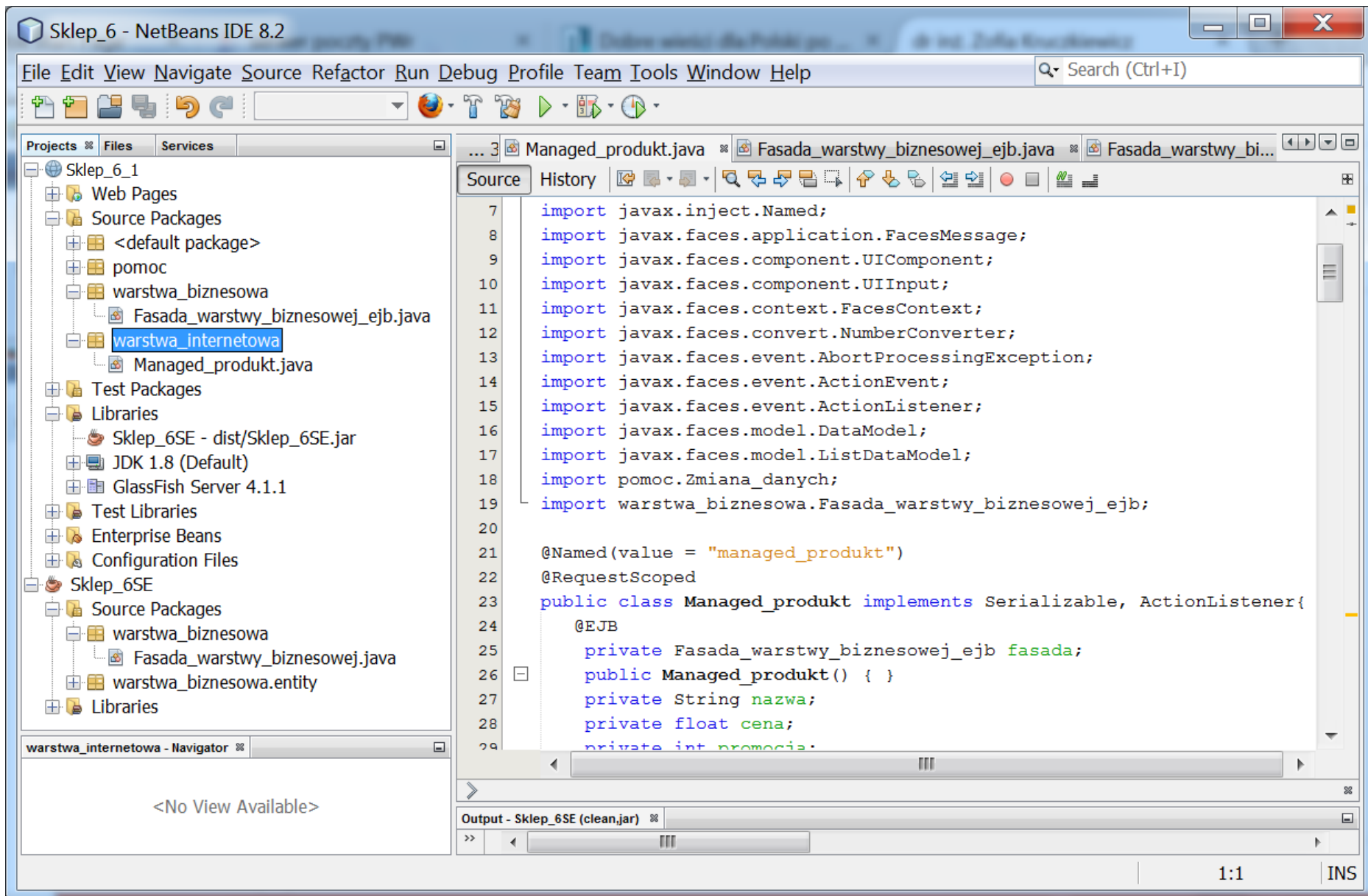
1.10. Wynik



1.11. Zmiana nazwy dotychczasowego komponentu EJB typu **Fasada_warstwy_biznesowej** np. na **Fasada_warstwy_biznesowej_ejb** (prawym klawiszem wybrać ten komponent, i z listy wybrać kolejno **Refactor/Rename** i podać nową nazwę)



1.12. W wyniku refaktoryzacji z p. 1.11 uległa modyfikacji nazwa typu komponentu EJB **fasada** w klasie **Managed_produk**t. Należy usunąć pakiet **warstwa_biznesowa.entity** z projektu Java Web: Sklep_6_1.



The screenshot displays the NetBeans IDE 8.2 interface. The left-hand 'Projects' pane shows the project structure for 'Sklep_6_1', with the 'warstwa_internetowa' package selected. The main editor window shows the source code of 'Managed_produk.java' in the 'warstwa_internetowa' package. The code includes several imports and a class definition for 'Managed_produk' that implements 'Serializable' and 'ActionListener'. The class is annotated with '@Named' and '@RequestScoped'. The code is as follows:

```
7   import javax.inject.Named;
8   import javax.faces.application.FacesMessage;
9   import javax.faces.component.UIComponent;
10  import javax.faces.component.UIInput;
11  import javax.faces.context.FacesContext;
12  import javax.faces.convert.NumberConverter;
13  import javax.faces.event.AbortProcessingException;
14  import javax.faces.event.ActionEvent;
15  import javax.faces.event.ActionListener;
16  import javax.faces.model.DataModel;
17  import javax.faces.model.ListDataModel;
18  import pomoc.Zmiana_danych;
19  import warstwa_biznesowa.Fasada_warstwy_biznesowej_ejb;
20
21  @Named(value = "managed_produk")
22  @RequestScoped
23  public class Managed_produk implements Serializable, ActionListener{
24      @EJB
25      private Fasada_warstwy_biznesowej_ejb fasada;
26      public Managed_produk() { }
27      private String nazwa;
28      private float cena;
29      private int promocja;
```

The bottom status bar shows the zoom level as 1:1 and the cursor as INS.

1.13. Należy zmodyfikować kod metod komponentu EJB, który teraz może być zdefiniowany za pomocą metod klasy **Fasada_warstwy_biznesowej**

```
package warstwa_biznesowa;
```

```
import java.util.ArrayList;
```

```
import java.util.Date;
```

```
import javax.ejb.Stateless;
```

```
@Stateless
```

```
public class Fasada_warstwy_biznesowej_ejb {
```

```
Fasada_warstwy_biznesowej fasada=new Fasada_warstwy_biznesowej();
```

```
public void utworz_produkt(String dane[], Date data) {
```

```
    fasada.utworz_produkt(dane, data);
```

```
}
```

```
public String[] dane_produktu() {
```

```
    return fasada.dane_produktu();
```

```
}
```

```
public ArrayList<ArrayList<String>> items() {
```

```
    return fasada.items();
```

```
}
```

```
}
```

