

**Instrukcja tworzenia aplikacji EE na podstawie aplikacji z dostępem do bazy danych, prezentowanej na zajęciach lab.5 z PIO – umożliwiająca przez sieć dostęp wielu użytkownikom.**

**Projektowanie i wdrażanie systemów informatycznych**

Przekształcenie projektu typu Java Application działającego na platformie Javy Standard Edition do typu działającego na platformie Javy Enterprise Edition

– wg

<http://netbeans.org/kb/docs/javaee/entappclient.html>

Zofia Kruczkiewicz

1. Czynności przygotowujące aplikację działającą na platformie Java SE Biblioteka5 (należy ją pobrać z załącznika z p.1)

# 1. Utworzenie pustej bazy danych – w okienku Services należy prawym klawiszem myszy wybrać **Databases\Java DB\Create Database**

The screenshot shows the NetBeans IDE 7.2 interface. The 'Services' window is active, displaying a tree view of services. A context menu is open over the 'Java DB' service, with the 'Create Database...' option selected. The main editor area shows a design view for a database with several text input fields and buttons.

**Services Window:**

- Databases
  - Java DB (Context Menu: Start Server, Stop Server, **Create Database...**, Properties...)
  - Drive
  - jdbc: [Jsf]pa on JSFJPA
  - jdbc: [app on APP]
  - jdbc: e1 [app1 on APP1]
  - jdbc: [travel on TRAVEL]
  - jdbc:derby://localhost:1527/vir [vir on VIR]
  - jdbc:derby://localhost:1527/web\_jpa [web\_jpa on WEB\_JPA]
- Web Services
- Servers
- Android Devices
- Maven Repositories
- Cloud
- Hudson Builders
- Issue Trackers
- C/C++ Build Hosts

**Database Design Form:**

Use the Connection Mode button (in the toolbar) to establish a connection between components. x

Tytuł książki

Nazwisko autora książki

Imię autora książki

ISBN tytułu książki

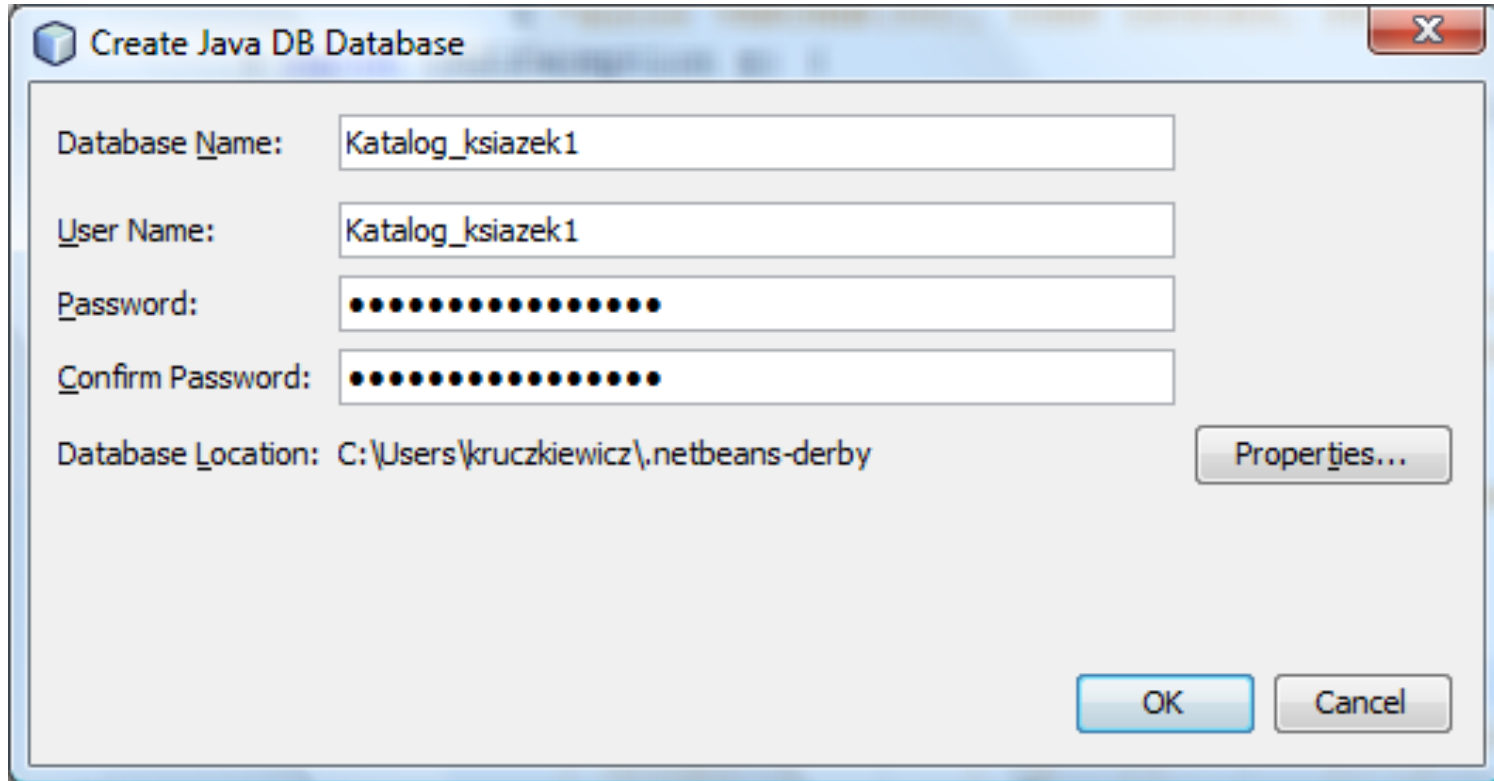
Wydawnictwo książki

Numer książki

Zapisz tytuł      Zapisz książkę      Wyświetl tytuły      Wyświetl książki

Tytuły do bazy      Książki do bazy      Tytuły z bazy      Książki z bazy

## 1.1. Wykonanie pustej bazy danych Katalog\_książek



Create Java DB Database

Database Name: Katalog\_ksiazek1

User Name: Katalog\_ksiazek1

Password: ●●●●●●●●●●●●●●●●

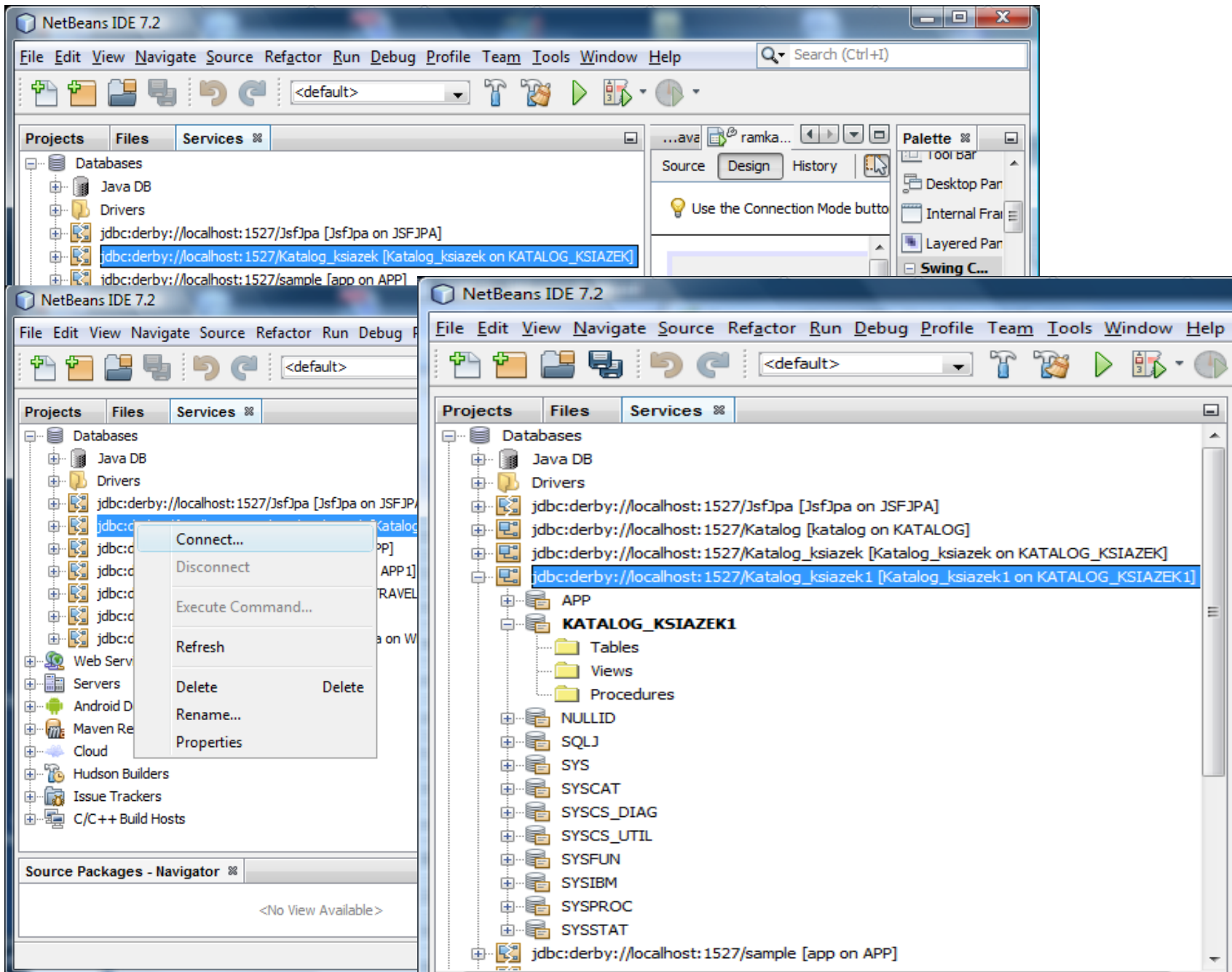
Confirm Password: ●●●●●●●●●●●●●●●●

Database Location: C:\Users\kruczkiewicz\.netbeans-derby

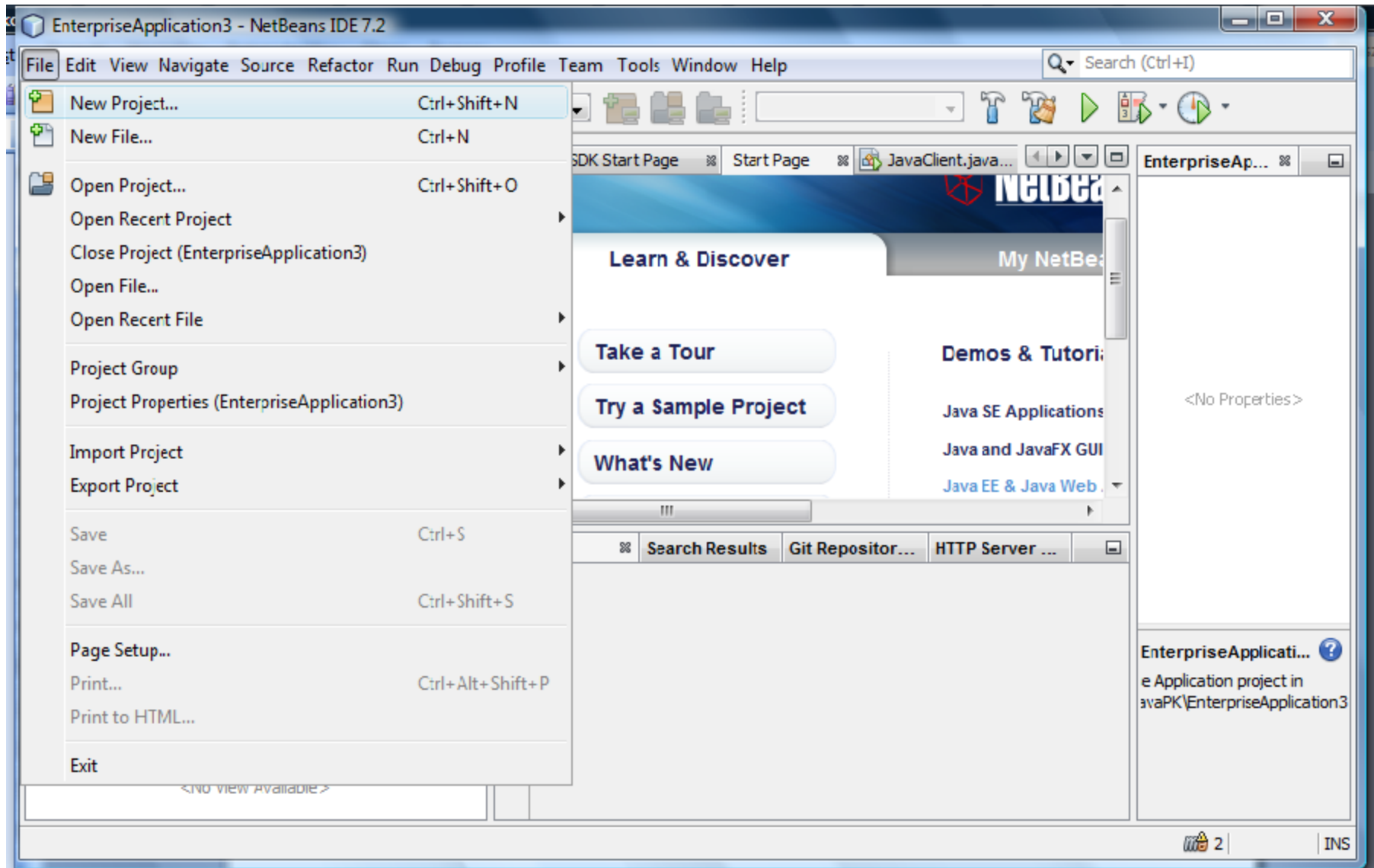
Properties...

OK Cancel

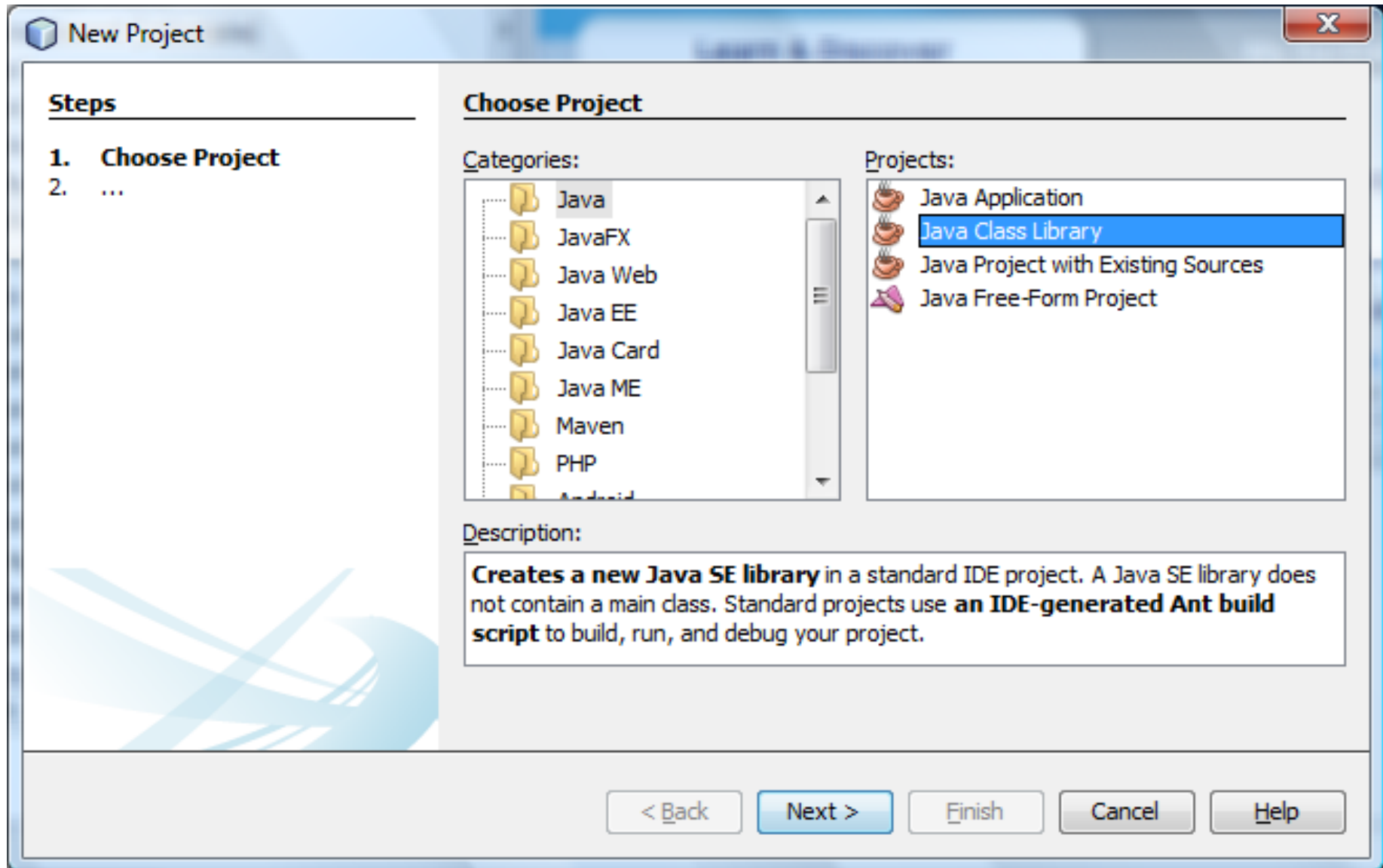
## 1.2. Połączenie z pustą bazą danych



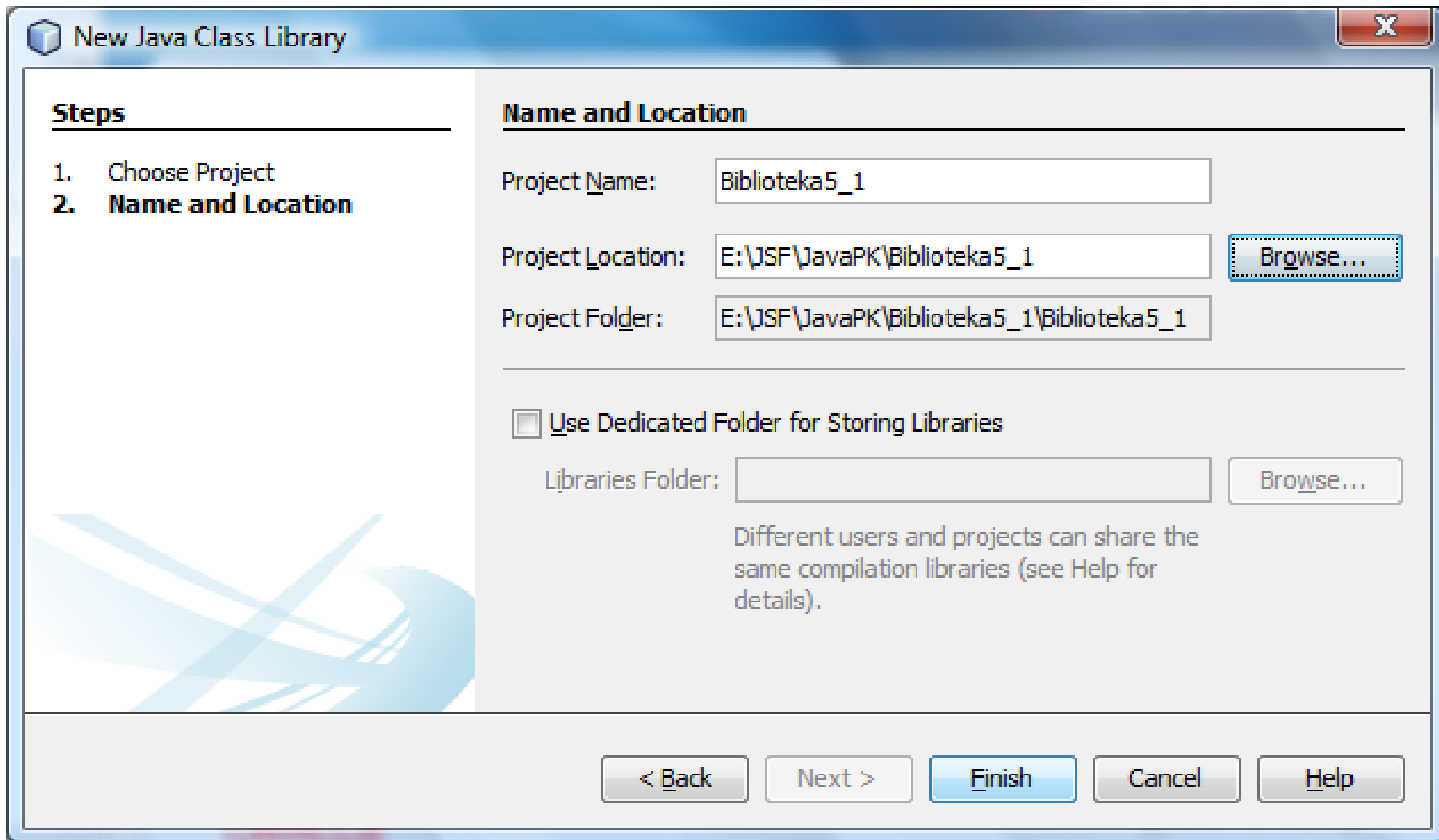
### 1.3. Wykonanie projektu typu **Java Class Library** – **File/New Project/** Wykonanie projektu typu JavaClassLibrary o nazwie Biblioteka5\_1, zawierającego warstwę biznesową oraz integracji



