

# Zastosowanie konwerterów

wg

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

## Technologie internetowe 5 (2)

# **Konwertery liczbowe i daty**

## **rozdziały 11-12**

# Konwertery - pakiet javax.faces.convert

Klasa w pakiecie javax.faces.convert	Converter ID
BigDecimalConverter	javax.faces.BigDecimal
BigIntegerConverter	javax.faces.BigInteger
BooleanConverter	javax.faces.Boolean
ByteConverter	javax.faces.Byte
CharacterConverter	javax.faces.Character
DateTimeConverter	javax.faces.DateTime
DoubleConverter	javax.faces.Double
EnumConverter	javax.faces.Enum
FloatConverter	javax.faces.Float
IntegerConverter	javax.faces.Integer
LongConverter	javax.faces.Long
NumberConverter	javax.faces.Number
ShortConverter	javax.faces.Short

# Konwertery liczbowe

Atrybuty znacznika **convertNumber** (obsługiwany przez komponent typu **NumberConverter**)

Atrybut	Typ	Opis
binding	NumberConverter	Używany do powiązania konwertera z właściwością obiektu typu Managed Bean
currencyCode	String	ISO 4217 - kod oznaczeń własnych programisty
currencySymbol	String	Własny symbol programisty
for	String	Używany w komponentach kompozytowych do powiązania z wybranym z wielu komponentów
groupingUsed	Boolean	Specyfikuje, czy w formatowanym wyjściu znajdują się separatory grupowania.
integerOnly	Boolean	Określa, czy tylko część całkowita wartości będzie konwertowana.

locale	String lub Locale	Kod, którego numer stylu jest używany do formatowania lub analizowania danych.
maxFractionDigits	int	Maksymalna liczba cyfr sformatowanych w części ułamkowej formatowanego wyjścia
maxIntegerDigits	Int	Maksymalna liczba cyfr sformatowanych w części całkowitej formatowanego wyjścia.
minFractionDigits	int	Minimalna liczba cyfr sformatowanych w części ułamkowej formatowanego wyjścia
minIntegerDigits	int	Minimalna liczba cyfr sformatowanych w części całkowitej formatowanego wyjścia.
pattern	String	Niestandardowy wzór formatowania, który określa sposób sformatowania i analizowania ciągu znaków liczby.
type	String	Określa, czy ciąg znaków jest analizowany i formatowane jako liczba, waluta, lub procent. Jeśli nie podano, używa się formatu liczby.

# Konwerter daty

## Atrybuty znacznika `convertDateTime`

Atrybut	Typ	Opis
<code>binding</code>	<code>DateTimeConverter</code>	Używany do przypisania konwertera do właściwości obiektu typu Managed Bean
<code>dateStyle</code>	<code>String</code>	Definiuje format daty lub fragment zawierający datę, podany przez klasę <code>java.text.DateFormat</code> . Stosowany tylko wtedy, gdy typ jest datą lub data i czasem lub jeśli szablon nie zdefiniowany Poprawne wartości formatu: domyślna, krótka, średnia, długa i pełna. Jeśli nie podano wartości formatu, używana jest domyślna wartość.
<code>for</code>	<code>String</code>	Używany w złożonych komponentach. Odnosi się do jednego z obiektów należących do złożonego komponentu, wewnątrz którego ten znacznik jest zagnieżdżony.

locale	String or Locale	Kod kraju, którego predefiniowane style dla dat i czasu są używane podczas formatowania lub przetwarzania. Jeśli nie podano, wykorzystywany jest kod kraju, zwrócony przez <a href="#"><b>javax.faces.context.FacesContext.getLocale</b></a> .
pattern	String	Niestandardowy wzór formatowania, który określa, jak łańcuch data / czas powinien być sformatowany i analizowany. Jeżeli ten atrybut jest określony, ignorowane są atrybuty <a href="#"><b>dateStyle</b></a> , <a href="#"><b>timeStyle</b></a> i <a href="#"><b>type</b></a> .
timeStyle	String	Definiuje format czasu lub część łańcucha, dotyczący czasu w łańcuchu daty, określony przez <a href="#"><b>java.text.DateFormat</b></a> . Stosowany tylko wtedy, kiedy typ jest czasem i szablon nie jest zdefiniowany. Prawidłowe wartości formatu: domyślny, krótki, średni, długi i pełny. Jeśli wartość nie jest określona, używana jest domyślna wartość formatu.
timeZone	String lub TimeZone	Strefa czasu, w której interpretuje się czas w łańcuchu daty
type	String	Określa, czy wartość ciągu będzie zawierać datę, czas, lub obie. Prawidłowe wartości to: data, czas, lub obie. Jeśli wartość nie jest określona, używana jest data.

# Wpływ atrybutu **Locale** na formatowanie danych podczas konwersji

Jeśli w komponentach atrybut **locale** nie jest ustawiony, wtedy przez komponent jest pobierana domyślna wartość tego atrybutu za pomocą [javax.faces.context.FacesContext.getLocale](#).

W celu uzyskania różnych formatów, należy ustawić wartość atrybutu locale indywidualnie w każdym komponencie, w przeciwnym wypadku zdefiniować domyślną wartość **locale** w pliku [faces-config](#).

---

```
public class Locales {
public static void main(String[] args) {
    String pattern = "#####.###";
    double value = 111111.111;
    Locale locales[] = Locale.getAvailableLocales();
    for (Locale loc : locales) {
        NumberFormat number_f = NumberFormat.getNumberInstance(loc);
        DecimalFormat decimal_f = (DecimalFormat) number_f;
        decimal_f.applyPattern(pattern);
        String output = decimal_f.format(value);
        System.out.println(pattern + " " + output + " " + loc.toString()); }
    }
}
```



<b>Pattern</b>	<b>Wyjściowy łańcuch dla wartości: 111111.111</b>	<b>locale</b>
<b>#####.###</b>	111111,111	<b>pl</b>
<b>#####.###</b>	111111,111	<b>pl_PL</b>
<b>#####.###</b>	111111,111	<b>fr</b>
<b>#####.###</b>	111111,111	<b>fr_BE</b>
<b>#####.###</b>	111111,111	<b>fr_CA</b>
<b>#####.###</b>	111111,111	<b>fr_FR</b>
<b>#####.###</b>	111111,111	<b>fr_LU</b>
<b>#####.###</b>	111111,111	<b>de</b>
<b>#####.###</b>	111111,111	<b>de_AT</b>
<b>#####.###</b>	111111,111	<b>de_DE</b>
<b>#####.###</b>	111111,111	<b>de_GR</b>
<b>#####.###</b>	111111,111	<b>de_LU</b>

<b>#####.###</b>	<b>111111.111</b>	<b>en</b>
<b>#####.###</b>	<b>111111.111</b>	<b>en_AU</b>
<b>#####.###</b>	<b>111111.111</b>	<b>en_CA</b>
<b>#####.###</b>	<b>111111.111</b>	<b>en_GB</b>
<b>#####.###</b>	<b>111111.111</b>	<b>en_IE</b>
<b>#####.###</b>	<b>111111.111</b>	<b>en_IN</b>
<b>#####.###</b>	<b>111111.111</b>	<b>en_MT</b>
<b>#####.###</b>	<b>111111.111</b>	<b>en_NZ</b>
<b>#####.###</b>	<b>111111.111</b>	<b>en_PH</b>
<b>#####.###</b>	<b>111111.111</b>	<b>en_SG</b>
<b>#####.###</b>	<b>111111.111</b>	<b>en_US</b>
<b>#####.###</b>	<b>111111.111</b>	<b>en_ZA</b>
<b>#####.###</b>	<b>111111.111</b>	<b>fr_CH</b>
<b>#####.###</b>	<b>111111.111</b>	<b>de_CH</b>

