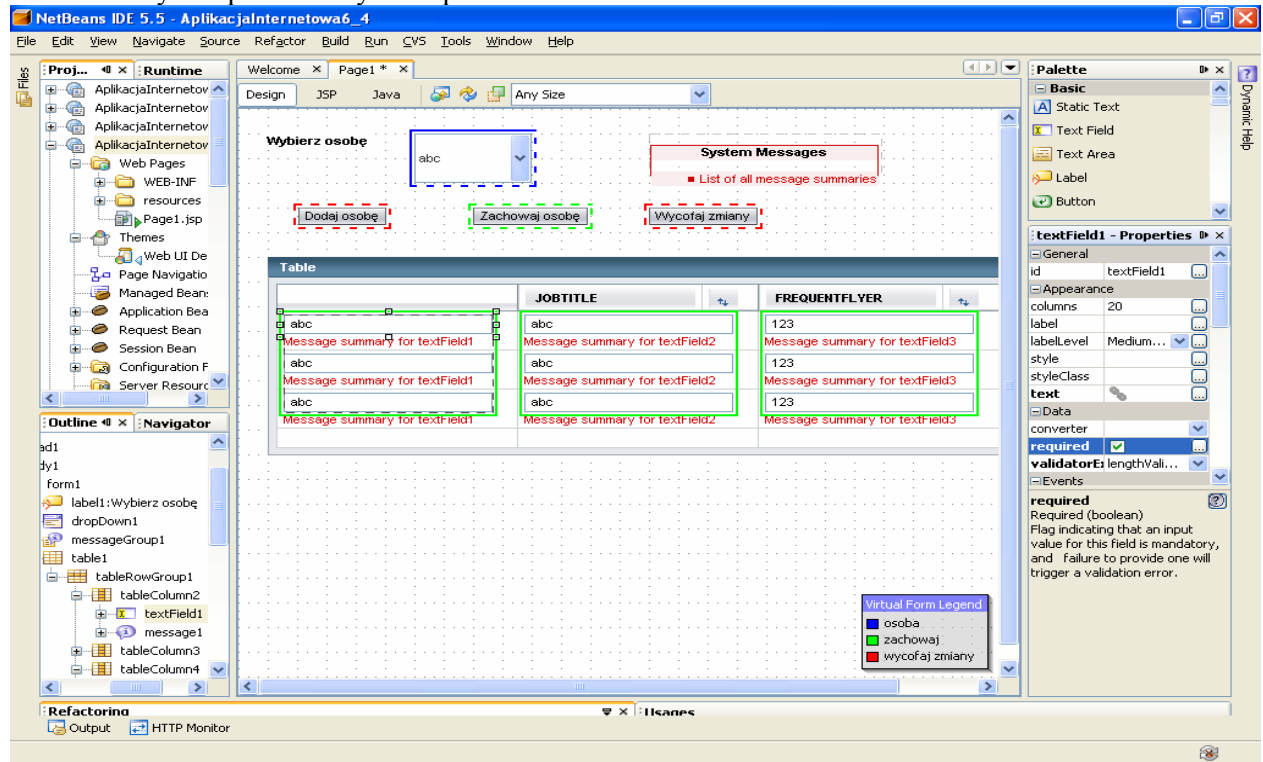


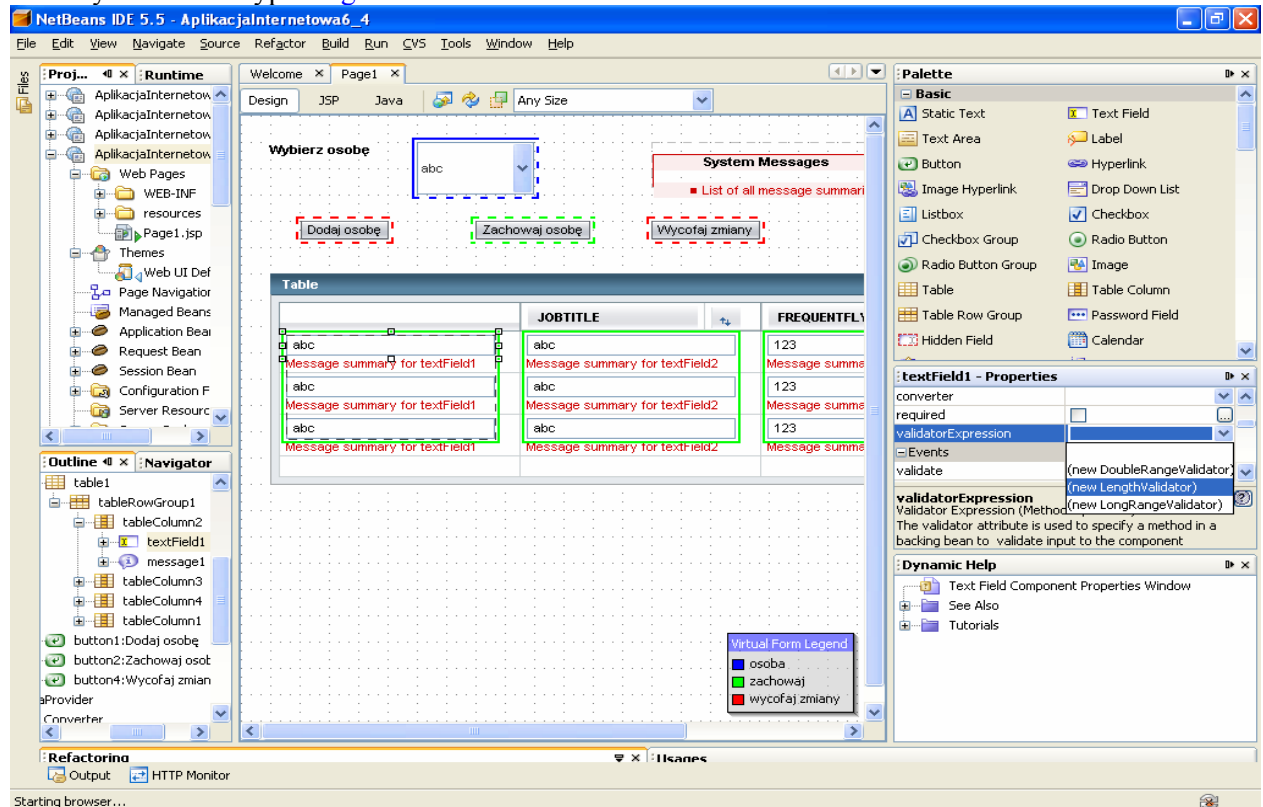
Lab5.

1. Ciąg dalszy laboratorium 4.

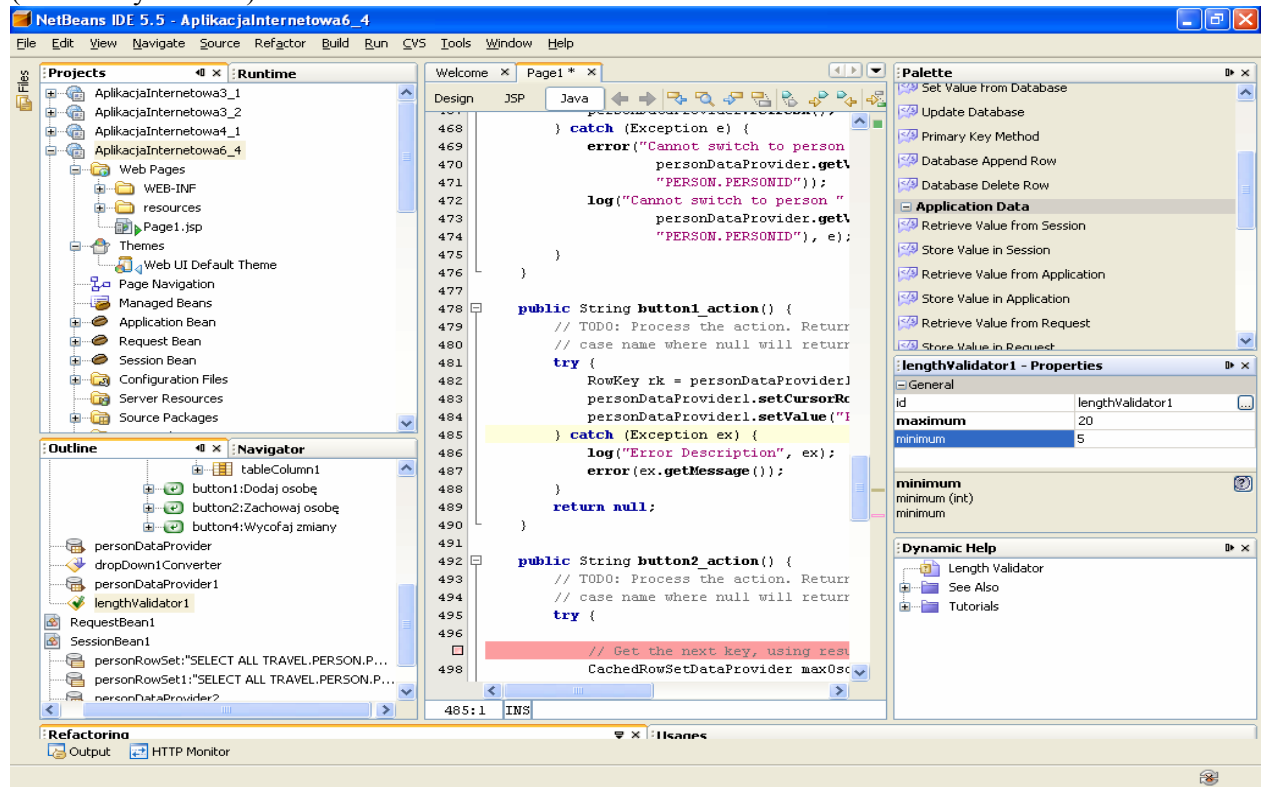
1.1 W trybie **Design** należy zaznaczyć komponent `textField1` w kolumnie **NAME** komponentu `table1`. W tabeli **Properties** komponentu `textField1` należy ustawić pole `required` typu `checkbox`. Zapobiega to przed brakiem danych wprowadzanych do pola **NAME** w tabeli **TRAVEL.PERSON**.



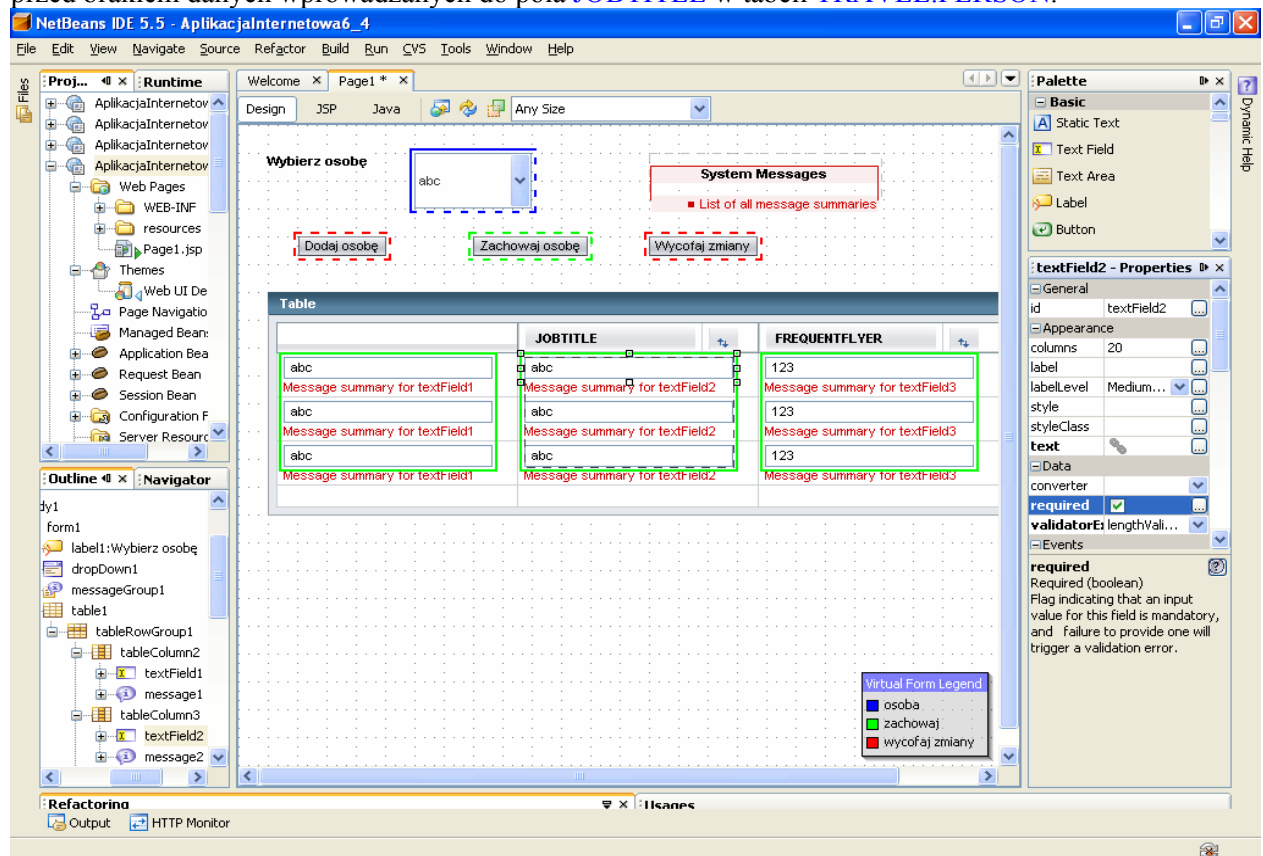
1.2 Wstaw walidator typu `lengthValidator` do sprawdzania długości łańcuchów wprowadzanych w kolumnie **Name** w komponencie `table1`. Należy w tabeli **Properties** wybrać listę `validatorExpression` i utworzyć walidator typu `lengthValidator`.



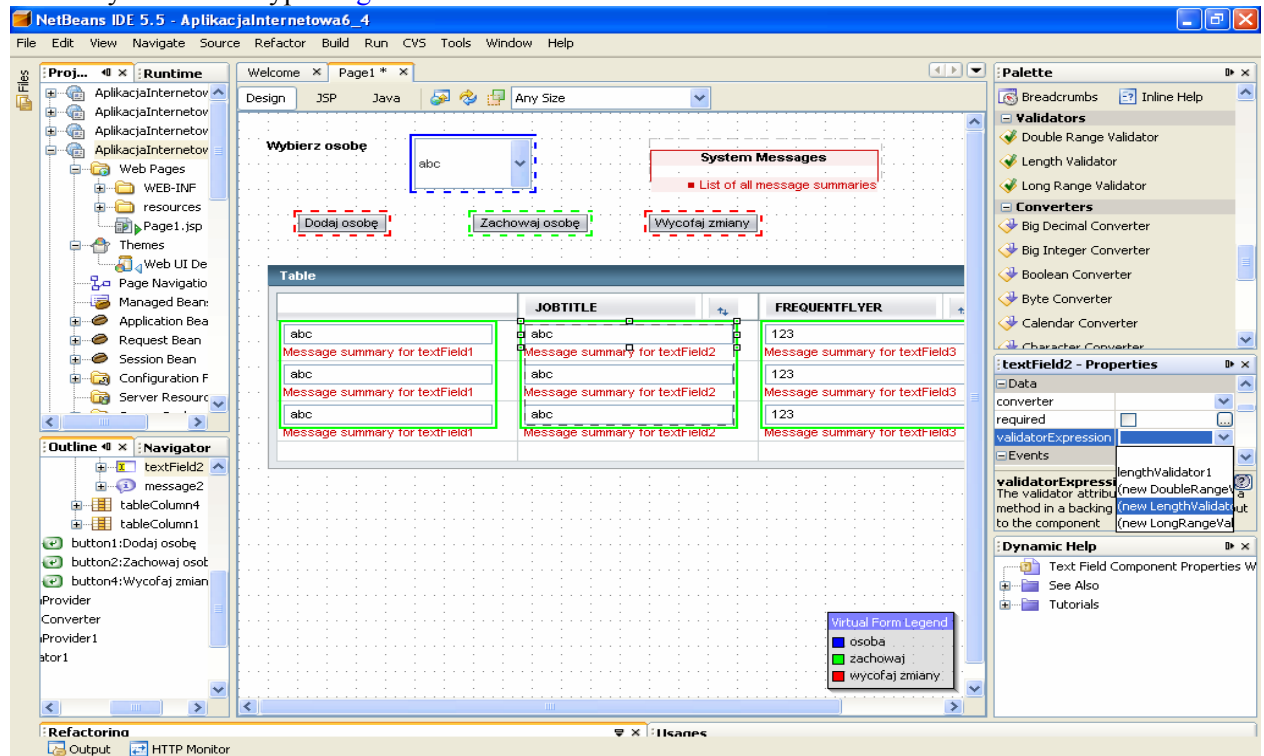
W tabeli **Properties** tego walidatora, wybranej przez kliknięcie w oknie **Outline** na ten komponent, należy w polu **maximum** wpisać wartość 20 (maksymalny łańcuch), natomiast w polu **minimum** należy wpisać 5 (minimalny łańcuch).