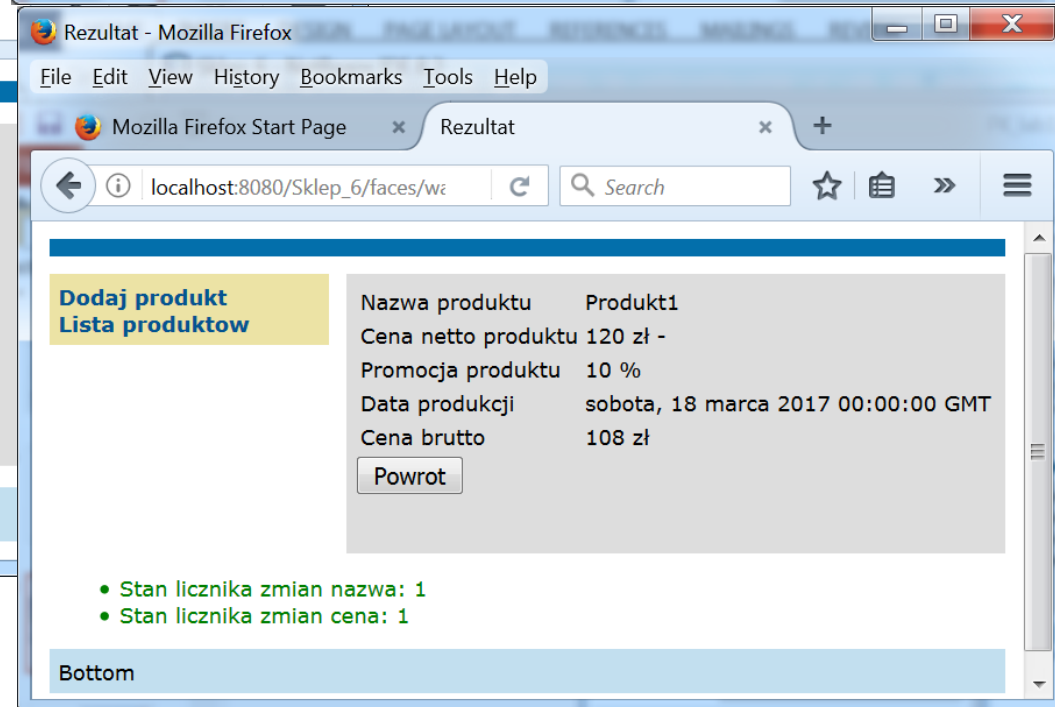
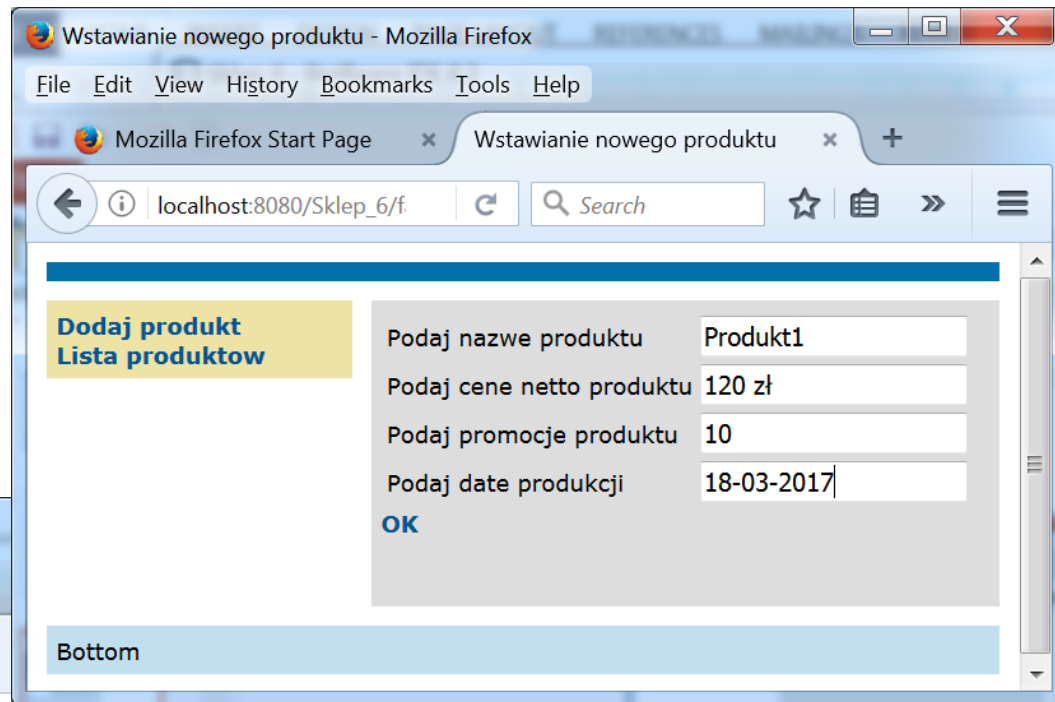
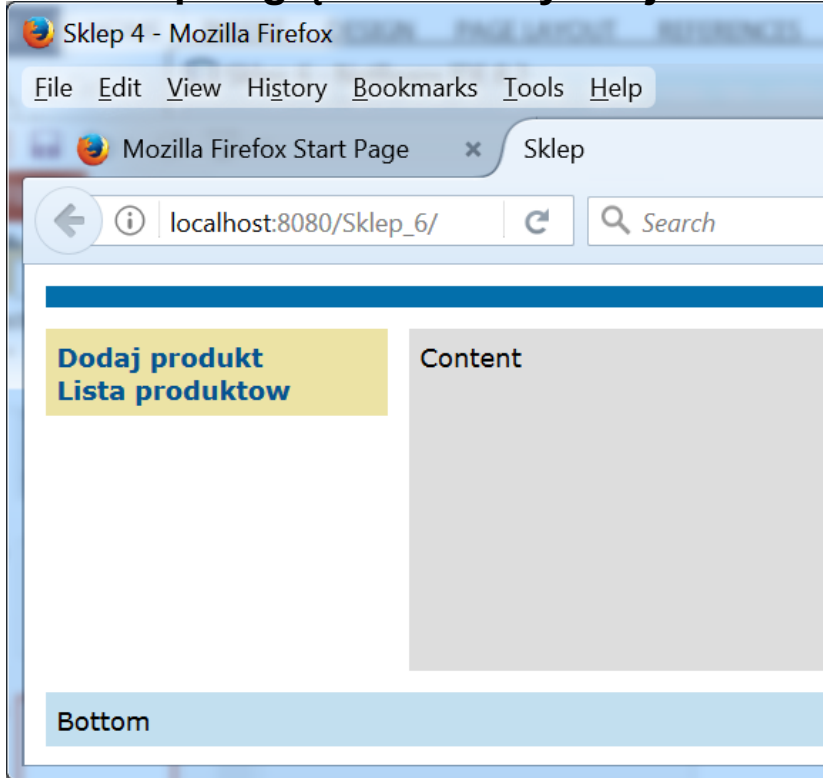
1.14. Wynik po zakończeniu refaktoryzacji kodu – wykonana aplikacja używa komponentu EJB typu Session- rodzajStateless

The screenshot displays the NetBeans IDE 8.2 interface for a project named 'Sklep_6'. The left-hand 'Projects' pane shows a tree view of the project structure, with the file 'Fasada_warstwy_biznesowej_ejb.java' selected under the 'warstwa_biznesowa' package. The main editor window shows the source code of this file, which is a Stateless EJB. The code includes imports for 'java.util.ArrayList', 'java.util.Date', and 'javax.ejb.Stateless'. The class 'Fasada_warstwy_biznesowej_ejb' is annotated with '@Stateless' and contains three methods: 'utworz_produkt', 'dane_produktu', and 'items'. The 'items' method is currently selected, and its return type is 'ArrayList<ArrayList<String>>'. The bottom status bar shows the application is running on 'GlassFish Server 4.1.1'.

```
1 package warstwa_biznesowa;
2
3 import java.util.ArrayList;
4 import java.util.Date;
5 import javax.ejb.Stateless;
6
7 @Stateless
8 public class Fasada_warstwy_biznesowej_ejb {
9     Fasada_warstwy_biznesowej fasada=new Fasada_warstwy_biznesowej();
10
11     public void utworz_produkt(String dane[], Date data) {
12         fasada.utworz_produkt(dane, data);
13     }
14     public String[] dane_produktu() {
15         return fasada.dane_produktu();
16     }
17     public ArrayList<ArrayList<String>> items() {
18         return fasada.items();
19     }
20 }
```

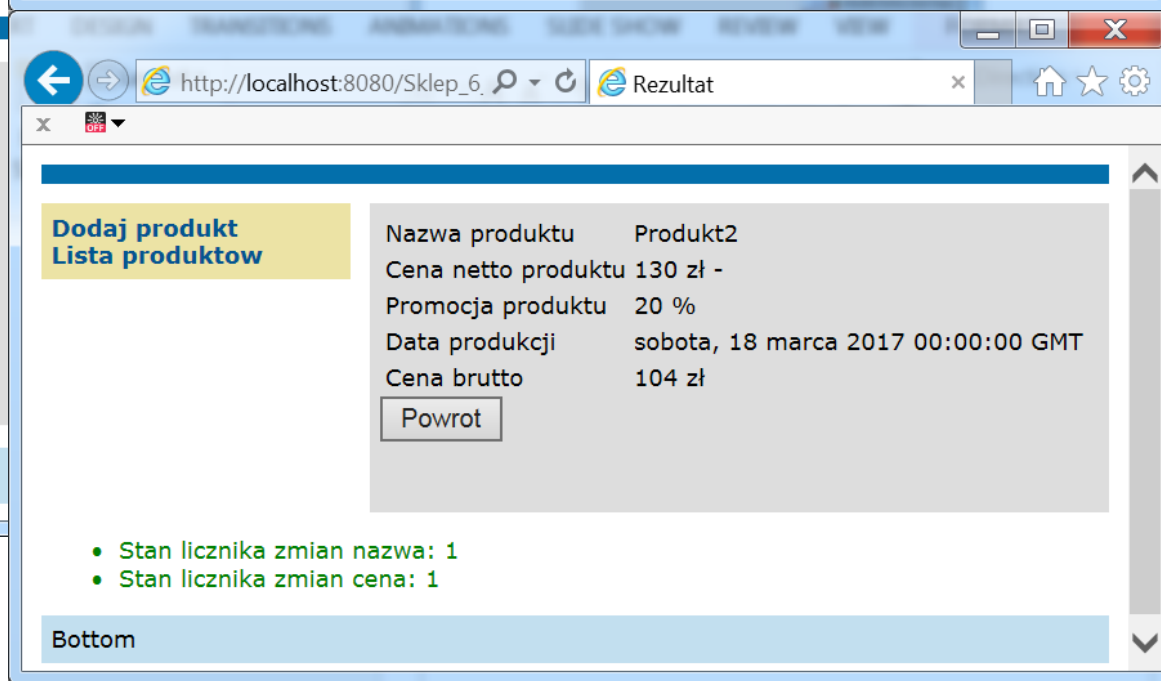
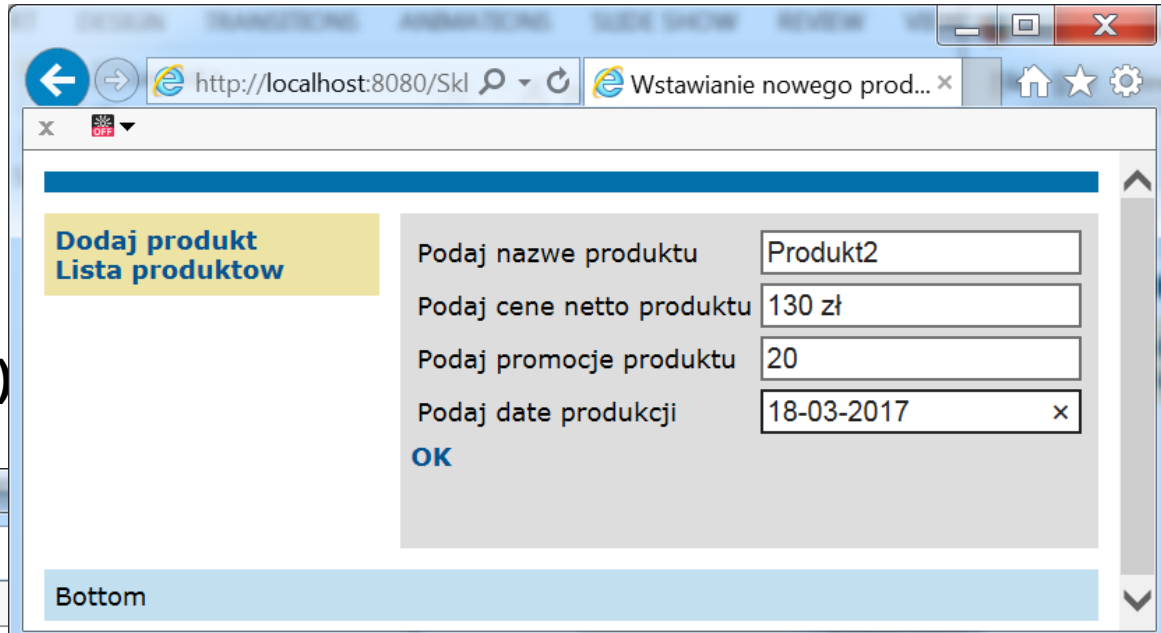
1.15. Uruchomienie aplikacji opartej na komponencie EJB typu Session, rodzaj Stateless - Clean and Build/Deploy/Run

Uruchomienie 1-ej instancji warstwy klienta aplikacji i wprowadzenie danych produktu – może być uruchomiona automatycznie w przeglądarce domyślnej.