## 1.4. Wykonanie projektu typu **Java Class Library** – **File/New Project/Java/Java Class Library** i **Next**

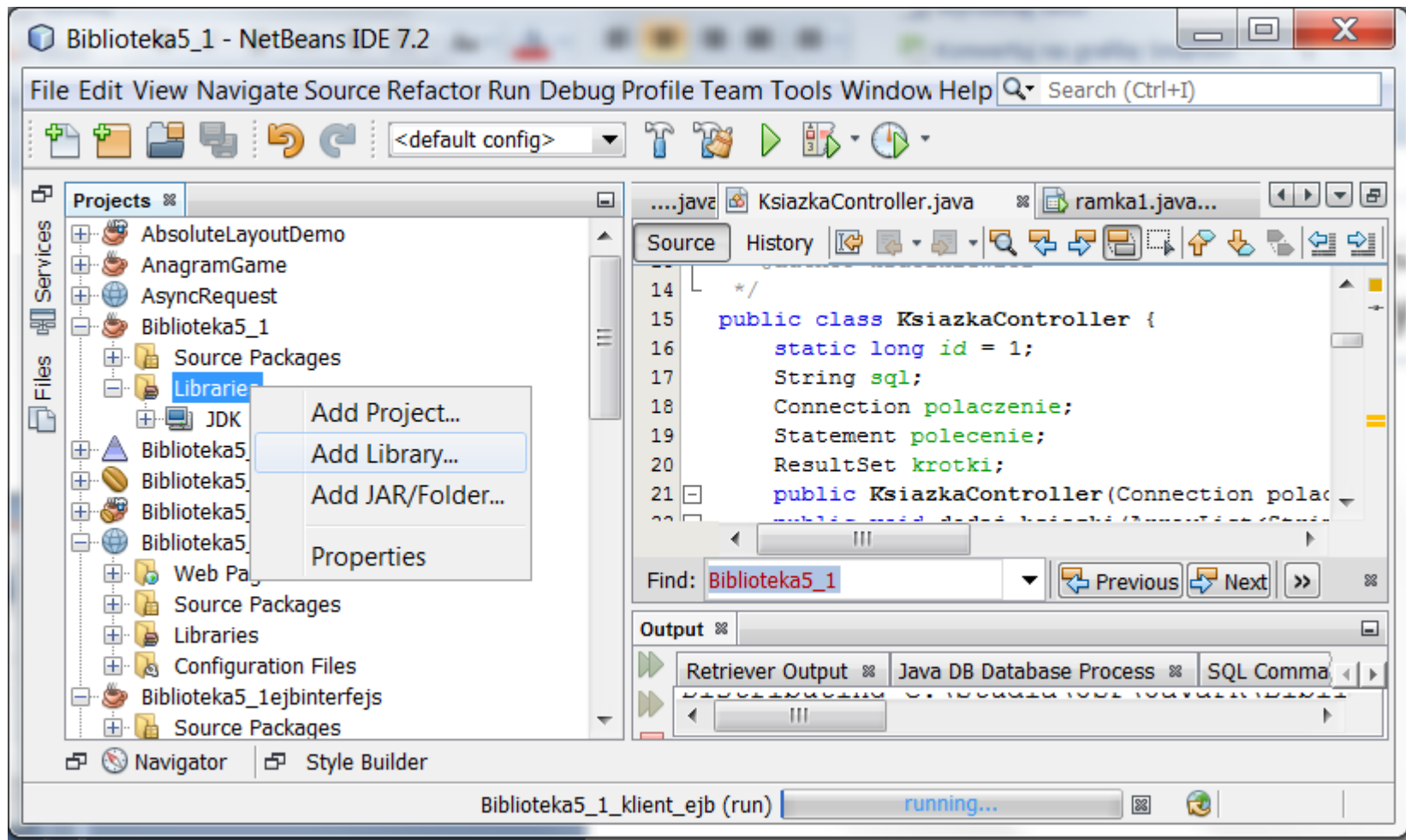


## 1.5. Nadanie nazwy **Biblioteka5\_1** utworzonemu projektowi w polu **Project Name**

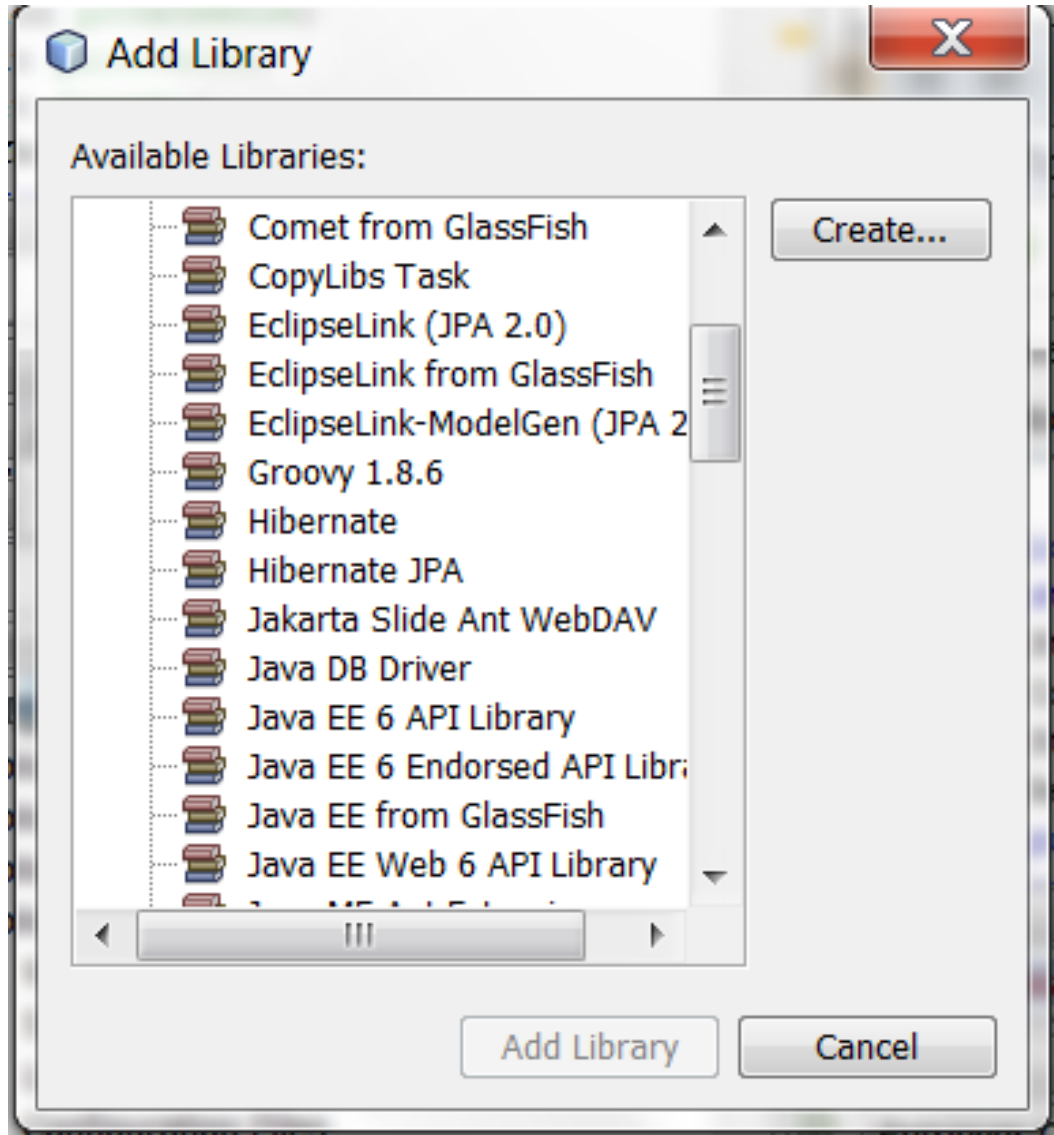




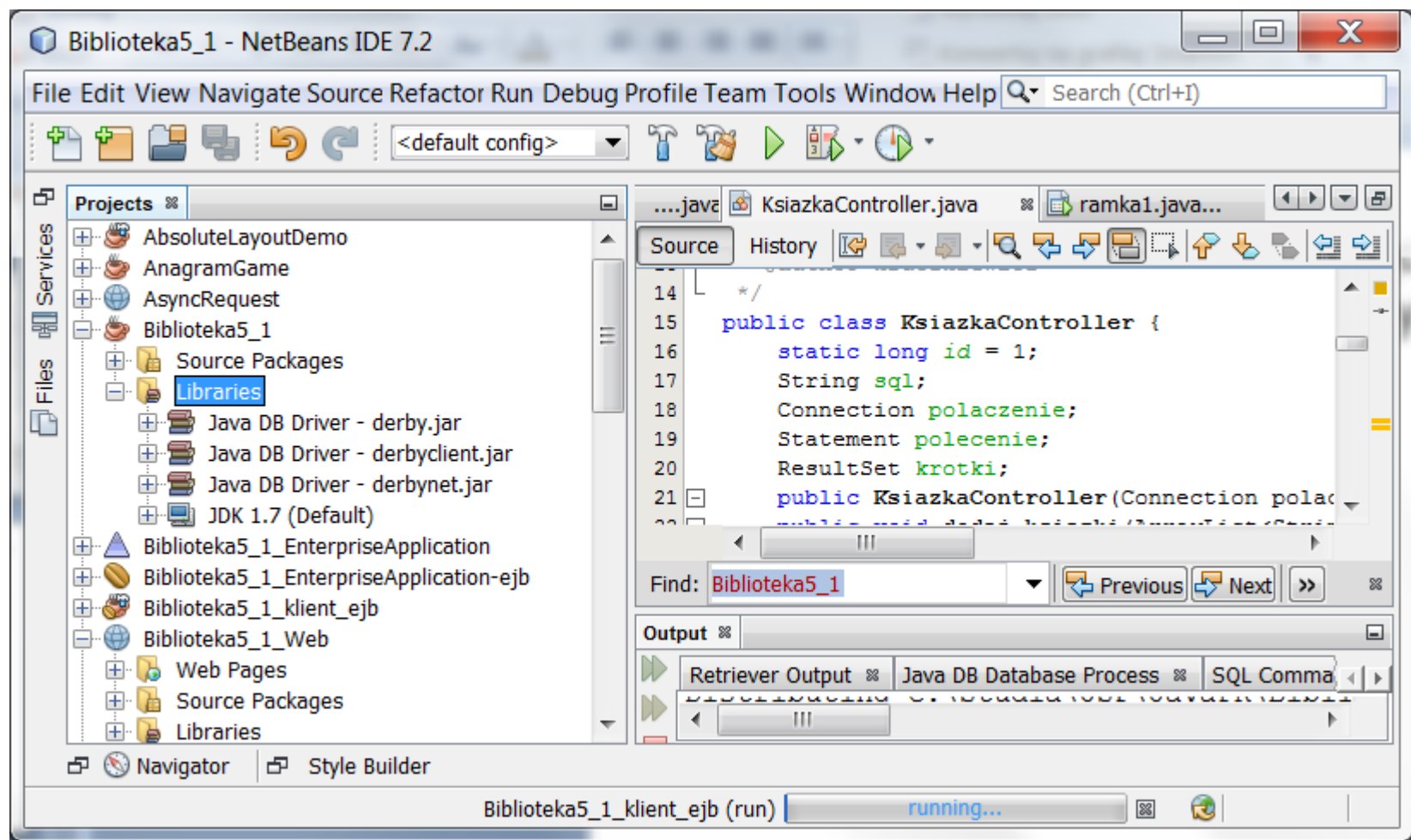
- 1.6. Dodanie biblioteki zawierającej definicję sterowników typu jdbc dla bazy danych typu Derby – w zakładce Projects należy prawym klawiszem myszy wybrać nazwę folderu **Libraries** w projekcie **Biblioteka5\_1** i wybrać z listy pozycję **Add Library**



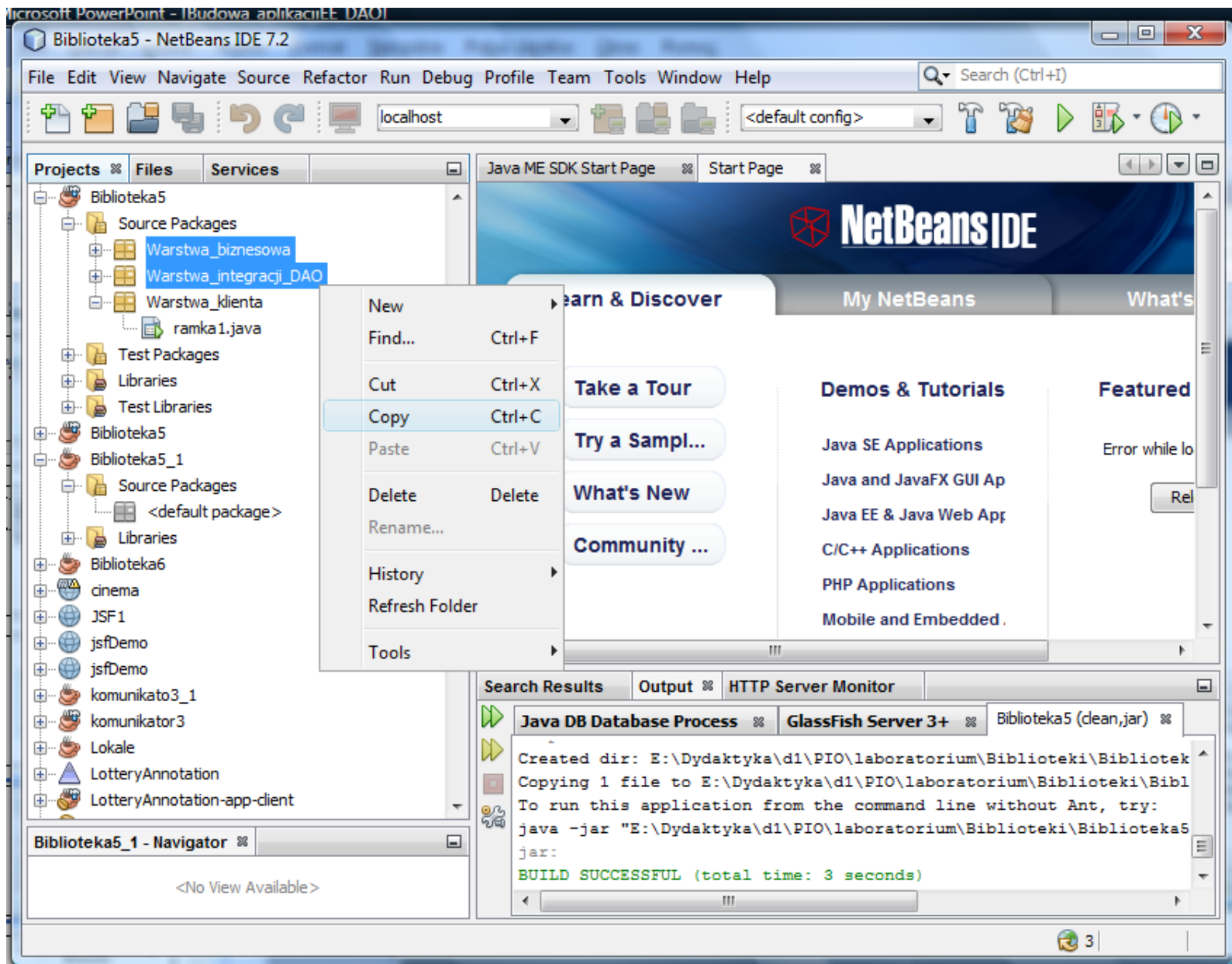
## 1.7. Dodanie biblioteki zawierającej definicję sterowników typu jdbc dla bazy danych typu Derby – Java DB Driver



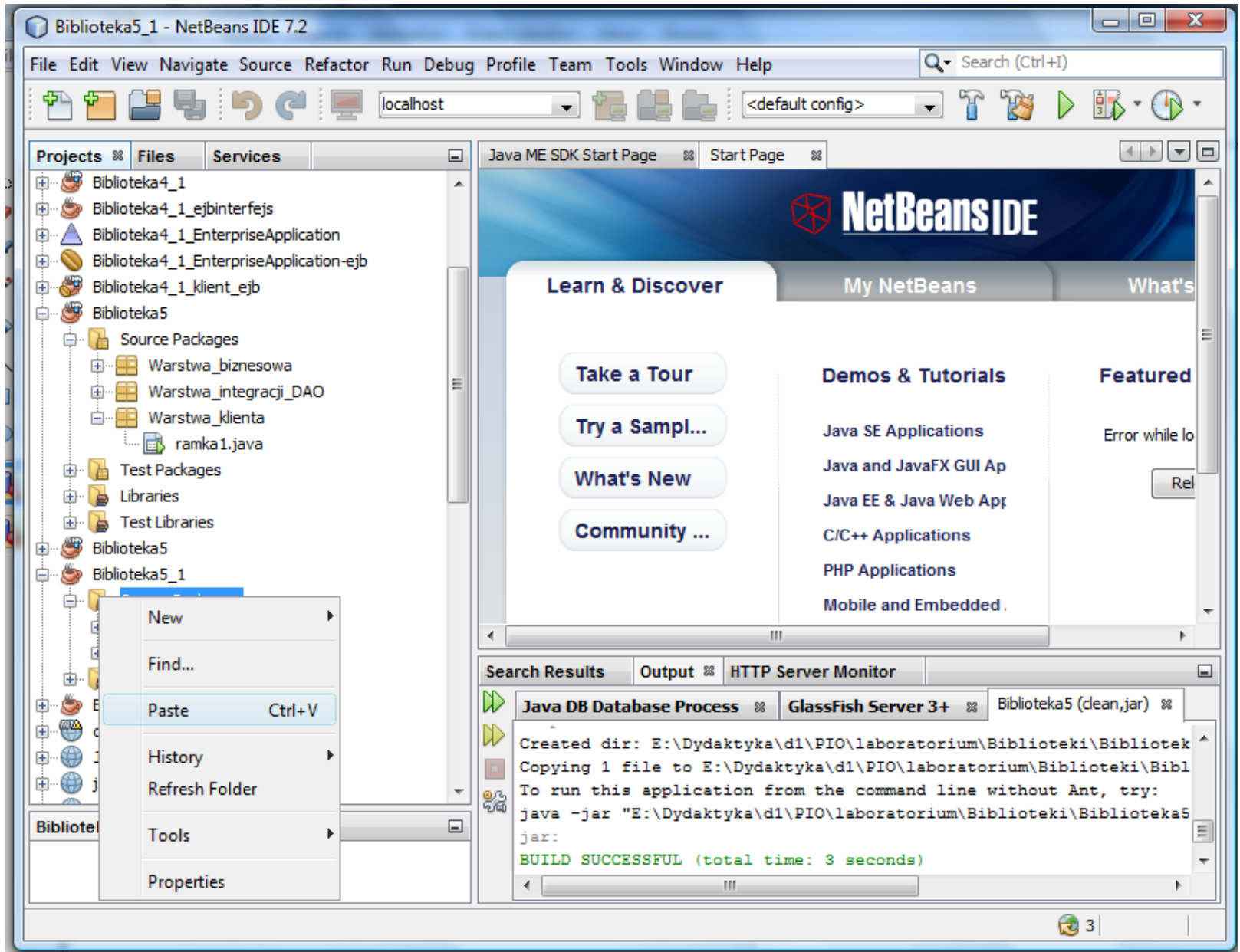
## 1.8. Dodanie biblioteki zawierającej definicję sterowników typu jdbc dla bazy danych typu Derby



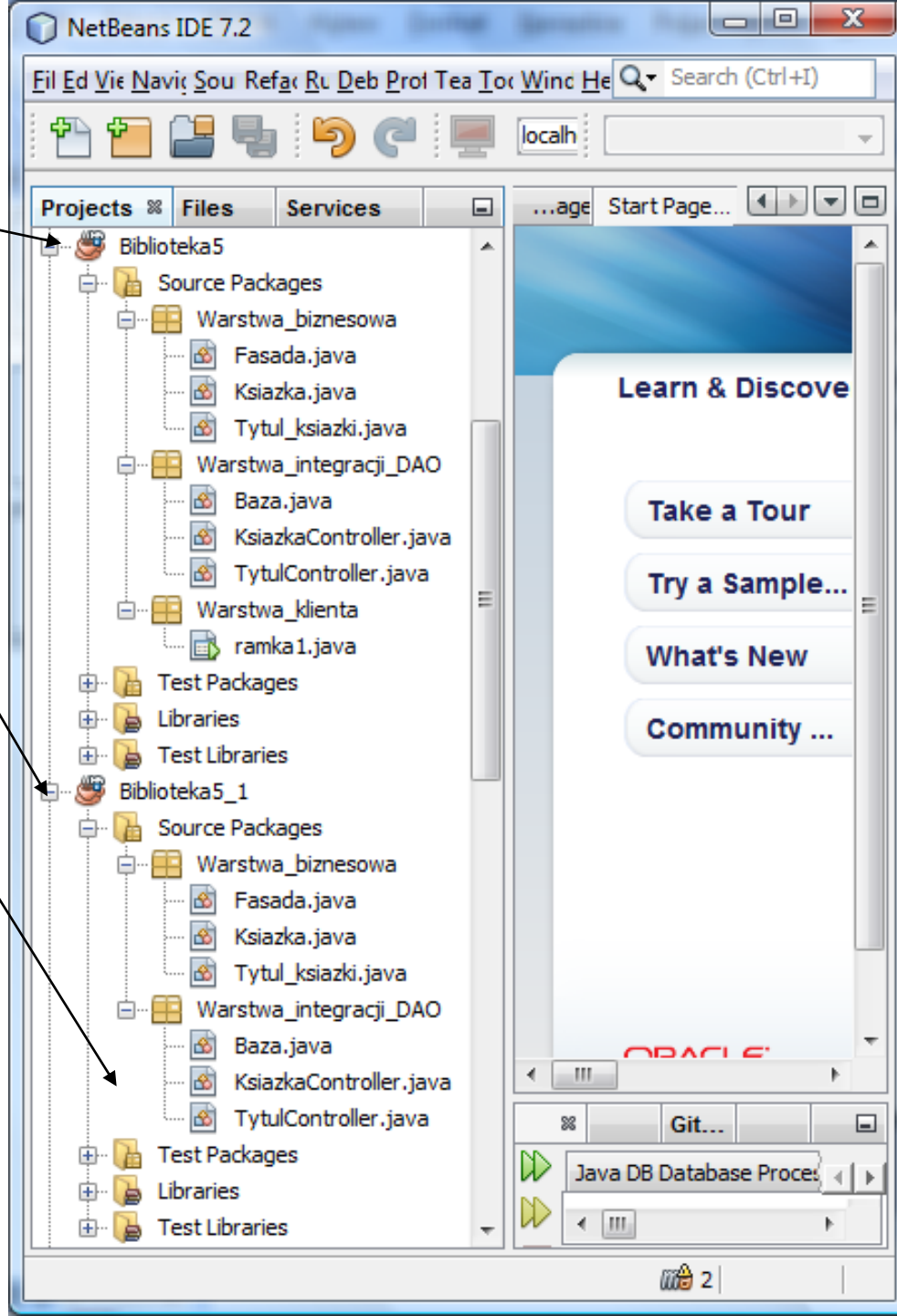
## 1.9. Wykonanie kopii pakietów **Warstwa\_biznesowa** oraz **Warstwa\_integracji\_DAO** z projektu **Biblioteka5**