# Rejestrowanie konwerterów

# Sposoby rejestrowania konwerterów do komponentu (SR)

1. Zbindowanie atrybutu **value** komponentów do **właściwości (metody typu set i get)** obiektów typu Managed Bean, takiego samego typu jak konwerter.
2. Zagnieżdżenie znaczników jednego ze standardowych konwerterów **f:convertNumber** i **f:convertDateTime** wewnątrz znaczników komponentów
3. Zagnieżdżenie znacznika **f:converter** wewnątrz znacznika komponentu i użycie albo jego atrybutu **converterId** lub jego atrybutu **binding** powiązanego z instancją konwertera zdefiniowanego w obiekcie typu **Managed Bean (lub standardowym)**.
4. Powiązanie atrybutu **converter** w komponencie z instancją konwertera zdefiniowanego w obiekcie typu **Managed Bean (lub standardowym)**.

# Przykład 1

## Definicja typów liczbowych właściwości obiektu **Managed\_produk**t

@ManagedBean

@RequestScoped

```
public class Managed_produk {
```

@EJB

```
private Fasada_warstwy_biznesowej fasada;
```

```
private DataModel items;
```

```
private int stan = 1;
```

```
private Produkt_dto produkt_dto = new Produkt_dto();
```

Sposób 1

```
public String dodaj_produk() {  
    fasada.utworz_produk(produkt_dto);  
    dane_produk();  
    return "rezultat2";  
}
```

```
public void dane_produk() {  
    stan = 1;  
    produkt_dto = fasada.dane_produk();  
    if (produkt_dto == null) {  
        stan = 0;  
        produkt_dto=new Produkt_dto();  
    }  
}
```

Metody **właściwości obiektu Managed\_produkt**, używanych w atrybutach value komponentów wejściowych i wyjściowych

```
public Float getCena() {  
    return produkt_dto.getCena(); }  
public void setCena(Float cena) {  
    this.produkt_dto.setCena(cena); }
```

Właściwość definiująca konwerter typu **Float**

```
public Integer getPromocja() {  
    return produkt_dto.getPromocja(); }  
public void setPromocja(Integer promocja) {  
    this.produkt_dto.setPromocja(promocja); }
```

Właściwość definiująca konwerter typu **Integer**

```
public Float getCena_brutto() {  
    return produkt_dto.getCena_brutto(); }  
public void setCena_brutto(Float cena_brutto) {  
    this.produkt_dto.setCena_brutto(cena_brutto); }
```

Metody definiujące właściwości obiektu **Managed\_produkt** umożliwiające bindowanie danych w formularzach JSF – danymi są pola obiektu **produkt\_dto**

Właściwość definiująca konwerter typu **Float**

```
public Date getData_produkcji() {  
    return produkt_dto.getData_produkcji(); }  
public void setCena_brutto(Date data_produkcji) {  
    this.produkt_dto.setData_produkcji(data_produkcji); }
```

Właściwość definiująca konwerter **DateTimeConverter**

# Obiekt typu **Produkt\_dto** – implementacja wzorca projektowego Transfer Object

```
public class Produkt_dto {  
    protected long id;  
    protected String nazwa;  
    protected float cena;  
    protected int promocja;  
    protected Date data_produkcji;  
    protected float cena_brutto;
```

Sposób 1



```
public long getId() { return id; }  
public void setId(long id) { this.id = id; }  
public String getNazwa() { return nazwa; }  
public void setNazwa(String nazwa) { this.nazwa = nazwa; }  
public float getCena() { return cena; }  
public void setCena(float cena) { this.cena = cena; }  
public int getPromocja() { return promocja; }  
public void setPromocja(int promocja) { this.promocja = promocja; }  
public Date getData_produkcji() { return data_produkcji; }  
public void setData_produkcji(Date data_produkcji) { this.data_produkcji = data_produkcji; }  
public float getCena_brutto() { return cena_brutto; }  
public void setCena_brutto(float cena_brutto) { this.cena_brutto = cena_brutto; }  
}
```



## Zmiana kodu w klasie **Fasada\_warstwy\_biznesowej** (1)

```
public void utworz_produkt(Produkt_dto produkt_dto) {
    Produkt produkt = new Produkt();
    klucz++;
    produkt.setId(klucz);
    produkt.setNazwa(produkt_dto.getNazwa());
    produkt.setCena(produkt_dto.getCena());
    produkt.setPromocja(produkt_dto.getPromocja());
    produkt.setData_produkcji(produkt_dto.getData_produkcji());
    dodaj_produkt(produkt);
}

protected void dodaj_produkt(Produkt produkt) {
    if (!produkty.contains(produkt)) {
        produkty.add(produkt);
        stan = true;
    } else {
        stan = false;
    }
}
```

## Zmiana kodu w klasie `Fasada_warstwy_biznesowej` (2)

```
public Produkt_dto dane_produktu() {
    if (stan) {
        Produkt produkt = produkty.get(produkty.size() - 1);
        return produkt_transfer(produkt); }
    return null;
}

public Produkt_dto produkt_transfer(Produkt produkt) {
    Produkt_dto pom = new Produkt_dto();
    pom.setId(produkt.getId());
    pom.setNazwa(produkt.getNazwa());
    pom.setCena(produkt.getCena());
    pom.setPromocja(produkt.getPromocja());
    pom.setData_produkcji(produkt.getData_produkcji());
    pom.setCena_brutto(produkt.cena_brutto());
    return pom;
}

public ArrayList<Produkt_dto> items() {
    ArrayList<Produkt_dto> dane = new ArrayList();
    for (Produkt produkt : produkty) {
        dane.add(produkt_transfer(produkt)); }
    return dane;
}
```

## Pierwszy sposób rejestrowania konwerterów (SR1)

Fragment pliku jsf, służący do wprowadzania danych, korzystający z **właściwości obiektu Managed\_produk** – typ liczbowy właściwości pozwala na formatowanie danych liczbowych przy wprowadzaniu danych (atrybut **value** komponentów **h:inputText**)

```
<h:outputLabel for="cena" value="#{bundle['dodaj_produk2.podaj_cena']}" />
<h:inputText
  id="cena"
  title="#{bundle['dodaj_produk2.podaj_cena']}"
  value="#{managed_produk.cena}"
  required="true"
  requiredMessage="#{bundle['dodaj_produk2.podaj_cena_blad']}" >
</h:inputText>
<h:outputLabel value="#{bundle['dodaj_produk2.podaj_promocja']}"
  for="promocja" />
<h:inputText
  id="promocja"
  title="#{bundle['dodaj_produk2.podaj_promocja']}"
  value="#{managed_produk.promocja}"
  required="true"
  requiredMessage="#{bundle['dodaj_produk2.podaj_promocja_blad']}" >
</h:inputText>
```