1.16. Uruchomienie 2-ej instancji warstwy klienta aplikacji i wprowadzenie danych kolejnego produktu (prze wprowadzenie adresu:

http://localhost:8080/Sklep_6_1)



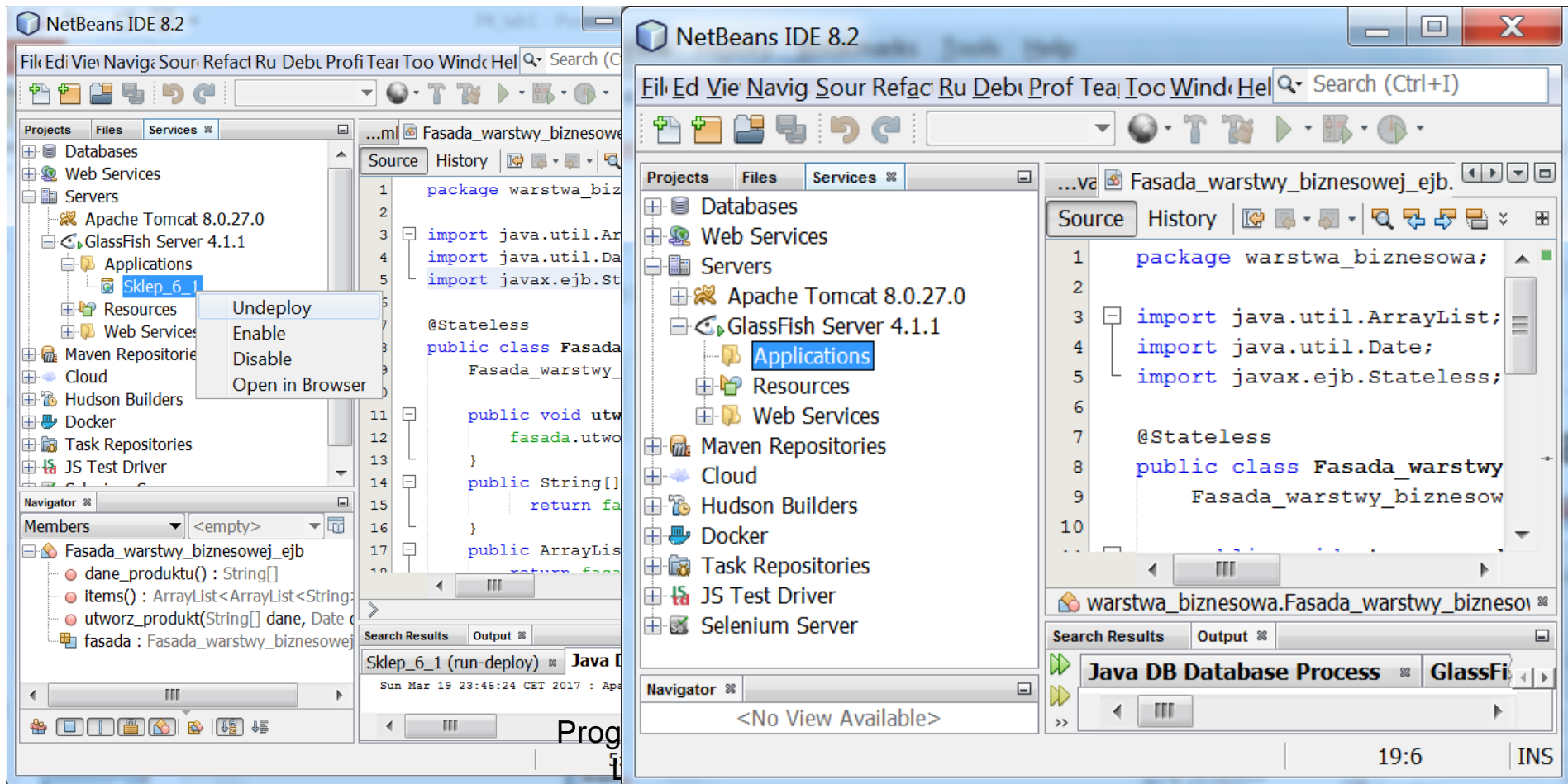
1.17. Po uruchomieniu formularza **Lista produktów** w obu instancjach klientów aplikacji widać, że korzystają **z tej samej instancji** komponentu typu **Fasada_warstwy_biznesowej_ejb**, czyli tej samej instancji obiektu **Fasada_warstwy_biznesowej**.

Id produktu	Nazwa produktu	Cena netto produktu	Promocja produktu	Data produkcji	Cena brutto
1	Produkt1	120.0	10	Sat Mar 18 01:00:00 CET 2017	108.0
2	Produkt2	130.0	20	Sat Mar 18 01:00:00 CET 2017	104.0

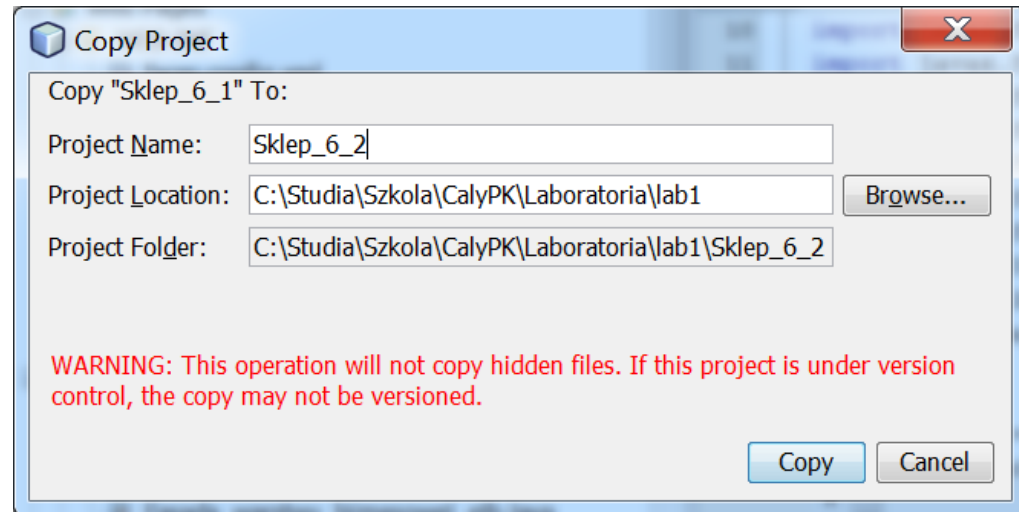
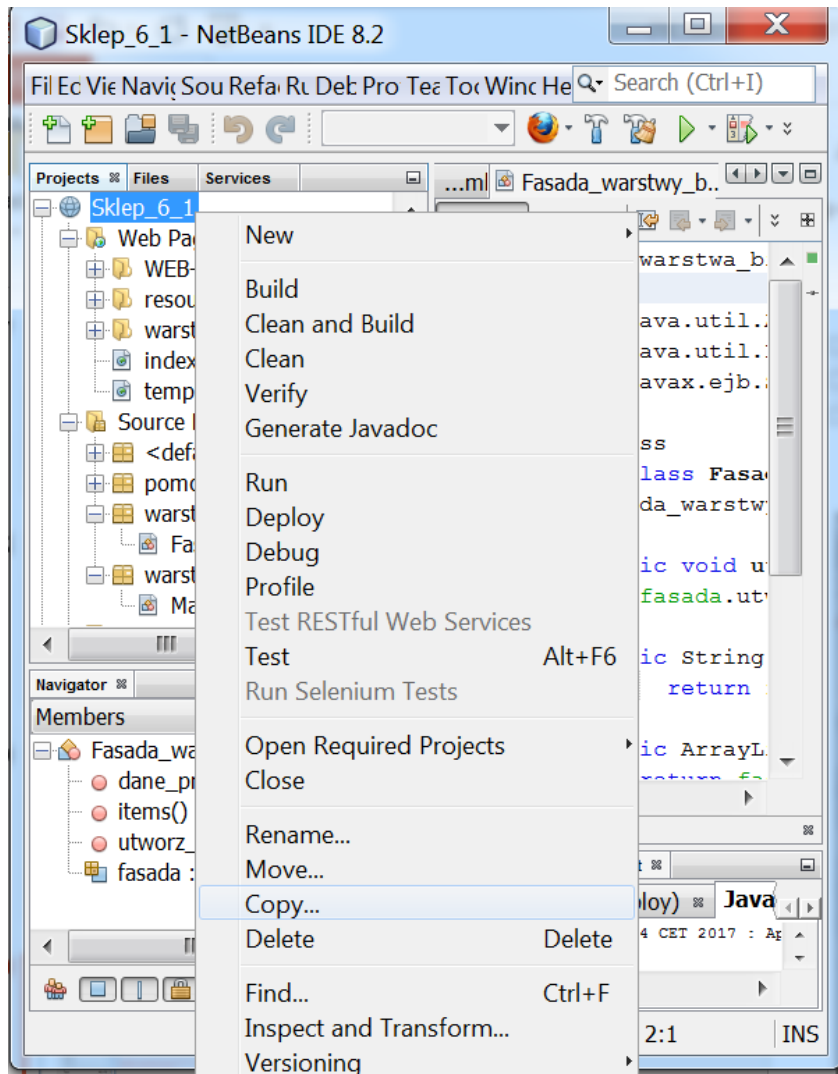
Id produktu	Nazwa produktu	Cena netto produktu	Promocja produktu	Data produkcji	Cena brutto
1	Produkt1	120.0	10	Sat Mar 18 01:00:00 CET 2017	108.0
2	Produkt2	130.0	20	Sat Mar 18 01:00:00 CET 2017	104.0