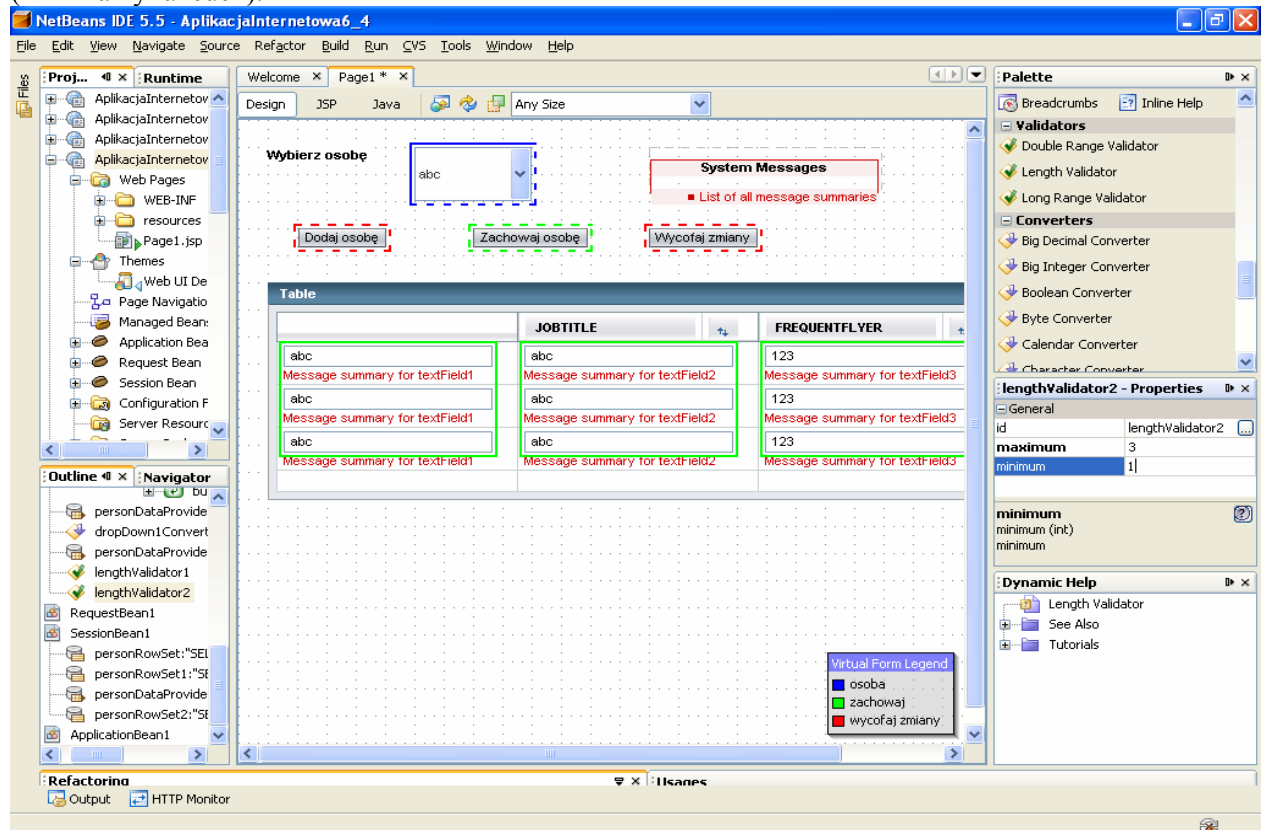
1.3. W trybie **Design** należy zaznaczyć komponent **textField2** w kolumnie **JOBTITLE** komponentu **table1**. W tabeli **Properties** komponentu **textField2** należy ustawić pole **required** typu **checkbox**. Zapobiega to przed brakiem danych wprowadzanych do pola **JOBTITLE** w tabeli **TRAVEL.PERSON**.



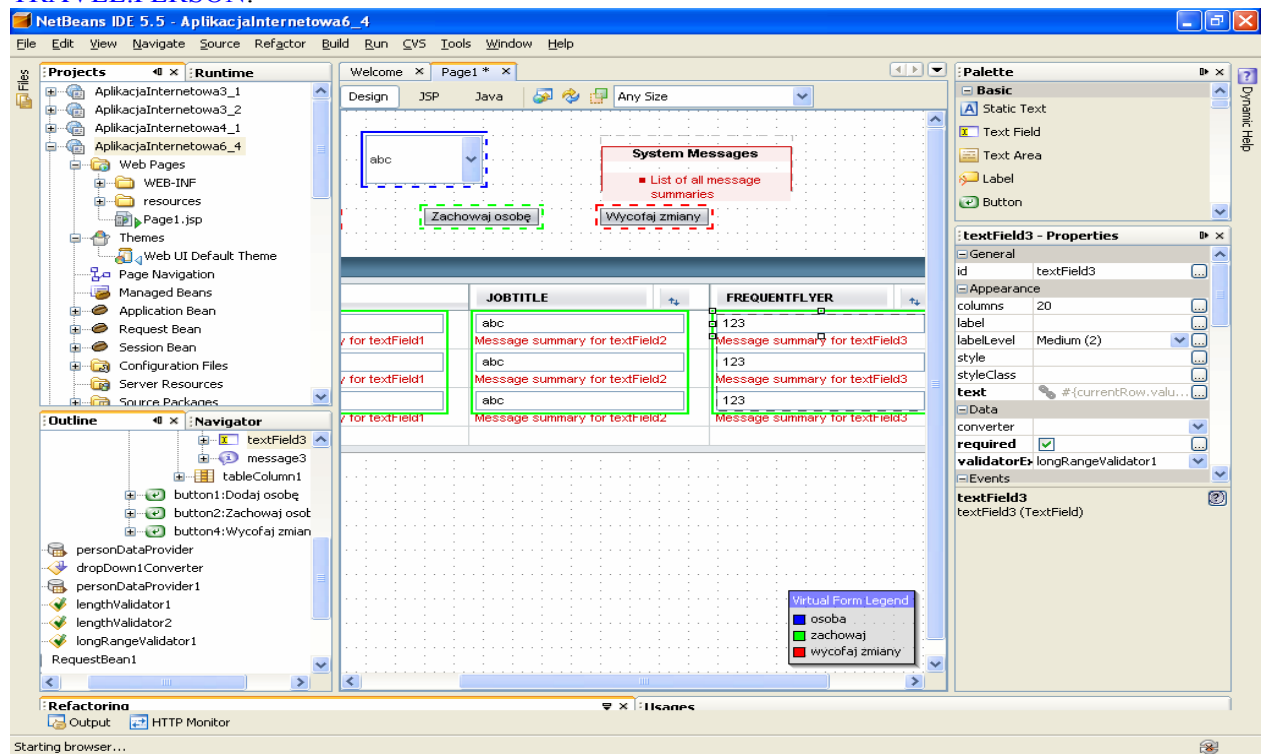
1.4. Wstaw walidator typu `lengthValidator` do sprawdzania długości łańcuchów wprowadzanych w kolumnie `JOBTITLE` w komponencie `table1`. Należy w tabeli `Properties` wybrać listę `validatorExpression` i utworzyć walidator typu `lengthValidator`.



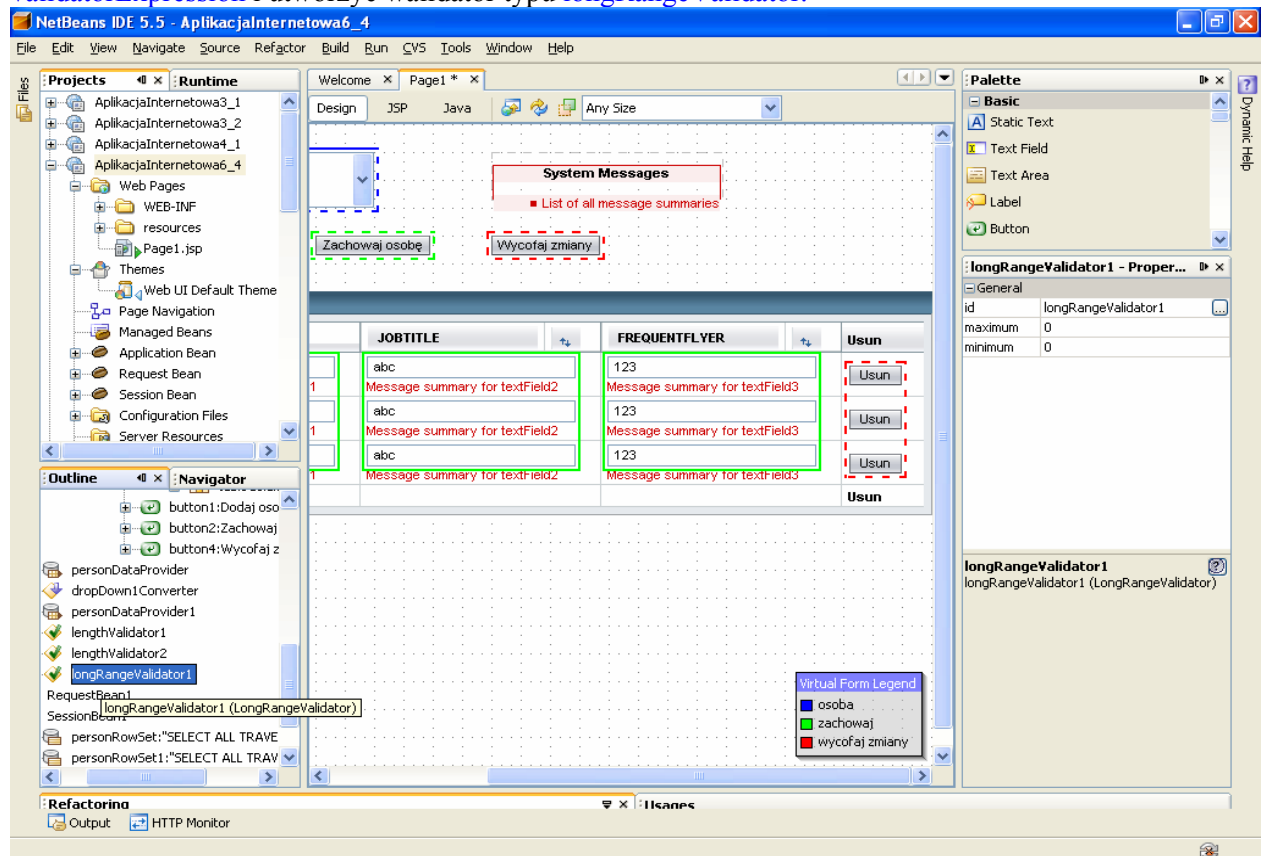
W tabeli `Properties` tego walidatora, wybranej przez kliknięcie w oknie `Outline` na ten komponent, należy w polu `maximum` wpisać wartość 3 (maksymalny łańcuch), natomiast w polu `minimum` należy wpisać 1 (minimalny łańcuch).



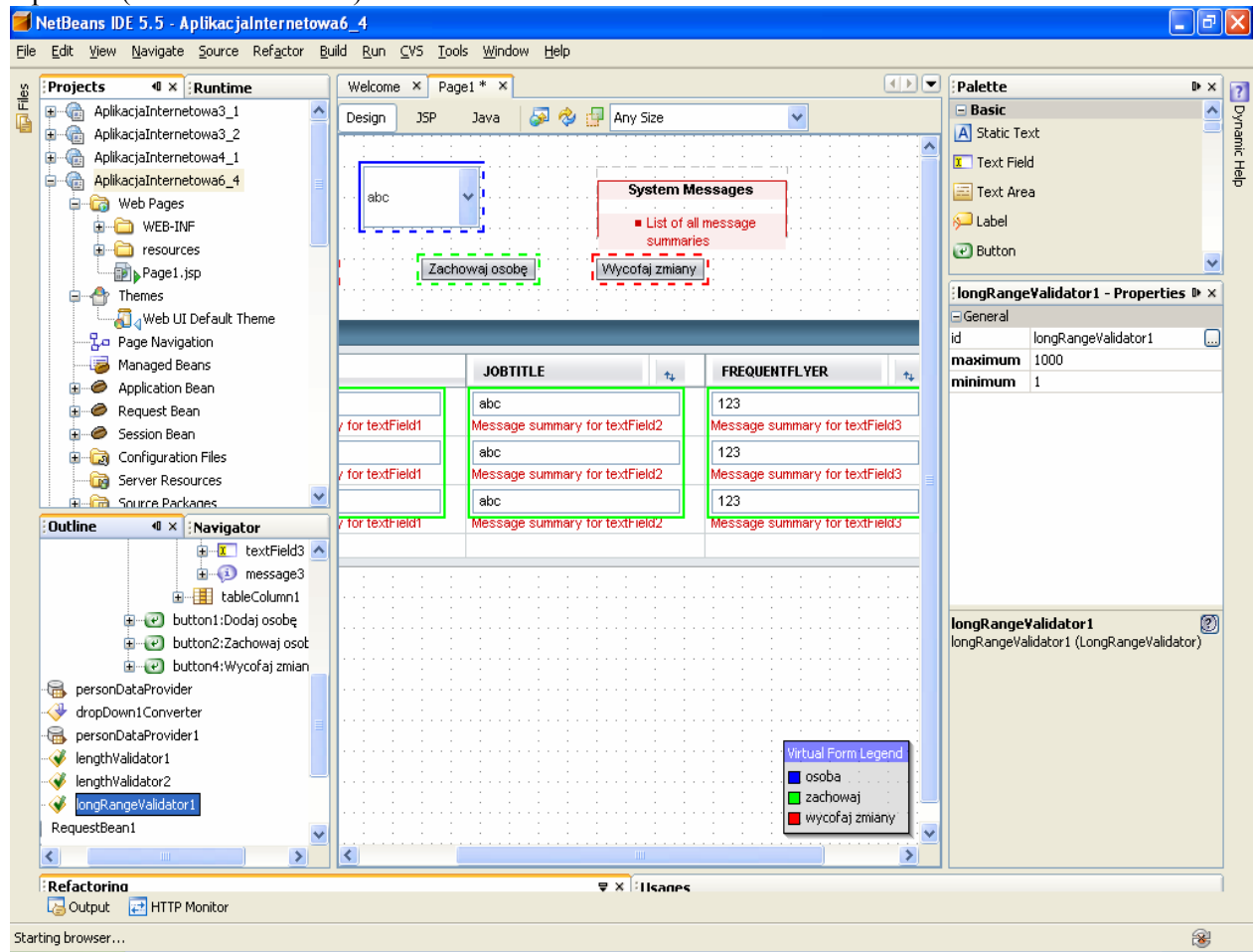
1.5. W trybie Design należy zaznaczyć komponent `textField3` w kolumnie `FREQUENTFLYER` komponentu `table1`. W tabeli `Properties` komponentu `textField3` należy ustawić pole `required` typu `checkbox`. Zapobiega to przed brakiem danych wprowadzanych do pola `FREQUENTFLYER` w tabeli `TRAVEL.PERSON`.