## Drugi sposób rejestrowania konwerterów (SR2)

Fragment pliku jsf, służący do wprowadzania danych, korzystający z właściwości obiektu `Managed_produkt` oraz znacznika zagnieżdżonego `f:convertDateTime`, który pozwala na formatowanie danych oraz przy wprowadzaniu danych (atrybut `pattern`)

```
<h:outputLabel value="#{bundle['dodaj_produkt2.podaj_data']}"  
               for="data" />
```

```
<h:inputText  
  id="data,,  
  title="#{bundle['dodaj_produkt2.podaj_data']}"  
  value="#{managed_produkt.data_produkcji}"  
  required="true"  
  requiredMessage="#{bundle['dodaj_produkt2.podaj_data_blad']}" >  
  <f:convertDateTime pattern="dd-MM-yyyy" />
```

```
</h:inputText>
```

## Pierwszy sposób rejestrowania konweterów (SR1)

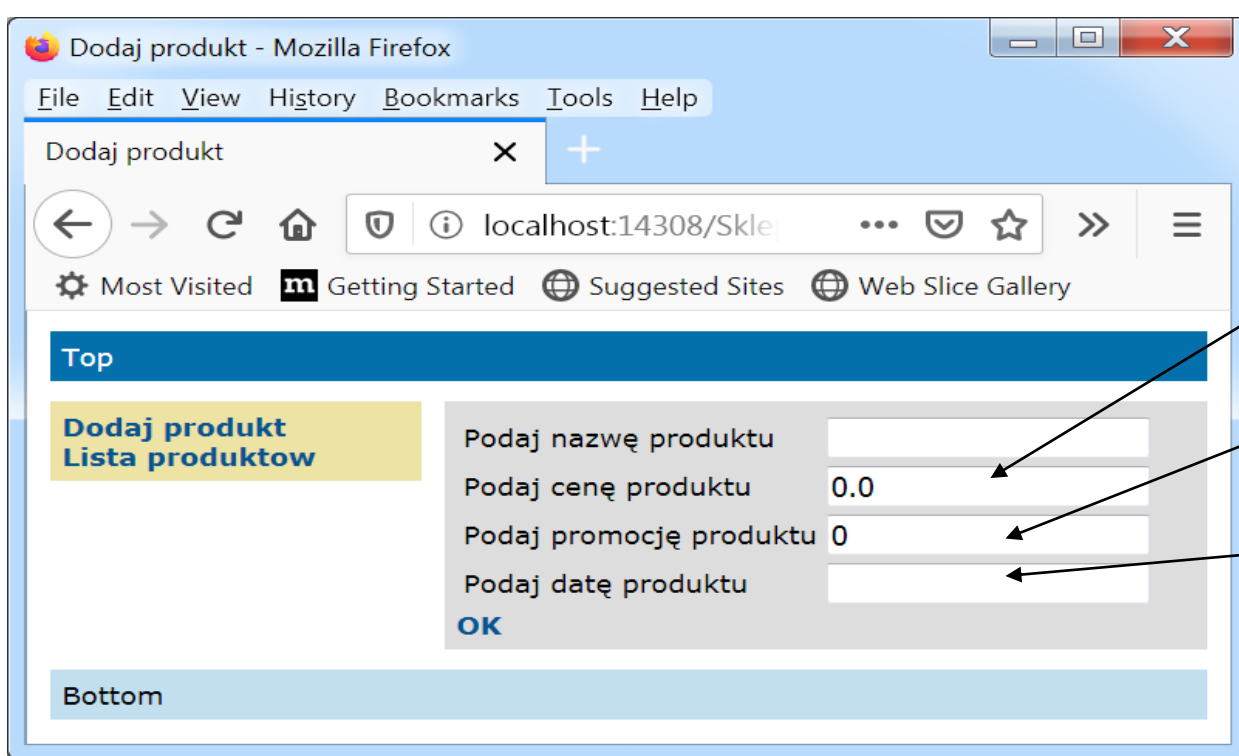
Fragment pliku jsf służący do wyświetlania danych, korzystający z **właściwości obiektu Managed\_produk** – typ liczbowy właściwości pozwala na formatowanie danych liczbowych przy wyprowadzaniu danych (atrybut **value** komponentów **h:outputText**)

```
<h:outputLabel for="cena" value="#{bundle['lista_produkow.cena']}" />
<h:outputText id="cena"
    value="#{managed_produk.cena}"/>
<h:outputLabel for="promocja" value="#{bundle['lista_produkow.promocja']}" />
<h:outputText id="promocja"
    value="#{managed_produk.promocja}"/>
<h:outputLabel for="data " value="#{bundle['lista_produkow.data']}" />
<h:outputText id="data"
    value="#{managed_produk.data_produkcji}"/>
<h:outputLabel for="brutto" value="#{bundle['lista_produkow.cenabrutto']}" />
<h:outputText id="brutto"
    value="#{managed_produk.cena_brutto}" />
```

## Pierwszy sposób rejestrowania konwerterów (SR1)

Fragment pliku jsf, służący do wyświetlania zbioru danych w formie tabeli, korzystający z komponentu `h:dataTable`, gdzie atrybut `value="#{managed_produkt.items}"` oraz `var="item"`, gdzie `item` jest obiektem typu `Produkt_dto`

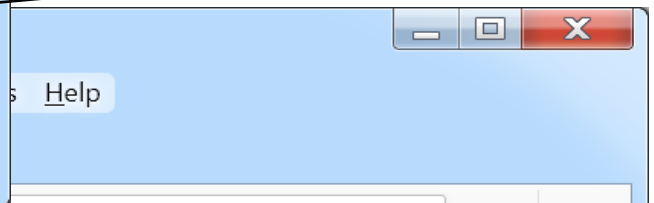
```
<h:column>
  <f:facet name="header"> <h:outputText value="#{bundle['lista_produktow.id']}" /> </f:facet>
  <h:outputText value="#{item.id}" />
</h:column>
<h:column>
  <f:facet name="header"> <h:outputText value="#{bundle['lista_produktow.cena']}" /> </f:facet>
  <h:outputText value="#{item.cena}" />
</h:column>
<h:column>
  <f:facet name="header"><h:outputText value="#{bundle['lista_produktow.promocja']}" /></f:facet>
  <h:outputText value="#{item.promocja}" />
</h:column>
<h:column>
  <f:facet name="header"> <h:outputText value="#{bundle['lista_produktow.data']}" /></f:facet>
  <h:outputText value="#{item.data_produkcji}" />
</h:column>
<h:column>
  <f:facet name="header"> <h:outputText value="#{bundle['lista_produktow.data']}" /></f:facet>
  <h:outputText value="#{item.cena_brutto}" />
</h:column>
```