## 1.10. Wklejenie skopiowanych pakietów **Warstwa\_biznesowa** oraz **Warstwa\_integracji\_DAO** do projektu **Biblioteka5\_1** typu **JavaClassLibrary**



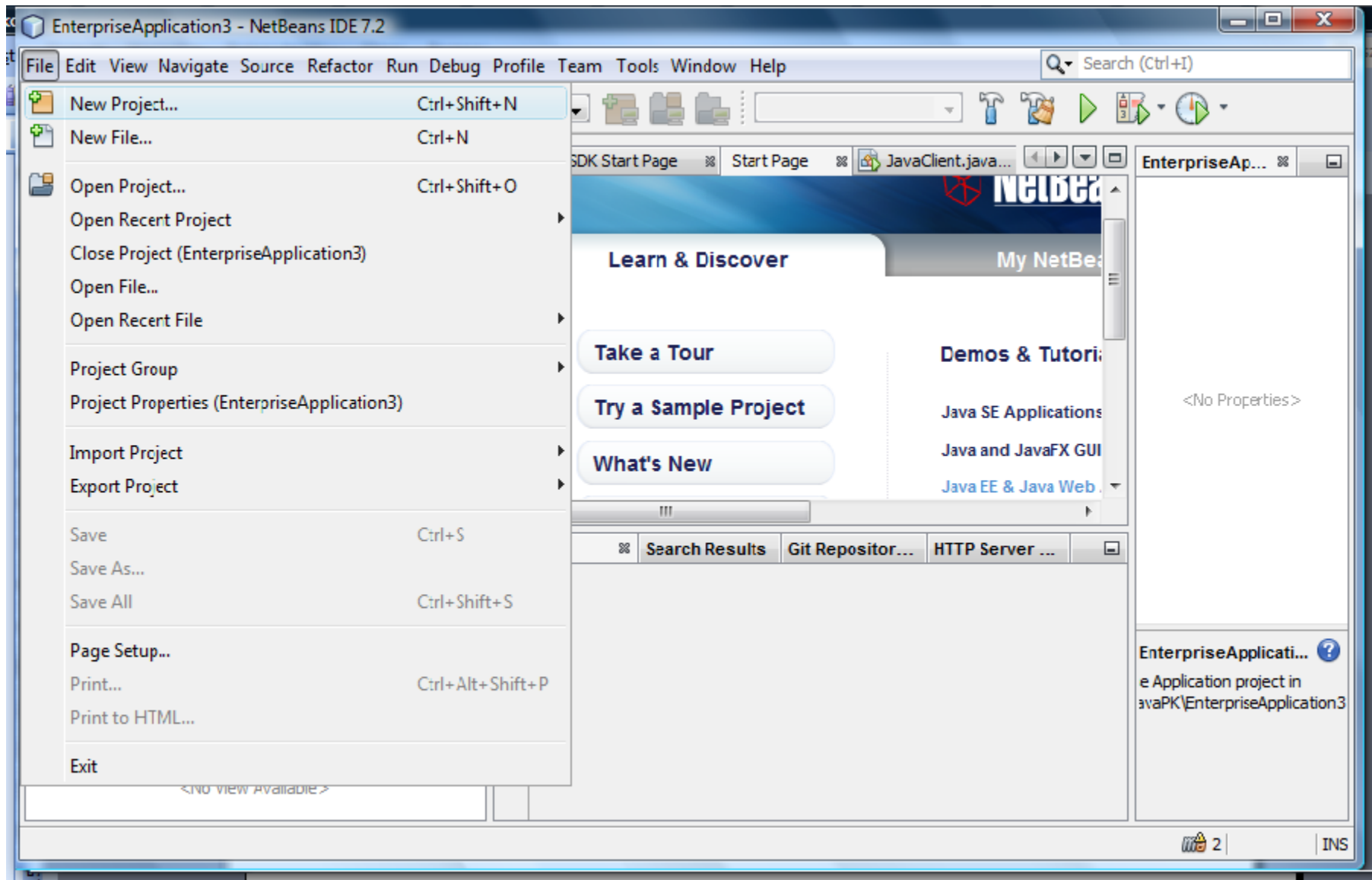
## 1.11. Rezultat kopiowania



## 2. Przygotowanie interfejsu fasady usług biznesowych

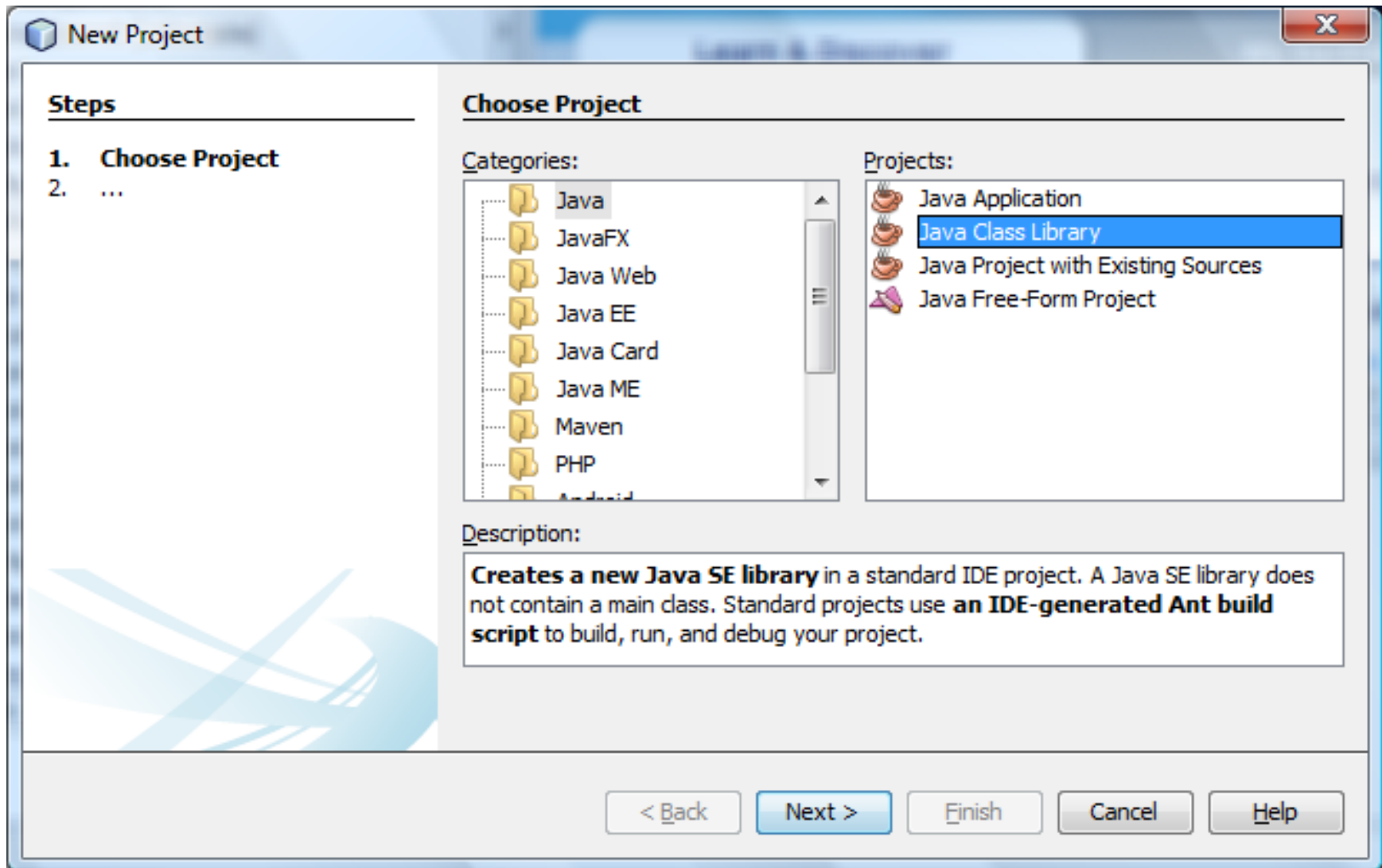


# (1) Wykonanie projektu typu **Java Class Library** – **File/New Project/**

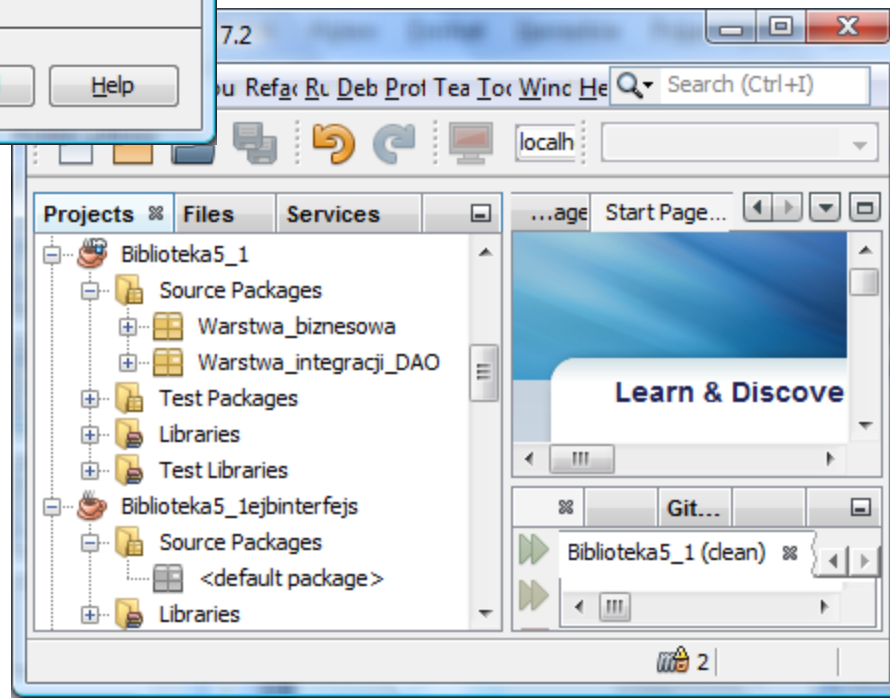
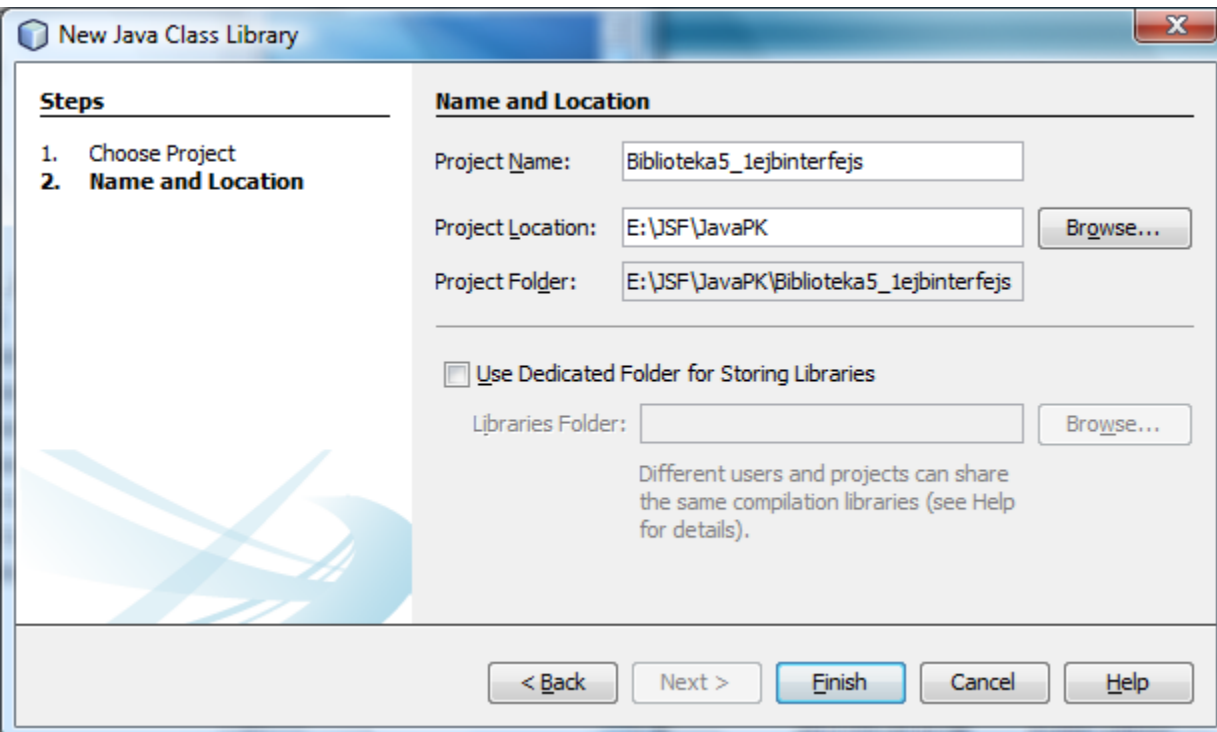




(2) Wykonanie projektu typu **Java Class Library** – **File/New Project/Java/Java Class Library** i **Next**

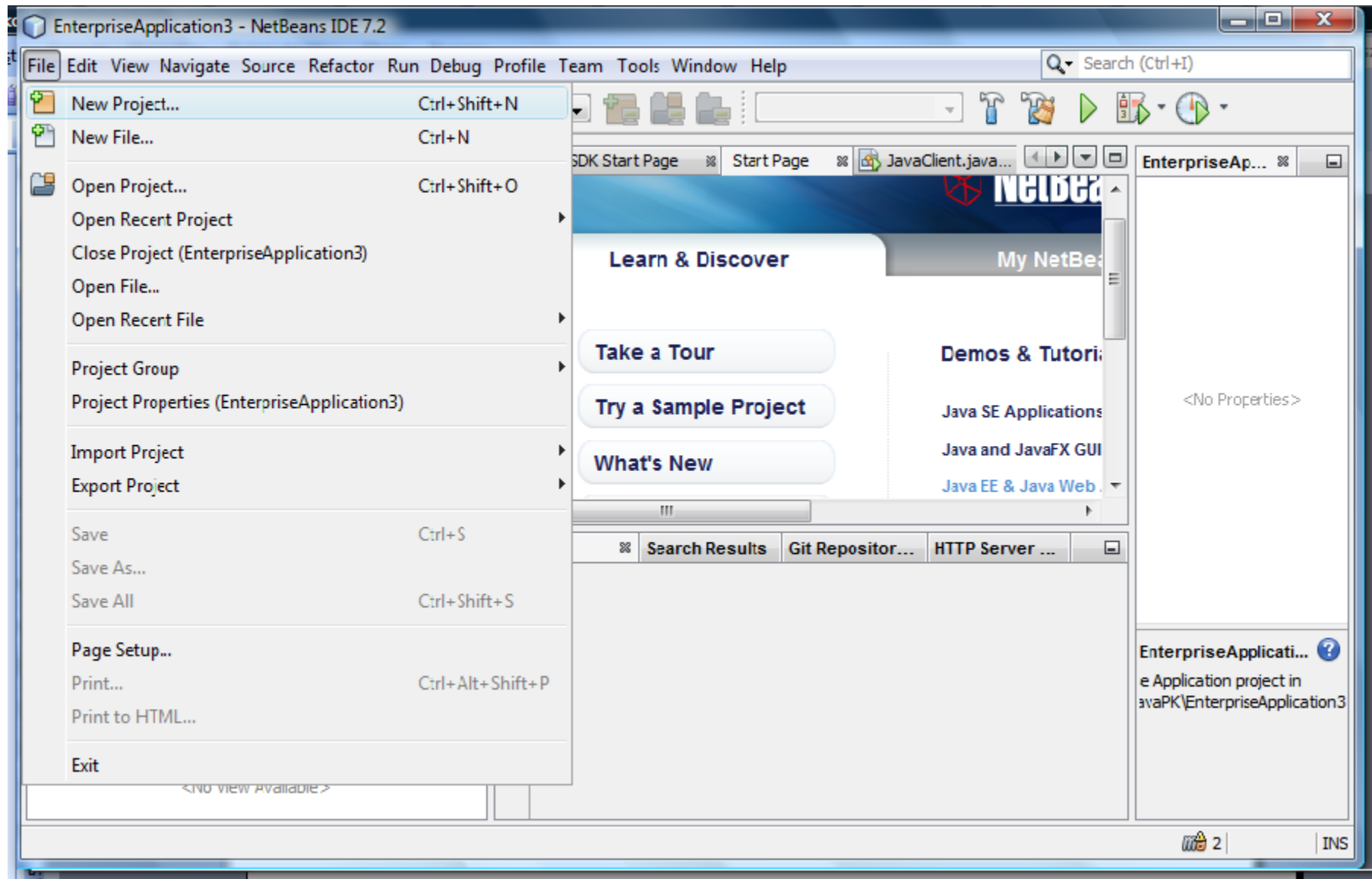


(3) Wykonanie projektu typu **Java Class Library** – **File/New Project/Java/Java Class Library** o nazwie **Biblioteka5\_1ejbinterfejs** podanej w polu **Project Name**

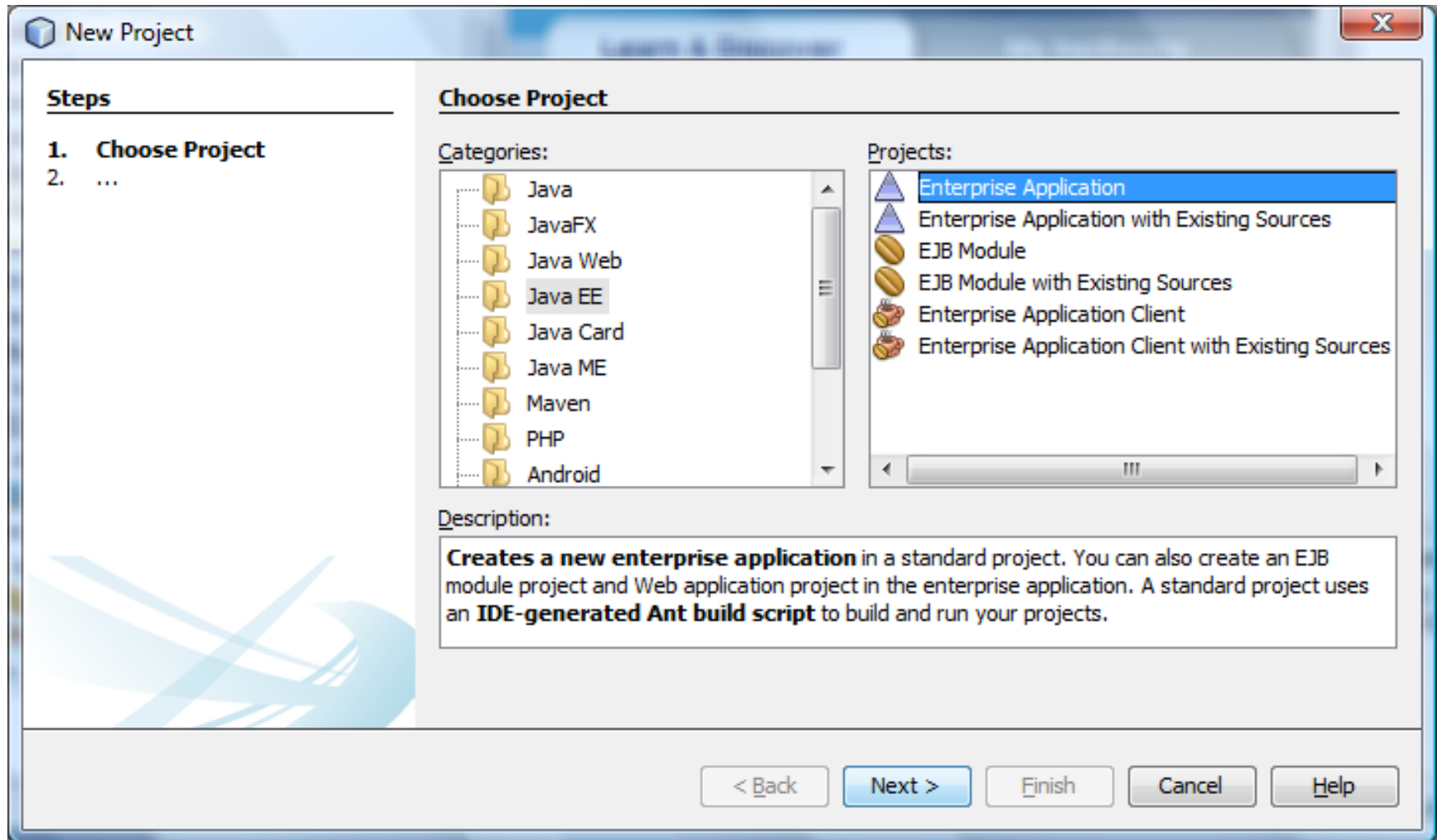


# 3. Wykonanie aplikacji typu Enterprise Edition Java EE 6

# (1) Utworzenie aplikacji typu Enterprise – **File/New Project**



## (2) Utworzenie aplikacji typu Enterprise – **File/New Project/JavaEE/Enterprise Application**



(3) Utworzenie aplikacji typu Enterprise - **File/New Project/JavaEE/Enterprise Application** o nazwie **Biblioteka5\_1\_EnterpriseApplication** podanej w polu **Project Name**