1.6. Wstaw walidator typu `longRangeValidator` do sprawdzania zakresu wartości wprowadzanych w kolumnie `FREQUENTFLYER` w komponentcie `table1`. Należy w tabeli `Properties` wybrać listę `validatorExpression` i utworzyć walidator typu `longRangeValidator`.

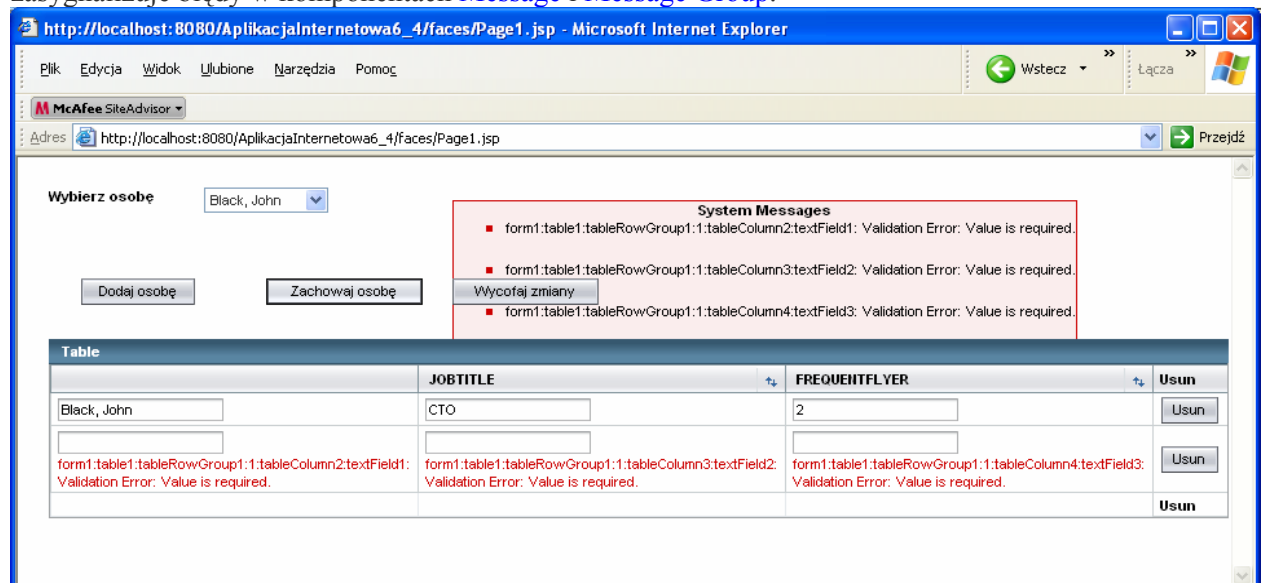


W tabeli **Properties** tego walidatora, wybranej przez kliknięcie w oknie **Outline** na ten komponent, należy w polu **maximum** wpisać wartość 1000 (maksymalna wartość), natomiast w polu **minimum** należy wpisać 1 (minimalna wartość).



1.7. Uruchom aplikację klikając prawym klawiszem myszy na nazwę projektu w okienku **Project**. Uruchom kolejno **Build/Deploy/Run** lub tylko **Run**, co powoduje uruchomienie tej całej sekwencji działań i uruchomienie aplikacji w domyślnej przeglądarce.

Wykonaj test wstawiania pustych pól- naciśnij klawisz **Dodaj osobę** i potem **Zachowaj osobę**. Aplikacja zasygnalizuje błędy w komponentach **Message** i **Message Group**.



Wykonaj test rozmiaru łańcucha w pierwszej kolumnie

Wybierz osobę: Black, John

Dodaj osobę Zachowaj osobę Wycofaj zmiany

System Messages

- form1.table1.tableRowGroup1:1.tableColumn2:textField1: Validation Error: Value is less than allowable minimum of '5'

	JOBTITLE	FREQUENTFLYER	Usun
Black, John	CTO	2	Usun
aaa	aa	1	Usun
			Usun

Wykonaj test rozmiaru łańcucha w drugiej kolumnie.

Wybierz osobę: Black, John

Dodaj osobę Zachowaj osobę Wycofaj zmiany

System Messages

- form1.table1.tableRowGroup1:1.tableColumn3:textField2: Validation Error: Value is greater than allowable maximum of '3'

	JOBTITLE	FREQUENTFLYER	Usun
Black, John	CTO	2	Usun
aaaaaaaa	dolddd	1	Usun
			Usun

Wykonaj test trzeciej kolumny

Wybierz osobę: Black, John

Dodaj osobę Zachowaj osobę Wycofaj zmiany

System Messages

- form1:table1:tableRowGroup1:1:tableColumn4:textField3: Validation Error: Value is greater than allowable maximum of '1 000'

	JOBTITLE	FREQUENTFLYER	Usun
Black, John	CTO	2	Usun
aaaaaaaa	bb	1000000	Usun
			Usun

Wybierz osobę: Kent, Richard

Dodaj osobę Zachowaj osobę Wycofaj zmiany

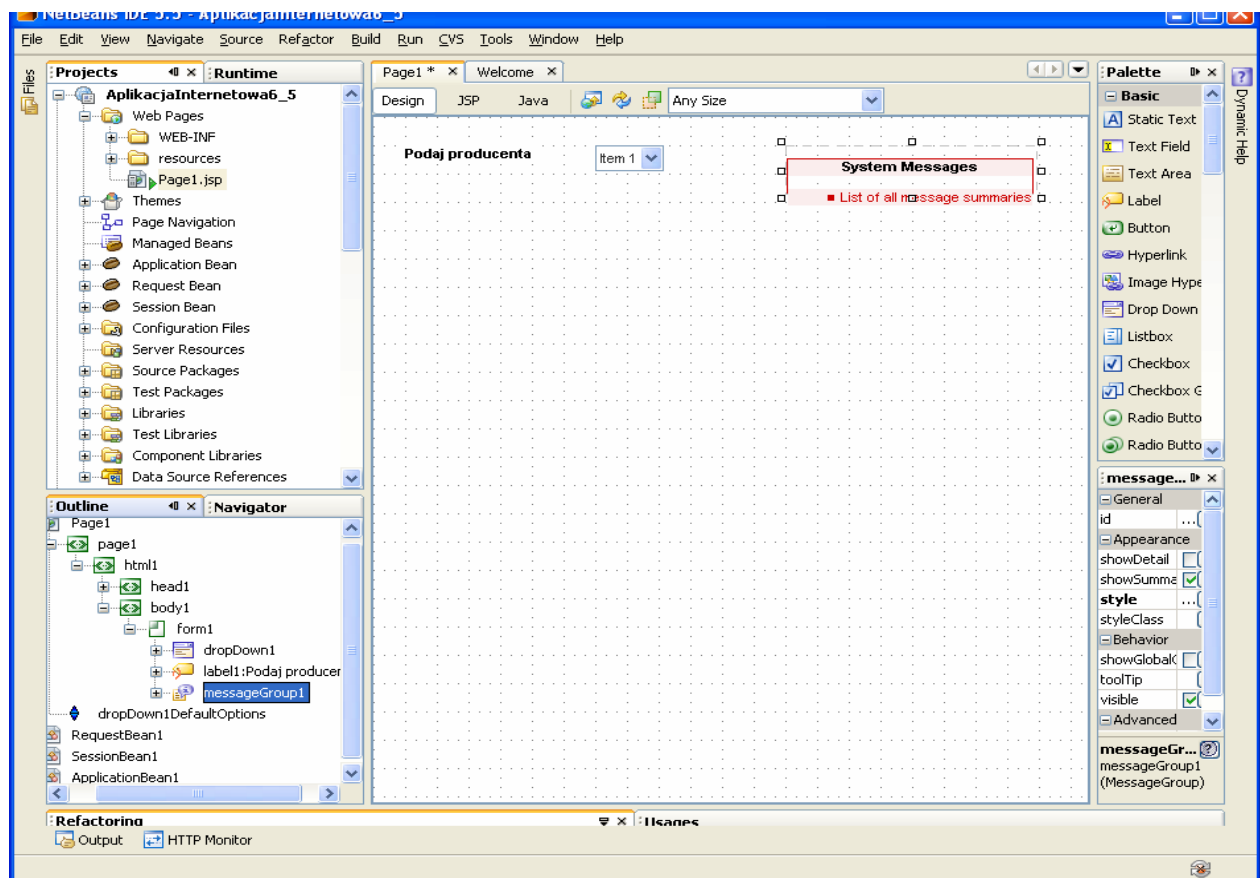
System Messages

- form1:table1:tableRowGroup1:1:tableColumn4:textField3: Validation Error: Specified attribute is not between the expected values of 1 and 1 000.

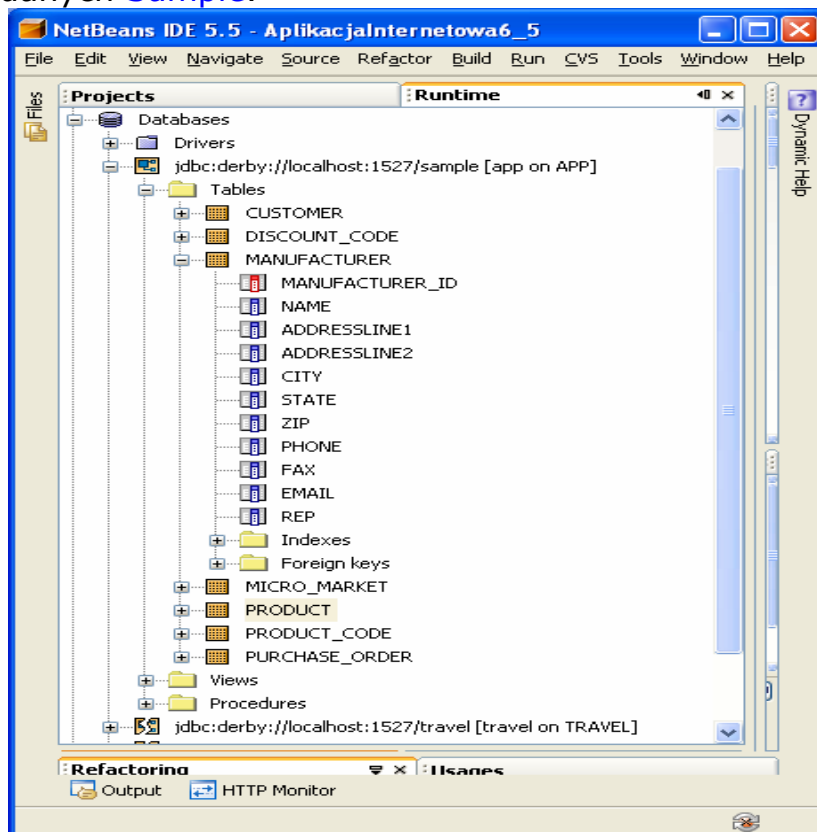
	JOBTITLE	FREQUENTFLYER	Usun
Kent, Richard	VP	1	Usun
ccccccc	aa	0	Usun
			Usun

Zad. 2. Relacja jeden do wiele (patrz laboratorium 3)

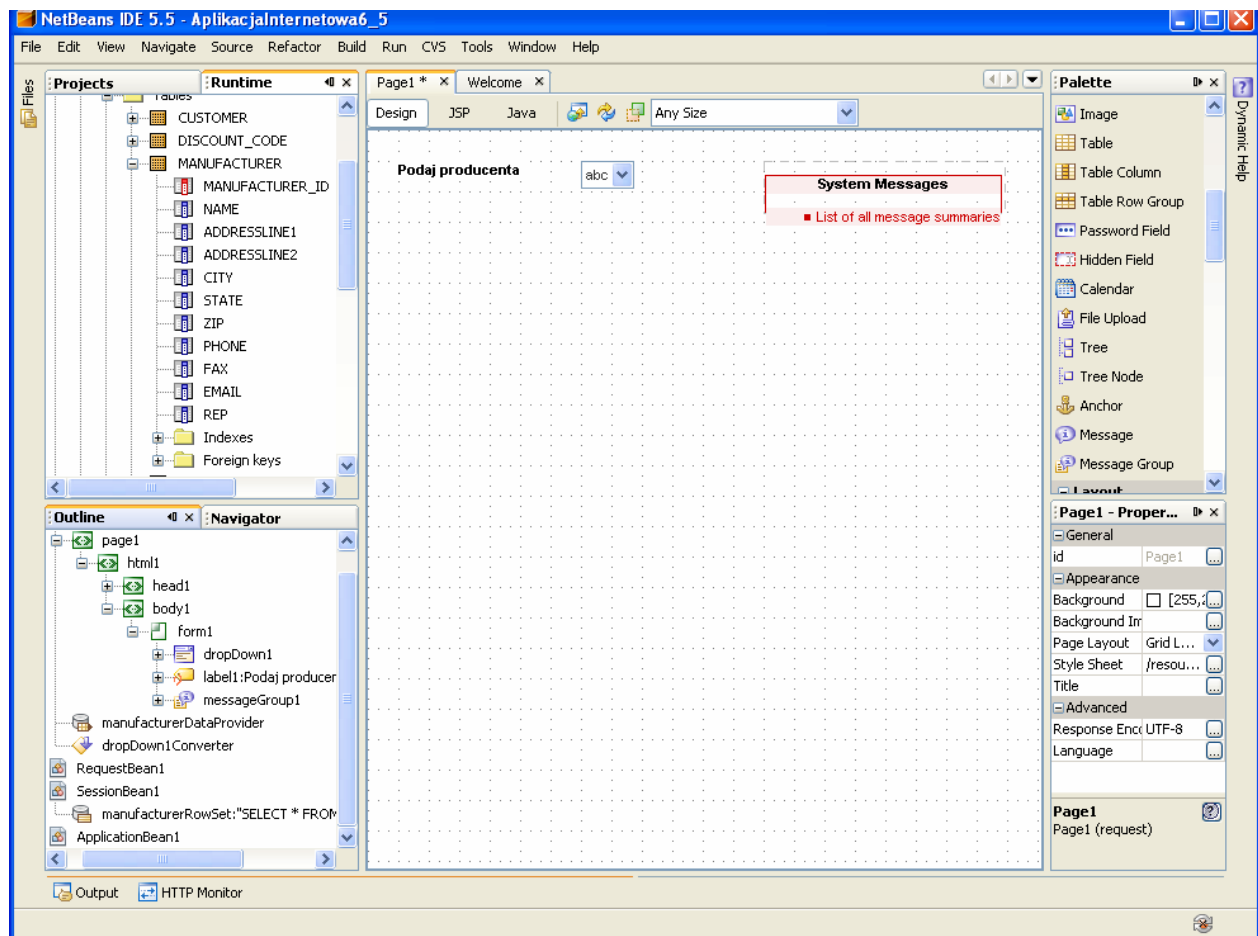
1. Wybierz opcję **File/New Project**. Wybierz kategorię projektu **Web**, a rodzaj projektu **Visual Web Application**
2. Kliknij na **Next**
3. Podaj nazwę projektu (**Project name**), wybierz katalog (**Project Location**)- np. **AplikacjaInternetowa6_5**
4. Wybierz serwer aplikacji (**Server**) oraz wersję Javy Enterprise (**Java EE Version**).
5. Okno projektu (**Projects**) zawiera układ plików typu **BluePrints**. Plik **Page1.jsp** jest stroną startową napisaną w języku JSP.
6. Okno **Files** zawiera układ fizyczny plików. Plik **Page1.jsp** oraz **Page1.java** stanowią całość- plik **Page1.java** obsługuje główną stronę jsp.
7. Zaprojektuj stronę **Page1** korzystając z **Palette Basic**
 - 7.1. Przeciągnij komponenty **Label** i **Drop Down List**. W oknach **Properties** obu komponentów nazwy ich są standardowe: **label1** i **dropDown1**. Nazwy te można zmieniać. Na komponencie **Label** napisz np. *Podaj producenta*. Naciśnij klawisze **Ctrl+Shift+przeciągnij** utworzoną etykietę **label1** na komponent typu **Drop Down List**. Pole **for** w oknie **Properties** komponentu **label1** zawiera nazwę **dropDown1**.
 - 7.2. Przeciągnij komponent **Message Group**, który będzie wyświetlał komunikaty metod **info(String)**, **error(String)**, **warn(String)**, lub **fatal(String)**. Domyślnie, wyświetlane są komunikaty typu błędy wykonania, błędy walidacji oraz błędy konwersji.