2. Przygotowanie środowiska platformy Java EE do uruchomienia kolejnej aplikacji – opartej na komponentach EJB typu Session – rodzaj **Stateful**

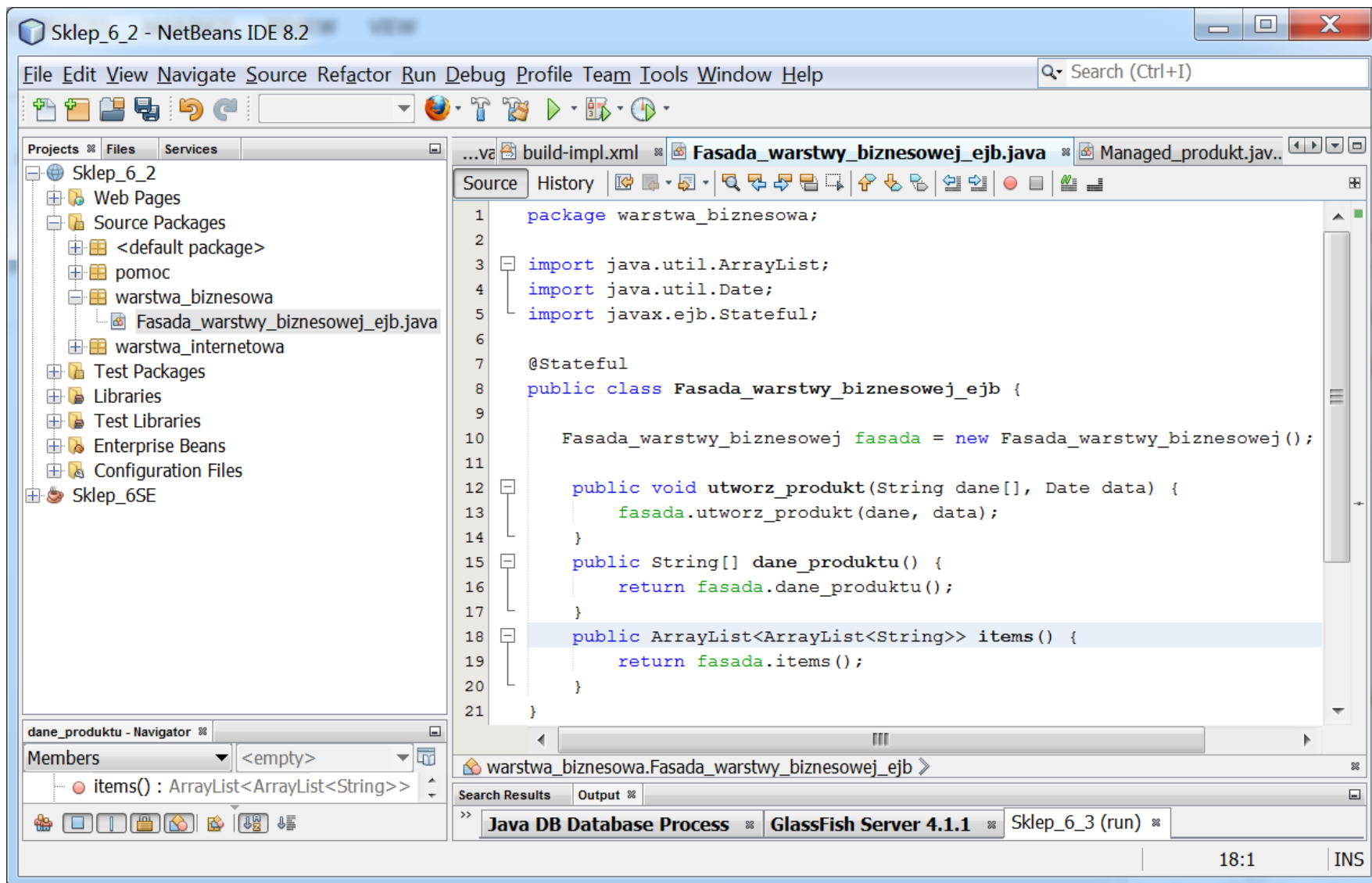
2.1. Przygotowanie środowiska platformy Java EE do uruchomienia kolejnej aplikacji – **zamknięcie aplikacji Sklep_6_1**: w oknie zakładki Services należy otworzyć pozycję **Servers/GlassFish Server 4.1.1/Applications** i prawym klawiszem wybrać pozycję Sklep_6_1. Z listy wybrać pozycję **Undeploy**.



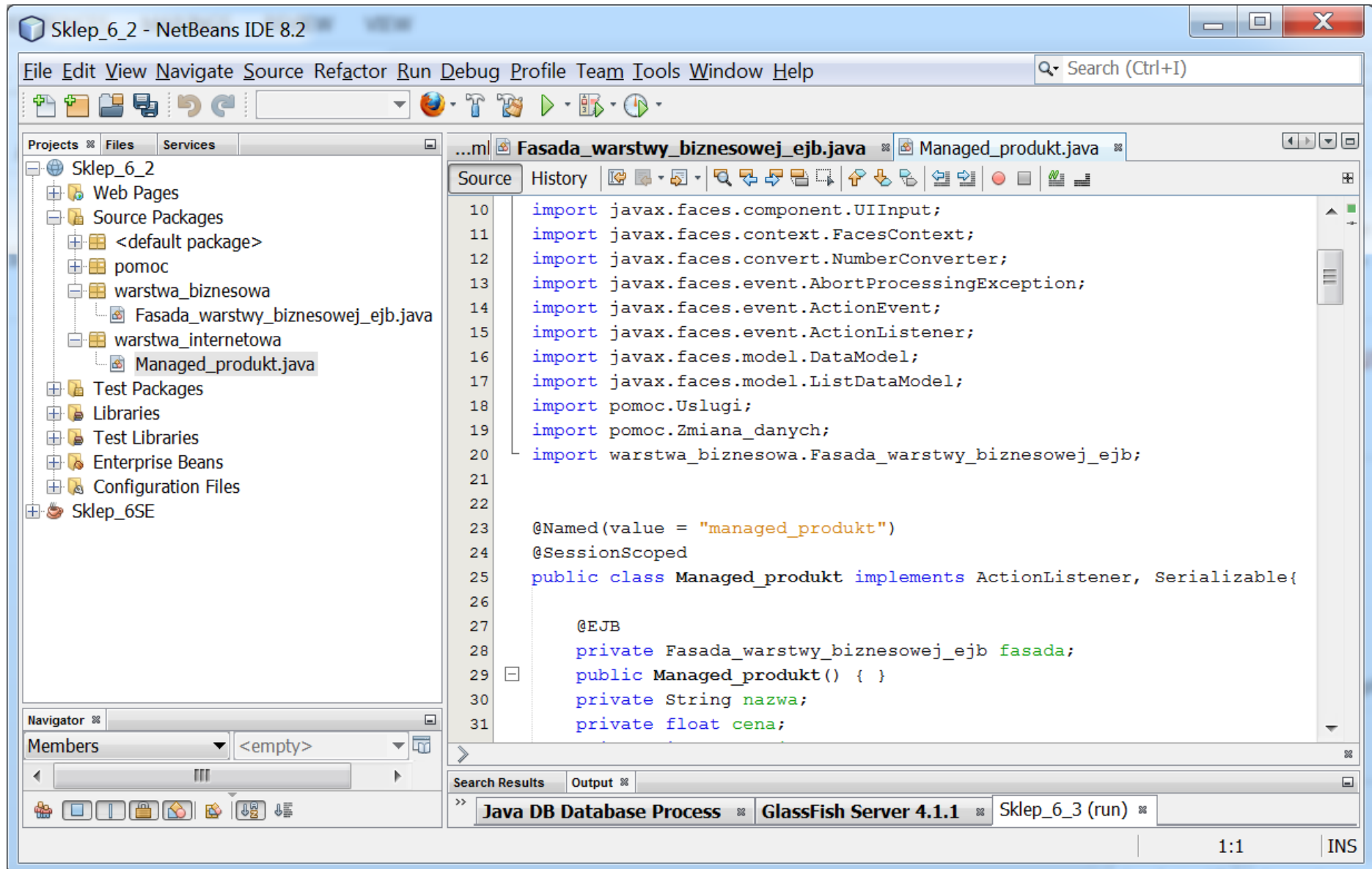
2.2. Wykonanie kopii programu Sklep_6_1 jako programu Sklep_6_2.



2.3. Wynik kopiowania. Po wykonaniu kopiowania należy zamknąć projekt Sklep_6_1. Następnie, należy zmodyfikować adnotację **@Stateless** na adnotację **@Stateful** – aby usunąć błędy należy kliknąć prawym klawiszem na powierzchnię edytora i z listy wybrać pozycję **Fix Imports**



2.4. Należy zmodyfikować adnotację **@RequestScoped** na **@SessionScope** i dodać implementację interfejsu **Serializable** w klasie **Managed_produk**t. W celu usunięcia błędów składni należy wybrać pozycję **Fix Imports**.



The screenshot shows the NetBeans IDE 8.2 interface. The main editor window displays the source code for `Managed_produk.java`. The code is as follows:

```
10 import javax.faces.component.UIInput;
11 import javax.faces.context.FacesContext;
12 import javax.faces.convert.NumberConverter;
13 import javax.faces.event.AbortProcessingException;
14 import javax.faces.event.ActionEvent;
15 import javax.faces.event.ActionListener;
16 import javax.faces.model.DataModel;
17 import javax.faces.model.ListDataModel;
18 import pomoc.Uslugi;
19 import pomoc.Zmiana_danych;
20 import warstwa_biznesowa.Fasada_warstwy_biznesowej_ejb;
21
22
23 @Named(value = "managed_produk")
24 @SessionScoped
25 public class Managed_produk implements ActionListener, Serializable{
26
27     @EJB
28     private Fasada_warstwy_biznesowej_ejb fasada;
29     public Managed_produk() { }
30     private String nazwa;
31     private float cena;
```

The IDE interface includes a Project Explorer on the left showing the project structure for `Sklep_6_2`, with `Managed_produk.java` selected under `warstwa_biznesowa`. The bottom status bar shows the current project is `Sklep_6_3 (run)`.