**New Enterprise Application**

**Steps**

1. Choose Project
- 2. Name and Location**
3. Server and Settings

**Name and Location**

Project Name: Biblioteka5\_1\_EnterpriseApplication

Project Location: E:\JSF\JavaPK

Project Folder: E:\JSF\JavaPK\Biblioteka5\_1\_EnterpriseApplication

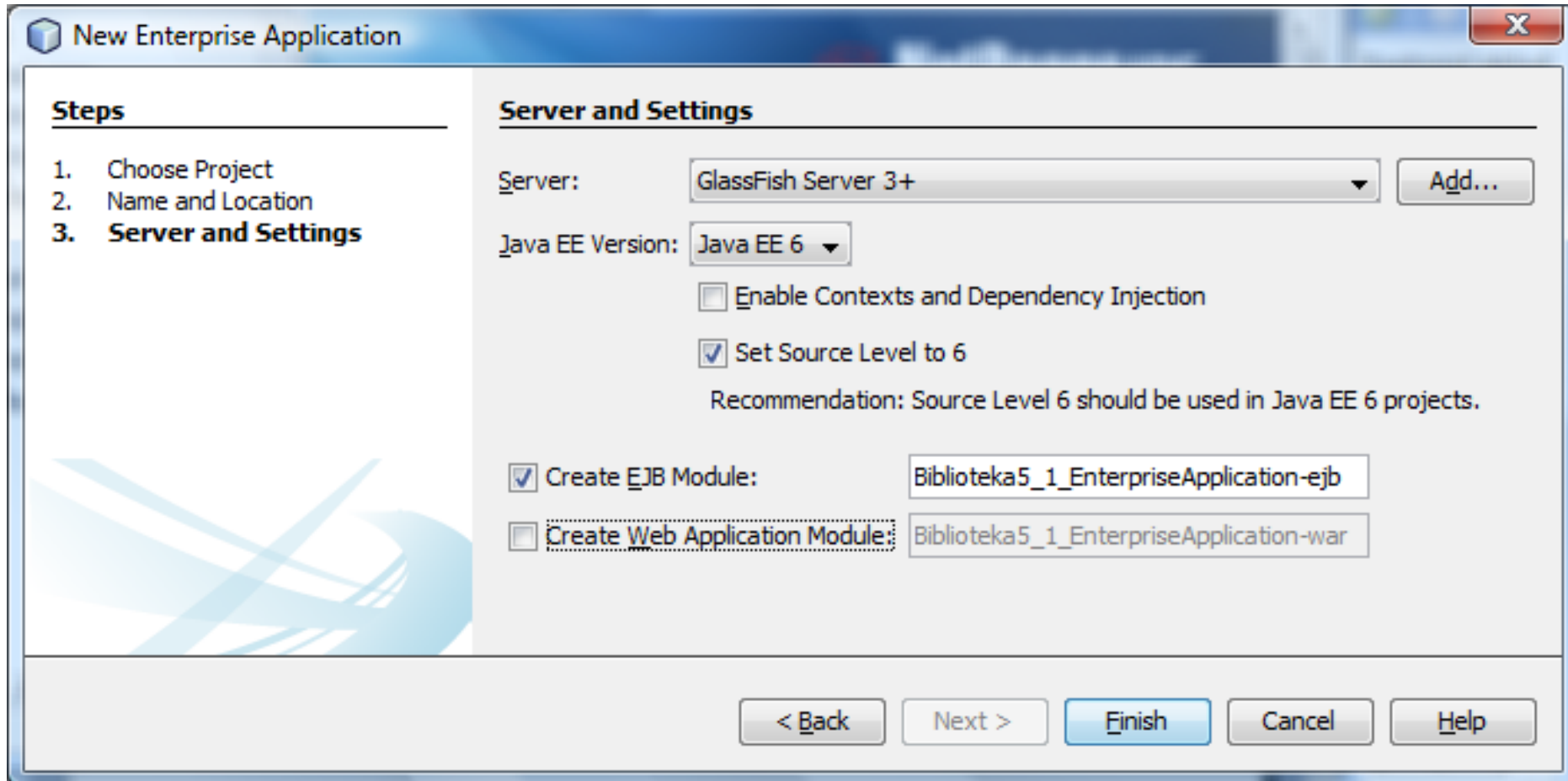
Use Dedicated Folder for Storing Libraries

Libraries Folder:

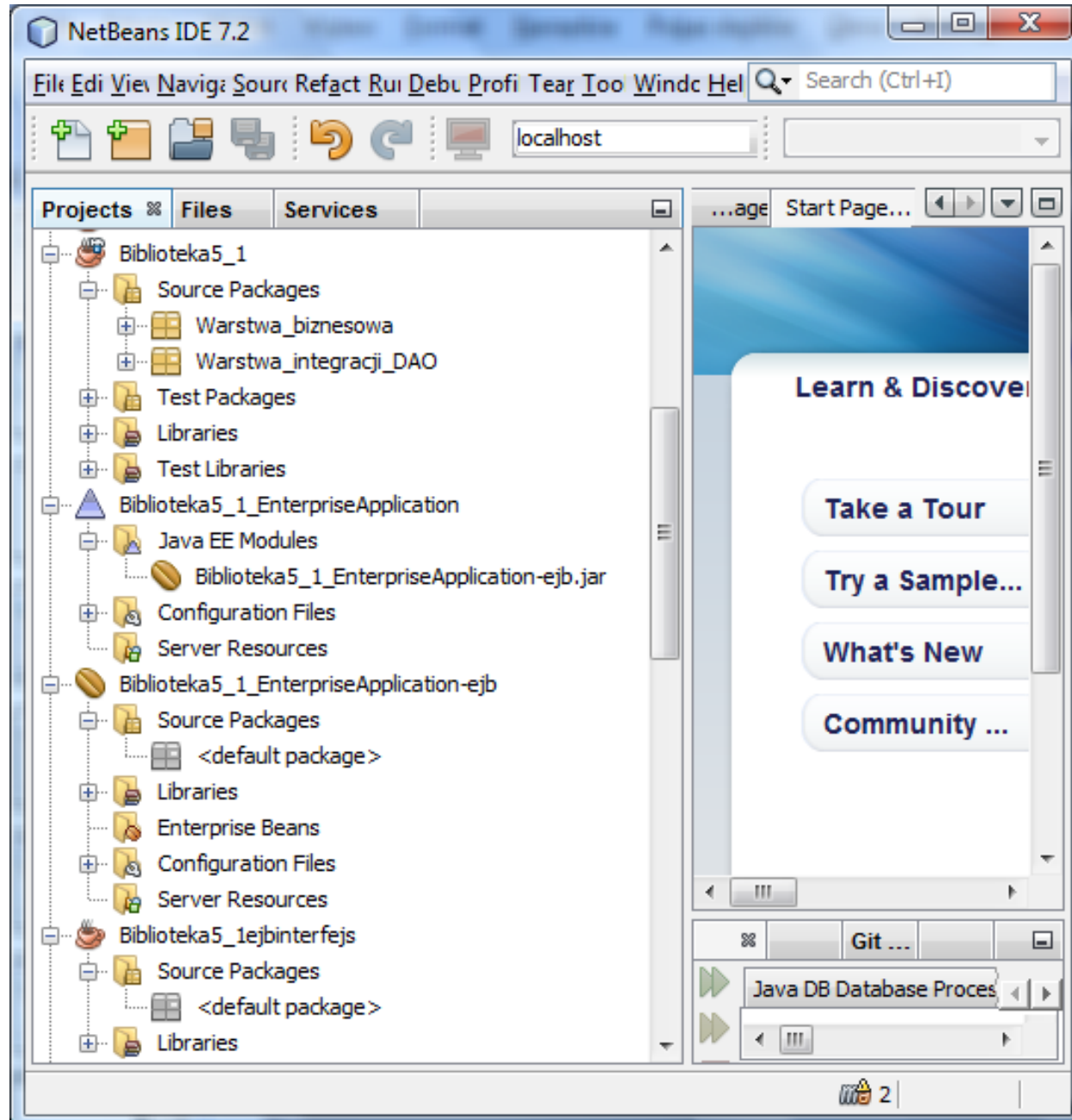
Different users and projects can share the same compilation libraries (see Help for details).

< Back **Next >** Finish Cancel Help

(4) Utworzenie aplikacji typu Enterprise - **File/New Project/JavaEE/Enterprise Application** o nazwie **Biblioteka5\_1\_EnterpriseApplication** z modułem EJB, ustawionym za pomocą **Create EJB Module** o nazwie **Biblioteka5\_1\_EnterpriseApplication-ejb**



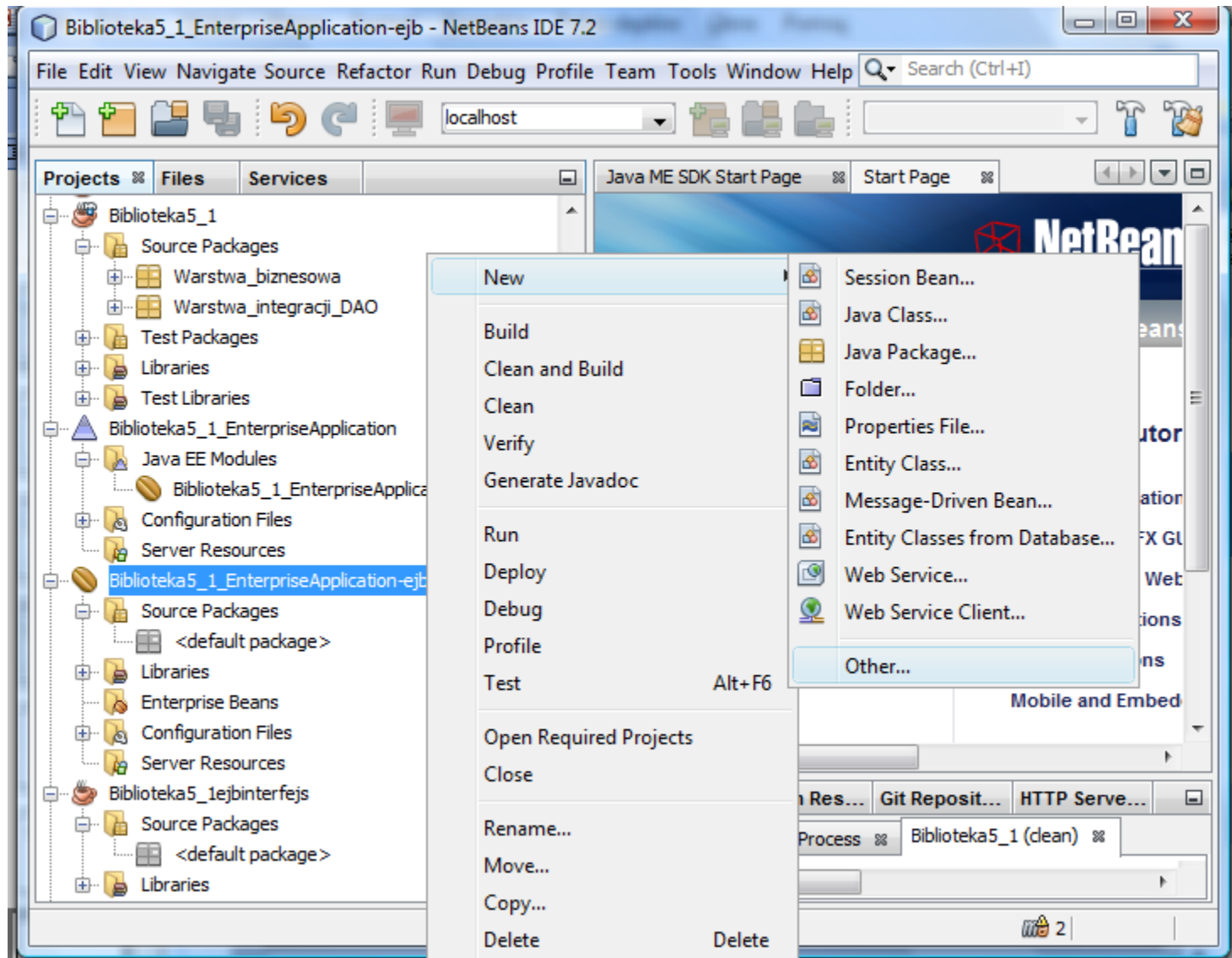
## Utworzone projekty, pokazane w zakładce **Projects**



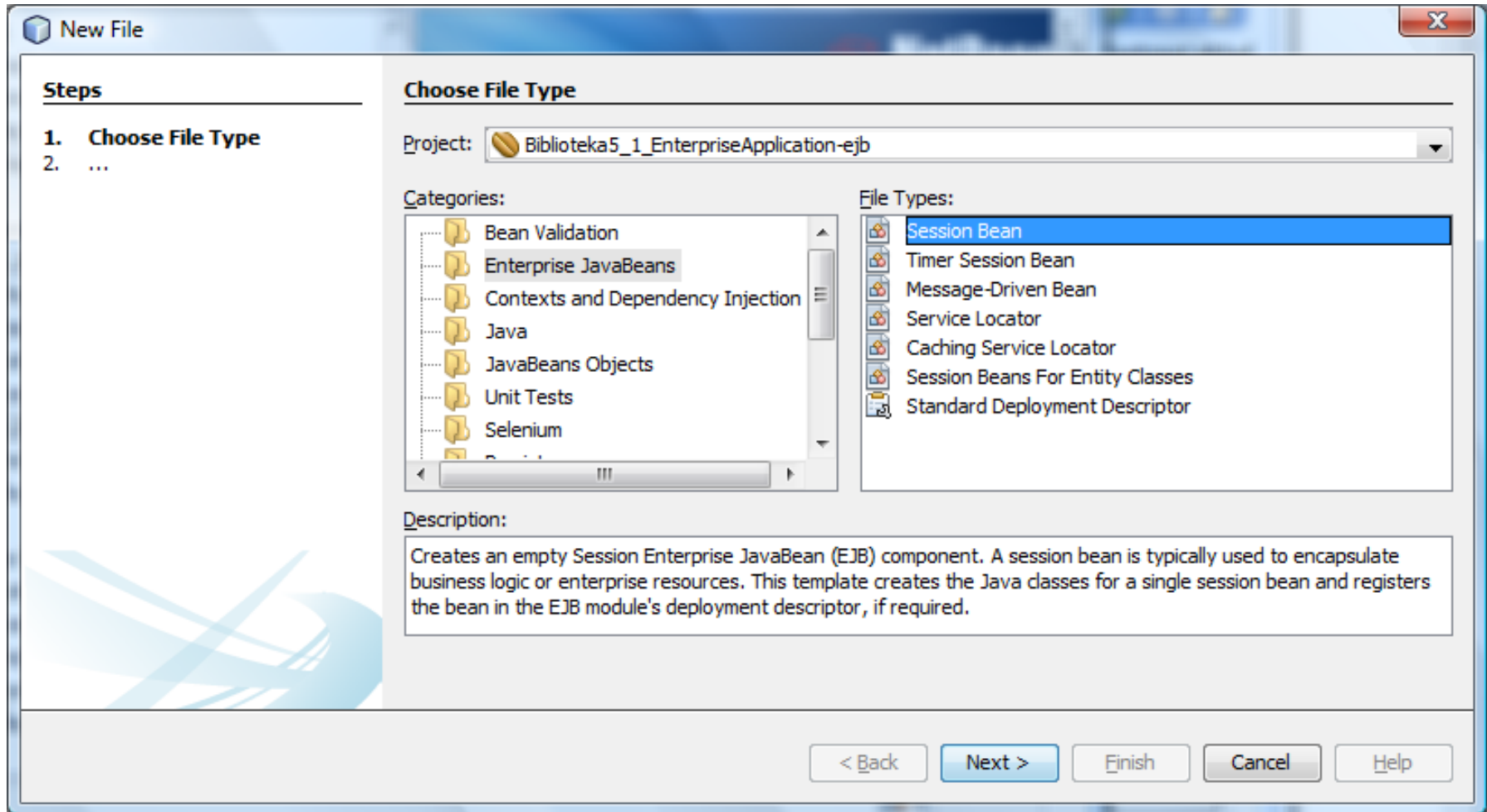


4. Wykonanie interfejsu zdalnego dostępu w projekcie typu Java Class Library (p. 2) dla klas Fasada i Baza oraz implementacji interfejsu w module EJB aplikacji typu Enterprise Edition Java EE 6 (p. 3)

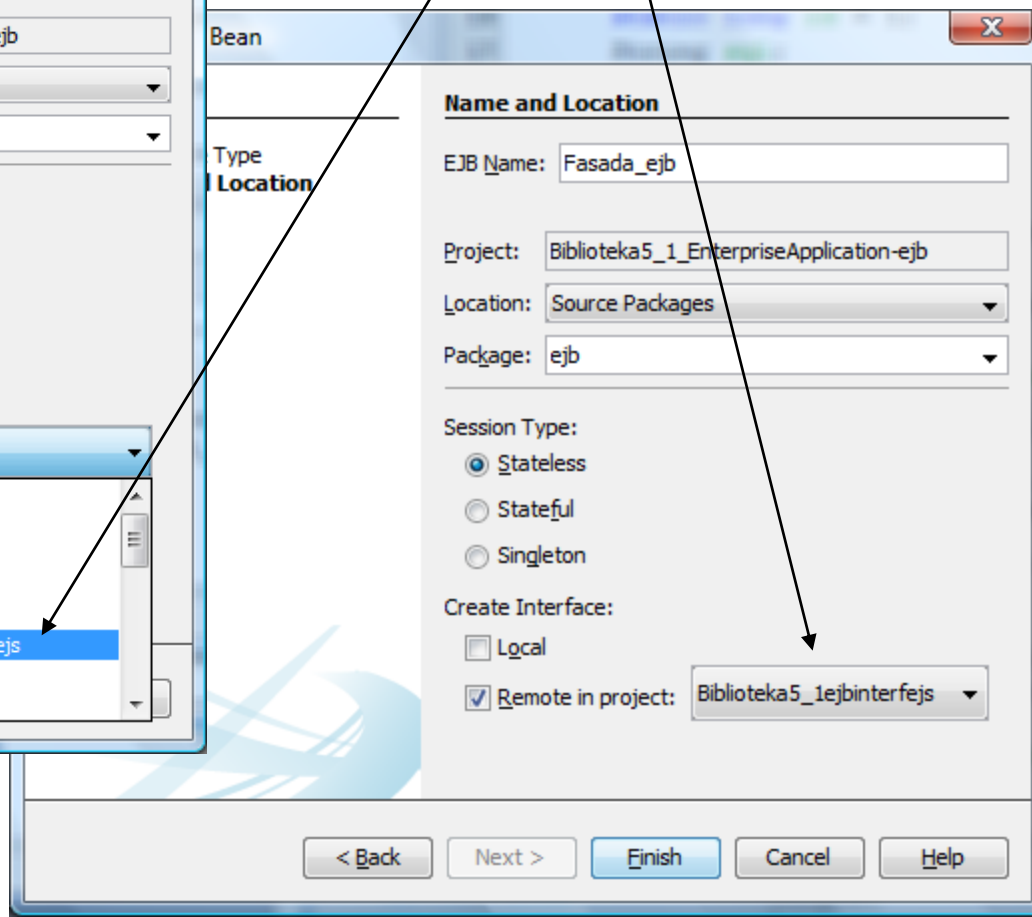
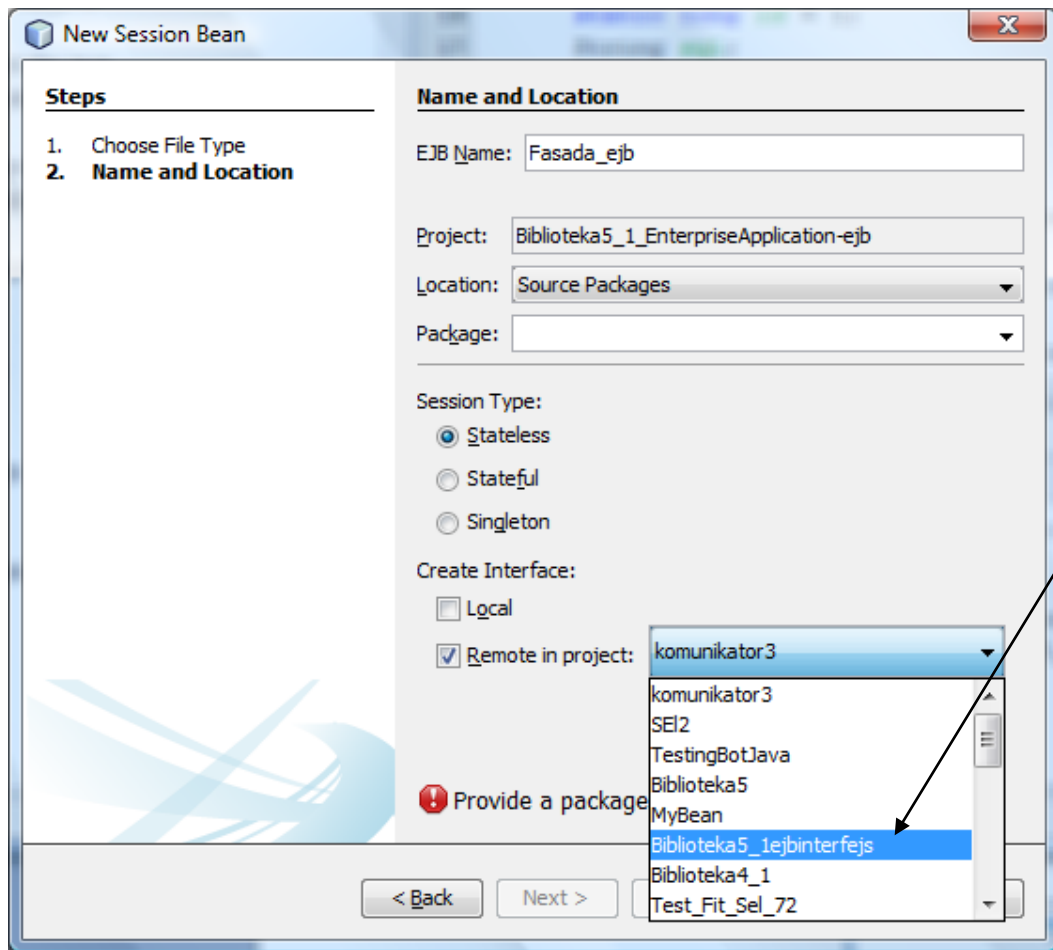
(1) Utworzenie fasady typu EJB wartwy biznesowej w projekcie **Biblioteka5\_1\_EnterpriseApplication-ejb** – prawy klawisz myszy na nazwę projektu w zakładce **Projects/New/Other**



(2) Utworzenie fasady typu EJB wartwy biznesowej w projekcie **Biblioteka5\_1\_EnterpriseApplication-ejb** – prawy klawisz myszy na nazwę projektu w zakładce **Projects/New/Other/Enterprise JavaBeans/SessionBean** i **Next**



(3) Nadanie nazwy **Fasada\_ejb** ziarnu typu **SessionBean** w tworzonym pakiecie **ejb** jako ziarno typu **Stateless** o dostępie **Remote in project** dla aplikacji z warstwy klienta. Wybór projektu typu **Biblioteka5\_1ejbinterfejs**, w którym zostanie utworzony interfejs dla ziarna **Fasada\_ejb**.