- 7.3. Połącz komponent `dropDown1` z bazą danych `Sample`. W tym celu należy kliknąć na zakładkę `Runtime`, następnie rozwinąć opcję `Databases` (kliknąć na przycisk `+`) i prawym klawiszem myszy kliknąć na bazę danych `jdbc:derby://localhost:1527/sample [app on App]`. Na wyskakującym menu kliknąć na opcję `Connect`. W kolejnym okienku wpisać hasło `app` w polu `password`. Po połączeniu z bazą danych uzyskuje się dostęp do tabel (`Tables`), widoków (`Views`) oraz składowanych procedur (`Procedures`) po kliknięciu na „przycisk `+`” bazy danych `Sample`.

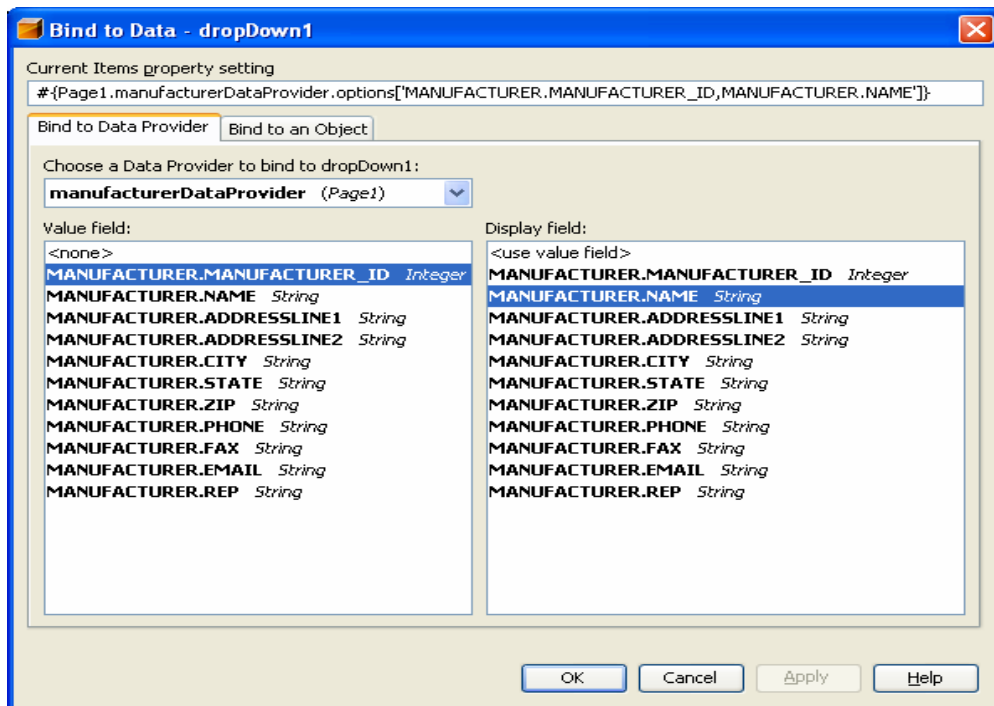


- 7.4. Przeciągnij tabelę `MANUFACTURER` z zakładki `Runtime` na komponent `dropDown1`. Na komponencie pokazały się elementy listy typu `abc`, co oznacza połączenie z łańcuchami typu `varchar` kolumn wybranej tabeli `MANUFACTURER`.

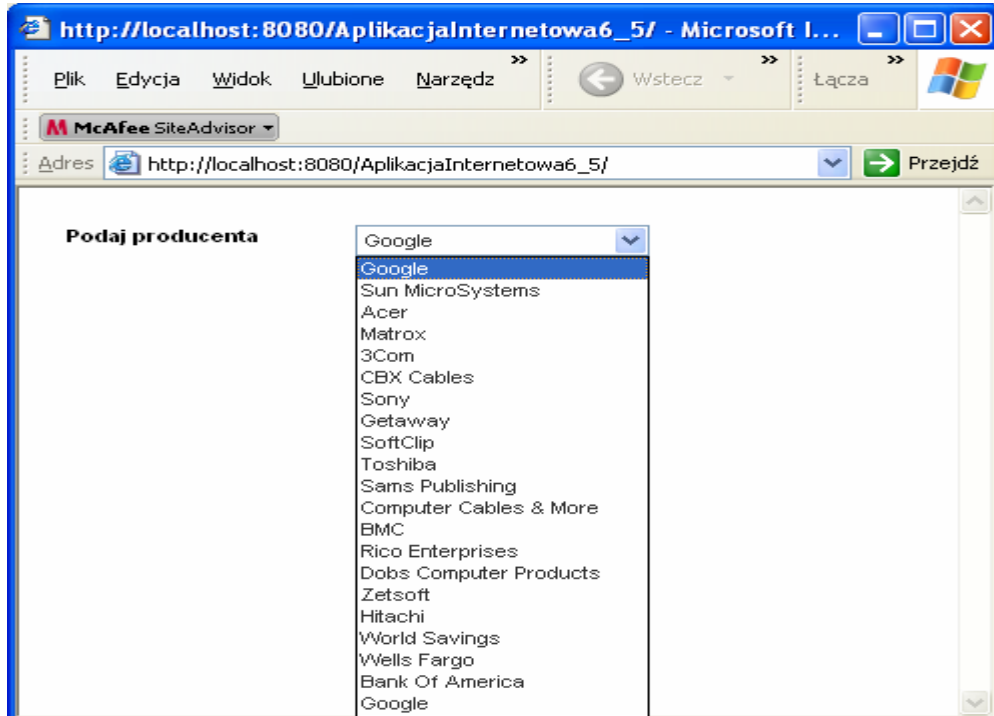


Pojawił się komponent `manufacturerDataProvider`, odpowiedzialny za połączenia z warstwa biznesową (obiekty EJB, tablice Arrays) oraz właściwość `manufacturerRowSet` w obiekcie `SessionBean1`, odpowiedzialny za połączenie i obsługę zapytań do tabel w bazie danych (wykonanie zapytania oraz zarządzanie jego wynikiem).

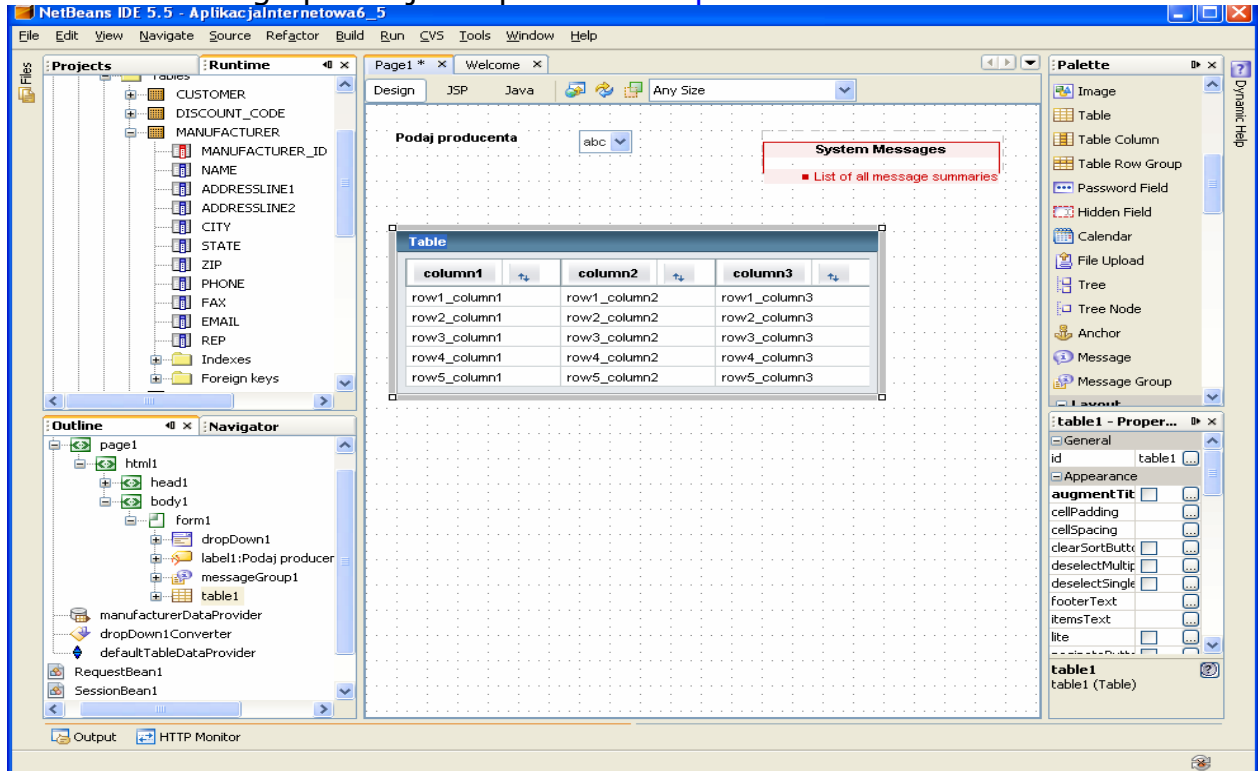
- 7.5. Prawym klawiszem myszy kliknąć na komponencie `dropDownList` i z wyskakującego menu wybrać `Bind To Data`. W wywołanym formularzu wybrać zakładkę `Value Field` i zaznaczyć `MANUFACTURER.MANUFACTURER_ID` jako domyślna wartość w programie oraz w zakładce `Display` wybrać `MANUFACTURER.NAME` jako kolumnę do wyświetlania na pozycjach komponentu `dropDown1`.



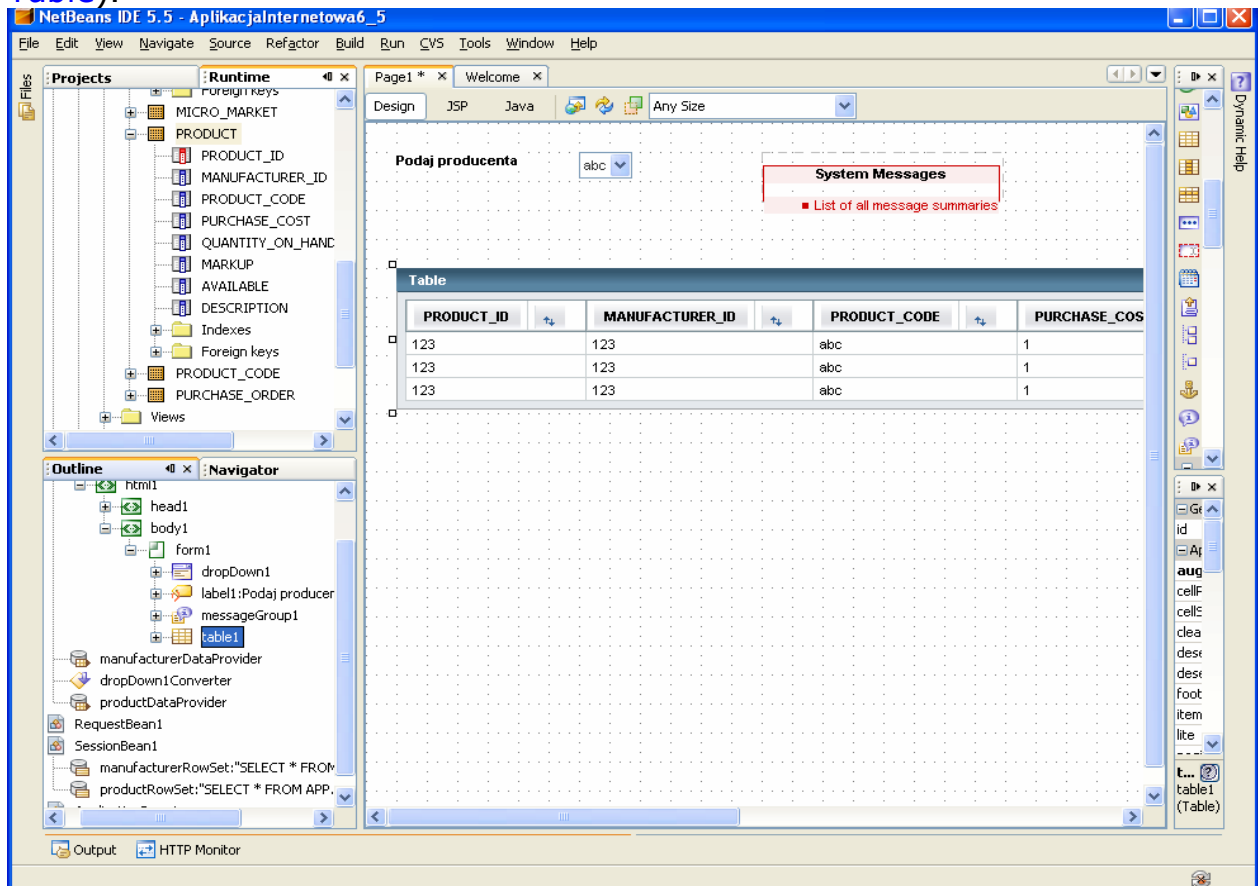
7.6. Uruchom aplikację (Kliknij prawym klawiszem myszy w oknie [Project](#) na nazwę projektu, w ukazanym oknie uruchom kolejno [Build Project](#), [Deploy Project](#), [Run Project](#) lub tylko [Run](#)) i uruchom poszczególne funkcje aplikacji.)



