

SOA – laboratorium nr 4

JavaServer Faces

Celem laboratorium jest nauczenie się projektowa warstwę prezentacyjną aplikacji WWW wykorzystującej technologię JavaServer Faces w wersji 2.0.

JSF 2 – wprowadzenie

Konfiguracja Eclipse - dodanie szablonu XHTML dla potrzeb JSF

1. Otwórz menu Window/Preferences. Następnie z drzewka wybierz Web/HTML Files/Editor/Templates.
2. Wybierz i zaznacz New XHTML File (1.0 transitional). W oknie preview zaznacz tekst i skopiuj do schowka.
3. Wybierz New... i jako nazwę wpisz New XHTML for JSF i jako opis tekst Tworzy nowy plik XHTML dla JSF, w Context wybierz New HTML.
4. Dodaj do elementu html przestrzeń nazw i dodaj przedrostek h: do tagów head i body.
<html xmlns="http://www.w3.org/1999/xhtml" xmlns:h="http://java.sun.com/jsf/html"> <h:head> ... </h:head> <h:body> ... </h:body>
5. Zamknij klikając OK i następnie Apply.

Pierwsza prosta aplikacja

1. W projekcie typu JSF utwórz trzy pliki xhtml (index.xhtml, wygrana.xhtml, przegrana.xhtml) korzystając z wcześniej przygotowanego wzorca. W tym celu Kliknij prawym przyciskiem na nazwie projektu JSF w Project Explorer i wybierz HTML File. Nadaj nazwę, np. index.xhtml. Kliknij Next. W Templates wybierz wcześniej utworzony szablon, kliknij Finish. Podobnie utwórz kolejne pliki.
2. W pliku web.xml zdefiniuj domyślne pliki, które będą wybierane gdy w URL nie będzie podana nazwa zasobu. Zmodyfikuj welcome-file-list jak niżej. Dzięki temu wpisując URL w stylu localhost:8080/aplikacja/ będzie uruchomiona domyślnie strona index.xhtml.

```
<welcome-file-list>
  <welcome-file>index.xhtml</welcome-file>
</welcome-file-list>
```

3. W plikach xhtml zamieść następujące treści

index.xhtml

```
<h1>Sprawdź szansę na wygraną!</h1>
<h:form >
Podaj swoje dane:
PESEL <h:inputText /><br/>
Imię <h:inputText /><br/>
Nazwisko <h:inputText /><br/>
<h:commandButton value="Wyślij" action="#{szansa.wyslij}" />
</h:form>
```

wygrana.xhtml

```
<h1>Wygrałeś!! Gratulacje</h1>
```

przegrana.xhtml

```
<h:form>
<h1>Przegrałeś!</h1>
```

```
<p>Spróbuj jeszcze raz <h:link value="kliknij" outcome="index" /> </p>
</h:form>
```

4. Utwórz klasę Javy `com.example.faces.Szansa` z następującym kodem:

```
@ManagedBean
public class Szansa {
    public String wyslij() {
        if (Math.random() < 0.2 ) {
            return "wygrana";
        } else {
            return "przegrana";
        }
    }
}
```

5. Uruchom aplikację i przetestuj jej działanie.

Praca własna.

Zadanie 1

Utwórz aplikację, która po uruchomieniu losuje liczbę z przedziału 1-5. Zapisz ją jako zmienną statyczną w klasie obsługującej `ManagedBean`, np. tak:

```
static int losuj = (int)(Math.random() * 5 + 1);
```

Na stronie głównej (`index.xhtml`) utwórz 5 przycisków, które powinny przenosić do pięciu stron o nazwach `1.xhtml`, `2.xhtml`, `3.xhtml`, `4.xhtml` i `5.xhtml`. Na stronach tych pojawia się jej numer.

`index.xhtml`

`1.xhtml`

Wciśnięcie przycisku z odpowiednią liczbą przenosi do strony o danym numerze. Jednak, gdy numer strony jest taki sam jak wybranej liczby, wówczas przeniesie na stronę o nazwie `trafiony.xhtml`, która wyświetla informację o trafieniu liczby wylosowanej przez komputer. Jednocześnie losowana jest nowa liczba.

Zadanie 2

Napisz aplikację, która przelicza wartość złotówki na wybraną przez użytkownika walutę i odwrotnie. Założenia:

1. Aplikacja składa się z 2 części: obliczeniowej oraz administracyjnej.
2. W części obliczeniowej na stronie znajduje się pole tekstowe z możliwością wpisania kwoty, rozwijana lista z nazwą waluty, w jakiej podana została kwota, oraz lista walut, na którą ma zostać dokonane przeliczenie oraz przycisk dokonujący obliczeń i przenoszący na nową stronę w wynikami. (w wersji prostszej listy są generowane statycznie, a wartości pobierane z odpowiedniej klasy. W wersji rozbudowanej do generowania list potrzebna jest druga część programu).
3. Część administracyjna polega na zaprogramowaniu jeszcze jednej strony, na której użytkownik może dodawać waluty, które mają się pojawiać w listach rozwijanych. Domyślnie mają być dwie waluty- PLN i EUR.