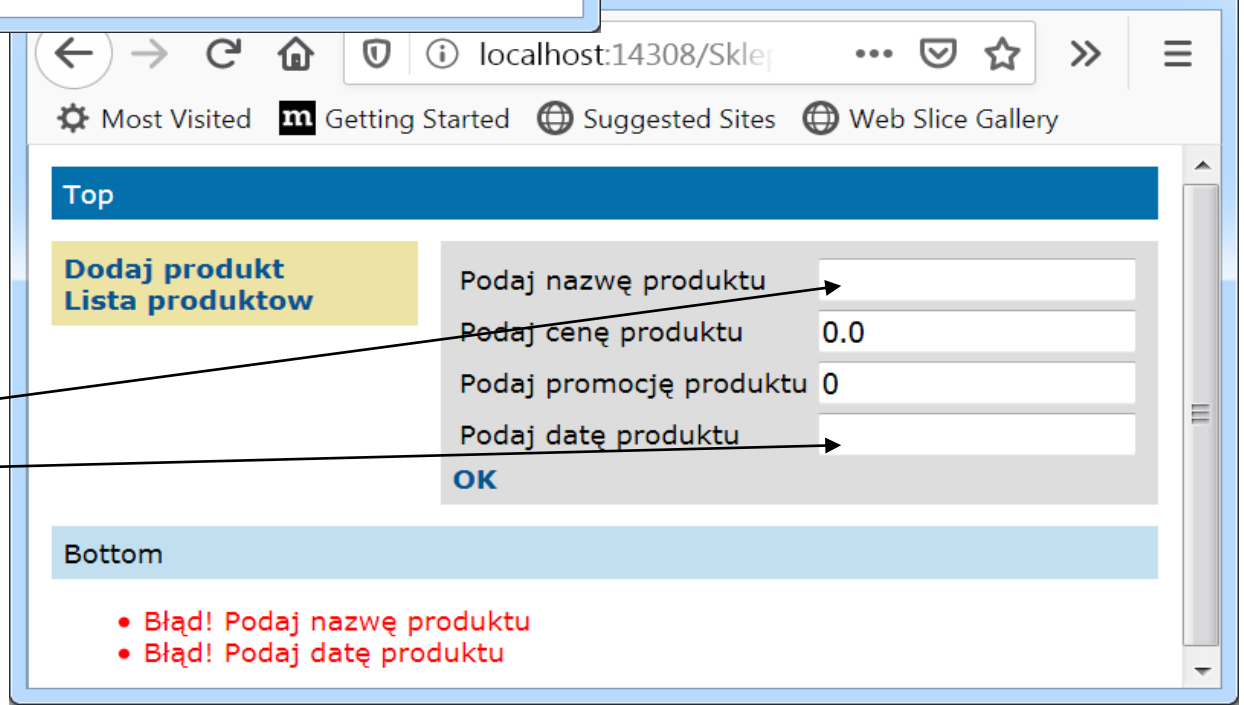
Sposób 1

Sposób 1

Sposób 2



Widok strony po kliknięciu na przycisk **OK**.  
Efekt uzyskany dzięki domyślnym wartościom pól wejściowych po zastosowaniu domyślnych konwerterów typu **Float i Integer** oraz polom **required i requiredMessage** w pozostałych polach wejściowych



Wynik działania domyślnych konwerterów: **Float, Integer, Date** oraz **Float**

Dodaj produkt

localhost:14308/Sklep

Most Visited Getting Started Suggested Sites Web Slice Gallery

Top

**Dodaj produkt**  
**Lista produktów**

Podaj nazwę produktu Produkt1

Podaj cenę produktu 100.50

Podaj promocję produktu 10

Podaj datę produktu 05-04-2013

OK

Bottom

Most Visited Getting Started Suggested Sites Web Slice Gallery

Top

**Dodaj produkt**  
**Lista produktów**

Nazwa produktu	Produkt1
Cena netto produktu	100.5
Promocja produktu	10
Data produkcji produktu	Fri Apr 05 02:00:00 CEST 2013
Cena brutto produktu	90.45

Powrót

Bottom



# Dane po konwersji w komponencie typu **dataTable**

Lista produktów - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Lista produktów × +

localhost:14308/Sklep\_3.DTO/f

Most Visited Getting Started Suggested Sites Web Slice Gallery

Top

**Dodaj produkt**  
Lista produktów

Id produktu	Nazwa produktu	Cena netto produktu	Promocja produktu	Data produkcji produktu	Cena brutto produktu
1	Produkt1	100.5	10	Fri Apr 05 02:00:00 CEST 2013	90.45

Bottom

Wynik działania domyślnych konwerterów:  
**Long, Float, Integer, Date** oraz **Float**

# Przykład 2

## Drugi sposób rejestrowania konwerterów (SR2)

**(P1) Wyświetlanie danych** - Konwerter typu **convertNumber** z atrybutem **pattern** określającym sposób prezentowania wartości i oznaczeń wartości konwertowanej (zł w kodzie UTF-8)

```
<h:column>
  <f:facet name="header"> <h:outputText value="#{bundle['lista_produkow.cena']}/> </f:facet>
  <h:outputText value="#{item.cena}">
    <f:convertNumber pattern="####.## z&#322;"/>
  </h:outputText>
</h:column>
```

**(P2) Wyświetlanie danych** - Konwerter typu **convertNumber** z atrybutami **currencySymbol** (%) i **type** do specyfikowania własnego typu (programisty) i oznaczeń wartości konwertowanej

```
<h:column>
  <f:facet name="header"> <h:outputText value="#{bundle['lista_produkow.promocja']}/> </f:facet>
  <h:outputText value="#{item.promocja}">
    <f:convertNumber currencySymbol="%" type="currency"/>
  </h:outputText>
</h:column>
```

**(P3) Wyświetlanie danych** - Konwerter typu **convertNumber** z atrybutami **currencySymbol** (zł w kodzie UTF-8) i **type** do specyfikowania własnego typu i oznaczeń wartości konwertowanej **(SR2)**

```
<h:column>
```

```
<f:facet name="header"> <h:outputText value="#{bundle['jsf.lista_produkow.cenabrutto']}" /> </f:facet>
```

```
<h:outputText value="#{item.cena_brutto}">
```

```
<f:convertNumber currencySymbol="zł" type="currency" />
```

```
</h:outputText>
```

```
</h:column>
```

**(P4) Wyświetlanie danych** - Konwerter **convertDateTime**, który przekształca datę na **nazwa\_dnia, numer\_dnia-numer\_miesiąca-rok** za pomocą atrybutu **pattern**. Nazwa dnia podawana jest w języku strefy czasowej podawanej przez **javax.faces.context.FacesContext.getLocale**, jeśli nie został zdefiniowany atrybut **locale**. **(SR2)**

```
<h:column>
```

```
<f:facet name="header"> <h:outputText value="#{bundle['lista_produkow.data']}" /> </f:facet>
```

```
<h:outputText value="#{item.data_produkcji}">
```

```
<f:convertDateTime pattern="EEEEEEE, dd-MM-yyyy" />
```

```
</h:outputText>
```

```
</h:column>
```