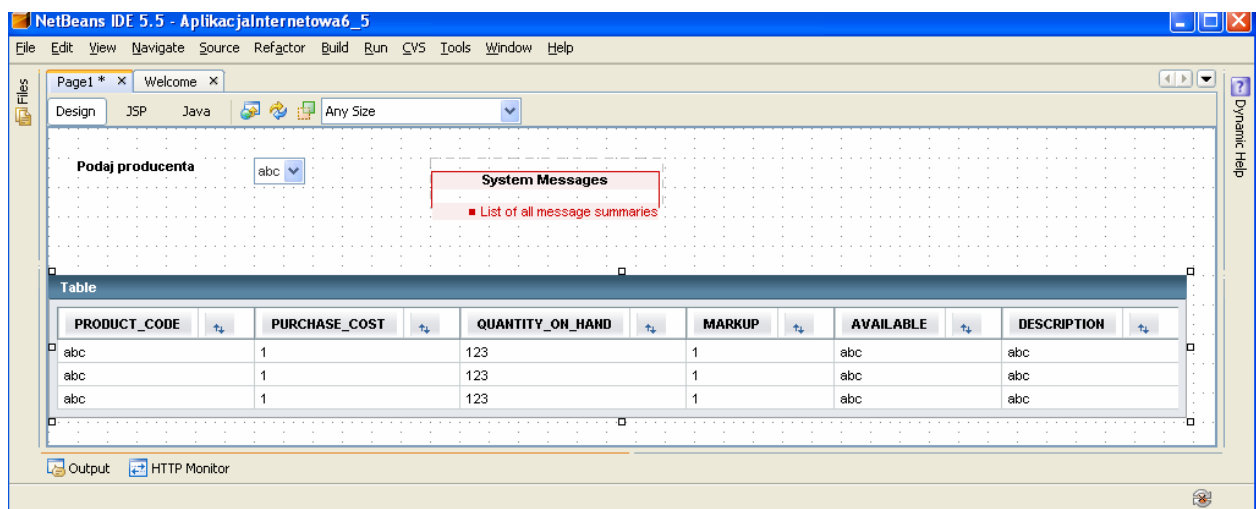
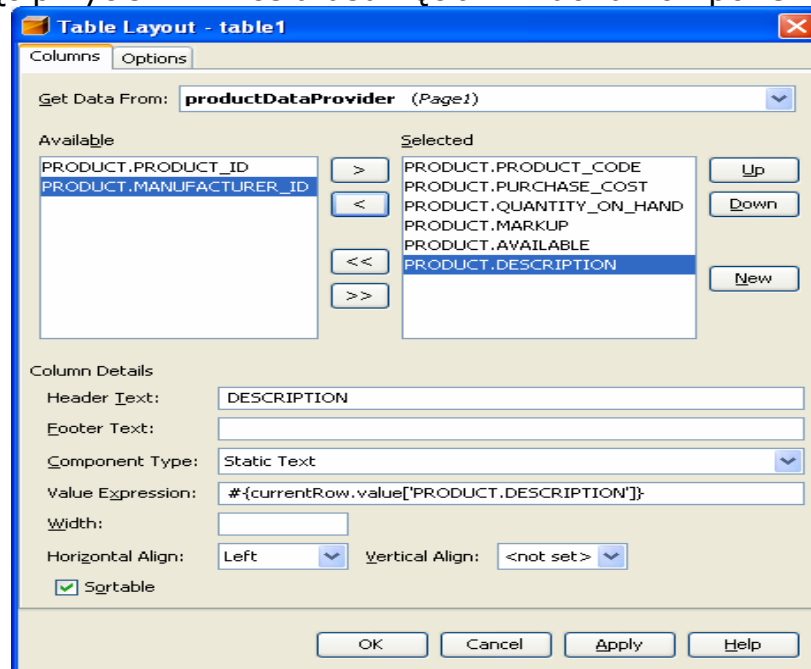
7.7. Przeciągnij komponent **Table** na stronę **Page1** w trybie **Design** i umieść go poniżej komponentu **dropDown1**.



W oknie **Runtime** przeciągnij tabelę **PRODUCT** na komponent **table1** (typu **Table**).



7.8. Kliknij prawym klawiszem na komponent `table1` i wybierz **Table Layout**. W ukazanym formularzu należy wybrać kolumny tabeli do wyświetlenia. Za pomocą **Ctrl+kliknięcie** wybrać następujące kolumny: `PRODUCT.MANUFACTURES_ID` i `PRODUCT.PRODUCT_ID` i nacisnąć przycisk **<** w celu usunięcia z widoku komponentu `table1`



7.9. Podgląd i możliwość edycji zapytania SQL dla warstwy bazodanowej `productRowSet` skojarzonej z `SessionBean1`. Należy wybrać w okienku **Outline** komponent `SessionBean1` i wybrać `productRowSet` prawym klawiszem myszy. Z wyskakującego menu wybrać opcję **Edit SQL Statement**.

NetBeans IDE 5.5

File Edit View Navigate Source Refactor Build Run CVS Tools Window Help

Projects: Runtime
 MICRO_MARKET
 PRODUCT
 PRODUCT_ID
 MANUFACTURER_ID
 PRODUCT_CODE
 PURCHASE_COST
 QUANTITY_ON_HAND
 MARKUP
 AVAILABLE
 DESCRIPTION
 Indexes
 Foreign keys
 PRODUCT_CODE
 PURCHASE_ORDER

Outline: Navigator
 html1
 head1
 body1
 form1
 dropDown1
 label1: Podaj produc
 messageGroup1
 table1
 manufacturerDataProvider
 dropDown1Converter
 productDataProvider
 RequestBean1
 SessionBean1
 manufacturerRowSet:"SELECT * FR
 productRowSet:"SELECT ALL APP.PR
 ApplicationBean1

Page1 * x Welcome x
 Design JSP Java Any Size

Podaj producenta abc

System Messages
 List of all message summaries

PRODUCT_CODE	PURCHASE_COST	QUANTITY_ON_HAND	MARKUP
abc	1	123	1
abc	1	123	1
abc	1	123	1

Palette
 Image
 Table
 Table Column
 Table Row Group
 Password Field
 Hidden Field
 Calendar
 File Upload
 Tree
 Tree Node
 Anchor
 Message
 Message Group
 Layout
 Grid Panel

productRowSet: "...
 General
 id prod...
 command SELE...
 concurrency CO...
 dataSource sam...
 fetchSize 0
 maxRows 0
 pageSize 0
 password
 showDeleted
 productRowSet: "SE...
 cachedRowSetXImpl3
 (CachedRowSetImpl)

Output HTTP Monitor

NetBeans IDE 5.5

File Edit View Navigate Source Refactor Build Run CVS Tools Window Help

Projects: Runtime
 MICRO_MARKET
 PRODUCT
 PRODUCT_ID
 MANUFACTURER_ID
 PRODUCT_CODE
 PURCHASE_COST
 QUANTITY_ON_HAND
 MARKUP
 AVAILABLE
 DESCRIPTION
 Indexes
 Foreign keys
 PRODUCT_CODE
 PURCHASE_ORDER

Outline: Navigator
 <No View Available>

Page1 * x Welcome x productRowSet (SessionBean1) x

PRODUCT
 PRODUCT_ID
 MANUFACTURER_ID
 PRODUCT_CODE
 PURCHASE_COST

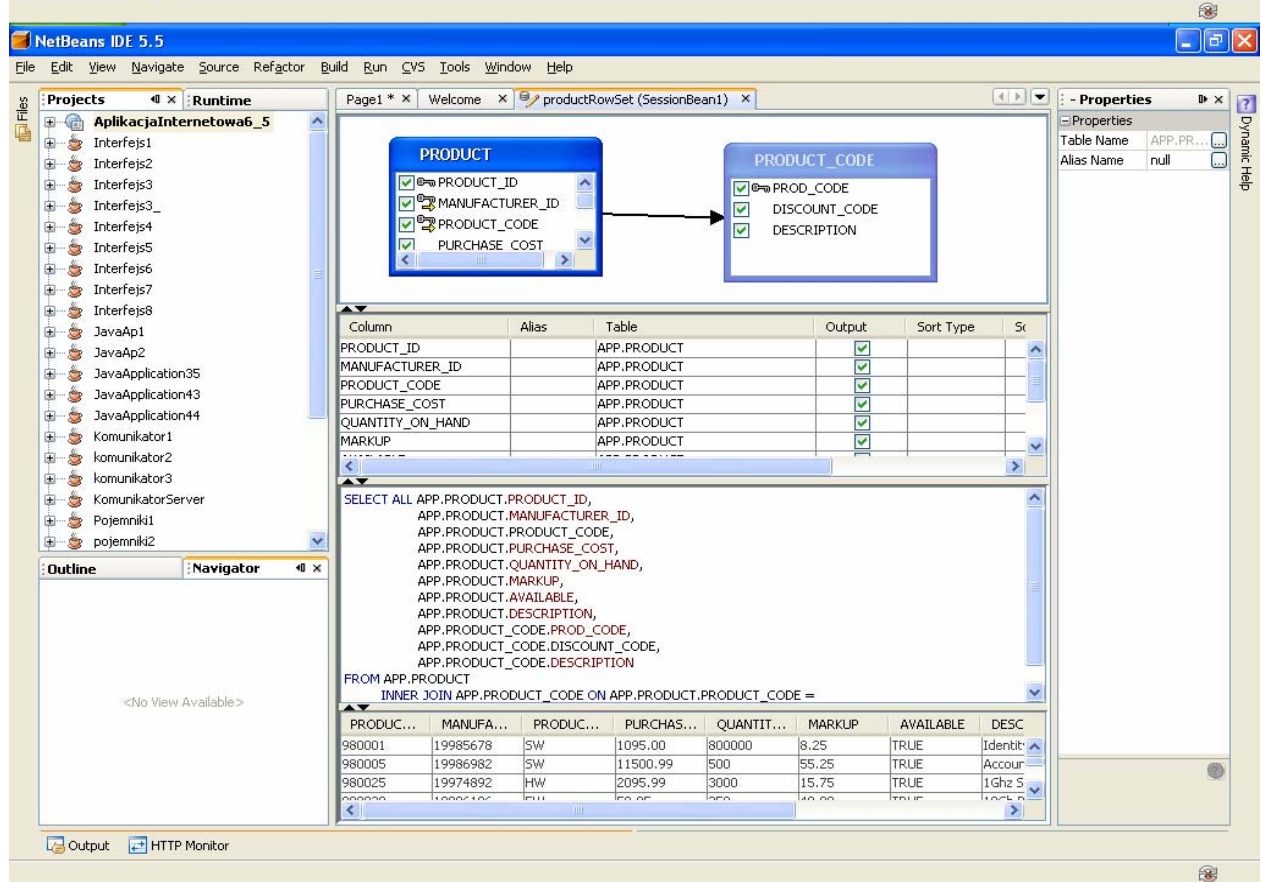
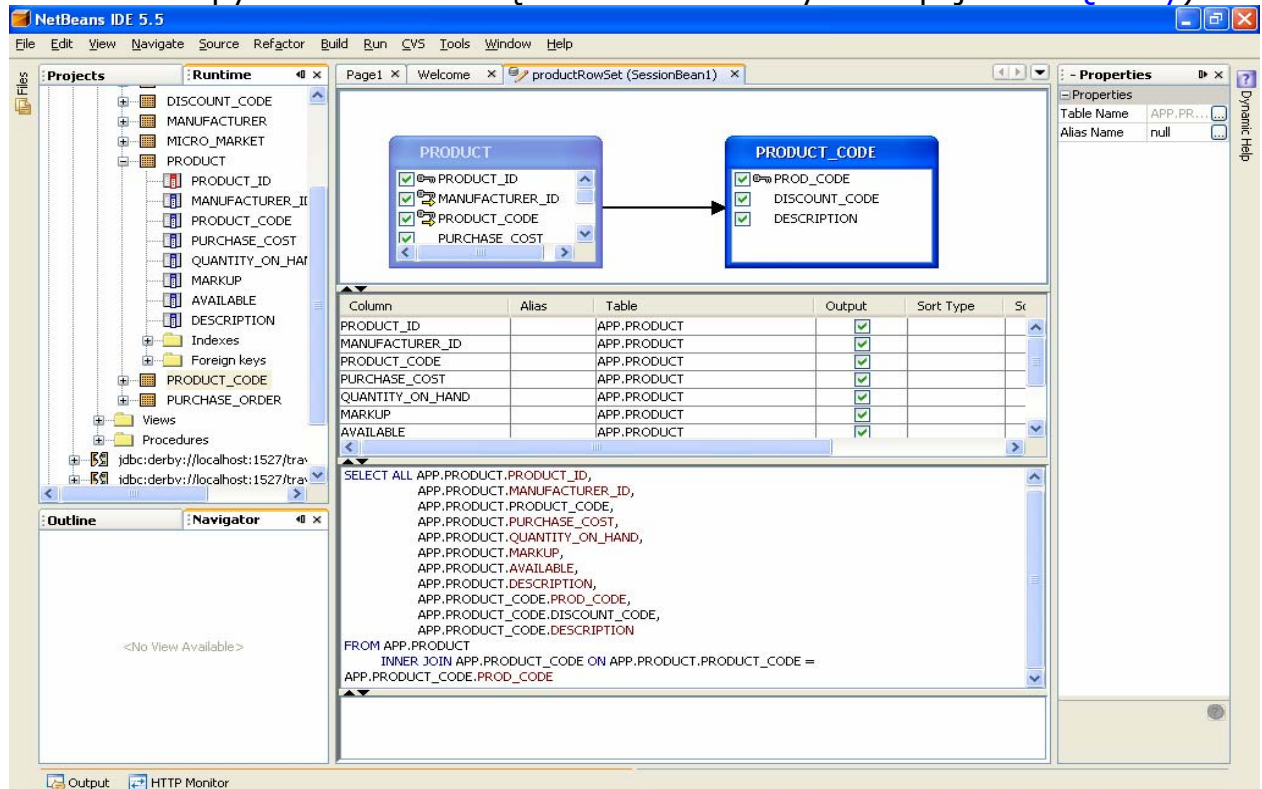
Column	Alias	Table	Output	Sort Type	Sort
PRODUCT_ID		APP.PRODUCT	<input checked="" type="checkbox"/>		
MANUFACTURER_ID		APP.PRODUCT	<input checked="" type="checkbox"/>		
PRODUCT_CODE		APP.PRODUCT	<input checked="" type="checkbox"/>		
PURCHASE_COST		APP.PRODUCT	<input checked="" type="checkbox"/>		
QUANTITY_ON_HAND		APP.PRODUCT	<input checked="" type="checkbox"/>		
MARKUP		APP.PRODUCT	<input checked="" type="checkbox"/>		
AVAILABLE		APP.PRODUCT	<input checked="" type="checkbox"/>		
DESCRIPTION		APP.PRODUCT	<input checked="" type="checkbox"/>		

```
SELECT ALL APP.PRODUCT.PRODUCT_ID,
APP.PRODUCT.MANUFACTURER_ID,
APP.PRODUCT.PRODUCT_CODE,
APP.PRODUCT.PURCHASE_COST,
APP.PRODUCT.QUANTITY_ON_HAND,
APP.PRODUCT.MARKUP,
APP.PRODUCT.AVAILABLE,
APP.PRODUCT.DESCRPTION
FROM APP.PRODUCT |
```