Wartość kursu walut ma być średnią wartością ściągniętą z NBP. Skorzystaj z klasy `NBPConnector`.

Wartość waluty otrzymuje się wywołując funkcję `exchangeRate("skrótowa_nazwa_valuty")`:
`String euro = NBPConnector.exchangeRate("EUR").`

Uwaga! Wartości waluty ściągnij tylko raz w czasie działania całej aplikacji.

Zadania do oddania

Zadanie 3

Zbuduj formularz do wyświetlania informacji o wybranym samochodzie w komisie. Formularz zawiera następujące pola:

- Marka samochodu (lista rozwijalna)
- Model (lista rozwijalna)

Zawartość pola model zależy od wybranej marki. Zmiana marki skutkuje zmianą dostępnych w polu model listy modeli.

- pola określające zakres cenowy szukanego samochodu
 - pole wyboru typ silnika (ON lub benzyna)
 - pola do wprowadzenia swojego imiennika oraz numeru telefonu.

oraz przycisk zatwierdzający i czyszczący formularz. Przycisk staje się aktywny dopiero gdy wszystkie pola są poprawnie wypełnione.

Jeśli wszystkie pola zostały poprawnie wypełnione, to po naciśnięciu przycisku akceptacji na dole ekranu wyświetla się lista pojazdów (marka model cena) ofert spełniających zdefiniowane warunki.

Zadanie 4.

Jesteś już specjalistą w zakresie ankiet, więc otrzymujesz coraz więcej zleceń. Twój kolejny klient poprosił Cię o przygotowanie ankiety dla internetowego sklepu z odzieżą. Ma ona być skierowana zarówno do nowych klientów użytkowników, badając ich preferencje zakupowe, jak również do obecnych klientów, sprawdzając ich poziom zadowolenia z towarów i obsługi. Twoim zadaniem jest zaproponowanie i wykonanie ankiety z uwzględnieniem części wspólnej, która zawiera:

- imię (wymagane)
- adres e-mail (wymagane, poprawny adres)
- wiek (wartość między 10 a 100)
- płeć
- wykształcenie
- wzrost (dla kobiet w zakresie 150 -185 dla mężczyzn 165-200)

Na podstawie płci należy wyświetlić dodatkowe informacje:

w przypadku kobiet:

- obwód biustu
- wielkość miseczki
- talia
- biodra
- długość nogi

w przypadku mężczyzn:

- klatka
- pas
- długość nogi

Dodatkowo po zaakceptowaniu powyższej części formularza wyświetla się dodatkowa sekcja pytań:

Pytania wspólne dla nowych klientów:

- Ile jesteś w stanie przeznaczyć miesięcznie na zakup ubrania? (Lista wyboru, odpowiedzi: „do 100 zł”, „100-500 zł”, „500-1000 zł”, „powyżej 1000 zł”.)
- Jak często dokonujesz zakupu ubrania? (Lista wyboru, odpowiedzi: „Codziennie”, „Raz w tygodniu”, „Raz w miesiącu”, „Kilka razy w roku”.)
- W jakich kolorach preferujesz ubrania? (Lista wielokrotnego wyboru, odpowiedzi: „Kolorowo-jaskrawych”, „Stonowanych w szarościach”, „W czerni i bieli”, „W samej czerni”.)
- Jakiego rodzaju ubrania najchętniej kupujesz?

W przypadku, gdy klient jest kobietą, pobierz odpowiedź z listy (Lista wielokrotnego wyboru, odpowiedzi: „garsonki”, „bluzki”, „spódniczki”, „spodnie”.)

Gdy mężczyzną: (Lista wielokrotnego wyboru, odpowiedzi: „spodnie”, „spodenki”, „garnitury”, „koszule”, „krawaty”.)

W ankiecie badania satysfakcji klienta powinienś dodatkowo wyświetlić odpowiednio sformułowane pytania w zależności od płci:

- Kiedy ostatnio robiłaś/robiłeś u nas zakupy (Lista wyboru, odpowiedzi: „Wczoraj”, „W zeszłym tygodniu”, „W zeszłym miesiącu”, „Kilka miesięcy temu”, „Nie pamiętam”.)
- Czy jesteś zadowolona/zadowolony z zakupu (Lista wyboru, odpowiedzi: „Tak”, „Raczej tak”, „Raczej nie”, „Nie”, „Nie wiem”.)
- Czy jesteś zadowolona/zadowolony z naszych pracowników? (Lista wyboru, odpowiedzi: „Tak”, „Raczej tak”, „Raczej nie”, „Nie”, „Nie wiem”.)
- Wpisz swoje uwagi. (Pole tekstowe z możliwością wpisania tekstu w wielu liniach.)

Konieczne sprawdzić poprawność wpisywanych danych, a w podsumowaniu ankiety wyświetl informacje wypełnione przez użytkownika. Klient zażyczył sobie również, by na stronie była możliwość losowego wyświetlania reklam wraz z rejestracją ilości kliknięć banera.