2.5. Należy ustawić czas sesji 2 min w pliku web.xml

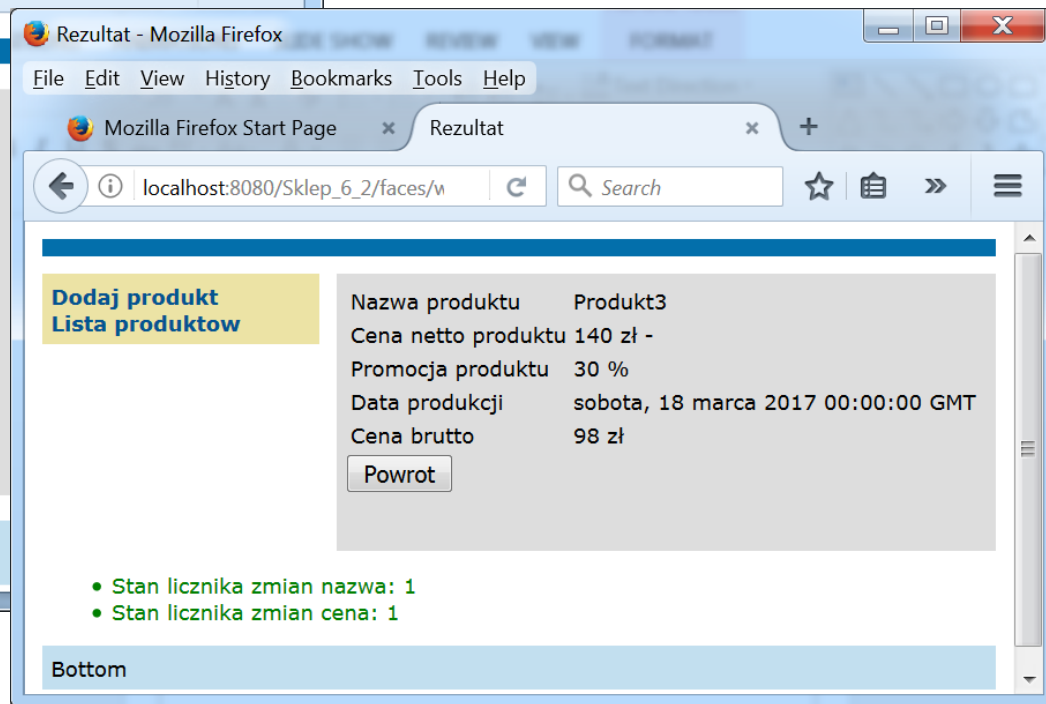
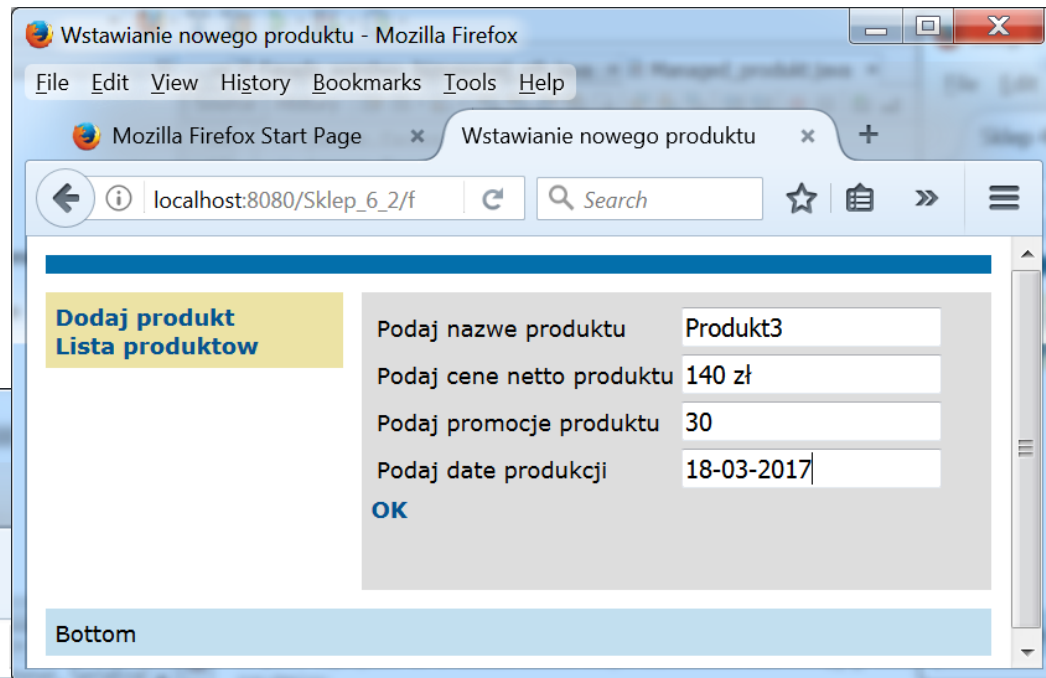
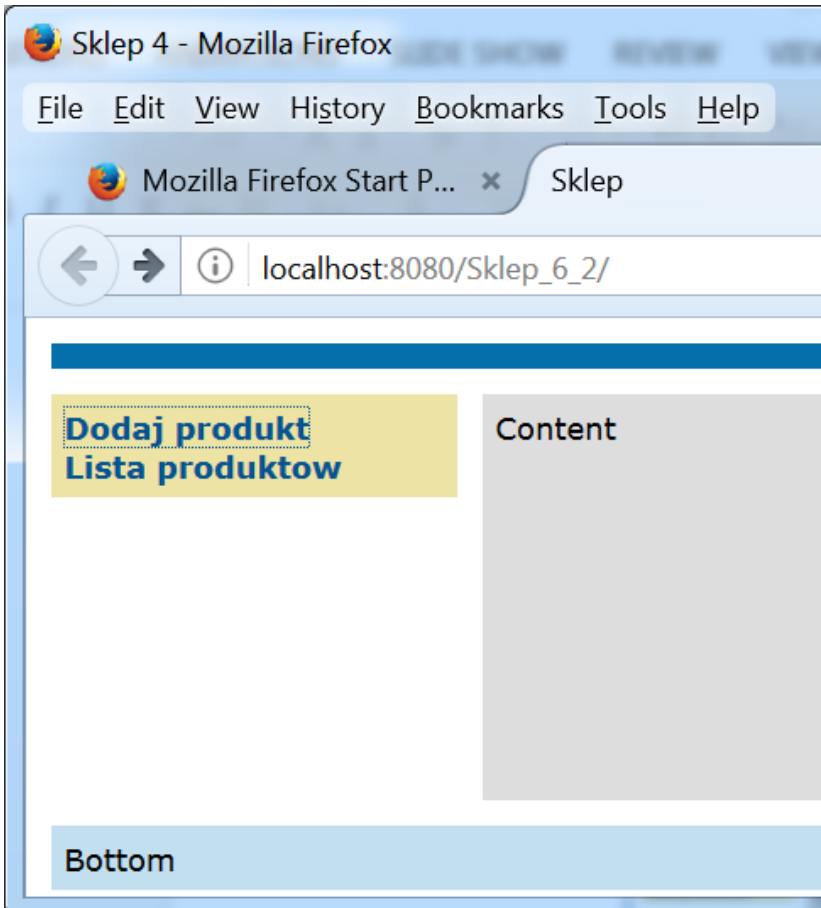
The screenshot shows the NetBeans IDE 8.2 interface. The main editor window displays the `web.xml` file with the following XML content:

```
11 </servlet>
12 <servlet-mapping>
13     <servlet-name>Faces Servlet</servlet-name>
14     <url-pattern>/faces/*</url-pattern>
15 </servlet-mapping>
16 <session-config>
17     <session-timeout>
18         2
19     </session-timeout>
20 </session-config>
21 <welcome-file-list>
22     <welcome-file>faces/index1.xhtml</welcome-file>
23 </welcome-file-list>
```

The `<session-timeout>` element is highlighted, and its value is set to `2`. The left sidebar shows the project structure for `Sklep_6_2`, including `Web Pages`, `WEB-INF`, `resources`, `warstwa_inter`, `index1.xhtml`, and `template.xhtml`. The bottom status bar shows the time `18:14` and the text `INS`.