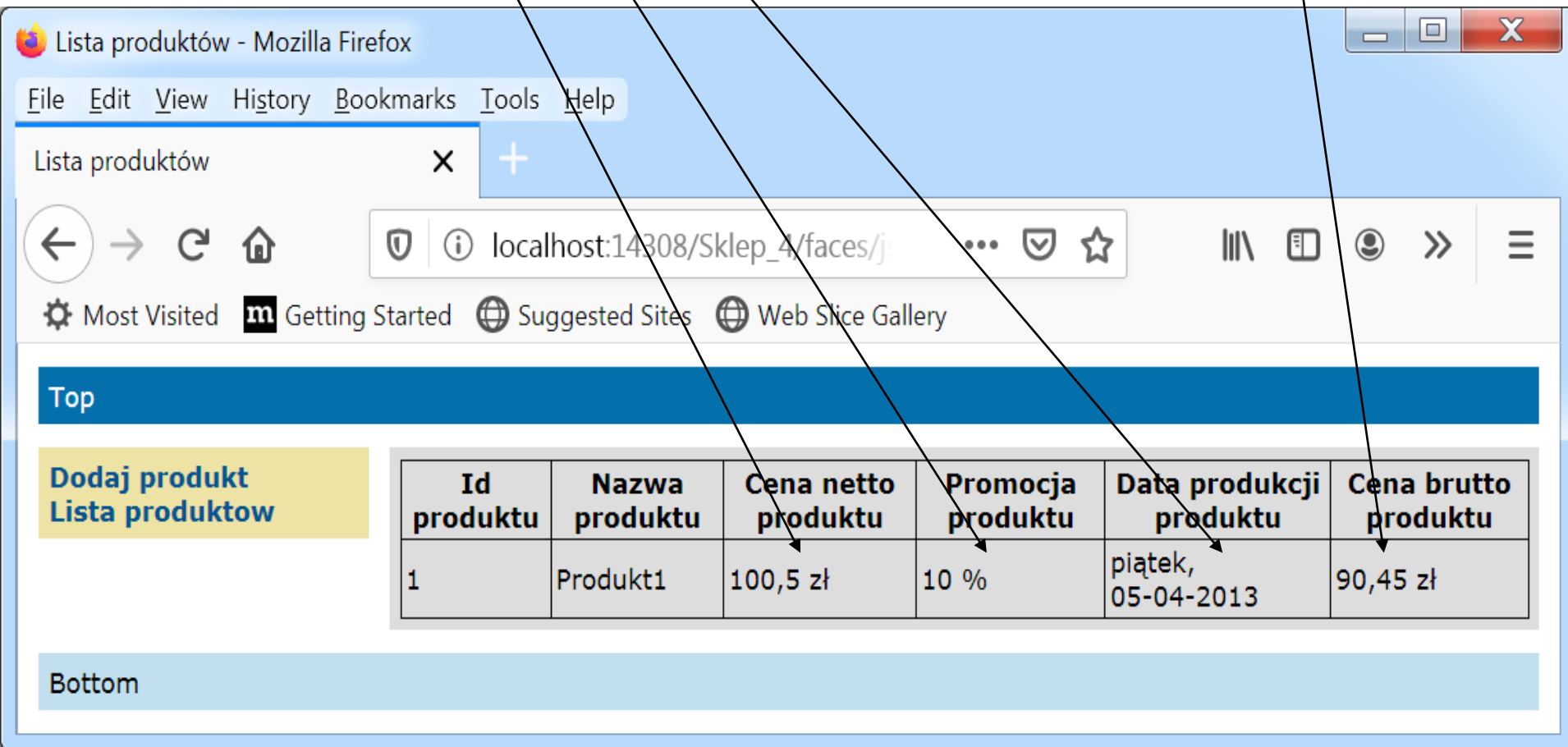
# Zawartość pliku typu faces-config.xml, określająca domyślną wartość atrybutu **locale** aplikacji (**pl\_PL**)

The screenshot displays the NetBeans IDE 8.2 interface. The left sidebar shows a project tree for 'Sklep\_4' with various files and folders. The main editor window is open to the 'faces-config.xml' file, showing the following XML content:

```
1 <?xml version='1.0' encoding='UTF-8'?>
2
3 <!-- ===== FULL CONFIGURATION FILE ===== -->
4
5 <faces-config version="2.1"
6     xmlns="http://java.sun.com/xml/ns/javaee"
7     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
8     xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
9         http://java.sun.com/xml/ns/javaee/web-facesconfig_2_1.xsd">
10     <application>
11         <resource-bundle>
12             <base-name>/Bundle</base-name>
13             <var>bundle</var>
14         </resource-bundle>
15         <locale-config>
16             <default-locale>pl_PL</default-locale>
17             <supported-locale>pl_PL</supported-locale>
18         </locale-config>
19     </application>
20 </faces-config>
21
```

The bottom status bar shows the application is running, with tabs for 'Java DB Database Process', 'GlassFish Server', and 'Sklep\_4 (run)'. The system tray at the bottom right indicates the time is 21:1 and the language is INS.

# Działanie konwerterów typu **convertNumber (P1), (P2), (P3)** oraz **convertDateTime (P4) – pl\_PL**



Lista produktów - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Lista produktów

localhost:14308/Sklep\_4/faces/j

Most Visited Getting Started Suggested Sites Web Slice Gallery

Top

**Dodaj produkt**  
Lista produktów

Id produktu	Nazwa produktu	Cena netto produktu	Promocja produktu	Data produkcji produktu	Cena brutto produktu
1	Produkt1	100,5 zł	10 %	piątek, 05-04-2013	90,45 zł

Bottom

# Przykład 3

## Drugi (SR2) i trzeci (SR3) sposób rejestrowania konwerterów

Należy dodać konwerter typu **NumberConverter**, który pozwala na wprowadzanie danych typu liczba rzeczywista, gdzie część ułamkowa jest oddzielona **przecinkiem** od części całkowitej liczby (**pl\_PL**) podczas wprowadzania ceny i dodawany jest symbol wartości **zł**

```
@ManagedBean
@RequestScoped
public class Managed_produk {

    @EJB
    private Fasada_warstwy_biznesowej fasada;
    private DataModel items;
    private int stan = 1;
    private Produkt_dto produkt_dto = new Produkt_dto();
    private NumberConverter number_convert=new NumberConverter();

    public NumberConverter getNumber_convert() {
        this.number_convert.setPattern("#####.## zł");
        return number_convert;
    }

    public void setNumber_convert(NumberConverter Number_convert) {
        this.number_convert = Number_convert;
    }
```



**(P5) Wprowadzanie danych** – znacznik konwertera typu **converter** z atrybutem **pattern** określającym sposób wprowadzania danych ceny (w kodzie konwertera). **(SR3)**

```
<h:outputLabel for="cena" value="#{bundle['dodaj_produkt2.cena']}" />
<h:inputText
    id="cena"    title="#{bundle['dodaj_produkt2.cena1']}"
    value="#{managed_produkt.cena}"
    required="true" requiredMessage="#{bundle['dodaj_produkt2.blad_cena']}" >
    <f:converter binding="#{managed_produkt.number_convert}" />
</h:inputText>
```

**(P6) Wprowadzanie danych** – znacznik konwertera typu **convertDateTime** z atrybutem **pattern** określającym sposób wprowadzania daty: **numer\_dnia-numer\_miesiaca-rok. (SR2)**

```
<h:outputLabel for="data" value="#{bundle['dodaj_produkt2.podaj_data']}" />
<h:inputText
    id="data"    title="#{bundle['dodaj_produkt2.podaj_data']}"
    value="#{managed_produkt.data_produkcji}"
    required="true" requiredMessage="#{bundle['dodaj_produkt2.podaj_data_blad']}" >
    <f:convertDateTime pattern="dd-MM-yyyy" />
</h:inputText>
```

**(P7) Wyświetlanie danych** - Konwerter typu **convertDateTime** z atrybutami **dateStyle**, **locale**, **timeStyle** oraz **type**, gdzie dla wybranych wartości atrybutów uzyskano następującą postać wyświetlanej daty:  
np. **Saturday, February 2, 2013 12:00:00 AM GMT (SR2)**

```
<h:outputLabel value="#{bundle['lista_produkow.data']}"  
              for="data"/>
```

```
<h:outputText id="data" value="#{managed_produk.data_produkcji}">  
  <f:convertDateTime dateStyle="full"  
                    locale="en_US" timeStyle="long" type="both"/>
```

```
</h:outputText>
```

**(P8) Wyświetlanie danych** - Konwerter typu **convertNumber** z przykładu **(P5)** z atrybutami **binding** oraz **pattern** (przesłania **pattern** z kodu w **Managed\_produk**), gdzie dla wybranych wartości atrybutów uzyskano następującą postać wyświetlanej ceny: np. **120,5 zł – lub 120.5 zł - (SR2)**