(4) Definicja interfejsu **Fasada\_ejbRemote** w projekcie **Biblioteka5\_1ejbinterfejs**

```
package ejb;
```

```
import java.util.ArrayList;  
import javax.ejb.Remote;
```

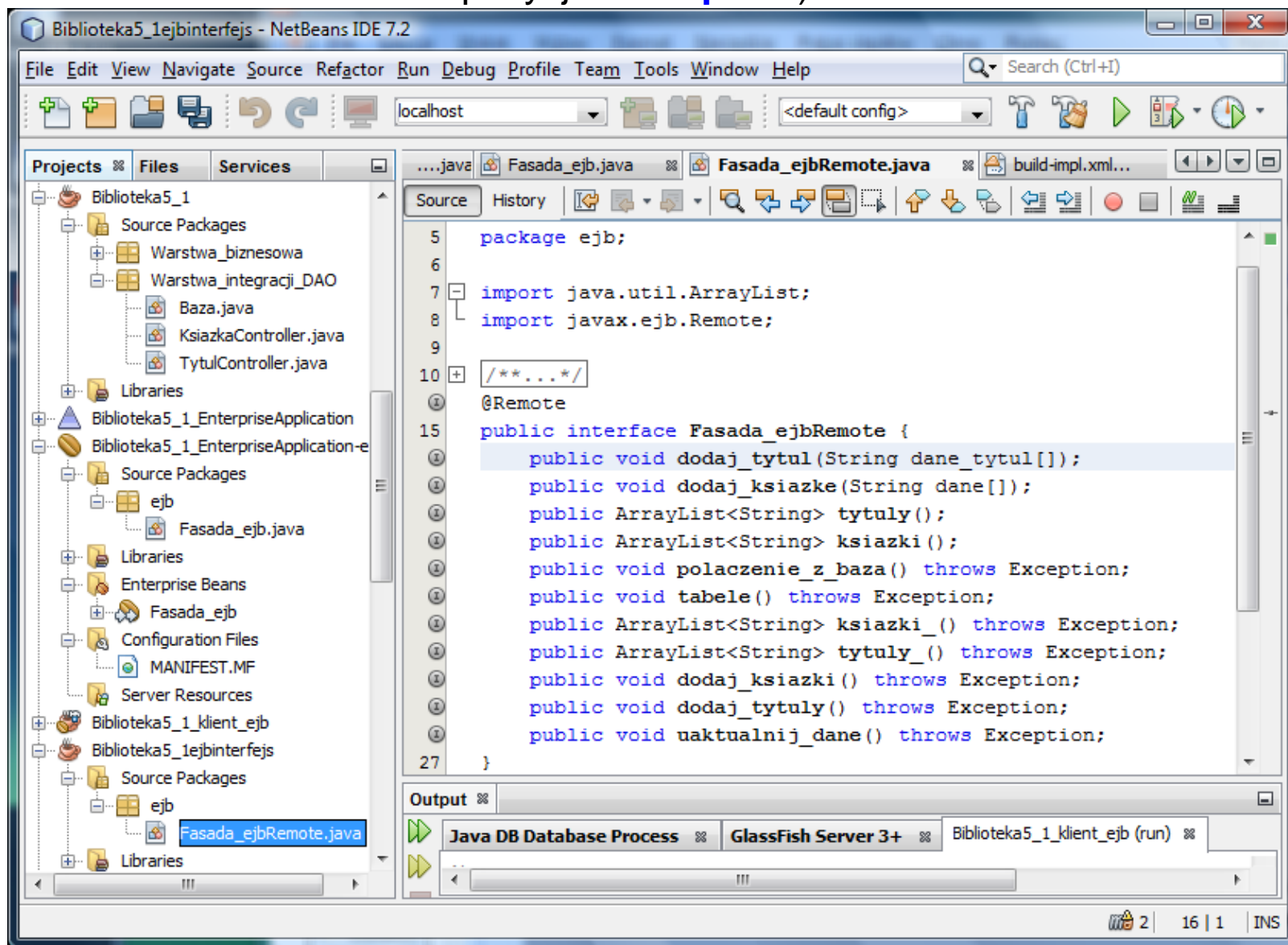
```
@Remote
```

```
public interface Fasada_ejbRemote {  
    public void dodaj_tytul(String dane_tytul[]);  
    public void dodaj_ksiazke(String dane[]);  
    public ArrayList<String> tytuly();  
    public ArrayList<String> ksiazki();  
    public void polaczenie_z_baza() throws Exception;  
    public void tabele() throws Exception;  
    public ArrayList<String> ksiazki_() throws Exception;  
    public ArrayList<String> tytuly_() throws Exception;  
    public void dodaj_ksiazki() throws Exception;  
    public void dodaj_tytuly() throws Exception;  
    public void uaktualnij_dane() throws Exception;  
}
```

Metody zwracające kolekcje obiektów typu **String** zawierających dane obiektów typu **Tytul\_ksiazki** oraz typu **Ksiazka**, przechowywane w pamięci aplikacji

Metody zwracające kolekcje obiektów typu **String** zawierających dane obiektów typu **Tytul\_ksiazki** oraz typu **Ksiazka**, przechowywane w bazie danych

(5) Utworzenie w projekcie **Biblioteka5\_1ejbinterfejs** pliku, który jest interfejsem **Fasada\_ejbRemote** dla ziarna **Fasada\_ejb** i dokonanie importu pakietu **java.util.ArrayList** (po kliknięciu prawym klawiszem myszy w oknie edytora i wybór pozycji **Fix Imports**)



(6) Definicja klasy **Fasada\_ejb** w projekcie **Biblioteka5\_1\_EnterpriseApplication-ejb**

```
package ejb;
```

```
import Warstwa_biznesowa.Fasada;  
import Warstwa_integracji_DAO.Baza;  
import java.util.ArrayList;  
import javax.ejb.Stateless;
```

```
@Stateless
```

```
public class Fasada_ejb implements Fasada_ejbRemote {
```

```
    public Fasada fasada = new Fasada();  
    public Baza baza = new Baza(fasada);
```

```
    public void dodaj_tytul(String dane_tytul[]) {  
        fasada.dodaj_tytul(dane_tytul);    }  
}
```

```
    public void dodaj_ksiazke(String dane[]) // your code here  
    {    fasada.dodaj_ksiazke(dane);    }  
}
```

```
    public ArrayList<String> tytuly() {  
        return fasada.tytuly();    }
```

```
    public ArrayList<String> ksiazki() {  
        return fasada.ksiazki(); }
```

```
public void polaczenie_z_baza() throws Exception
    { baza.polaczenie_z_baza(); }
```

```
public void tabele() throws Exception
    { baza.tabele(); }
```

```
public ArrayList<String> ksiazki_() throws Exception
    { return baza.ksiazki(); }
```

```
public ArrayList<String> tytuly_() throws Exception
    { return baza.tytuly(); }
```

```
public void dodaj_ksiazki() throws Exception
    { baza.dodaj_ksiazki();}
```

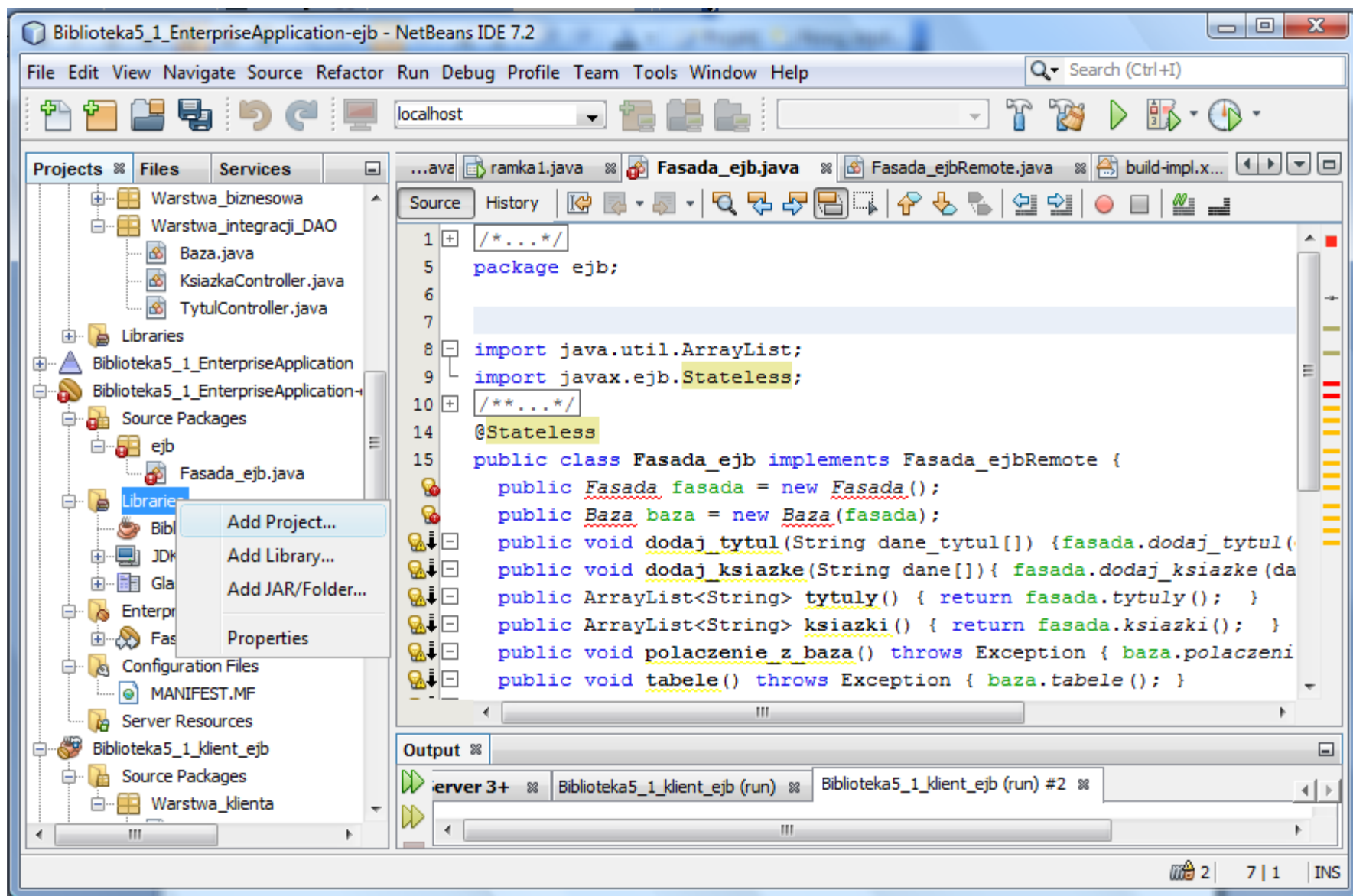
```
public void dodaj_tytuly() throws Exception
    { baza.dodaj_tytuly(); }
```

```
public void uaktualnij_dane() throws Exception
    { baza.uaktualnij_dane(); }
```

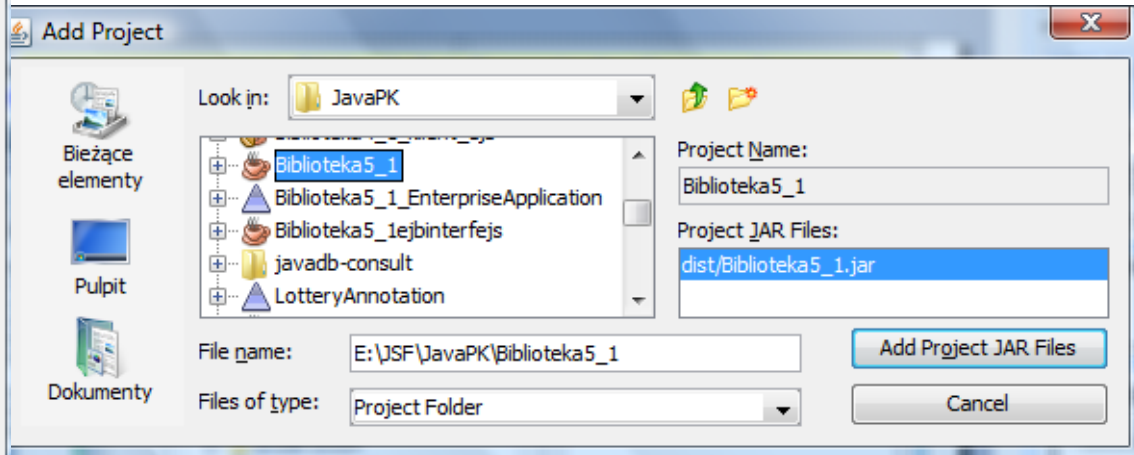
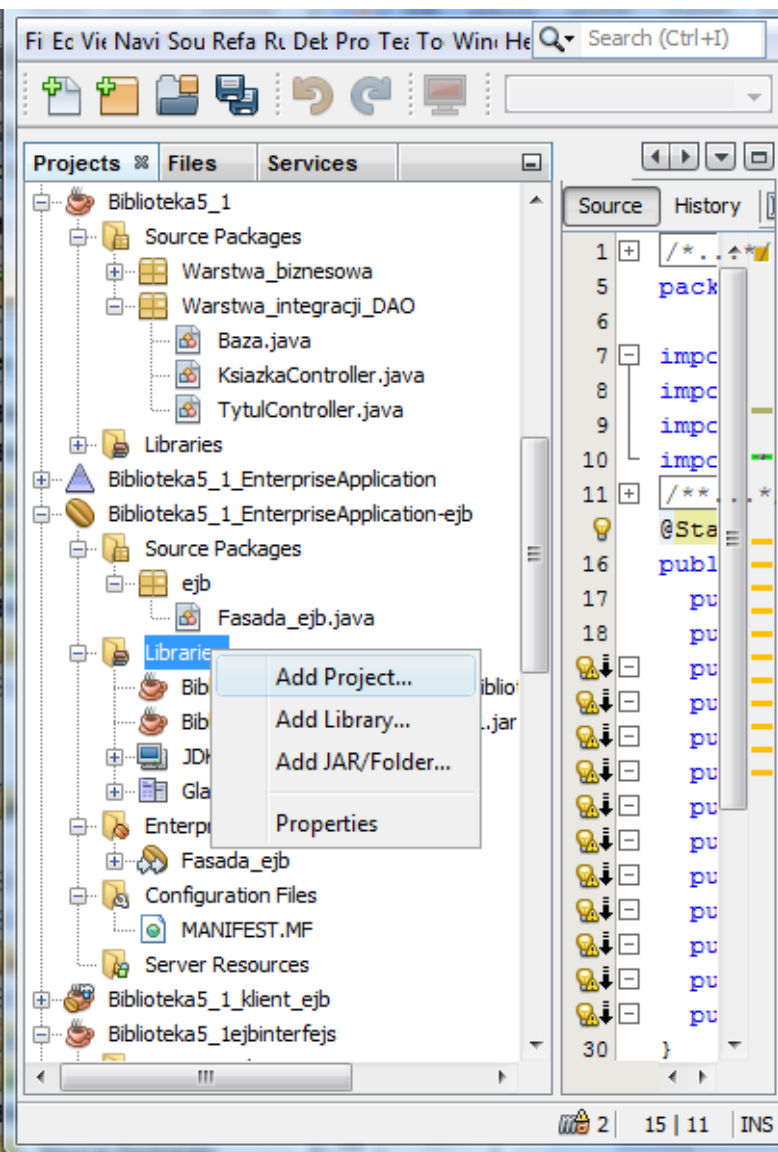
```
}
```



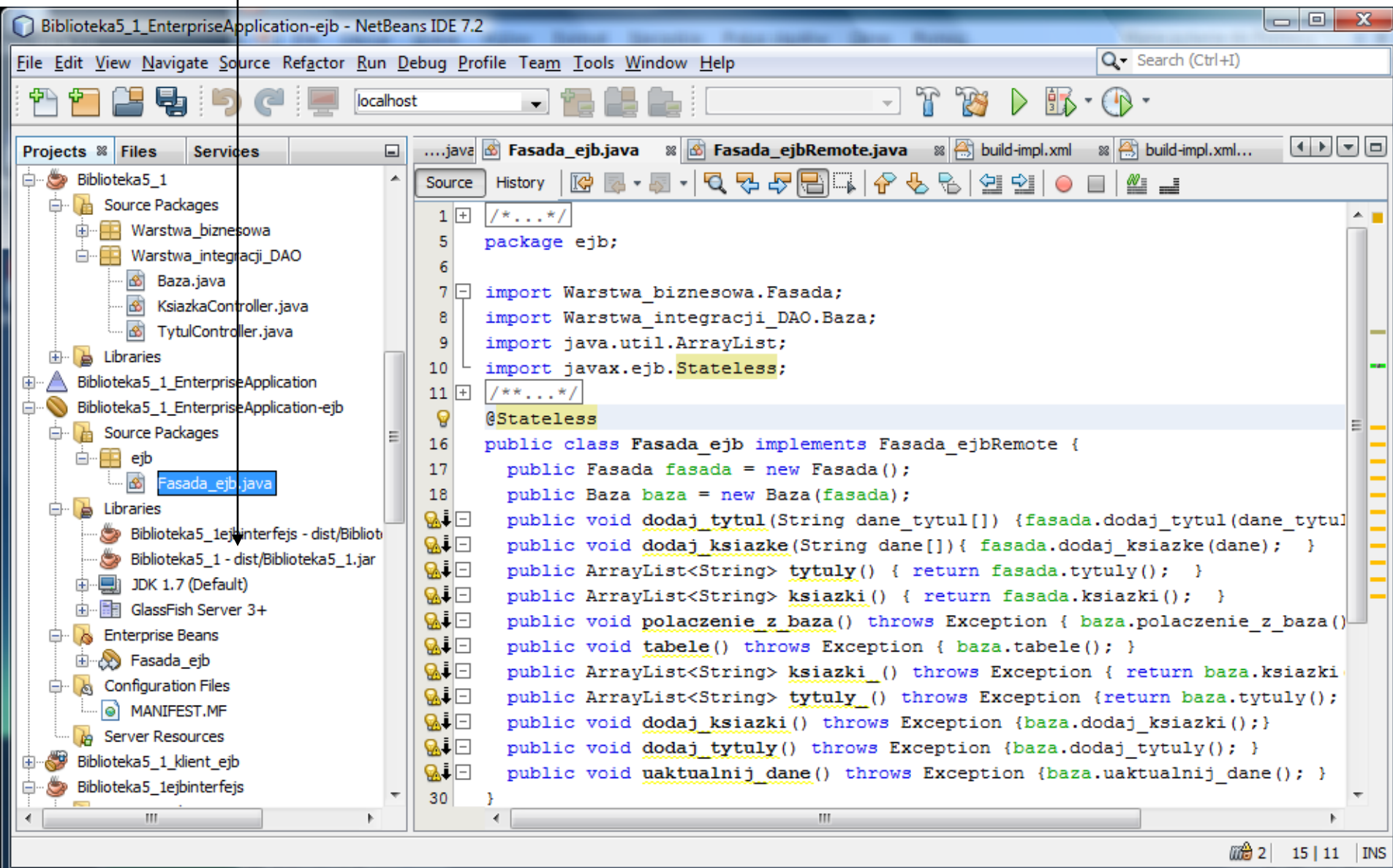
(7) Dodanie w projekcie **Biblioteka5\_1\_EnterpriseApplication-ejb** dostępu do klas zdefiniowanych w projekcie **Biblioteka5\_1** (prawy klawisz na katalog **Libraries** i wybór z listy **Add Project...**) – w celu utworzenia fasady typu EJB dla klasy **Fasada i Baza** projektu Java SE



(8) Dodanie w projekcie **Biblioteka5\_1\_EnterpriseApplication-ejb** dostępu do klas zdefiniowanych w projekcie **Biblioteka5\_1** – w celu utworzenia fasady typu EJB dla klasy **Fasada** i **Baza** projektu Java SE



(9) Dodanie w projekcie **Biblioteka5\_1\_EnterpriseApplication-ejb** dostępu do klas zdefiniowanych w projekcie **Biblioteka5\_1** – wykonanie operacji **Fix Imports** (prawy klawisz na oknie edytora i wybór z listy pozycji **Fix Imports**)



## 5. Uruchomienie aplikacji typu Enterprise Edition Java EE 6 za pomocą operacji Deploy

(1) Należy wykonać operację **Build i Deploy** projektu **Biblioteka5\_1\_EnterpriseApplication**

The screenshot displays the NetBeans IDE 7.2 interface. The main window title is "Biblioteka5\_1\_EnterpriseApplication - NetBeans IDE 7.2". The menu bar includes File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, and Help. The toolbar shows various icons for file operations and running. The left sidebar shows the project structure with "Biblioteka5\_1\_EnterpriseApplication" selected. The central editor shows the source code for "ramka1.java", which includes package declarations, imports, and a class definition for "ramka1" extending "JFrame" and implementing "EJB". The bottom panel shows the "Output" window with the following log messages:

```
INFO: EJBS181:Portable JNDI names for EJB Fasada_ejb: [java:global
INFO: EJBS182:Glassfish-specific (Non-portable) JNDI names for EJB
INFO: Biblioteka5_1_EnterpriseApplication was successfully deployed
INFO: Nie mozna zalozyc tabeli Tytul
INFO: Nie mozna zalozyc tabeli Ksiazka
INFO: EJBS181:Portable JNDI names for EJB Fasada_ejb: [java:global
INFO: EJBS182:Glassfish-specific (Non-portable) JNDI names for EJB
INFO: Biblioteka5_1_EnterpriseApplication was successfully deployed
```

The status bar at the bottom right shows "2 | 28 | 6 | INS".