Properties
 Table Name APP.PR...
 Alias Name null

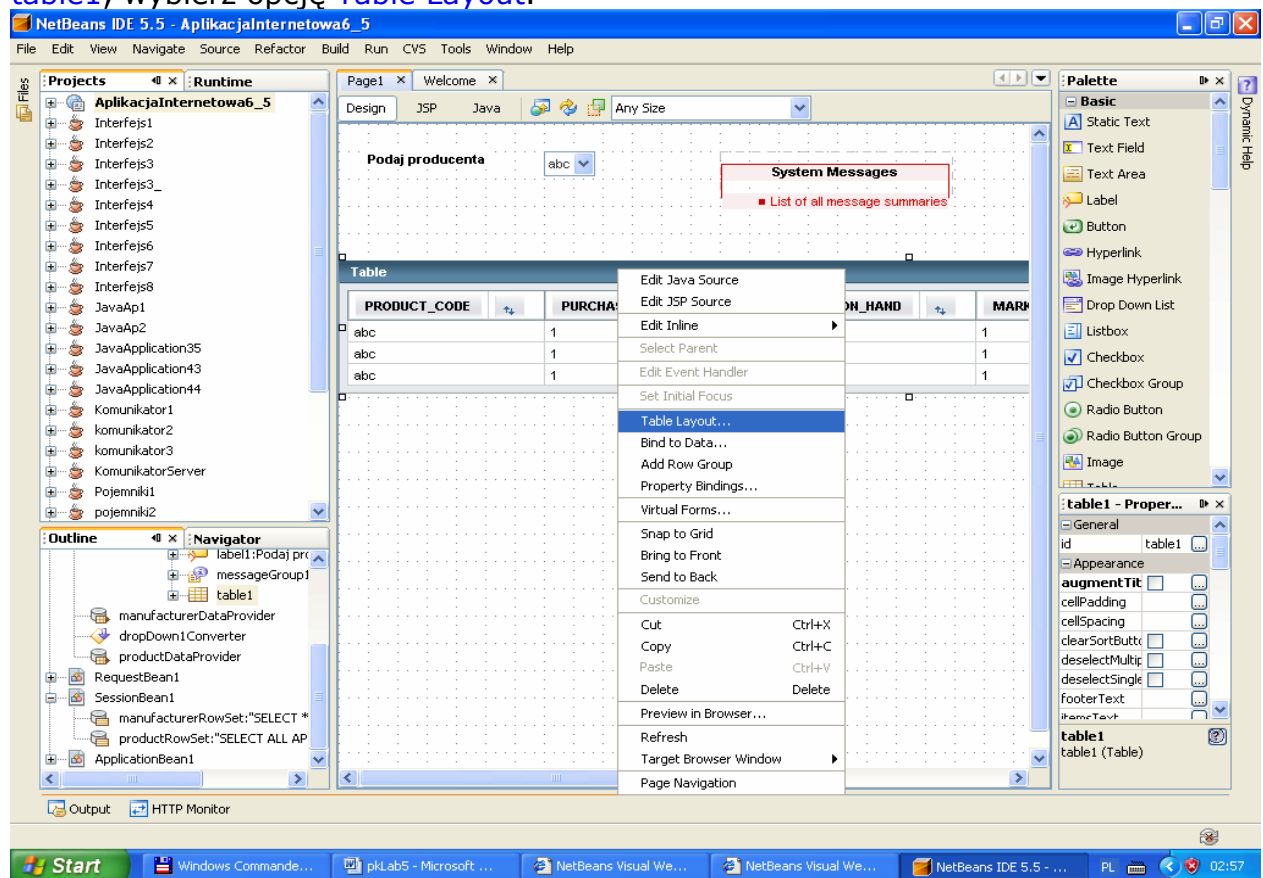
Output HTTP Monitor

Należy przeciągnąć tabelę **PRODUCT_CODE** i wykonać zapytanie bezpośrednio na bazie danych (prawym klawiszem myszy kliknąć na obszar zapytania – dolna część formularza i wybrać opcje **Run Query**)

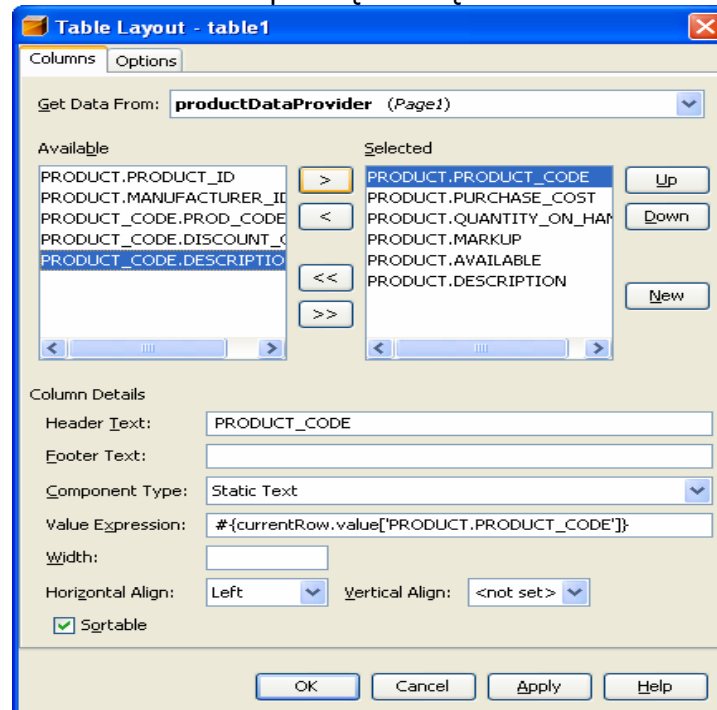


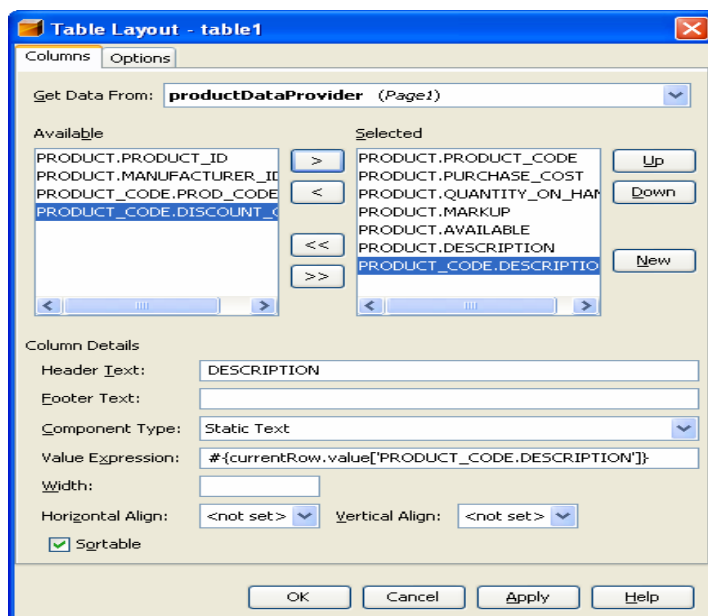
Zamknij formularz klikając na klawisz "X" zakładki **productRowSet(SssionBean1)**

7.10. Dodaj nową kolumnę w komponencie `table1`, wyświetlającą kolumnę `PRODUCT_CODE.DESCRPTION`. W tym celu zaznacz w trybie `Design` komponent `table1`, wybierz opcję `Table Layout`.



W formularzu należy przenieść z lewej strony z okienka `Available` kolumnę `PRODUCT_CODE.DESCRPTION` na prawą stronę do okienka `Selected`





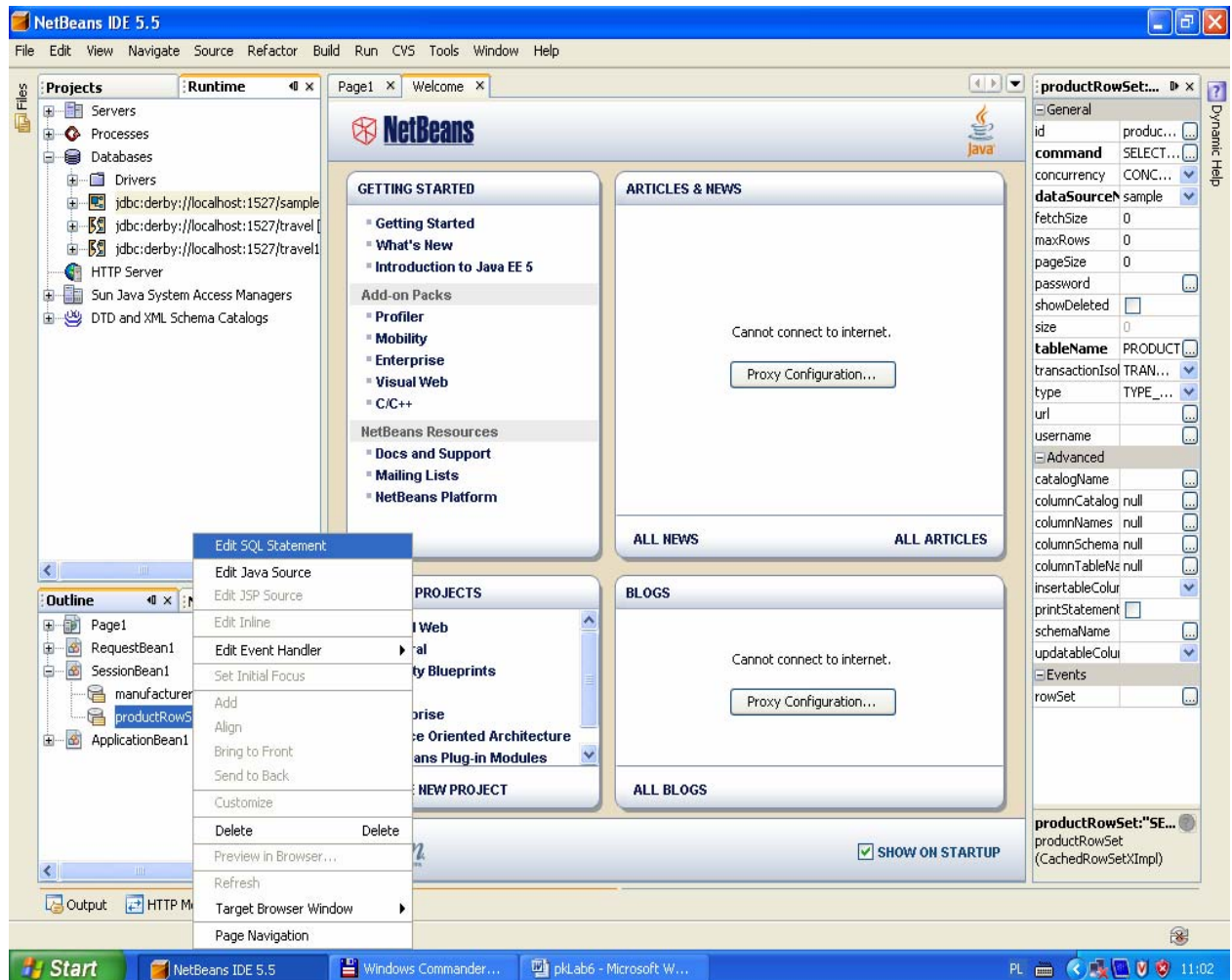
Uwaga: Uruchom aplikację (Kliknij prawym klawiszem myszy w oknie **Project** na nazwę projektu, w ukazanym oknie uruchom kolejno **Build Project**, **Deploy Project**, **Run Project** lub tylko **Run Project**). Po uruchomieniu aplikacji w komponencie table1 są wyświetlane niektóre kolumny wszystkich krotek tabeli **PRODUCT** i skojarzonych krotek z tabeli **PRODUCT_CODE** (nowa kolumna **DESCRIPTION** skrajna z prawej strony. Można zmienić **Header Text** na inny, gdyż teraz są dwie kolumny z tym samym nagłówkiem)

http://localhost:8080/AplikacjaInternetowa6_5/ - Microsoft Internet Explorer

Podaj producenta: Google

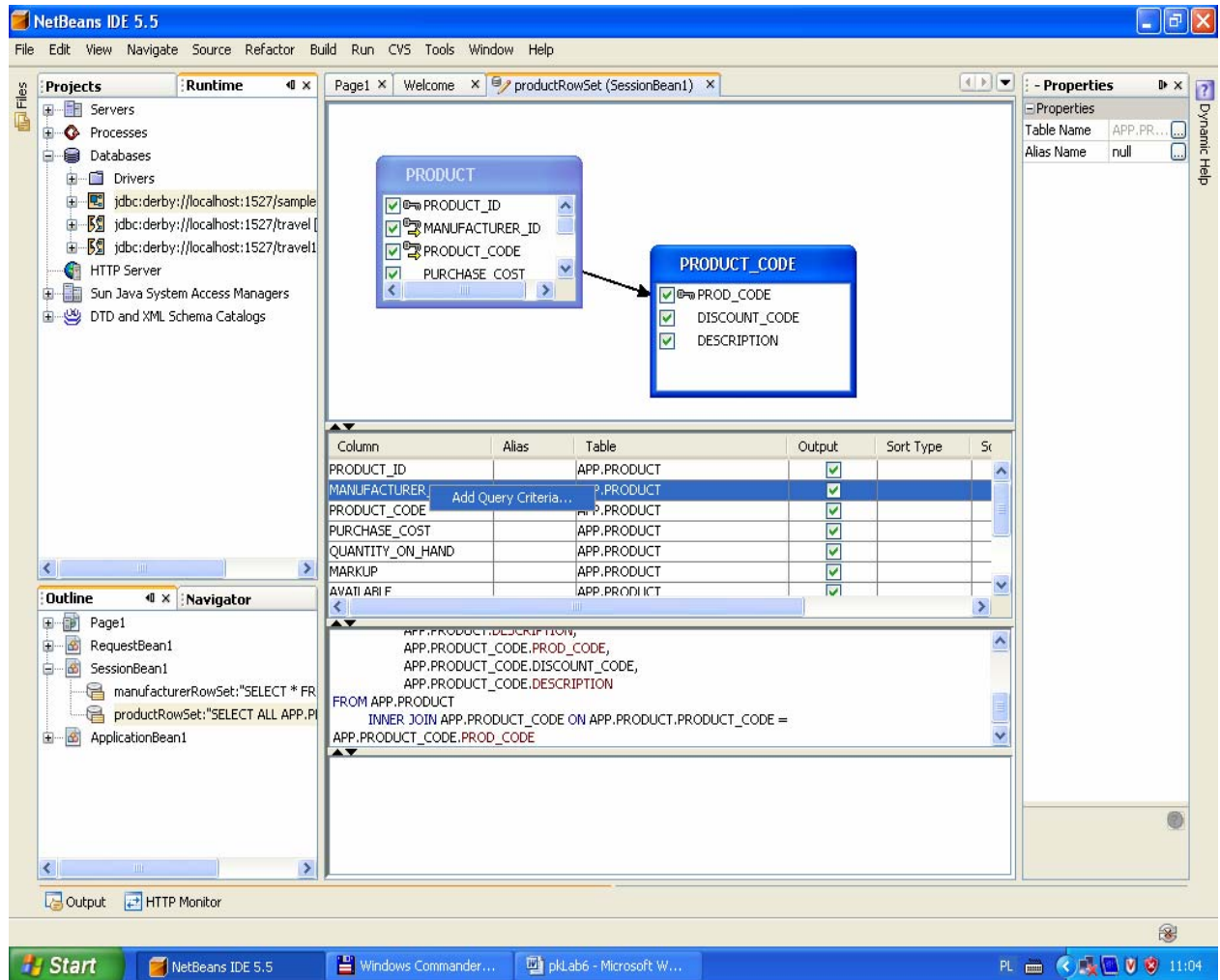
Table						
PRODUCT_CODE	PURCHASE_COST	QUANTITY_ON_HAND	MARKUP	AVAILABLE	DESCRIPTION	DESCRIPTION
SW	1095.00	800000	8.25	TRUE	Identity Server	Software
SW	11500.99	500	55.25	TRUE	Accounting Application	Software
HW	2095.99	3000	15.75	TRUE	1Ghz Sun Blade Computer	Hardware
FW	59.95	250	40.00	TRUE	10Gb Ram	Firmware
FW	39.95	50	25.50	TRUE	Sound Card	Firmware
CB	15.98	400	30.00	TRUE	Printer Cable	Cables
HW	595.00	800	5.75	TRUE	24 inch Digital Monitor	Hardware
HW	10.95	25	9.75	TRUE	104-Key Keyboard	Hardware
SW	49.95	0	5.25	FALSE	Directory Server	Software
HW	69.95	1000	10.50	TRUE	512X IDE DVD-ROM	Hardware
BK	29.98	25	5.00	TRUE	Learn Solaris 10	Books
CB	25.95	500	30.00	TRUE	Network Cable	Cables
HW	2000.95	2000	25.00	TRUE	300Mhz Pentium Computer	Hardware
BK	29.95	1000	33.00	TRUE	Learn NetBeans	Books
MS	75.00	0	12.00	FALSE	Corporate Expense Survey	Misc
SW	595.95	75	14.00	TRUE	Sun Studio C++	Software
BK	19.95	100	5.00	TRUE	Client Server Testing	Books
BK	18.95	43	4.00	TRUE	Learn Java in 1/2	Books