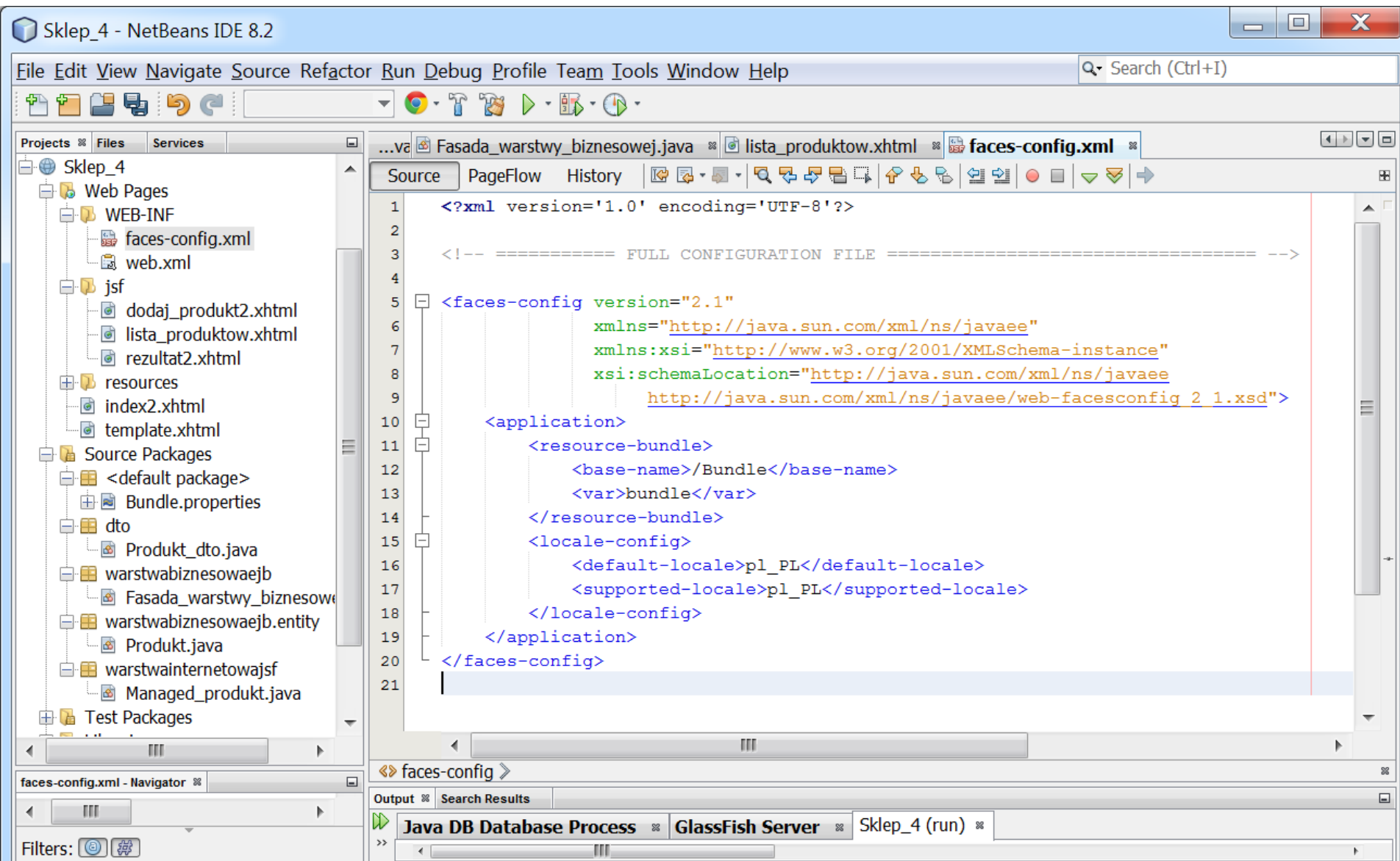
```
<h:outputLabel for="cena" value="#{bundle['lista_produkow.cena']}" />
```

```
<h:outputText id="cena" value="#{managed_produk.cena}">
```

```
  <f:convertNumber binding="#{managed_produk.number_convert}"  
                  pattern="####.### z&#322; -"/>
```

```
</h:outputText>
```

# Zawartość pliku typu faces-config.xml, określająca domyślną wartość atrybutu **locale** aplikacji (**pl\_PL**)



The screenshot displays the NetBeans IDE 8.2 interface. The main editor window shows the content of the `faces-config.xml` file. The XML code is as follows:

```
1 <?xml version='1.0' encoding='UTF-8'?>
2
3 <!-- ===== FULL CONFIGURATION FILE ===== -->
4
5 <faces-config version="2.1"
6     xmlns="http://java.sun.com/xml/ns/javaee"
7     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
8     xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
9         http://java.sun.com/xml/ns/javaee/web-facesconfig_2_1.xsd">
10
11     <application>
12         <resource-bundle>
13             <base-name>/Bundle</base-name>
14             <var>bundle</var>
15         </resource-bundle>
16         <locale-config>
17             <default-locale>pl_PL</default-locale>
18             <supported-locale>pl_PL</supported-locale>
19         </locale-config>
20     </application>
21 </faces-config>
```

The IDE interface includes a menu bar (File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help), a toolbar, a Projects view on the left showing the project structure for 'Sklep\_4', and a bottom status bar with tabs for 'Output', 'Search Results', 'Java DB Database Process', 'GlassFish Server', and 'Sklep\_4 (run)'.

Działanie konwertera typu **NumberConverter (P5)** oraz konwertera typu **convertDateTime (P6)**. Błąd wprowadzania ceny z kropką z powodu ustawienia języka **pl\_PL**

Dodaj produkt - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Dodaj produkt

localhost:14308/Sklep\_4/f

Most Visited Getting Started Suggested Sites Web Slice Gallery

Top

**Dodaj produkt**  
Lista produktów

Podaj nazwę produktu Produkt1

Podaj cenę produktu 100.50 zł

Podaj promocję produktu 10

Podaj datę produktu 05-04-2013

OK

Bottom

• j\_idt17:cena: '100.50 zł' is not a number pattern.

# Działanie konwertera typu **NumberConverter (P5)** oraz konwertera typu **convertDateTime (P6)**

The screenshot displays a Mozilla Firefox browser window titled "Dodaj produkt - Mozilla Firefox". The address bar shows the URL "localhost:14308/S". The page content includes a "Dodaj produkt" button and a form with the following fields:

Podaj nazwę produktu	Produkt1
Podaj cenę produktu	100,50 zł
Podaj promocję produktu	10
Podaj datę produktu	05-04-2013

An "OK" button is located below the date field. A tooltip "Bookmark this page" is visible over the browser's address bar. Two arrows from the title point to the "100,50 zł" and "05-04-2013" values.