(2) Widok w okienku Services (należy wykonać operację Refresh po kliknięciu prawym klawiszem na pozycję GlassFish Server lub Application) – uruchomiona aplikacja **Biblioteka5\_1\_EnterpriseApplication**

The screenshot displays the NetBeans IDE 7.2 interface. The 'Services' view on the left shows a tree structure with 'GlassFish Server 3+' selected, and 'Biblioteka5\_1\_EnterpriseApplication' highlighted under its 'Applications' folder. The main editor area shows the source code of 'Baza.java' with three methods: 'polaczenie\_z\_baza()', 'tabela()', and 'ksiazki()'. The bottom pane, 'HTTP Server Monitor', shows the deployment output for 'Biblioteka5\_1\_EnterpriseApplication', including the message 'BUILD SUCCESSFUL (total time: 45 seconds)'. An arrow points from the text above to the 'Biblioteka5\_1\_EnterpriseApplication' entry in the Services view.

```
public void polaczenie_z_baza() throws Exception {
    baza.polaczenie_z_baza();
}

public void tabela() throws Exception {
    baza.tabela();
}

public ArrayList<String> ksiazki() throws Exception {
    return baza.ksiazki();
}
```

Search Results | Output | HTTP Server Monitor

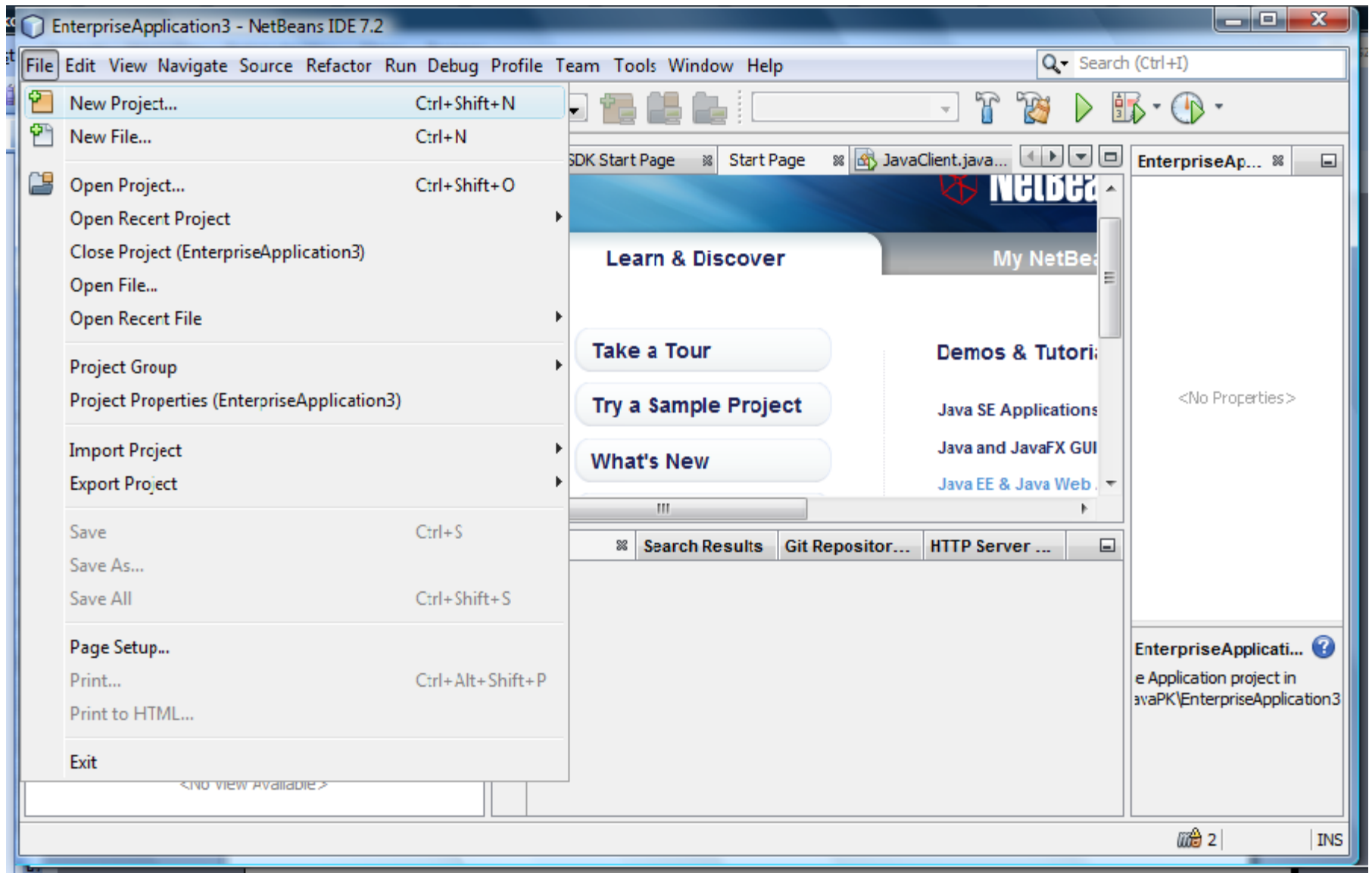
Java DB Database Process | GlassFish Server 3+ | Biblioteka5\_1\_EnterpriseApplication

Completed initial distribution of Biblioteka5\_1\_EnterpriseApplication  
Internal HTTP server running on port 8082  
post-run-deploy:  
run-deploy:  
BUILD SUCCESSFUL (total time: 45 seconds)



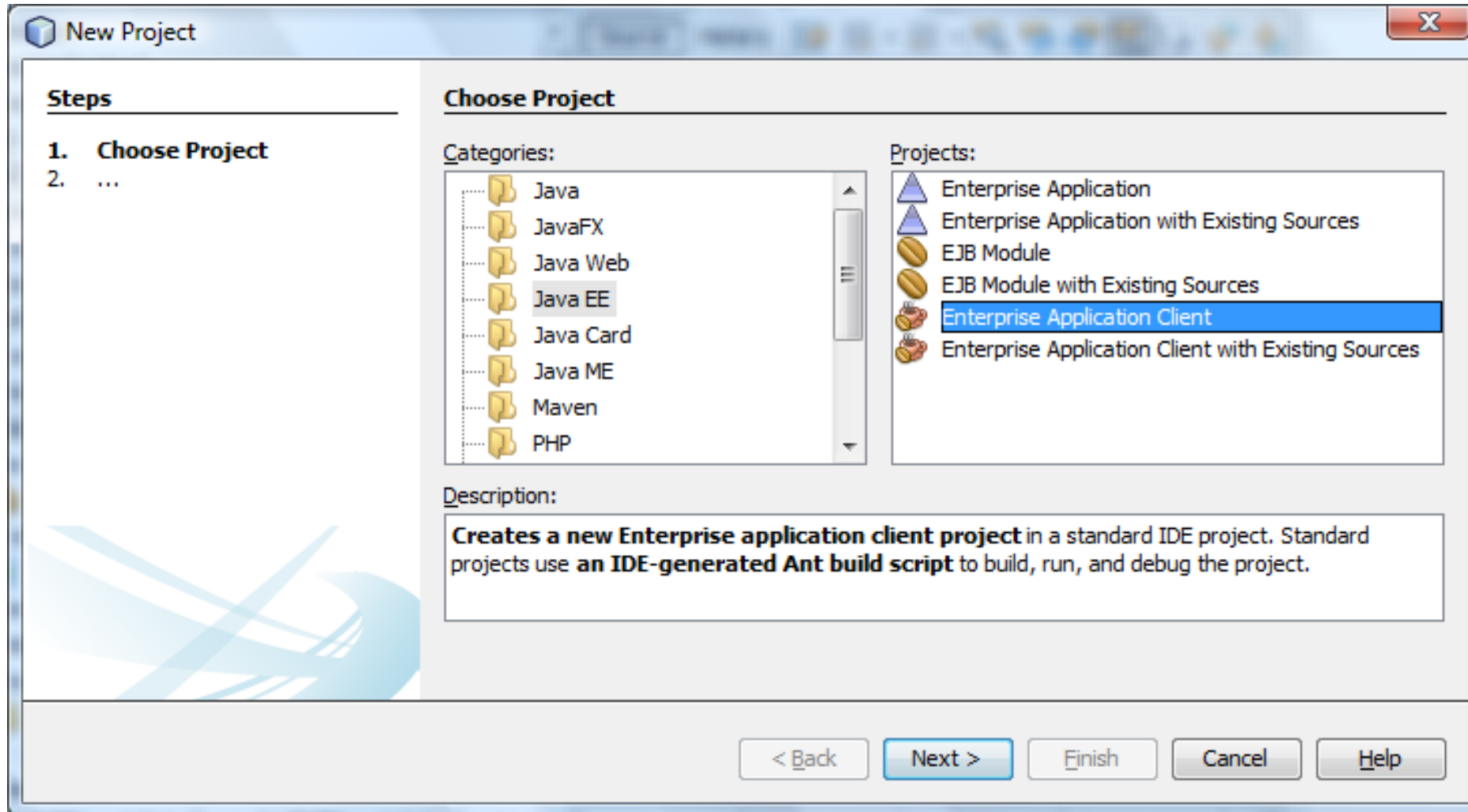
## 6. Wykonanie aplikacji klienta typu Enterprise Edition Java EE 6

# (1) Utworzenie aplikacji klienta opartej klasie **ramka1** z projektu **Biblioteka5** z lab5-PIO – **File/New Project**

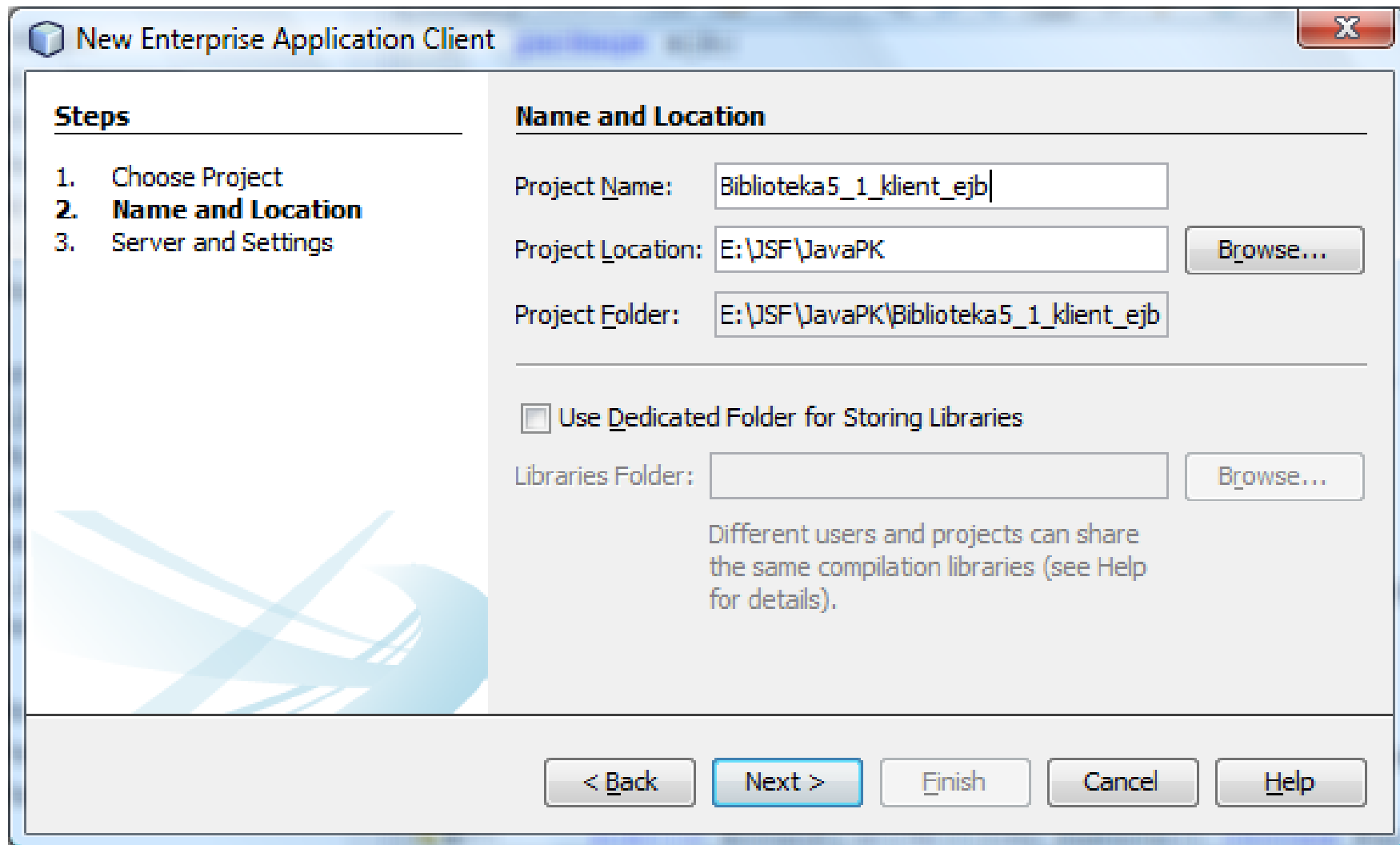




(2) Utworzenie aplikacji klienta opartej klasie ramka1 z projektu **Biblioteka5** z lab5-PIO – **File/New Project/JavaEE/Enterprise Application Client** i **Next**



(3) Utworzenie aplikacji klienta opartej klasie **ramka** z projektu **Biblioteka5** z lab5-PIO – **File/New Project/JavaEE/Enterprise Application Client** o nazwie **Biblioteka5\_1\_klient\_ejb** i **Next**



**New Enterprise Application Client**

**Steps**

1. Choose Project
- 2. Name and Location**
3. Server and Settings

**Name and Location**

Project Name: Biblioteka5\_1\_klient\_ejb

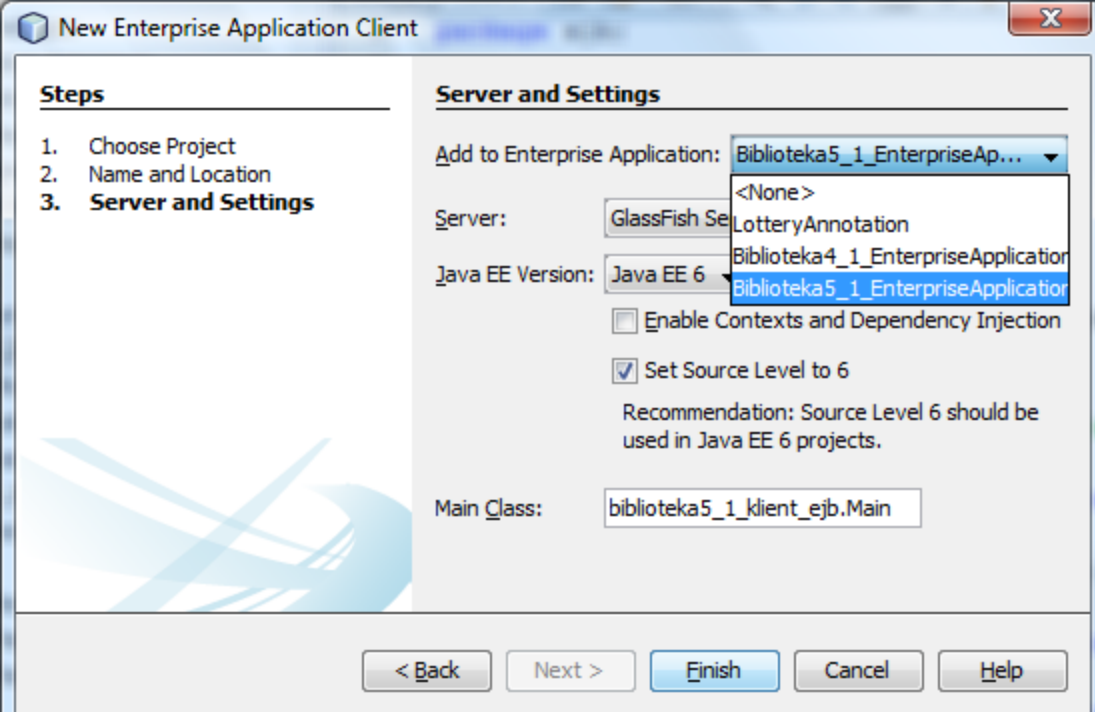
Project Location: E:\JSF\JavaPK

Project Folder: E:\JSF\JavaPK\Biblioteka5\_1\_klient\_ejb

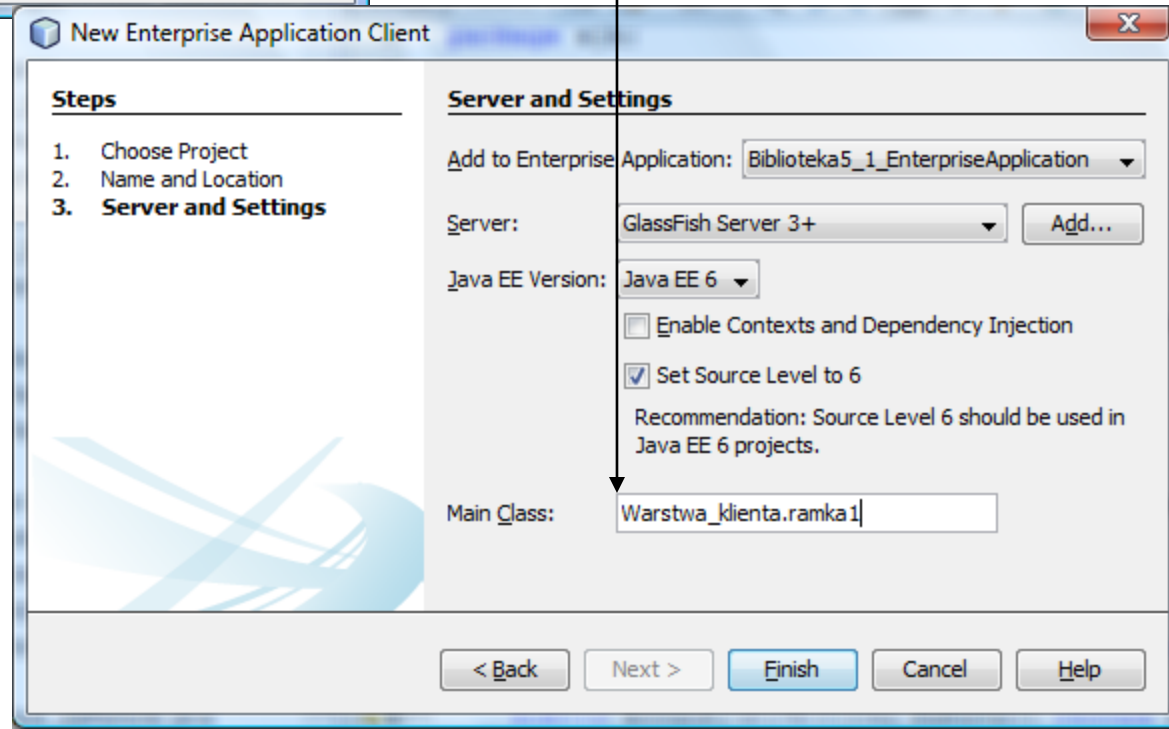
Use Dedicated Folder for Storing Libraries

Libraries Folder:

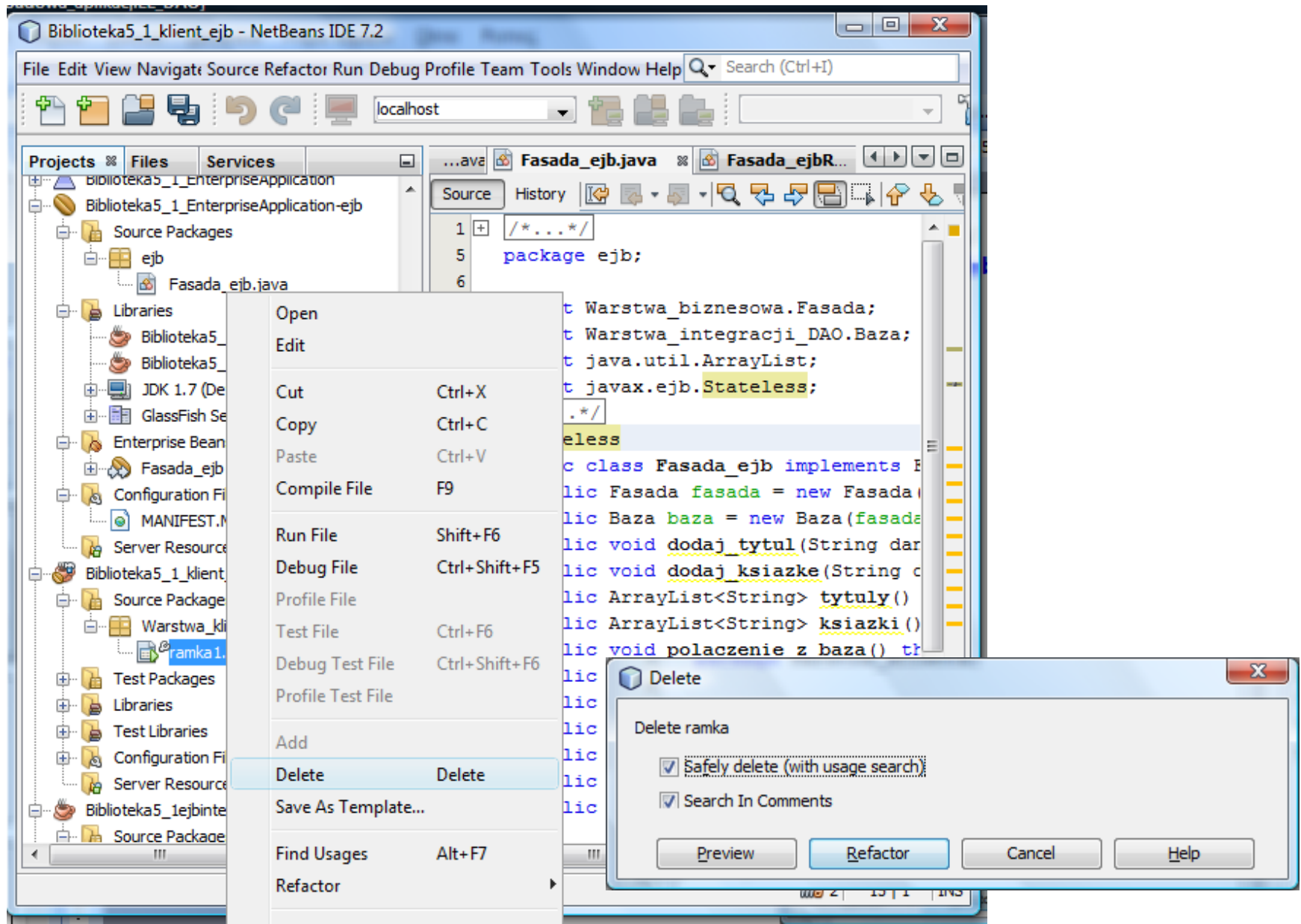
Different users and projects can share the same compilation libraries (see Help for details).



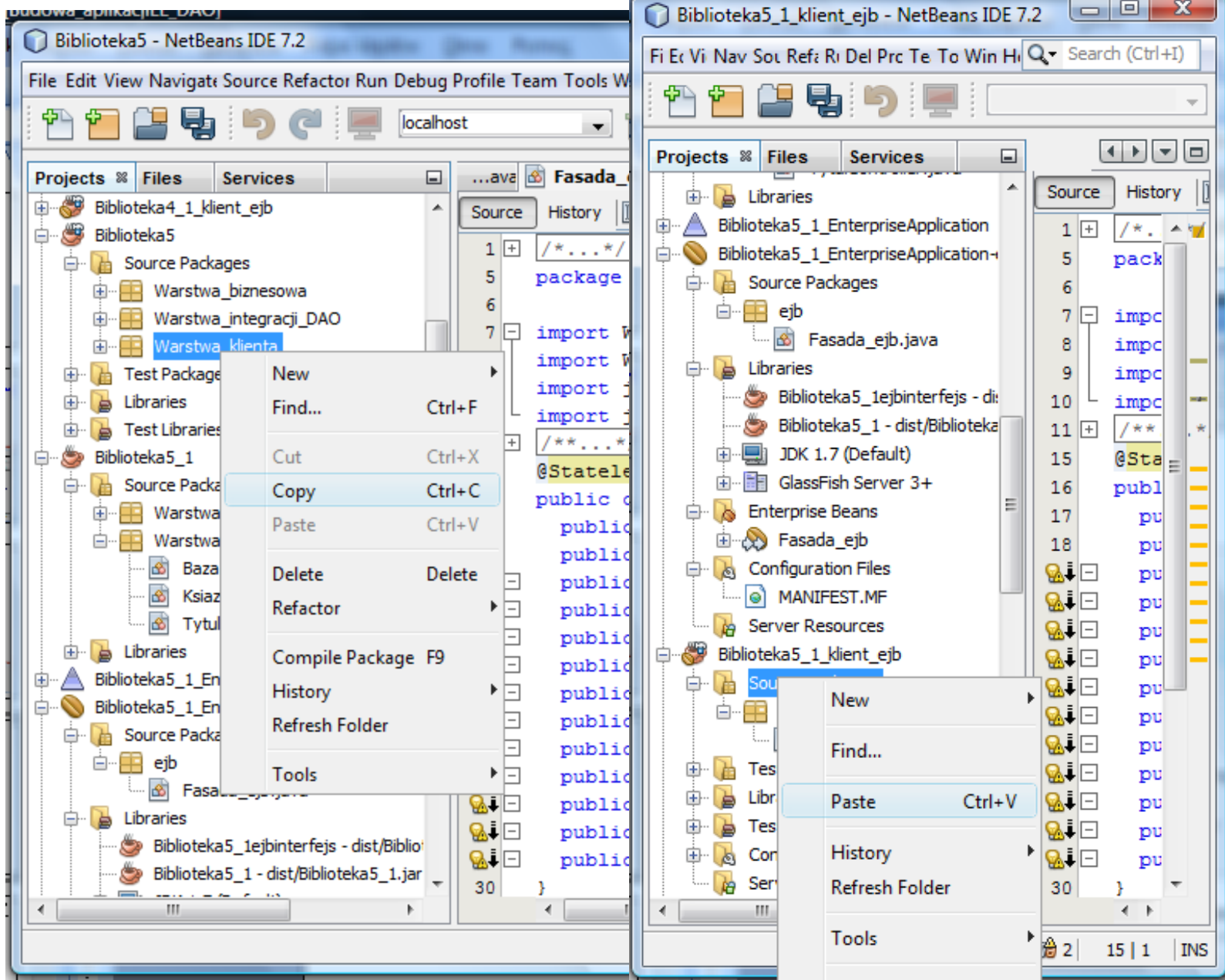
(5) Wybór aplikacji głównej typu Enterprise  
Biblioteka5\_1\_EnterpriseApplication,  
nadac nazwę Main  
Class: Warstwa\_klienta.ramka1 i  
Finish



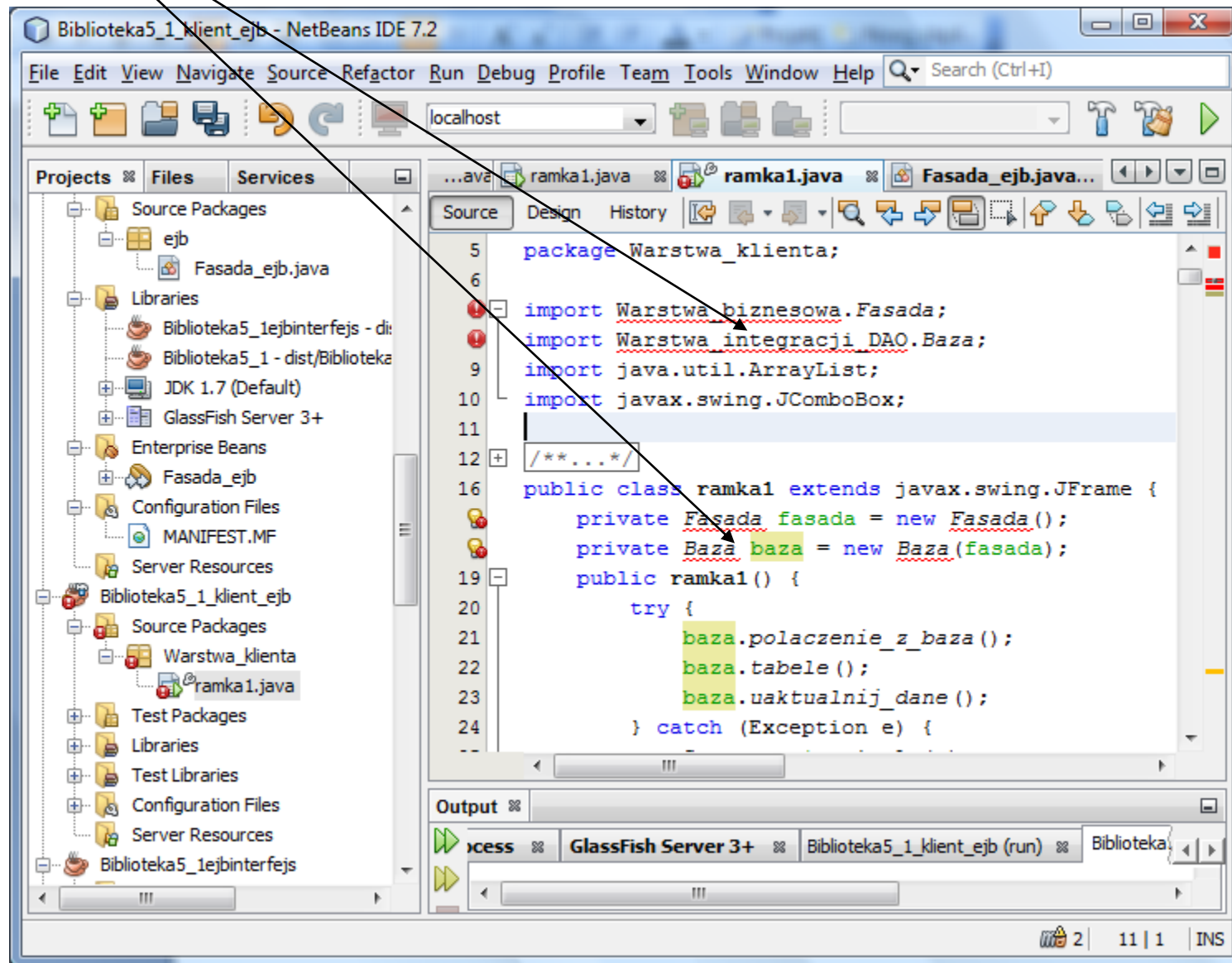
(6) Należy z projektu **Biblioteka5\_1\_klient\_ejb** usunąć plik **ramka1** (**Safely delete**) i wkleić pakiet **Warstwa\_klienta** z programu **Biblioteka5** z lab5-PIO



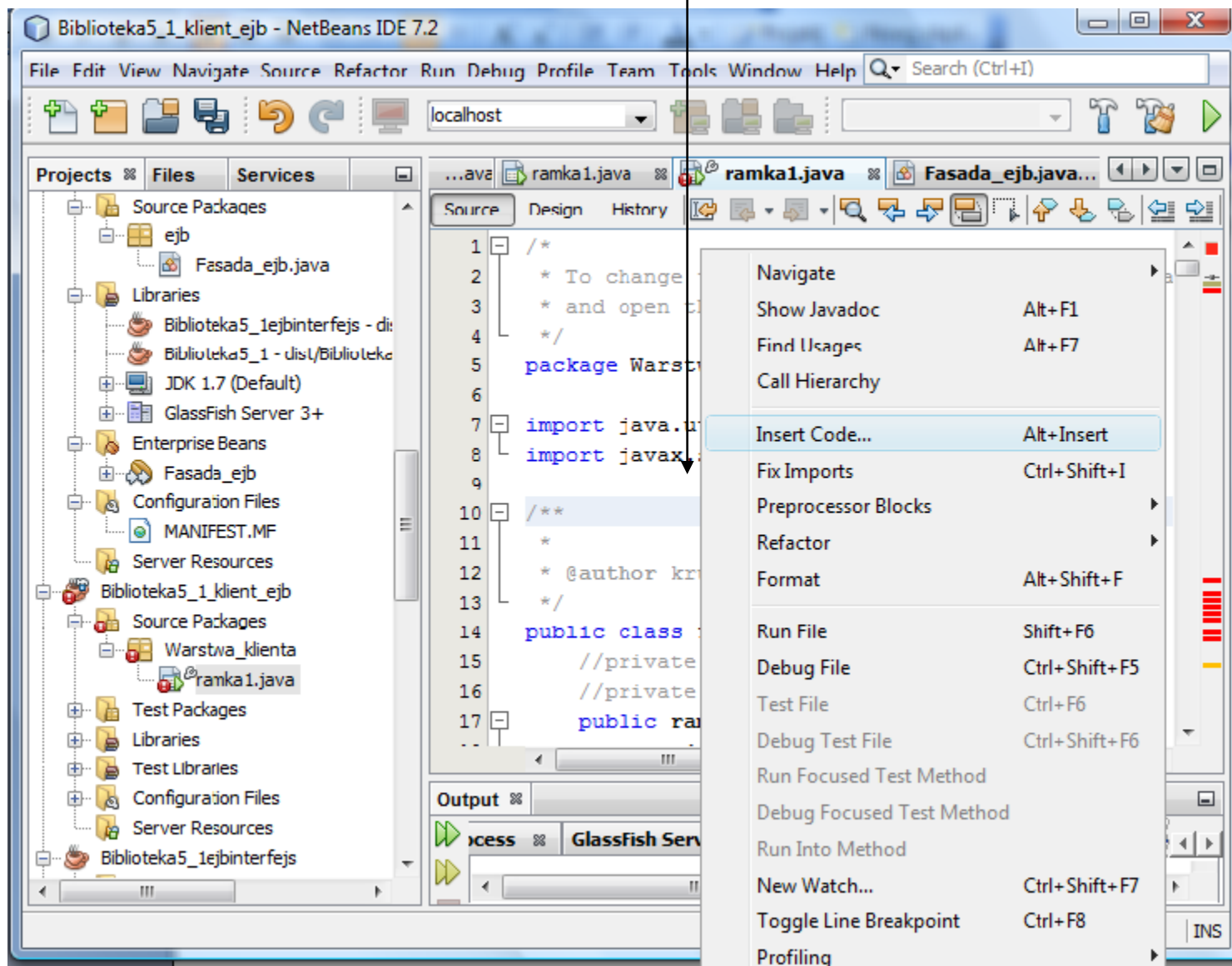
(7) Należy z projektu **Biblioteka5\_1\_klient\_ejb** usunąć plik **ramka1** (**Safely delete**) i wkleić pakiet **Warstwa\_klienta** z programu **Biblioteka5** z lab5-PIO



(8) Należy w projekcie **Biblioteka5\_1\_klient\_ejb** zastąpić fasadę aplikacji z warstwy klienta typu **Baza** oraz **Fasada** fasadą typu **Fasada\_ejb** umożliwiającą zdalny dostęp do warstwy integracji oraz biznesowej (z wykorzystaniem sieci)

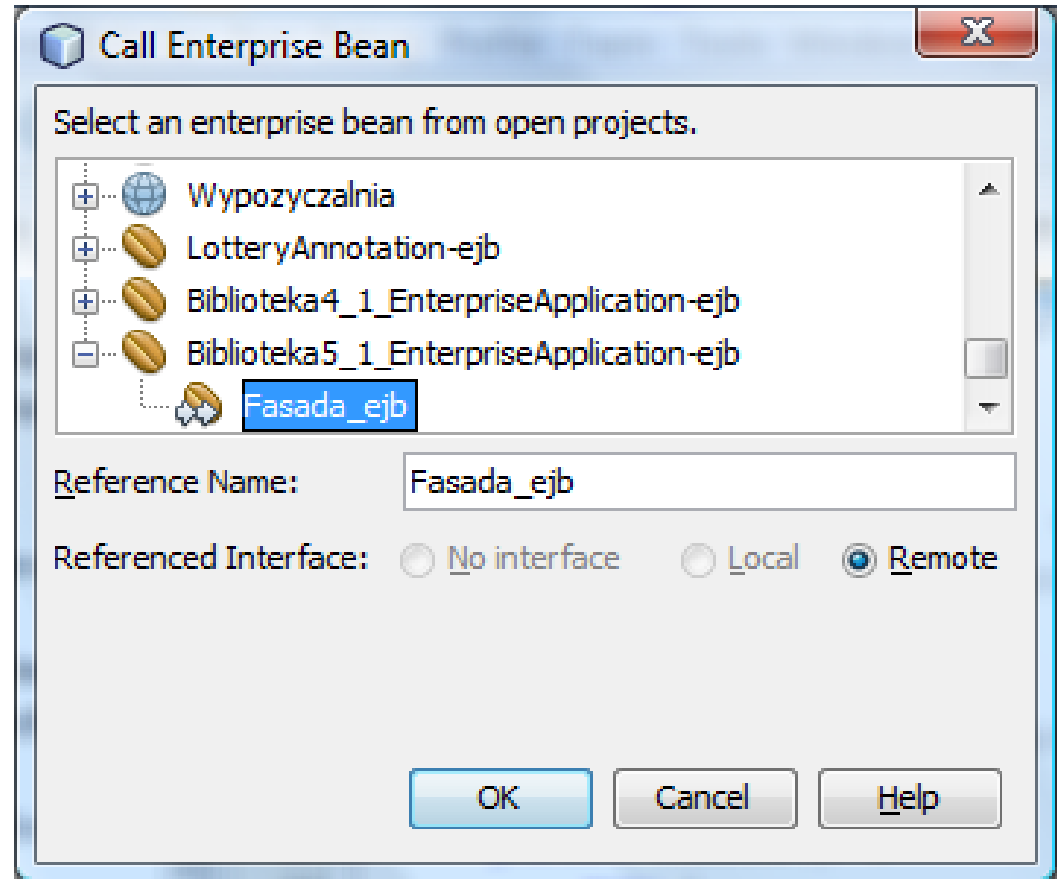
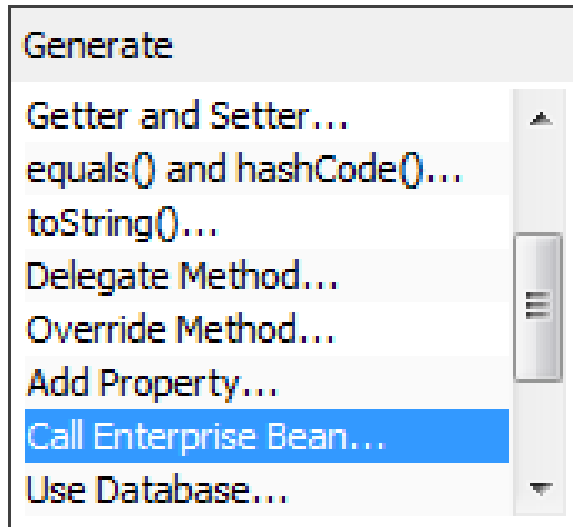


(9) Po usunięciu w projekcie **Biblioteka5\_1\_klient\_ejb** definicji **private Fasada fasada** oraz **private Baza baza** należy kliknąć prawym klawiszem na powierzchnię edytora i wybrać pozycję **Fix Imports** w celu usunięcia zbędnego importu klasy **Fasada i Baza** i następnie należy wybrać podobnie pozycję **Insert Code**





(10) Należy kliknąć prawym klawiszem na powierzchnię edytora i wybrać pozycję **Insert Code/Call Enterprise Bean...** w wybrać z listy ziarno **Fasada\_ejb** z projektu **Biblioteka5\_1\_EnterpriseApplication-ejb**





## (11) Kod metod obsługujących zdarzenia w klasie ramka1 przystawanych do interfejsu klasy Fasada\_ejb


```
package Warstwa_klienta;

import.ejb.Fasada_ejbRemote;
import java.util.ArrayList;
import javax.ejb.EJB;
import javax.swing.JComboBox;

public class ramka1 extends javax.swing.JFrame {
    @EJB
    private static Fasada_ejbRemote fasada;

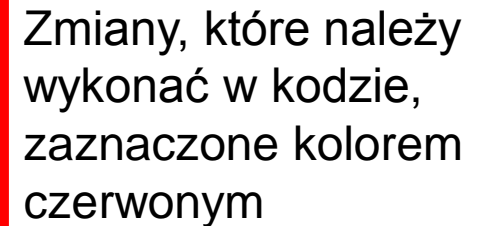
    //private Fasada fasada = new Fasada();
    //private Baza baza = new Baza(fasada);
    public ramka1() {
        try {
            fasada.polaczenie_z_baza();
            fasada.tabele();
            fasada.uaktualnij_dane();
        } catch (Exception e) {
            System.out.println(e);
        }
        initComponents();
    }
}
```