2.6. Uruchomienie aplikacji opartej na komponencie EJB typu Session, rodzaj Stateful - Clean and Build/Deploy/Run

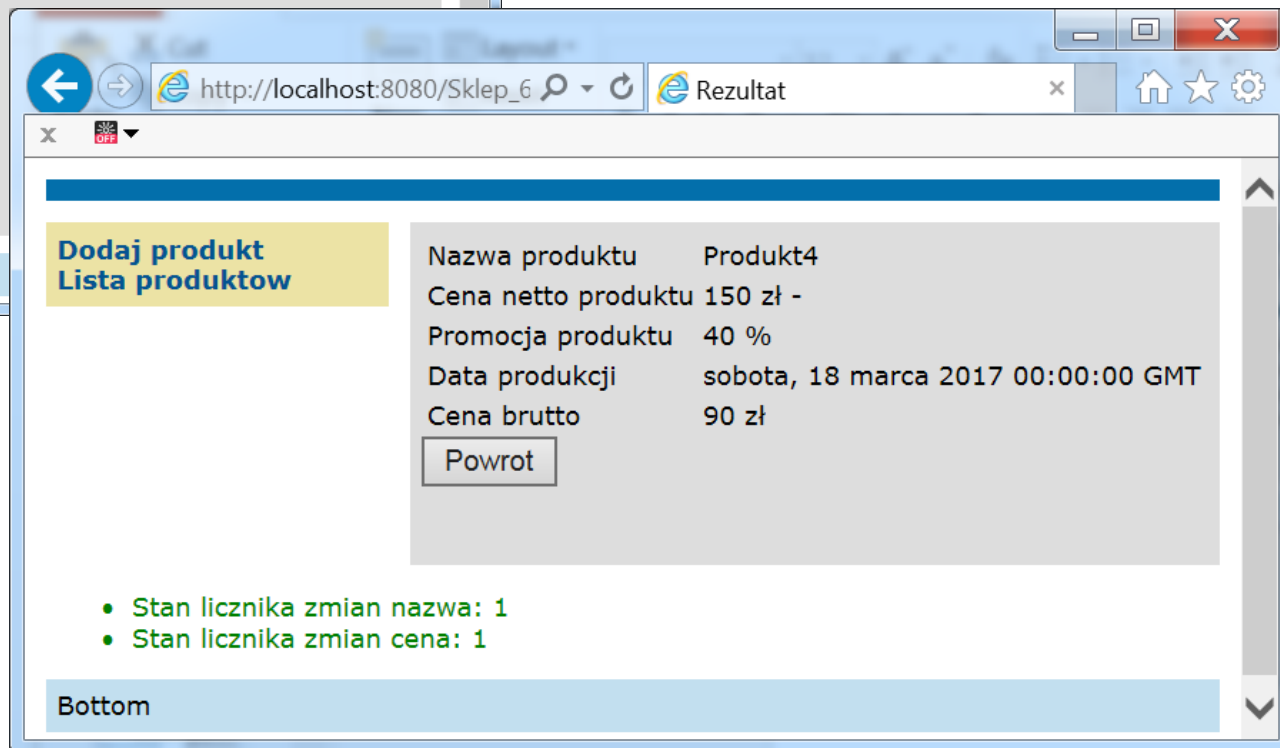
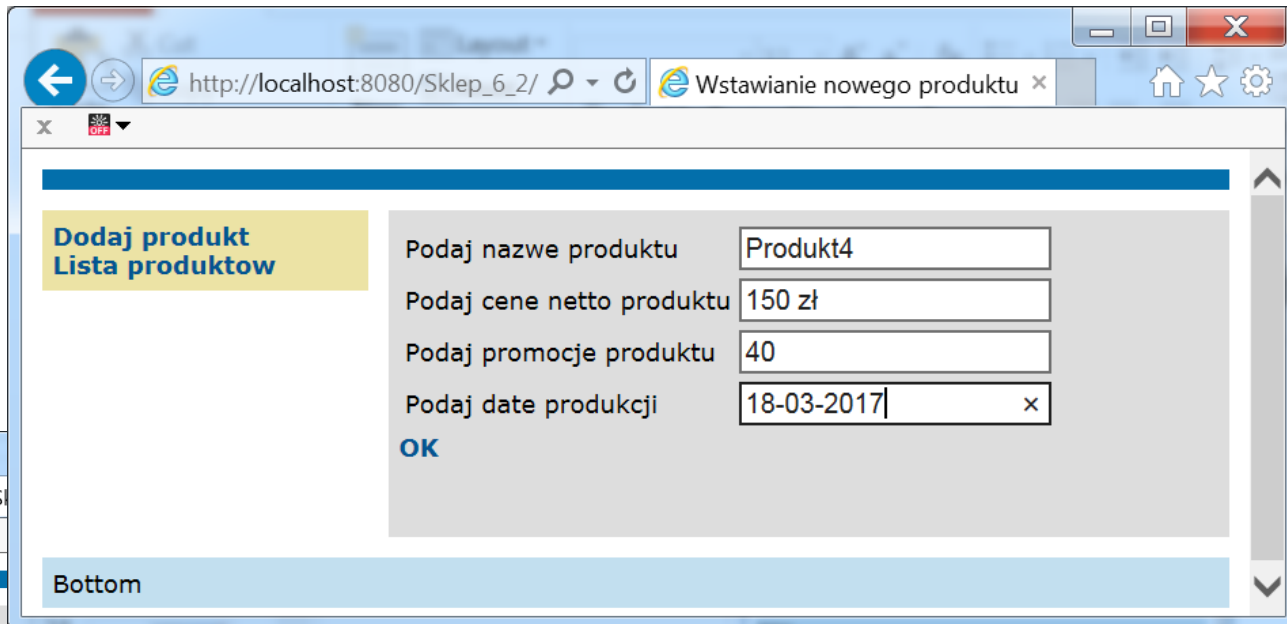
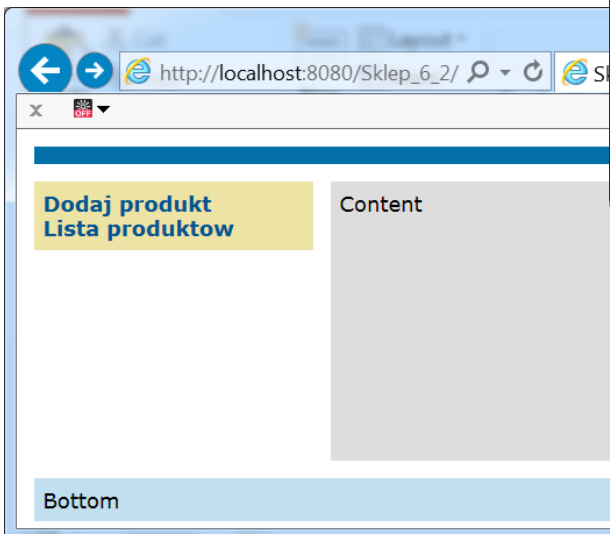
Uruchomienie 1-ej instancji warstwy klienta aplikacji i wprowadzenie danych produktu



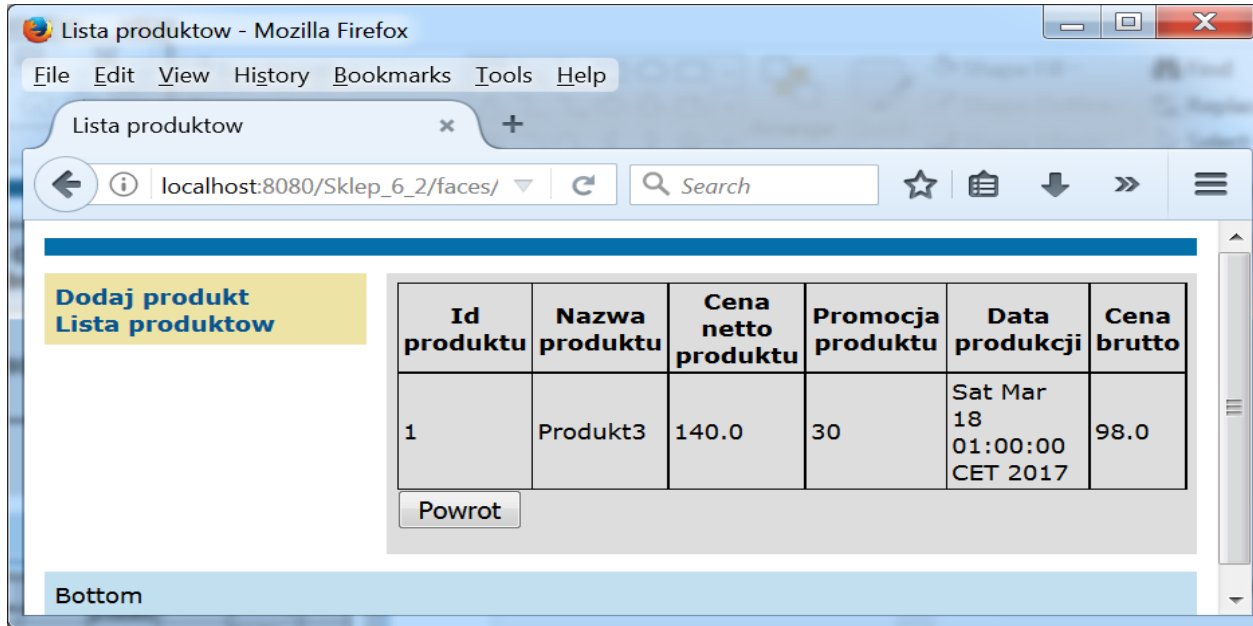
Programowanie komponentowe -
Lab1, Zofia Kruczkiewicz

2.7. Uruchomienie 2-ej instancji warstwy klienta aplikacji i wprowadzenie danych produktu (podobnie jak w p.1.16:

http://localhost:8080/Sklep_6_2)



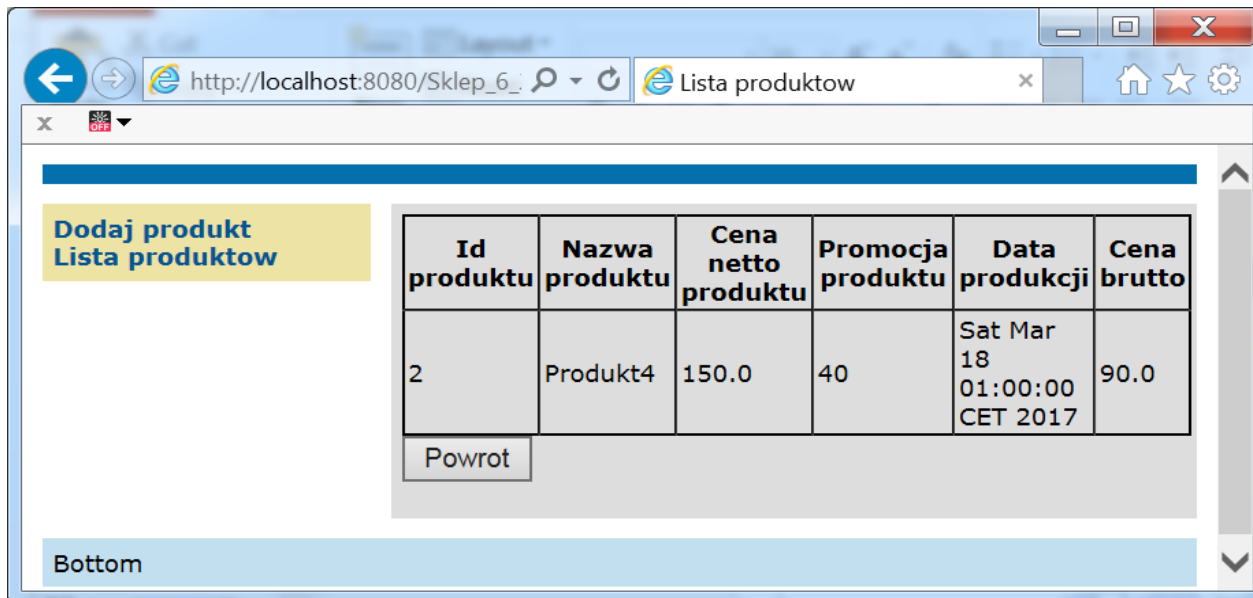
2.8. Po uruchomieniu formularza **Lista produktów** w obu instancjach klientów aplikacji widać, że korzystają **z różnych instancji** komponentu typu **Fasada_warstwy_biznesowej_ejb**, czyli różnych instancji obiektu **Fasada_warstwy_biznesowej**.