Działanie konwertera typu **convertDateTime (P7)** w wersji narzuconej **en\_US**  
oraz konwertera typu **NumberConverter (P8)** w wersji domyślnej **pl\_PL**

Rezultat dodawania nowego produktu - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Rezultat dodawania nowego produktu X +

localhost:14308/Sklep\_4/

Most Visited Getting Started Suggested Sites Web Slice Gallery

Top

**Dodaj produkt**  
**Lista produktów**

Nazwa produktu	Produkt1
Cena netto produktu	100,5 zł -
Promocja produktu	10
Data produkcji produktu	Friday, April 5, 2013 12:00:00 AM GMT
Cena brutto produktu	90.45

Powrót

Bottom

Działanie konwerterów typu  
**convertNumber (P1), (P2), (P3)** oraz **convertDateTime (P4)** – pl\_PL

Lista produktów - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Lista produktów

localhost:14308/Sklep\_4/

Most Visited Getting Started Suggested Sites Web Slice Gallery

Top

**Dodaj produkt**  
Lista produktów

Id produktu	Nazwa produktu	Cena netto produktu	Promocja produktu	Data produkcji produktu	Cena brutto produktu
1	Produkt1	100,5 zł	10 %	piątek, 05-04-2013	90,45 zł

Bottom

# Przykład 4



Zawartość pliku typu **faces-config.xml** , określająca domyślną wartość **locale** aplikacji (**en\_US**)

```
<?xml version='1.0' encoding='UTF-8'?>
<!-- ===== FULL CONFIGURATION FILE ===== -->
<faces-config version="2.1"
  xmlns="http://java.sun.com/xml/ns/javaee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
    http://java.sun.com/xml/ns/javaee/web-facesconfig_2_1.xsd">

  <application>
    <resource-bundle>
      <base-name>/Bundle</base-name>
      <var>bundle</var>
    </resource-bundle>
    <locale-config>
      <default-locale>en_US</default-locale>
      <supported-locale>pl_PL</supported-locale>
    </locale-config>
  </application>
</faces-config>
```

Działanie konwertera typu **NumberConverter (P5)** oraz konwertera typu **convertDateTime (P6)**. Błąd wprowadzania ceny z przecinkiem z powodu ustawienia języka **en\_US** jako domyślny.

Dodaj produkt - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Dodaj produkt

localhost:14308/Sklep\_4/f

Most Visited Getting Started Suggested Sites Web Slice Gallery

Top

**Dodaj produkt**  
**Lista produktów**

Podaj nazwę produktu Produkt1

Podaj cenę produktu 100,50 zł

Podaj promocję produktu 10

Podaj datę produktu 05-04-2013

OK

Bottom

• j\_idt17:cena: '100,50 zł' is not a number pattern.

# Działanie konwertera typu **NumberConverter (P5)** oraz konwertera typu **convertDateTime (P6)**

Dodaj produkt - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Dodaj produkt

localhost:14308/Sklep\_4/f

Most Visited Getting Started Suggested Sites Web Slice Gallery

Top

**Dodaj produkt**  
**Lista produktów**

Podaj nazwę produktu	Produkt1
Podaj cenę produktu	100.50 zł
Podaj promocję produktu	10
Podaj datę produktu	05-04-2013

OK

Bottom

# Efekt zmiany formatowania domyślnego zdefiniowanego za pomocą znacznika **default-locale** na **en\_US** w pliku **faces-config.xml**

Rezultat dodawania nowego produktu

localhost:14308/Sklep\_4/

**Dodaj produkt**  
**Lista produktów**

Nazwa produktu Produkt1  
Cena netto produktu 100.5 zł -  
Promocja produktu 10  
Data produkcji produktu Friday, April 5, 2013 12:00:00 AM GMT  
Cena brutto produktu 90.45

Powrót

Atrybut **locale** ustawiony indywidualnie w **convertDateTime (P7)**

Działanie konwerterów typu **convertNumber (P8), (P1), (P2), (P3)** oraz **convertDateTime (P4)**

Lista produktów - Mozilla Firefox

localhost:14308/Sklep\_4/

**Dodaj produkt**  
**Lista produktów**

Id produktu	Nazwa produktu	Cena netto produktu	Promocja produktu	Data produkcji produktu	Cena brutto produktu
1	Produkt1	100.5 zł	%10.00	Friday, 05-04-2013	zł90.45

Bottom

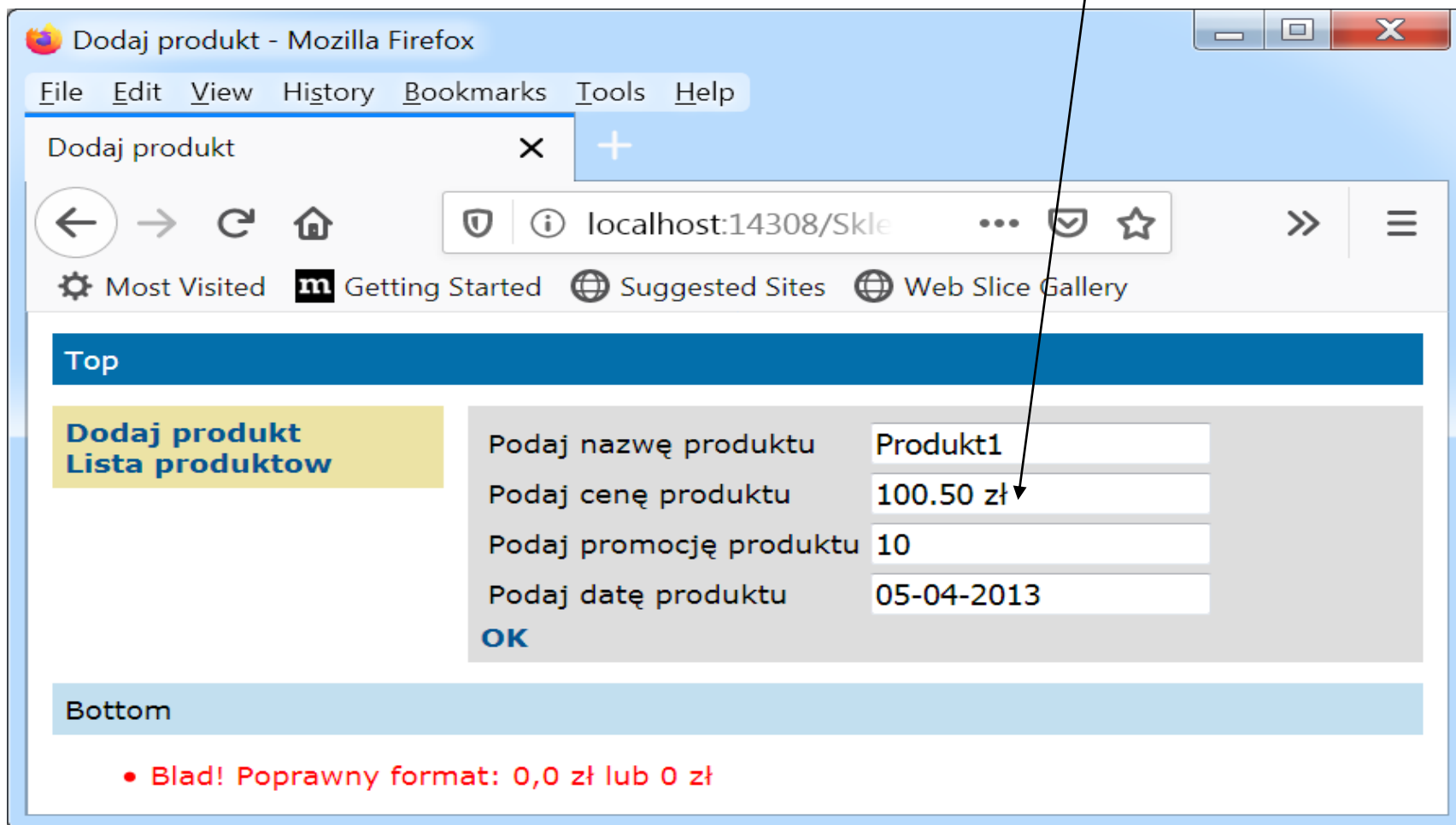
# Przykład 5

**(P9) Wprowadzanie danych** - Komponent typu **h:inputText** z atrybutem **converter**, odwołującym się do konwertera typu **NumberConverter** zdefiniowanego w klasie obiektu typu **Managed\_produkt (Przykład 3)**, gdzie nadano wartość jego atrybutu **pattern** metodą **setPattern**. Widok taki sam, jak dla konwertera **(P5)**.