Zmiany, które należy wykonać w kodzie, zaznaczone kolorem czerwonym



```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  
String s1, s2, s3, s4, s5;  
Object zrodlo = evt.getSource();  
if (zrodlo == jButton1) {  
    s1 = jTextField1.getText();  
    s2 = jTextField2.getText();  
    s3 = jTextField4.getText();  
    s4 = jTextField5.getText();  
    s5 = jTextField6.getText();  
    String[] tytul = {s1, s2, s3, s4, s5};  
    if (!s1.equals("") && !s2.equals("") && !s3.equals("")  
        && !s4.equals("") && !s5.equals("")) {  
        fasada.dodaj_tytul(tytul); }  
    }  
}  
  
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {  
String s1, s2, s3, s4, s5;  
Object zrodlo = evt.getSource();  
if (zrodlo == jButton2) {  
    s1 = jTextField5.getText();  
    s2 = jTextField7.getText();  
    String[] ksiazka = {s1, s2};  
    if (!s1.equals("") && !s2.equals("")) {  
        fasada.dodaj_ksiazke(ksiazka); }  
    }  
}
```

Zmiany, które należy  
wykonać w kodzie,  
zaznaczone kolorem  
czerwonym



```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    zawartosc_listy(fasada.tytuly(), jComboBox1);
}
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
    zawartosc_listy(fasada.ksiazki(), jComboBox2);
}
private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
    try {
        fasada.dodaj_tytuly();
    } catch (Exception e) {
        System.out.println(e);
    }
}
private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {
    try {
        fasada.dodaj_ksiazki();
    } catch (Exception e) {
        System.out.println(e);
    }
}
private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {
    try {
        zawartosc_listy(fasada.tytuly_(), jComboBox1);
    } catch (Exception e) {
        System.out.println(e);
    }
}
```

Zmiany, które należy  
wykonać w kodzie,  
zaznaczone kolorem  
czerwonym

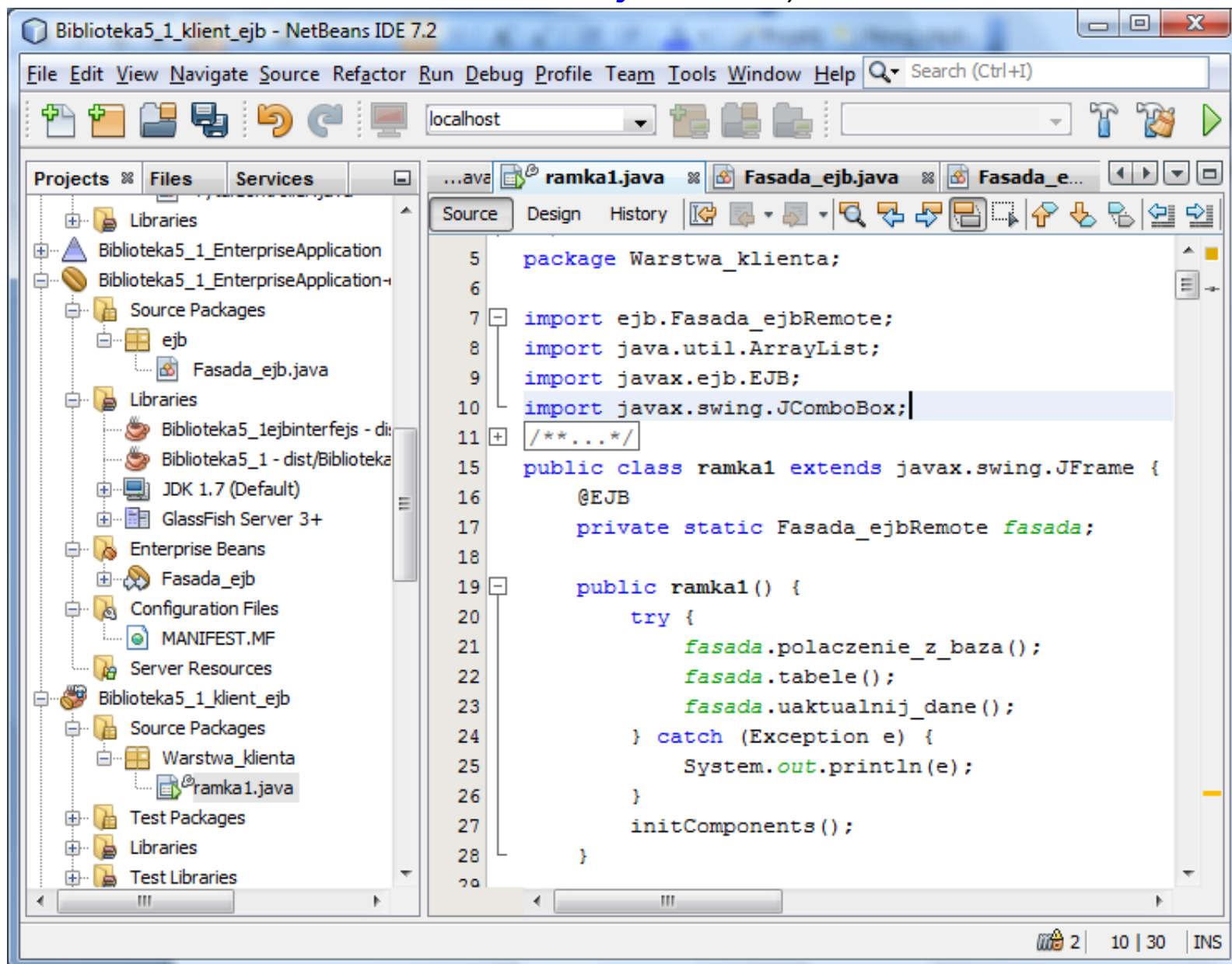
```
private void jButton8ActionPerformed(java.awt.event.ActionEvent evt) {
    try {
        zawartosc_listy(fasada.książki_(), jComboBox2);
    } catch (Exception e) {
        System.out.println(e);
    }
}
private void jButton9ActionPerformed(java.awt.event.ActionEvent evt) {
    jButton7ActionPerformed(evt);
    jButton8ActionPerformed(evt);
}
private void zawartosc_listy(ArrayList<String> kol, JComboBox lista) {
    lista.removeAllItems();
    for (String s : kol) {
        lista.addItem(s);
    }
}
```

Zmiany, które należy  
wykonać w kodzie,  
zaznaczone kolorem  
czerwonym

```
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new ramka1().setVisible(true);
        }
    });
}
```

```
// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JButton jButton5;
private javax.swing.JButton jButton6;
private javax.swing.JButton jButton7;
private javax.swing.JButton jButton8;
private javax.swing.JButton jButton9;
private javax.swing.JComboBox jComboBox1;
private javax.swing.JComboBox jComboBox2;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;
private javax.swing.JLabel jLabel9;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField4;
private javax.swing.JTextField jTextField5;
private javax.swing.JTextField jTextField6;
private javax.swing.JTextField jTextField7;
// End of variables declaration
}
```

(12) Aplikacja **Biblioteka5\_1\_klient\_ejb** z warstwy klienta ma teraz zdalny dostęp do kodu obiektu typu **Fasada i Baza** dzięki ziarnu typu **Fasada\_ejb** (czyli interfejsu **Fasada\_ejbRemote**)



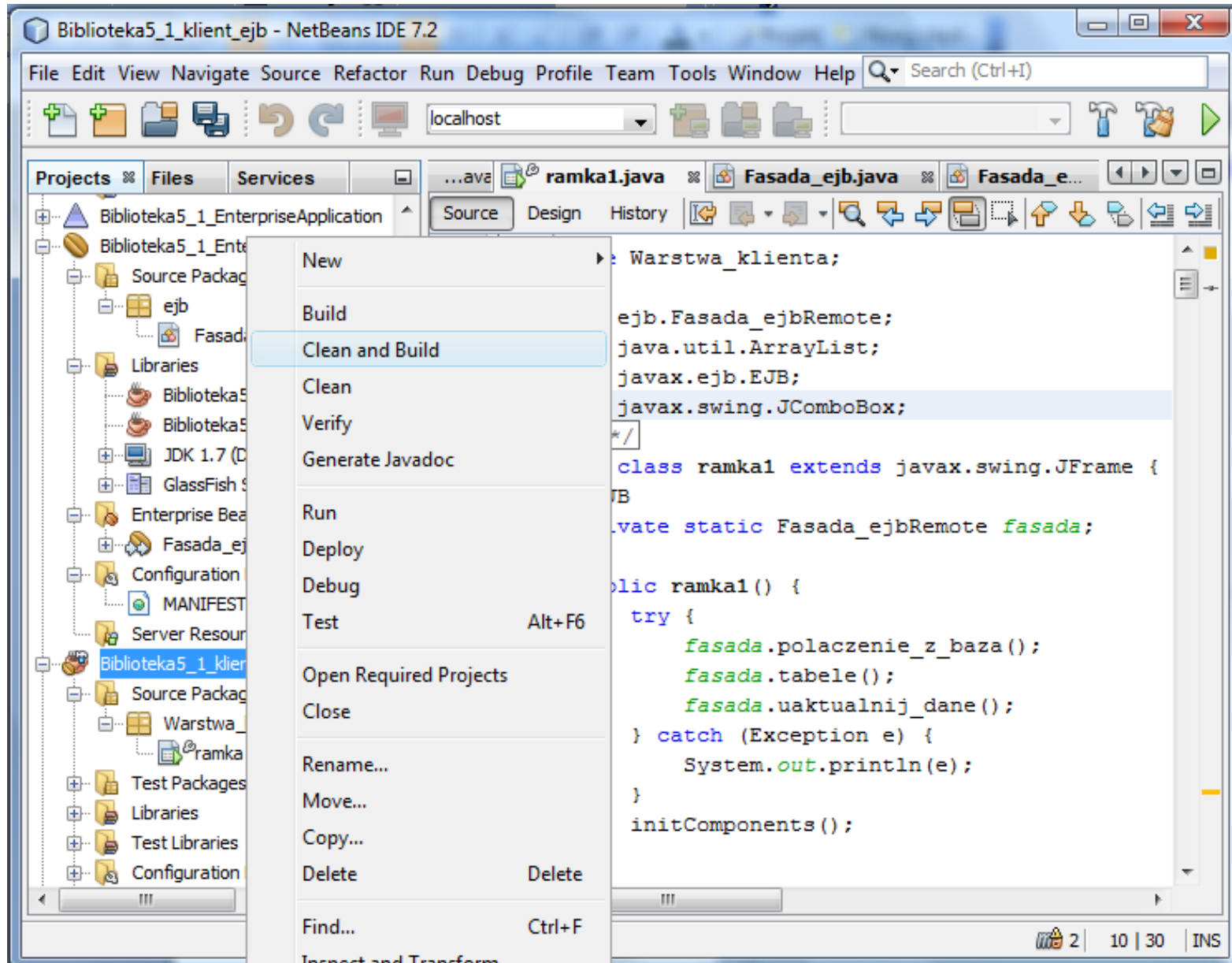
The screenshot shows the NetBeans IDE 7.2 interface. The left sidebar displays the project structure for 'Biblioteka5\_1\_klient\_ejb', including source packages like 'Warstwa\_klienta' and 'ejb'. The main editor window shows the source code of 'ramka1.java' with the following content:

```
5 package Warstwa_klienta;
6
7 import.ejb.Fasada_ejbRemote;
8 import java.util.ArrayList;
9 import javax.ejb.EJB;
10 import javax.swing.JComboBox;
11 /**...*/
15 public class ramka1 extends javax.swing.JFrame {
16     @EJB
17     private static Fasada_ejbRemote fasada;
18
19     public ramka1() {
20         try {
21             fasada.polaczenie_z_baza();
22             fasada.tabele();
23             fasada.uaktualnij_dane();
24         } catch (Exception e) {
25             System.out.println(e);
26         }
27         initComponents();
28     }
29 }
```

The status bar at the bottom indicates '2 | 10 | 30 | INS'.

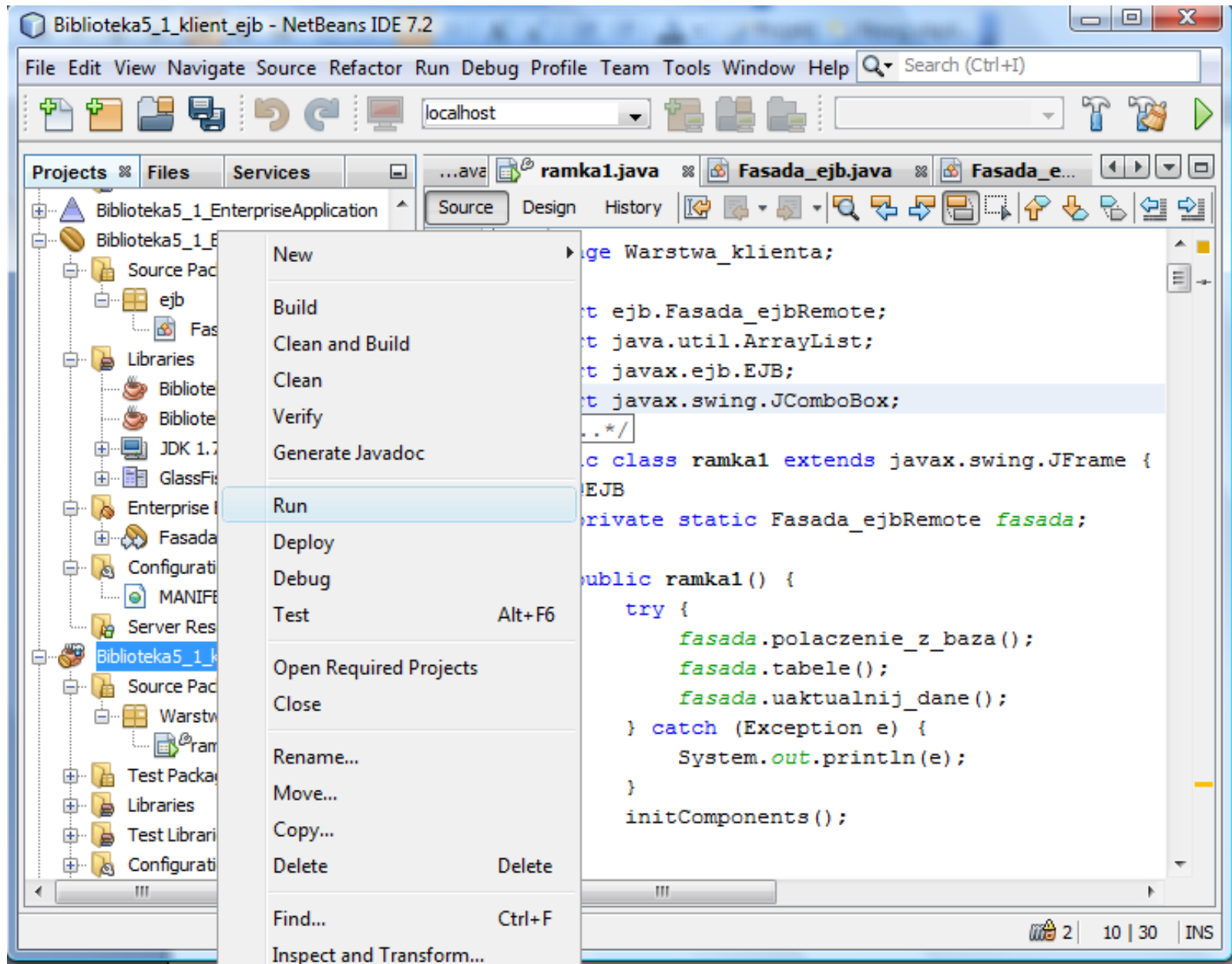
# 7. Uruchomienie aplikacji klienta typu Enterprise Edition Java EE 6

(1) Teraz można uruchomić aplikację klienta **Biblioteka5\_1\_klient\_ejb ()** - wykonanie operacji **Clean and Build**





(2) Po uruchomieniu aplikacji klienta **Biblioteka5\_1\_klient\_ejb** za pomocą **Run**



### (3) Działająca jedna instancja aplikacji **Biblioteka5\_1\_klient\_ejb** – dodawanie i wyświetlanie tytułów

The screenshot shows a window with a form for adding a book title. The form has the following fields and values:

Tytuł książki	22
Nazwisko autora książki	22
Imię autora książki	22
ISBN tytułu książki	222
Wydawnictwo książki	22
Numer książki	1

Buttons: Zapisz tytuł, Zapisz książkę, Wyświetl tytuły, Wyświetl książki, Tytuły do bazy, Książki do bazy, Tytuły z bazy, Książki z bazy.

Below the buttons, there are two sections: "Tytuły książek" and "Książki". The "Tytuły książek" section contains a list of entries, with the first one highlighted:

- id\_tytul 7 tytuł: 1 autor\_nazwisko: 1 autor\_imie: 1 ISBN: 1 wyd...

The screenshot shows the same window as the previous one, but with a dropdown menu open. The form fields and buttons are the same. The dropdown menu is open, showing a list of entries:

- Tytuł: 1 Autor: 1 1 ISBN: 1 Wydawnictwo: 1
- Tytuł: 1 Autor: 1 1 ISBN: 1 Wydawnictwo: 1
- Tytuł: 12 Autor: 21 12 ISBN: 2 Wydawnictwo: 2
- Tytuł: 2 Autor: 2 2 ISBN: 22 Wydawnictwo: 2
- Tytuł: 3 Autor: 3 3 ISBN: 33 Wydawnictwo: 3
- Tytuł: 4 Autor: 4 4 ISBN: 44 Wydawnictwo: 4
- Tytuł: 5 Autor: 5 5 ISBN: 55 Wydawnictwo: 5
- Tytuł: 22 Autor: 22 22 ISBN: 222 Wydawnictwo: 22

(4) Działająca jedna instancja aplikacji **Biblioteka5\_1\_klient\_ejb**– dodawanie i wyświetlanie książek

The screenshot shows a Java Swing application window with the following components:

- Input Fields:**
  - Tytuł książki: 22
  - Nazwisko autora książki: 22
  - Imię autora książki: 22
  - ISBN tytułu książki: 222
  - Wydawnictwo książki: 22
  - Numer książki: 1
- Buttons:**
  - Zapisz tytuł
  - Zapisz książkę
  - Wyświetl tytuły
  - Wyświetl książki
  - Tytuły do bazy
  - Książki do bazy
  - Tytuły z bazy
  - Książki z bazy
  - Dane z bazy
- Dropdown Menus:**
  - Tytuły książek:** Tytuł: 1 Autor: 1 1 ISBN: 1 Wydawnictwo: 1
  - Książki:** Tytuł: 1 Autor: 1 1 ISBN: 1 Wydawnictwo: 1 Numer: 1

The 'Książki' dropdown menu is expanded, showing a list of 10 items with their respective details:

- Tytuł: 1 Autor: 1 1 ISBN: 1 Wydawnictwo: 1 Numer: 11
- Tytuł: 1 Autor: 1 1 ISBN: 11 Wydawnictwo: 1 Numer: 1
- Tytuł: 1 Autor: 1 1 ISBN: 11 Wydawnictwo: 1 Numer: 2
- Tytuł: 2 Autor: 2 2 ISBN: 22 Wydawnictwo: 2 Numer: 2
- Tytuł: 3 Autor: 3 3 ISBN: 33 Wydawnictwo: 3 Numer: 1
- Tytuł: 4 Autor: 4 4 ISBN: 44 Wydawnictwo: 4 Numer: 1
- Tytuł: 5 Autor: 5 5 ISBN: 55 Wydawnictwo: 5 Numer: 1
- Tytuł: 22 Autor: 22 22 ISBN: 222 Wydawnictwo: 22 Numer: 1

(5) Uruchomione dwie aplikacje klienta **Biblioteka5\_1\_klient\_ejb** – aplikacja uruchomiona jako druga (za pomocą Run) ma dostęp do danych wprowadzonych przez pierwszą aplikację

Tytuł książki

Nazwisko autora książki

Imię autora książki

ISBN tytułu książki

Wydawnictwo książki

Numer książki

Zapisz tytuł    Zapisz książkę    Wyświetl tytuły    Wyświetl książki

Tytuły do bazy    Książki do bazy    Tytuły z bazy    Książki z bazy

Tytuły książek

Tytuł: 1 Autor: 1 1 ISBN: 1 Wydawnictwo: 1

Tytuł: 1 Autor: 1 1 ISBN: 1 Wydawnictwo: 1

Tytuł: 12 Autor: 21 12 ISBN: 2 Wydawnictwo: 2

Tytuł: 2 Autor: 2 2 ISBN: 22 Wydawnictwo: 2

Tytuł: 3 Autor: 3 3 ISBN: 33 Wydawnictwo: 3

Tytuł: 4 Autor: 4 4 ISBN: 44 Wydawnictwo: 4

Tytuł: 5 Autor: 5 5 ISBN: 55 Wydawnictwo: 5

Tytuł: 22 Autor: 22 22 ISBN: 222 Wydawnictwo: 22

Książki

Tytuł książki

Nazwisko autora książki

Imię autora książki

ISBN tytułu książki

Wydawnictwo książki

Numer książki

Zapisz tytuł    Zapisz książkę    Wyświetl tytuły    Wyświetl książki

Tytuły do bazy    Książki do bazy    Tytuły z bazy    Książki z bazy    Dane z bazy

Tytuły książek

Tytuł: 1 Autor: 1 1 ISBN: 1 Wydawnictwo: 1

Tytuł: 1 Autor: 1 1 ISBN: 1 Wydawnictwo: 1 Numer: 11

Tytuł: 1 Autor: 1 1 ISBN: 11 Wydawnictwo: 1 Numer: 1

Tytuł: 1 Autor: 1 1 ISBN: 11 Wydawnictwo: 1 Numer: 2

Tytuł: 2 Autor: 2 2 ISBN: 22 Wydawnictwo: 2 Numer: 2

Tytuł: 3 Autor: 3 3 ISBN: 33 Wydawnictwo: 3 Numer: 1

Tytuł: 4 Autor: 4 4 ISBN: 44 Wydawnictwo: 4 Numer: 1

Tytuł: 5 Autor: 5 5 ISBN: 55 Wydawnictwo: 5 Numer: 1

Tytuł: 22 Autor: 22 22 ISBN: 222 Wydawnictwo: 22 Numer: 1

Książki