The screenshot shows a Mozilla Firefox browser window titled "Lista produktow". The address bar displays "localhost:8080/Sklep_6_2/faces/". On the left, there is a yellow button labeled "Dodaj produkt Lista produktow". The main content area features a table with the following data:

Id produktu	Nazwa produktu	Cena netto produktu	Promocja produktu	Data produkcji	Cena brutto
1	Produkt3	140.0	30	Sat Mar 18 01:00:00 CET 2017	98.0

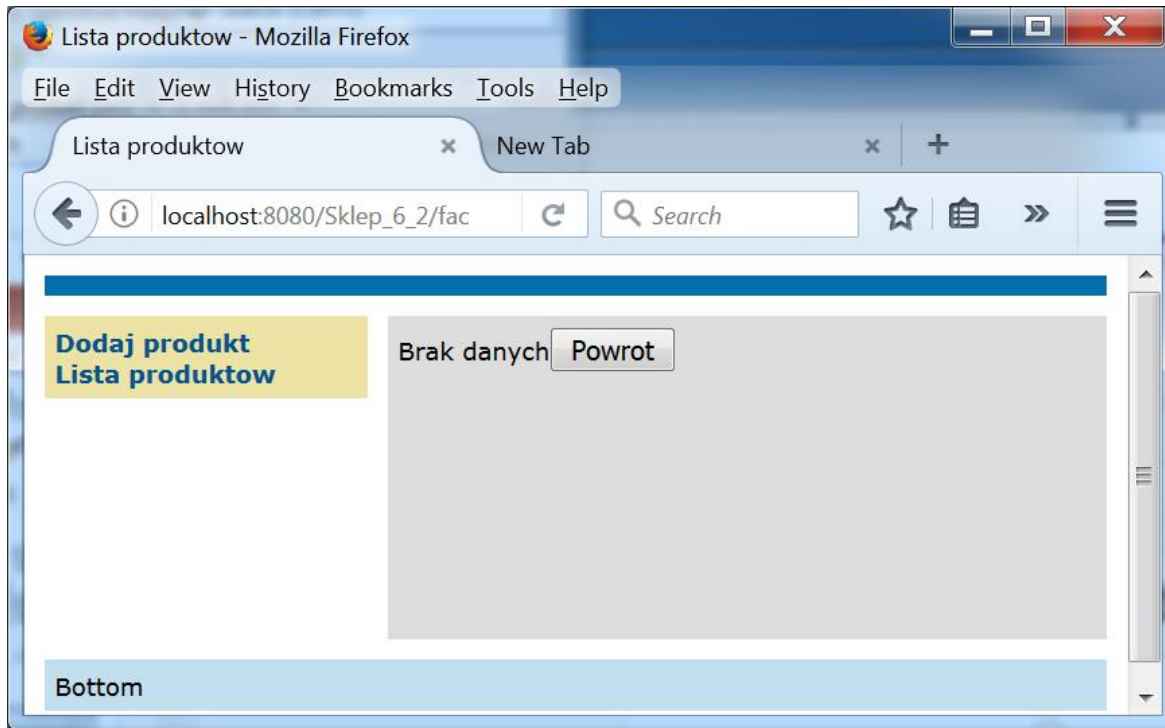
Below the table is a "Powrot" button. The footer of the page contains the text "Bottom".



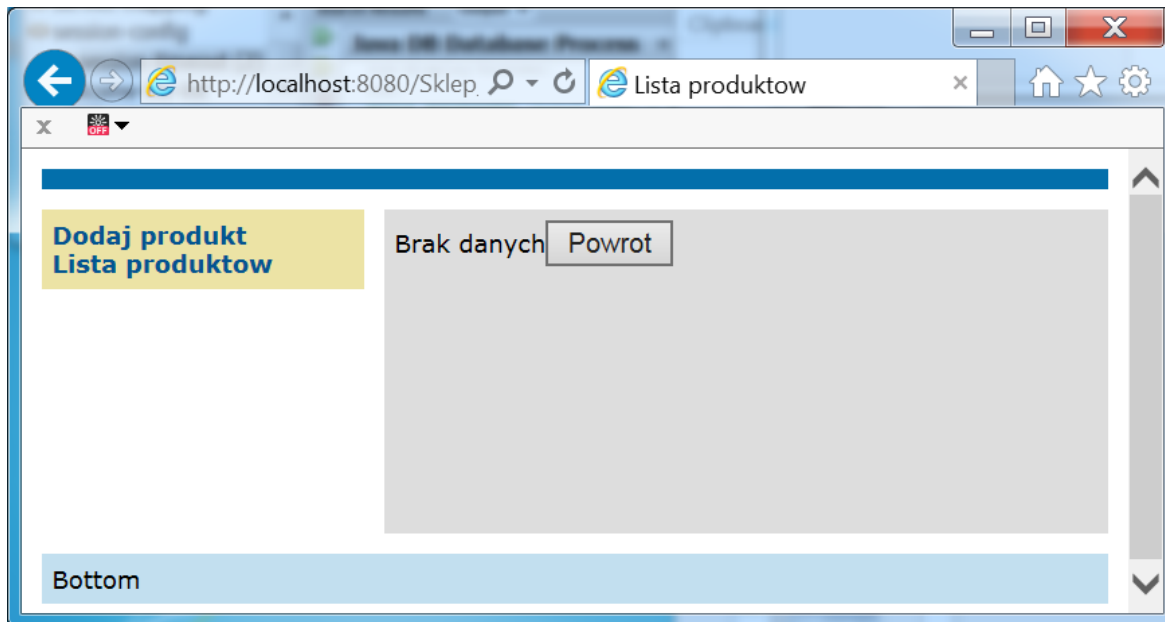
The screenshot shows a web browser window titled "Lista produktow". The address bar displays "http://localhost:8080/Sklep_6_2". On the left, there is a yellow button labeled "Dodaj produkt Lista produktow". The main content area features a table with the following data:

Id produktu	Nazwa produktu	Cena netto produktu	Promocja produktu	Data produkcji	Cena brutto
2	Produkt4	150.0	40	Sat Mar 18 01:00:00 CET 2017	90.0

Below the table is a "Powrot" button. The footer of the page contains the text "Bottom".



2.9. Po upływie 2 min próba wywołania strony Lista produktów za pomocą linku Lista produktów daje rezultat – Brak danych, ponieważ każdy z komponentów typu **Stateful** przeszedł w stan nieaktywny po upływie 2 min i nastąpiła utrata danych.



3. Przygotowanie środowiska platformy Java EE do uruchomienia kolejnej aplikacji – opartej na komponencie EJB typu Session – rodzaj **Singleton**

3.1. Przygotowanie środowiska platformy Java EE do uruchomienia kolejnej aplikacji – **zamknięcie aplikacji Sklep_6_2**: w oknie zakładki Services należy otworzyć pozycję **Servers/GlassFish Server 4.1.1/Applications** i prawym klawiszem wybrać pozycję Sklep_6_1. Z listy wybrać pozycję **Undeploy**.

3.2. Wykonanie kopii Sklep_6_2 jako programu Sklep_6_3. Po wykonaniu kopiowania należy zamknąć projekt Sklep_6_1. Następnie, należy zmodyfikować adnotację **@Stateful** na adnotację **@Singleton** – aby usunąć błędy, należy kliknąć prawym klawiszem na powierzchnię edytora i z listy wybrać pozycję **Fix Imports**

3.3. W pliku Managed_produkty należy zmienić adnotację **@SessionScoped** na adnotację **@RequestScoped** – aby usunąć błędy, należy kliknąć prawym klawiszem na powierzchnię edytora i z listy wybrać pozycję **Fix Imports**.