**(SR4)**

```
<h:outputLabel value="#{bundle['dodaj_produkt2.cena']}"
               for="cena" />
<h:inputText
  id="cena"
  title="#{bundle['dodaj_produkt2.podaj_cena']}"
  value="#{managed_produkt.cena}"
  converter="#{managed_produkt.number_convert}"
  converterMessage="Bład! Poprawny format: 0,0 zł lub 0 zł"
  required="true"
  requiredMessage="#{bundle['dodaj_produkt2.podaj_cena_blad']}" >
</h:inputText>
```

Efekt zastosowania definicji konwertera wg 4-go sposobu (SR4) definiowania konwertera wartości liczbowych za pomocą atrybutu **converter** komponentu (**zastosowano znacznik `<default-locale>pl_PL</default-locale>` w pliku `favces-config.xml`) – należało część ułamkową oddzielić przecinkiem od części całkowitej. Komunikat o błędzie jest realizowany za pomocą atrybutu **converterMessage** komponentu **typu `h:inputText`****

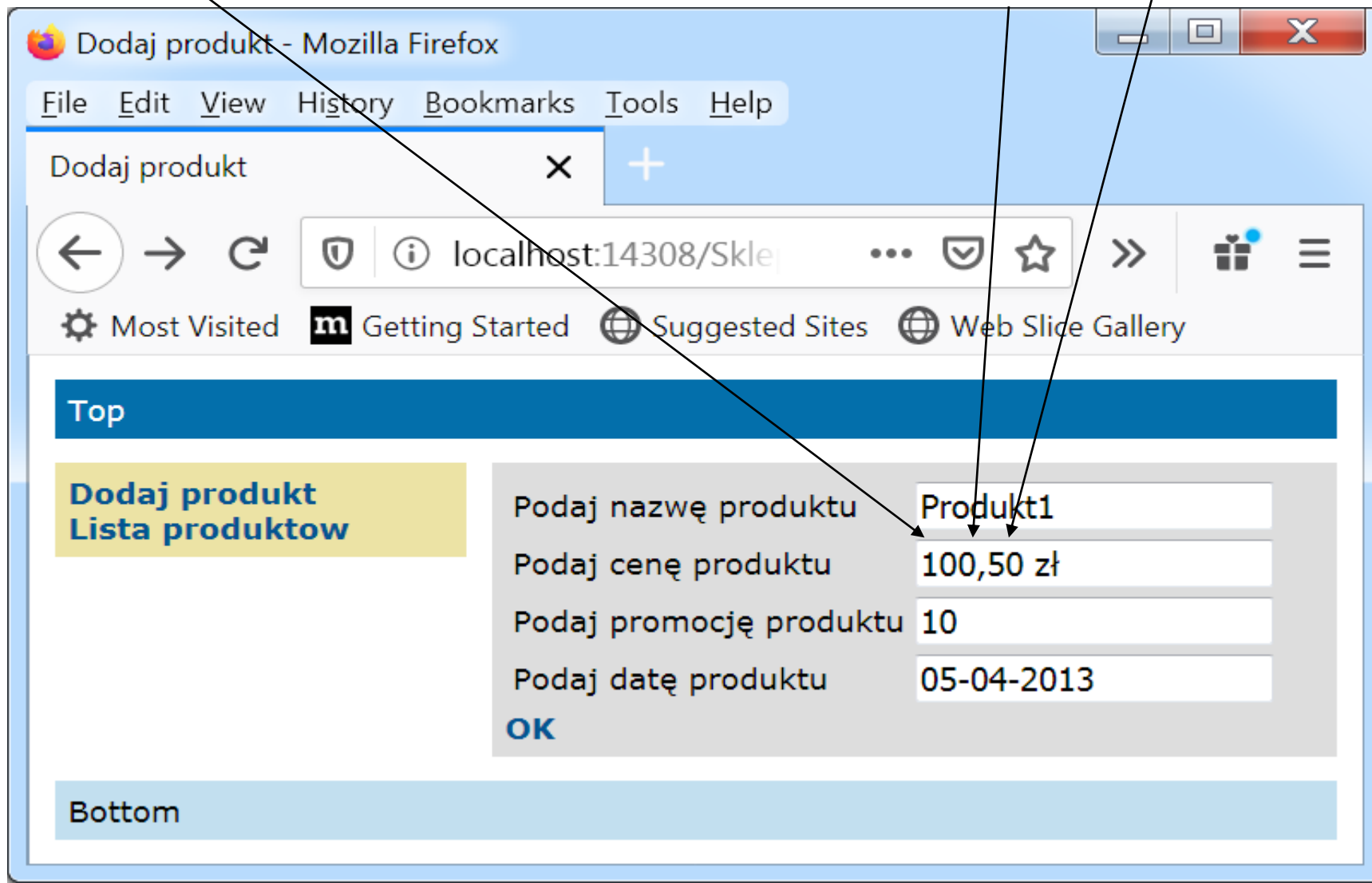


The screenshot shows a Mozilla Firefox browser window titled "Dodaj produkt - Mozilla Firefox". The address bar shows "localhost:14308/Skle". The page content includes a form with the following fields:

Podaj nazwę produktu	Produkt1
Podaj cenę produktu	100.50 zł
Podaj promocję produktu	10
Podaj datę produktu	05-04-2013

Below the form is an "OK" button. At the bottom of the page, a red error message is displayed: "• Bład! Poprawny format: 0,0 zł lub 0 zł". A black arrow points from the text in the first paragraph to the "100.50 zł" value in the form.

Działanie konwertera typu **NumberConverter**, zdefiniowanego w kodzie komponentu typu **Managed\_produk**t, zbindowanego **(SR2, SR3)** ze znacznikiem typu **f:convertNumber** lub **f:converter (P5)** lub **(SR4)** z atrybutem **converter (P9) -pl\_PL**





Działanie konwertera typu **NumberConverter (P8)**,  
zdefiniowanego w kodzie komponentu typu **Managed\_produk**  
zbindowanego (**SR4**) ze znacznikiem **f:convertNumber – pl\_PL**

Rezultat dodawania nowego produktu - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Rezultat dodawania nowego produ x +

localhost:14308/Sklep\_4/t...

Most Visited Getting Started Suggested Sites Web Slice Gallery

Top

**Dodaj produkt**  
**Lista produktow**

Nazwa produktu	Produkt1
Cena netto produktu	100,5 zł -
Promocja produktu	10
Data produkcji produktu	Friday, April 5, 2013 12:00:00 AM GMT
Cena brutto produktu	90.45

Powrót

Bottom

## Linki do materiałów zawierających informacje o formatowaniu

- <http://docs.oracle.com/javase/tutorial/i18n/format/simpleDateFormat.html>
- <http://docs.oracle.com/javase/tutorial/i18n/format/decimalFormat.html>