- 7.11. Ograniczenie wyświetlanych wierszy w tabeli do produktów firmy wybranej w komponencie `downDrop1`.
- Należy wybrać `productRowSet` skojarzony z komponentem `table1` prawym klawiszem myszy i wybrać `edytor SQL`

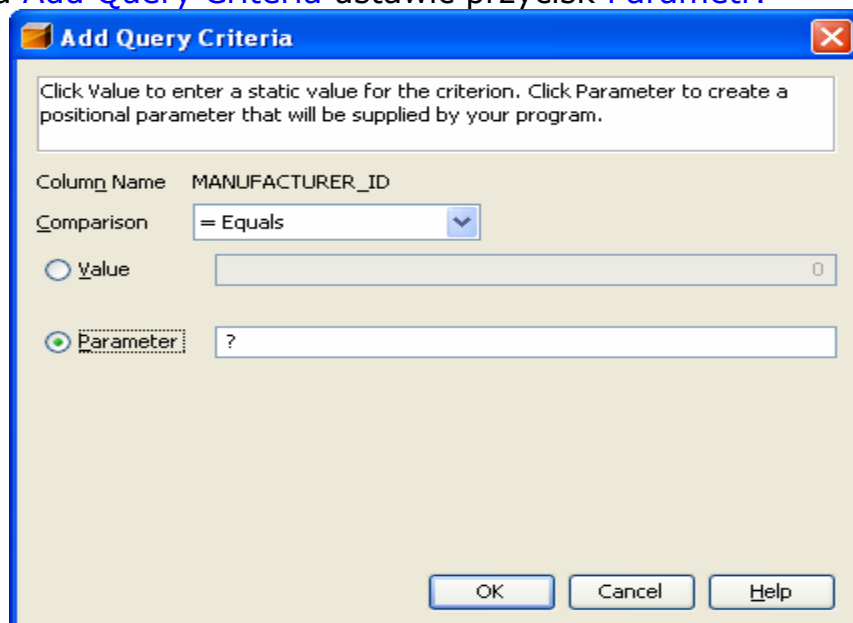


Uwaga: W tym rozwiązaniu kryterium dla zapytania `Select` jest kolumna `MANUFACTURER_ID`

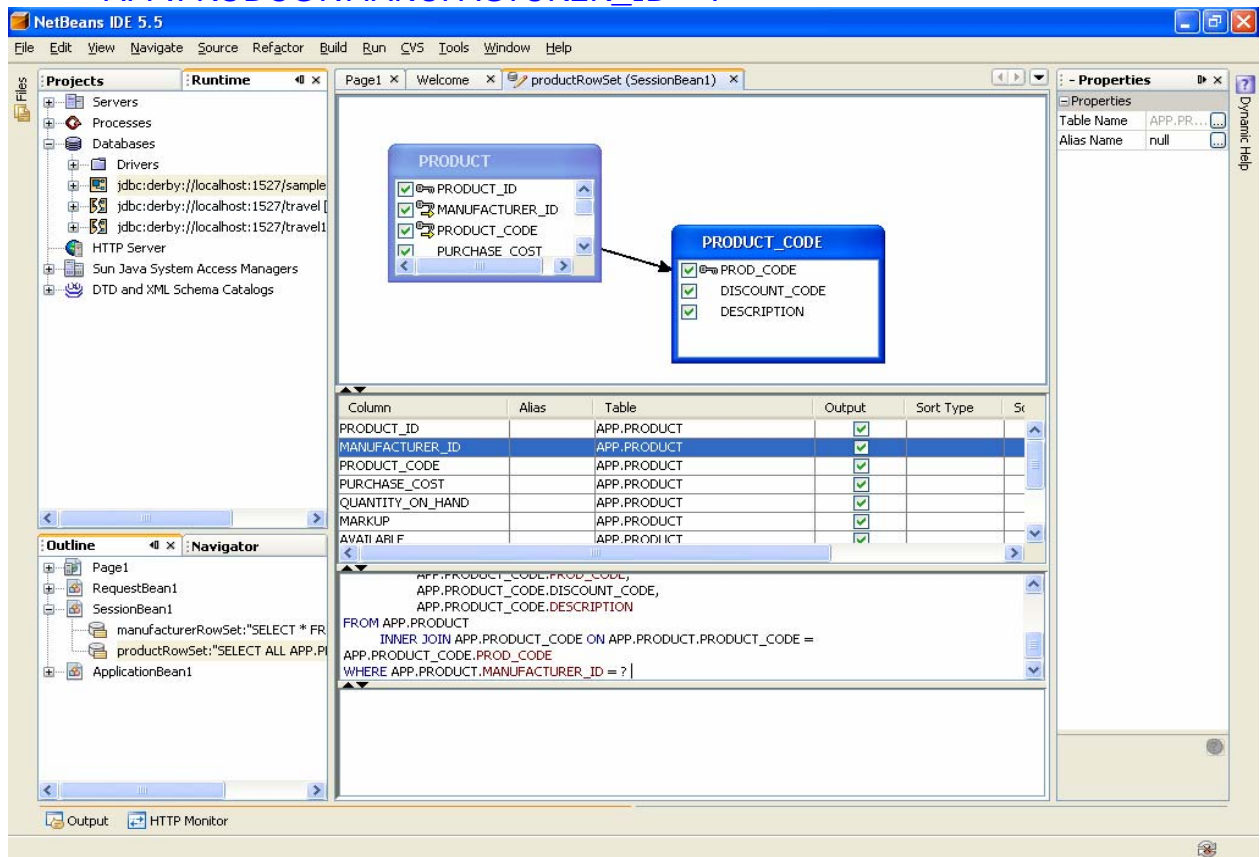
- W kolumnie **MANUFACTURER_ID** kliknąć prawym klawiszem myszy i wybrać **Add Query Criteria**



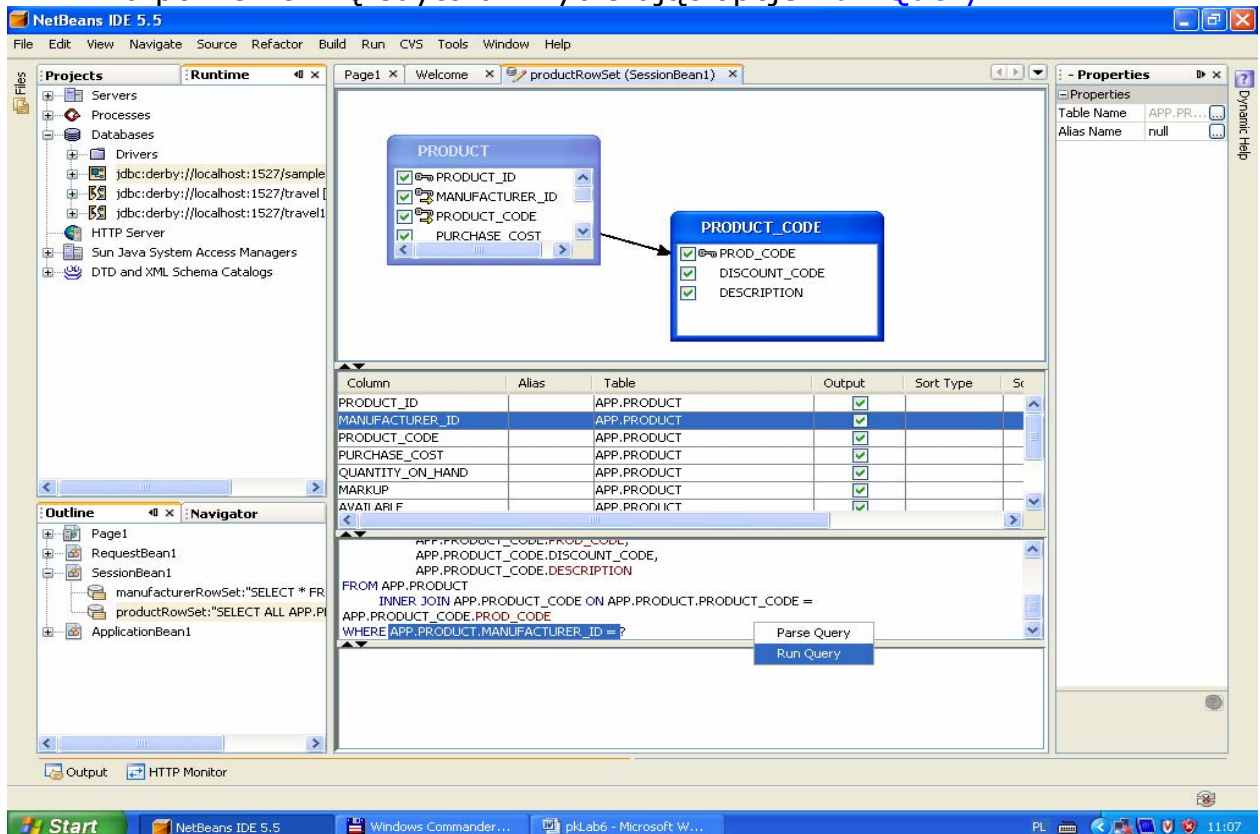
- W menu **Add Query Criteria** ustawić przycisk **Parametr**.



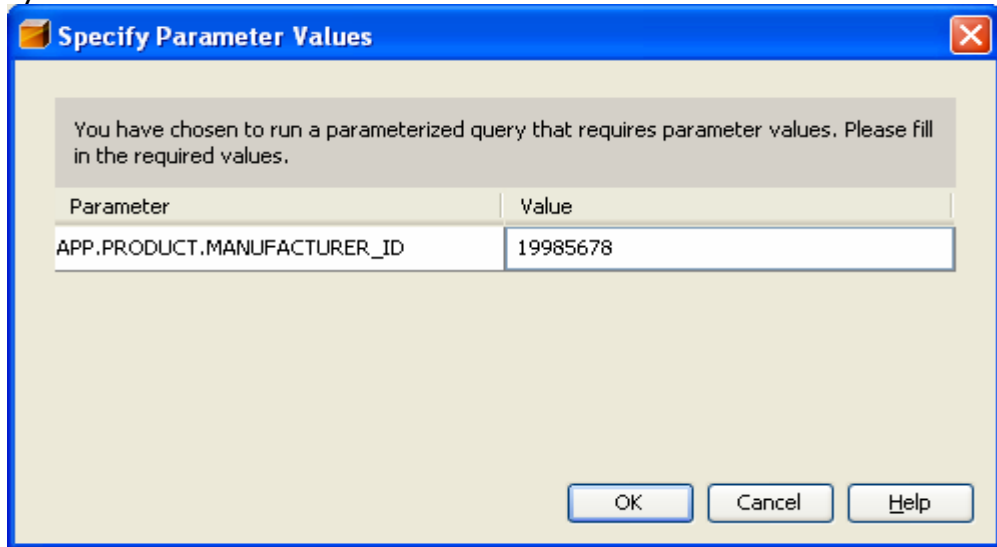
- Zapytanie zostanie zmodyfikowane o warunek **WHERE APP.PRODUCT.MANUFACTURER_ID = ?**



- Można sprawdzić działanie zapytania klikając prawym klawiszem myszy na powierzchnię edytora i wybierając opcje **Run Query**.



- Podanie parametru zapytania np. `APP.PRODUCT.MANUFACTURER_ID` równy 19985678



- Wynik zapytania

NetBeans IDE 5.5

File Edit View Navigate Source Refactor Build Run CVS Tools Window Help

Projects Runtime Page1 x Welcome x productRowSet(SessionBean1) x

Properties Table Name APP.PR... Alias Name null

PRODUCT

- PRODUCT_ID
- MANUFACTURER_ID
- PRODUCT_CODE
- PURCHASE_COST

PRODUCT_CODE

- PROD_CODE
- DISCOUNT_CODE
- DESCRIPTION

Column	Alias	Table	Output	Sort Type	Sc
PRODUCT_ID		APP.PRODUCT	<input checked="" type="checkbox"/>		
MANUFACTURER_ID		APP.PRODUCT	<input checked="" type="checkbox"/>		
PRODUCT_CODE		APP.PRODUCT	<input checked="" type="checkbox"/>		
PURCHASE_COST		APP.PRODUCT	<input checked="" type="checkbox"/>		
QUANTITY_ON_HAND		APP.PRODUCT	<input checked="" type="checkbox"/>		
MARKUP		APP.PRODUCT	<input checked="" type="checkbox"/>		
AVATARI F		APP.PRODUCT	<input checked="" type="checkbox"/>		

```

APP.PRODUCT_CODE.PROD_CODE,
APP.PRODUCT_CODE.DISCOUNT_CODE,
APP.PRODUCT_CODE.DESCRPTION
FROM APP.PRODUCT
INNER JOIN APP.PRODUCT_CODE ON APP.PRODUCT.PRODUCT_CODE =
APP.PRODUCT_CODE.PROD_CODE
WHERE APP.PRODUCT.MANUFACTURER_ID = ?

```

PRODUC...	MANUFA...	PRODUC...	PURCHAS...	QUANTIT...	MARKUP	AVAILABLE	DESCRIP
980001	19985678	SW	1095.00	800000	8.25	TRUE	Identity Se

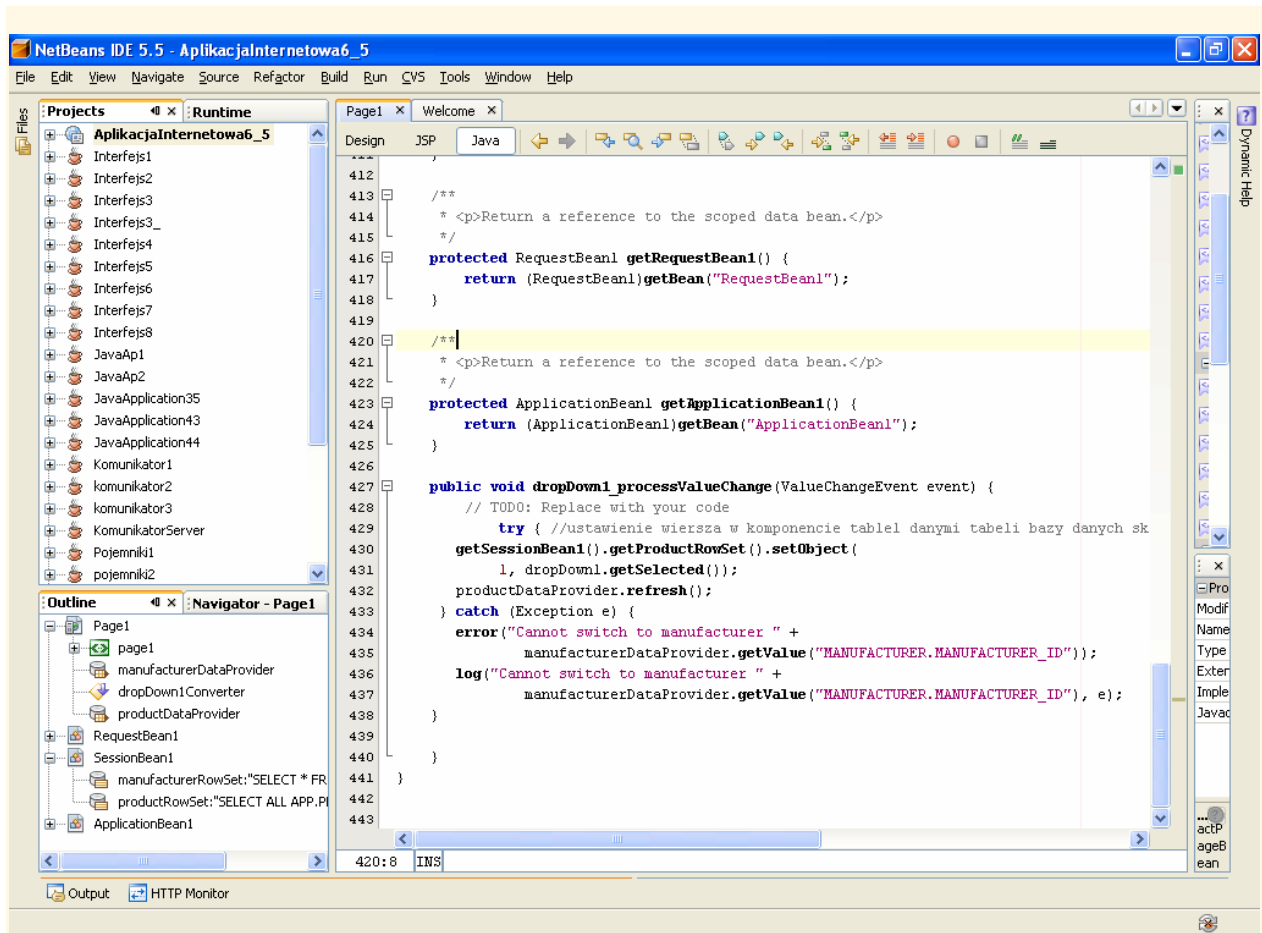
Output HTTP Monitor

SQL statement(s) executed successfully.

