

Praca domowa 11– nbp

Szlachetka Miłosz
Nr albumu. 114014

1.Cel zadania

W zadaniu należało zaimplementować usługę sieciową wyznaczającą stosunek jednej waluty, której kod przekazywany był w URL, do waluty referencyjnej, której kod zwracany był przez metodę `currencyId` komponentu `MdbManager`.

2.Struktura programu

@Path("/exchangeRate")

public class ExchRates - klasa będąca usługą sieciową w standardzie `WebService's` (JAX-RS). Klasa zawiera:

void registerMdbManager() - metoda rejestrująca i zestawiająca połączenie z komponentem EJB (Session bean) o nazwie `MdbManager`. Połączenie jest realizowane przy użyciu mechanizmu JNDI.

@GET

@Path("/{code}")

@Produces("text/plain")

public String calculateRatio(@PathParam("code") String currencyCode) - Endpoint (metoda) odpowiadający za obsługę żądań typu GET kierowanych na adres zdefiniowany w adnotacji `@Path`. Parametrem żądania (parametr jest fragmentem URL) jest kod waluty, która będzie poddana procesowi przeliczania. Metoda ta zwraca stosunek wartości jednostki walutowej, której kod podany jest w URL, do wartości waluty referencyjnej, której kod jest pobierany z komponentu EJB o nazwie `MdbManager`. Wynik zwracany jest z dokładnością do 4 miejsc po przecinku jako wartość typu `String`.

Document getNBPDocument() - metoda, która wysyła żądanie typu GET na adres: <http://www.nbp.pl/kursy/xml/LastA.xml> otrzymując w odpowiedzi dane dotyczące kursów walut w formacie XML. Następnie pobrane dane są konwertowane do postaci dokumentu, który może być przeszukiwany (typ `Document`). Następnie dokument ten jest zwracany przez tę metodę.

double roundDouble(double value, int decimalPlaces) - metoda, która zaokrągla wartość typu `double` do zadanej w parametrze **decimalPlaces**, liczby miejsc po przecinku. Tak zaokrąglona wartość jest następnie zwracana.

@Remote

interface IMdbManager - interjes typu Remote, który zawiera deklarację metody String currencyId(). Metoda currencyId zwraca kod waluty referencyjnej.

class CurrencyEvaluator - klasa, która zawiera implementację metod odpowiedzialnych za parowanie (wyszukiwanie kursów walut oraz przeliczników) dokumentu XML oraz wyliczanie stosunku waluty przeliczanej do waluty referencyjnej. Metody tej klasy to:

static double calculateRatioFromDocument(Document doc, String currencyCode, String referencedCurrencyCode) - metoda parsująca dokument (Document doc) podany w parametrze metody. Wyszukuje w nim kursy oraz przeliczniki dla zadanych kodów walut (currencyCode, referencedCurrencyCode). Następnie wyznacza stosunek waluty przeliczanej do waluty referencyjnej z uwzględnieniem przelicznika. Formuła wyznaczająca tens stosunek ma postać :

$$(\text{kurs_waluty_przeliczanej} / \text{przelicznik_waluty_przeliczanej}) / (\text{kurs_waluty_referencyjnej} / \text{przelicznik_waluty_referencyjnej}).$$

String getCodeFromXMLElement(Element element) - dla danego elementu (część dokumentu XML) pobiera kod waluty znajdujący się w tym elemencie.

String getExchangeRateFromXMLElement(Element element) - dla danego elementu (część dokumentu XML) pobiera kurs waluty znajdujący się w tym elemencie.

String getQuantityFromXMLElement(Element element) - dla danego elementu (część dokumentu XML) pobiera przelicznik waluty znajdujący się w tym elemencie.

Double stringToDouble(String val) - dla danej wartości typu String zamienia wszystkie przecinki na kropki a następnie konwertuje uzyskaną wartość do typu Double i zwraca ją.

3. Mechanizm wyszukiwania i zestawiania połączenia

Wyszukiwanie komponentu EJB zostało zrealizowane dzięki interfejsowi JNDI, który daje możliwość wyszukiwania i uzyskiwania dostępu do serwisów i katalogów przy użyciu przenośnych nazw. Aby uzyskać dostęp do komponentu EJB o nazwie MdbManager użyto następującej nazwy, która określa położenie komponentu:

java:global/mdb-project/MdbManager!pl.jrj.mdb.IMdbManager"

java:global - przestrzeń nazw, do wyszukiwania zdalnych EJB.

mdb-projec - deployment descriptor

MdbManager- nazwa komponentu EJB

!IMdbManager - interfejs komponentu

Połączenie z komponentem EJB było zrealizowane poprzez utworzenie kontekstu (InitialContext), a następnie wywołanie na nim metody lookup, z parametrem będącym adresem komponentu.

4. Algorytm

Początkowo z adresu URL żądania GET pobierana jest kod waluty przeliczanej. Następnie zestawiane jest połączenie z komponentem EJB typu remote o nazwie MdbManager. Korzystając z metody currencyId komponentu MdbManager pobierany jest kod waluty referencyjnej. Później wysyłane jest żądanie typu GET na adres <http://www.nbp.pl/kursy/xml/LastA.xml>, co skutkuje pobraniem zawartości dokumentu w formacie XML z kursami walut oraz ich przelicznikami. Następnie wykonywane jest parsowanie pobranej zawartości dokumentu. Przeszukiwanie to polega na iterowaniu po wszystkich elementach oznaczonych tagami <pozycja></pozycja>. Następnie dla każdego takiego elementu pobierana jest wartość między tagami <kod_waluty></kod_waluty>. Wartość ta jest następnie porównywana z kodem waluty pobranym z URL oraz kodem waluty pobranym z komponentu EJB o nazwie MdbManager. Jeśli kody się zgadzają (są takie same) to dla elementu pobierany jest kurs średni oraz przelicznik. Po sparsowaniu całego dokumentu XML sprawdzane jest czy zostały pobrane wszystkie wartości tzn. średni kurs dla waluty przeliczanej oraz referencyjnej oraz przeliczniki dla obu walut. Jeśli tak to wyznaczany jest stosunek waluty przeliczanej do waluty referencyjnej zgodnie z formułą:

$$\frac{(\text{kurs_waluty_przeliczanej} / \text{przelicznik_waluty_przeliczanej})}{(\text{kurs_waluty_referencyjnej} / \text{przelicznik_waluty_referencyjnej})}.$$

Obliczony stosunek walut jest następnie zaokrąglany do 4 miejsc po przecinku i konwertowany do typu String, a następnie jest zwracany jako odpowiedź na żądanie GET.