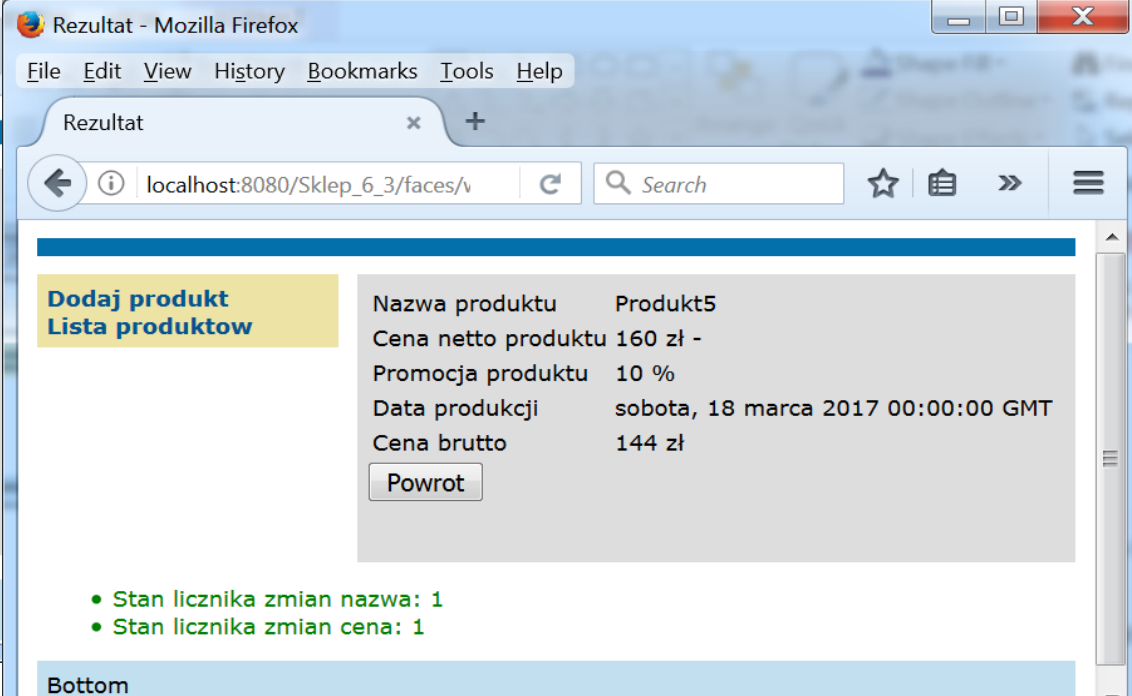
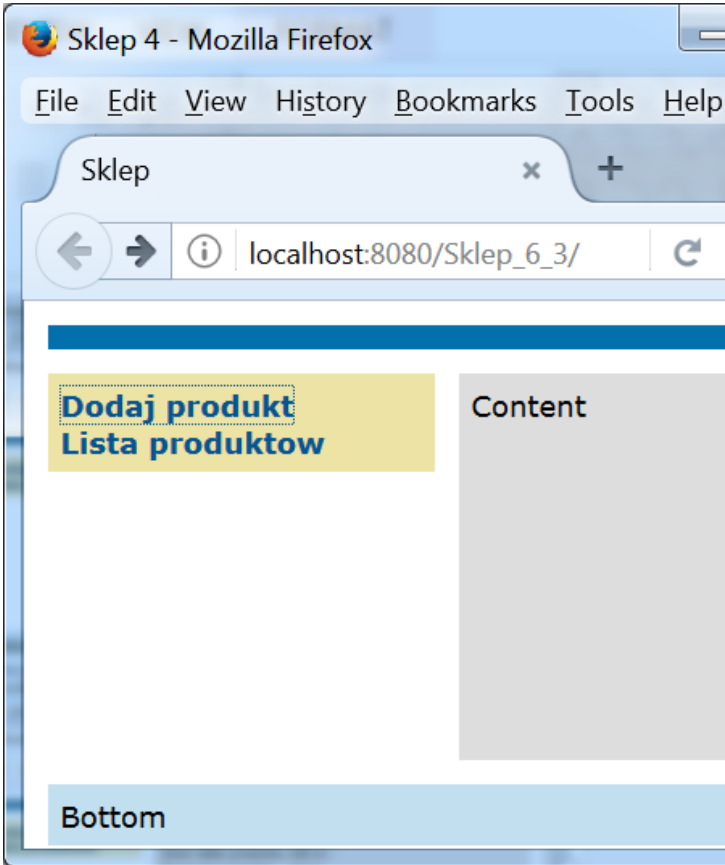
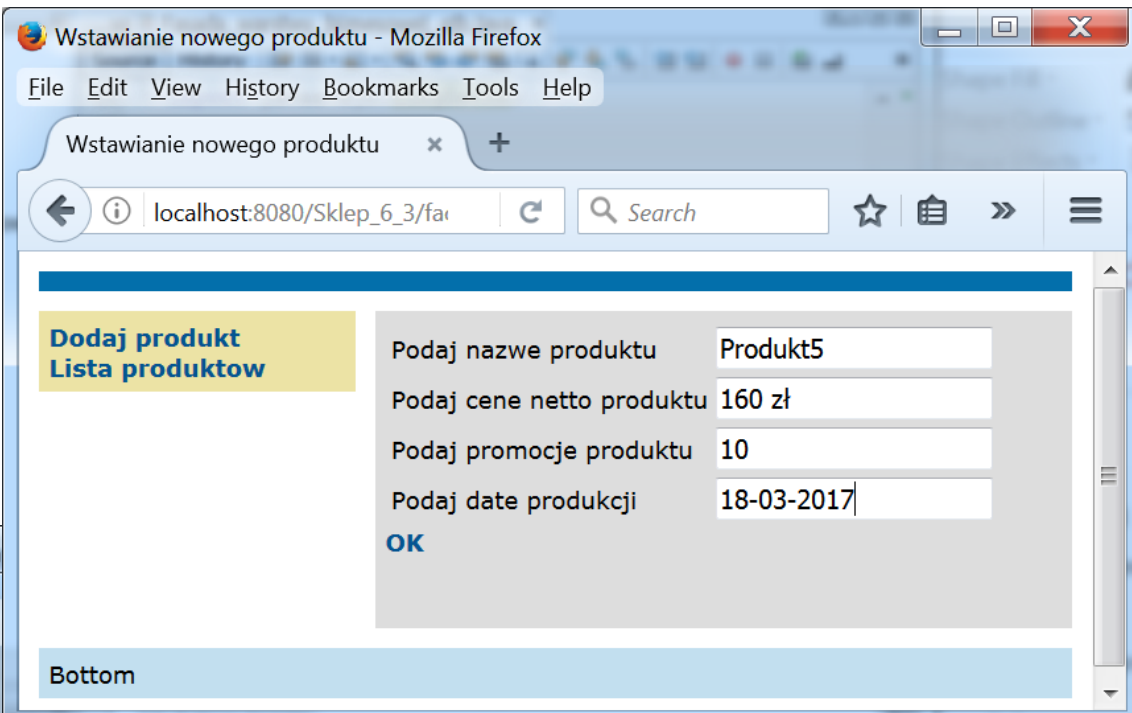
3.3. Przygotowanie środowiska platformy Java EE do uruchomienia kolejnej aplikacji – opartej na komponencie EJB typu Session – rodzaj **Singleton**

The screenshot displays the NetBeans IDE 8.2 interface. The main window shows the source code for the file `Fasada_warstwy_biznesowej_ejb.java`. The code is as follows:

```
1 package warstwa_biznesowa;
2
3 import java.util.ArrayList;
4 import java.util.Date;
5 import javax.ejb.Singleton;
6
7 @Singleton
8 public class Fasada_warstwy_biznesowej_ejb {
9
10     Fasada_warstwy_biznesowej fasada = new Fasada_warstwy_biznesowej();
11
12     public void utworz_produkt(String dane[], Date data) {
13         fasada.utworz_produkt(dane, data);
14     }
15     public String[] dane_produktu() {
16         return fasada.dane_produktu();
17     }
18     public ArrayList<ArrayList<String>> items() {
19         return fasada.items();
20     }
21 }
```

The IDE interface includes a Project Explorer on the left showing the project structure for `Sklep_6_3`, with `Fasada_warstwy_biznesowej_ejb.java` selected under the `warstwa_biznesowa` package. The bottom status bar shows the system tray with `Java DB Database Process`, `GlassFish Server 4.1.1`, and `Sklep_6_3 (run)`. The system clock indicates 18:1 and the user is identified as INS.

3.4. Uruchomienie aplikacji opartej na **komponencie EJB** typu **Session**, rodzaj **Singleton** - Clean and Build/Deploy/Run **Uruchomienie 1-ej instancji warstwy klienta aplikacji i wprowadzenie danych produktu**



3.5. Uruchomienie 2-ej instancji warstwy klienta aplikacji i wprowadzenie danych produktu produktu (podobnie jak w p.1.16: http://localhost:8080/Sklep_6_3)

Wstawianie nowego produktu

Dodaj produkt
Lista produktów

Podaj nazwe produktu

Podaj cene netto produktu

Podaj promocje produktu

Podaj date produkcji

OK

Bottom

Dodaj produkt
Lista produktów

Content

Bottom

Rezultat

Dodaj produkt
Lista produktów

Nazwa produktu Produkt6

Cena netto produktu 130 zł -

Promocja produktu 20 %

Data produkcji sobota, 18 marca 2017 00:00:00 GMT

Cena brutto 104 zł

- Stan licznika zmian nazwa: 1
- Stan licznika zmian cena: 1

Bottom

3.6. Po uruchomieniu formularza **Lista produktów** w obu instancjach klientów aplikacji widać, że korzystają **z tej samej instancji** komponentu typu **Fasada_warstwy_biznesowej_ejb**, czyli tej samej instancji obiektu **Fasada_warstwy_biznesowej**.

The image shows two screenshots of a web browser displaying a product list page. The browser window title is "Lista produktow - Mozilla Firefox". The address bar shows "localhost:8080/Sklep_6_3/faces/warstw". The page content includes a yellow button labeled "Dodaj produkt Lista produktow" and a table with the following data:

Id produktu	Nazwa produktu	Cena netto produktu	Promocja produktu	Data produkcji	Cena brutto
1	Produkt5	160.0	10	Sat Mar 18 01:00:00 CET 2017	144.0
2	Produkt6	130.0	20	Sat Mar 18 01:00:00 CET 2017	104.0

Below the table is a "Powrot" button. The browser window also shows a "Bottom" label at the bottom.

The second screenshot shows the same page, but the address bar shows "http://localhost:8080/Sklep_6_3/faces/warst" and the browser title is "Lista produktow". The table data is identical to the first screenshot, demonstrating that both instances are connected to the same data source.

4. Należy wykonać eksperymenty dotyczące zmiany czasu sesji z domyślnych 30 min na 2 min w pliku web.xml, w projektach Sklep_6_1 i Sklep_6_3

4.1. Wprowadzić dane produktu. Po upływie ponad 2 min wywołać stronę **Lista produktów** za pomocą linku z lewej strony formularza

4.2. opisać skutki eksperymentów.