8. Uzupełnij kod aplikacji w trybie Java strony [Page1](#) (klasa Page1 dziedziczy od klasy [AbstractPageBean](#))

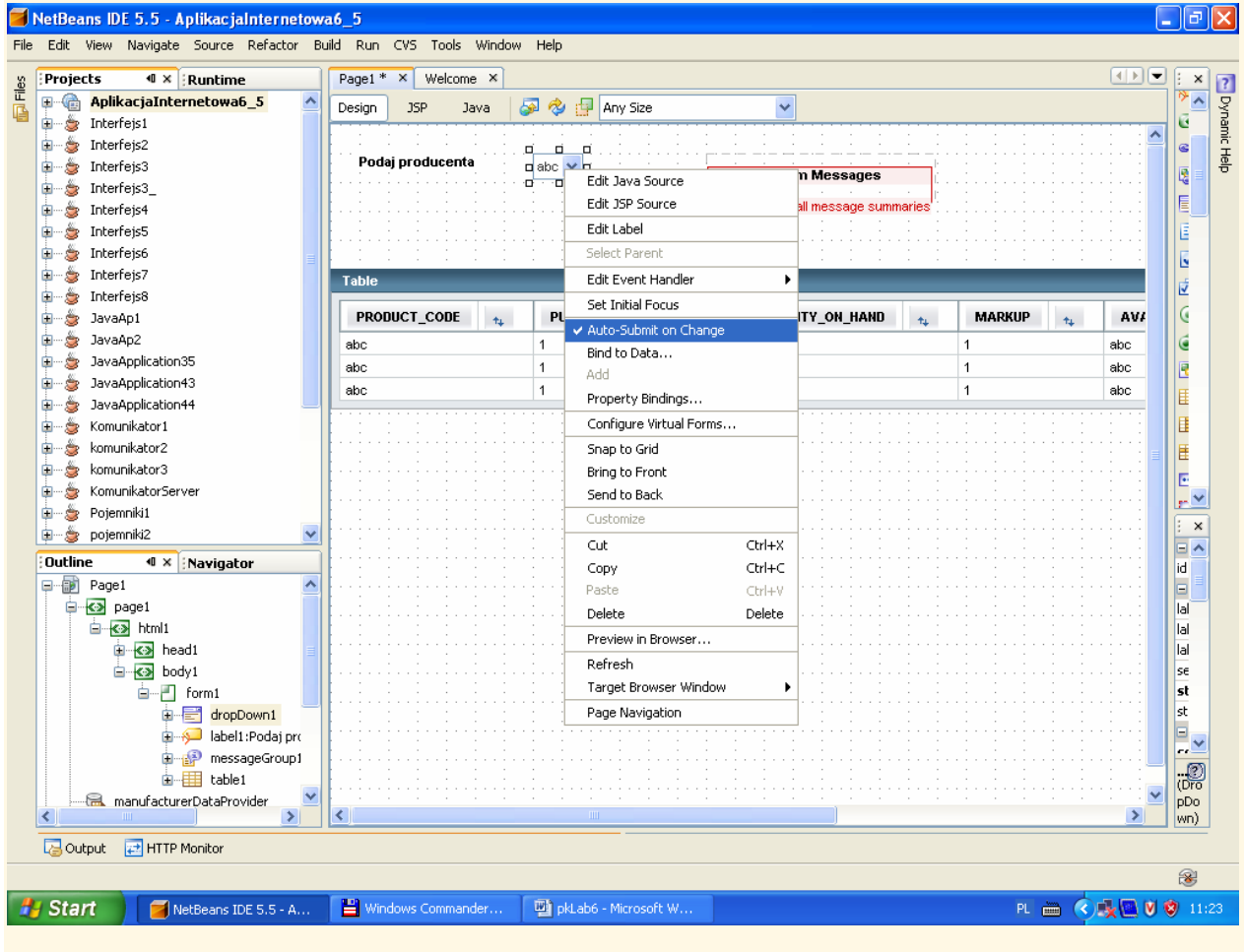
8.1. Należy kliknąć podwójnie na kontrolkę [dropDown1](#) i obsłużyć zdarzenie [ProcessValueChange](#) za pomocą metody [dropDown1_processValueChange](#)

```
public void dropDown1_processValueChange(ValueChangeEvent event)
{
    try { //ustawienie wiersza w komponencie table1 danymi tabeli bazy danych skojarzonej z komponentem dropDown1 i
//wybranymi w tym komponencie
        getSessionBean1().getProductRowSet().setObject(
            1, dropDown1.getSelected());
        productDataProvider.refresh();
    } catch (Exception e) {
        error("Cannot switch to manufacturer " +
            manufacturerDataProvider.getValue(
                "MANUFACTURER.MANUFACTURER_ID"));
        log("Cannot switch to manufacturer " +
            manufacturerDataProvider.getValue(
                "MANUFACTURER.MANUFACTURER_ID"), e);
    }
}
```



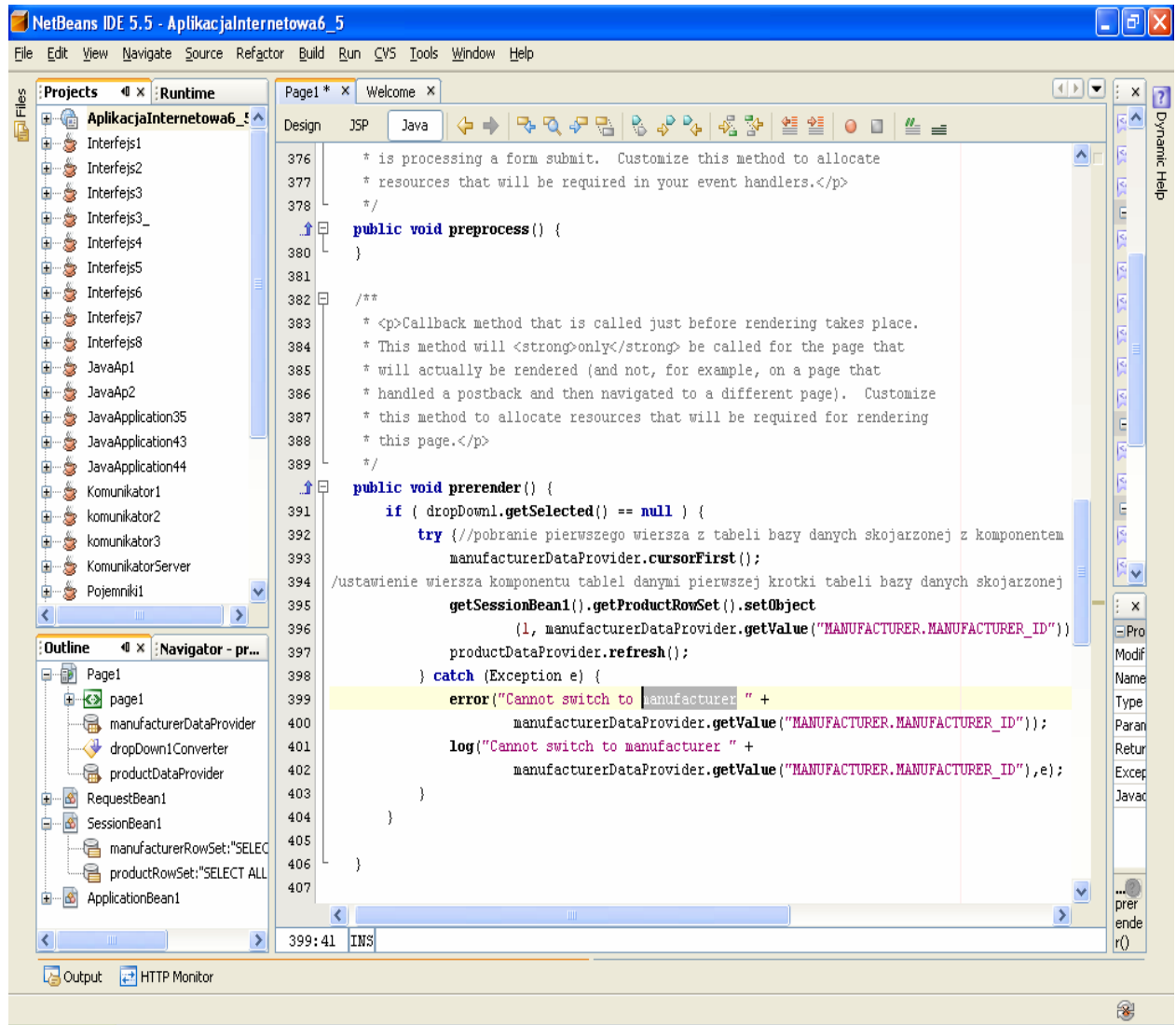
8.2. Przejdź do trybu **Design** strony **Page1**, zaznacz komponent **dropDown1** i prawym klawiszem myszy kliknij nad tym zaznaczonym komponentem. Z wyskakującego menu ustaw **Auto-Submit on Change**.

Pozwoli to obsługiwać zdarzenia wyboru pozycji z listy rozwijanej komponentu **dropDown1**, typu **ProcessValueChange**, za pomocą metody **dropDown1_processValueChange**.



8.3. metoda prerender() //inicjowanie zawartości komponentu `table1`

```
public void prerender()
{
    if ( dropdown1.getSelected() == null ) {
        try { //pobranie pierwszego wiersza z tabeli bazy danych skojarzonej z komponentem dropdown1
            manufacturerDataProvider.cursorFirst();
            //ustawienie wiersza komponentu table1 danymi pierwszej krotki tabeli bazy danych skojarzonej z komponentem dropdown1
            getSessionBean1().getProductRowSet().setObject
                (1, manufacturerDataProvider.getValue(
                    "MANUFACTURER.MANUFACTURER_ID"));
            productDataProvider.refresh();
        } catch (Exception e) {
            error("Cannot switch to manufacturer" +
                manufacturerDataProvider.getValue(
                    "MANUFACTURER.MANUFACTURER_ID"));
            log("Cannot switch to manufacturer " +
                manufacturerDataProvider.getValue(
                    "MANUFACTURER.MANUFACTURER_ID"),e);
        }
    }
}
```



8.3. Uruchom aplikację (Kliknij prawym klawiszem myszy w oknie **Project** na nazwę projektu, w ukazanym oknie uruchom kolejno **Build Project**, **Deploy Project**, **Run Project** lub tylko **Run Project**) i uruchom poszczególne funkcje aplikacji.

Podaj producenta

PRODUCT_CODE	PURCHASE_COST	QUANTITY_ON_HAND	MARKUP	AVAILABLE	DESCRIPTION	DESCRIPTION
HWV	10.95	25	9.75	TRUE	104-Key Keyboard	Hardware

Podaj producenta

PRODUCT_CODE	PURCHASE_COST	QUANTITY_ON_HAND	MARKUP	AVAILABLE	DESCRIPTION	DESCRIPTION
SW	209.95	300	41.00	TRUE	Jax VWS Application Development Environment	Software

Table 1: Input Components

Component	Description	Palette Section
Text Field	An input field for a single line of text.	Basic
Text Area	An input field for multiple lines of text.	Basic
Password Field	An input field that echoes the input characters with a replacement character to mask the input.	Basic
Calendar	An input field with a pop-up calendar that allows the user to select a date.	Basic
Drop Down List	A drop-down menu, also referred to as a combo box.	Basic
Listbox	A list from which the user can select either one item or multiple items, depending on how the component is configured.	Basic
Checkbox	A single-character box that the user can either select (check) or clear.	Basic
Radio Button	A single radio button that the user can select (check).	Basic
Add Remove List	Two lists (one for available options, one for selected options) with buttons to move the options between the lists, and to order the selected options.	Composite
File Upload	A component with a text input field and a Browse button that displays a file chooser for specifying a file to upload. The application uploads the specified file when the user submits the page.	Basic

Table 2: Display Components

Component	Description	Palette Section
Static Text	Field for displaying text.	Basic
Label	Text field that can be associated with an input field and for which you can specify a weak, medium, or strong font style.	Basic
Image	Inline image.	Basic
Message	Text field that is linked to a specific component for displaying validation errors and other messages about that component.	Basic
Message Group	Text field for displaying runtime error messages, program generated error messages, and, optionally, validation errors and other messages about components that are on the page.	Basic
Alert	Displays an icon and informational text such as a warning, an error, or the successful completion of some event.	Composite
Page Alert	Similar to an Alert component, but is intended for displaying the icon and information on a separate page.	Layout

Table 3: Grouping Components

Component	Description	Palette Section
Checkbox Group	Displays two or more check boxes in a grid layout.	Basic
Radio Button Group	Displays two or more radio buttons in a grid layout and ensures that only one button can be selected at a time.	Basic
Table, Table Row Group, and Table Column	Displays data from a composite data type such as a database table or an array.	Basic
Tree and Tree Node	Renders an expandable list in a tree structure.	Basic
Tab Set and Tab	Displays different layouts on the same page. Also can be used as a navigational tool.	Layout
Grid Panel	Organizes the components within a layout of rows and columns.	Layout
Group Panel	Groups a set of components in flow layout mode.	Layout
Layout Panel	Use to group a set of components in flow layout mode or grid layout mode.	Layout
Property Sheet, Property Sheet Section, and Property	Lays out a single column of labeled components quickly, and divides the components into sections.	Layout
Breadcrumbs	Lays out a series of link components separated by right angle brackets (>).	Composite
Page Fragment Box	Groups components that you want to consistently display on two or more pages.	Layout

Table 4: Action Components

Component	Description	Palette Section
Button	Button that submits the associated form.	Basic
Hyperlink	Text field that submits a URL or submits a form.	Basic
Image Hyperlink	Image that submits a URL or submits a form.	Basic
Tree Node	Subcomponent of a Tree or Tree Node. A leaf tree node can optionally submit a URL or submit a form.	Basic
Tab	Subcomponent of a Tab Set or a Tab. A tab can optionally submit a URL or submit a form.	Composite