

PES Domácí úkol 2 - Zpracování objednávek (kombinovaná forma)

Zadat úkol skupině č. 1 Zadat úkol skupině č. 2

Kontext

Cílem tohoto cvičení je procvičit si práci s XML soubory a s výjimkami.

Zadání

Napište program vyřizující objednávky zboží internetového obchodu. Budete pracovat s následujícími daty:

- Databáze produktů. Ta popisuje prodávané zboží.
- Sklad. Stav skladu říká, kolik kterého zboží máme aktuálně k dispozici.
- Objednávky. Popisují, kteří zákazníci si u nás objednali jaké zboží a v jakém počtu.

Všechna vstupní data budou ve formátu XML, pojďme si jednotlivé soubory popsat blíže. Než dokonalému formálnímu popisu jednotlivých XML dokumentů dáme přednost především ukázkám, ze kterých by měla být daná struktura zřejmá. Je ovšem potřeba dát pozor na to, že vstupní data mohou obsahovat chyby.

Databáze produktů

Ukázka:

```
<?xml version="1.0"?>
     cproducts>
         cproduct code="PE316">
4
             <name>Zvukotesna vlozka SPIRE pro redukci hluku</name>
             <price>104</price>
6
         </product>
         cproduct code="RI080">
8
             <name>FAN COOLER PASIVNI</name>
             <price>491</price>
10
         cproduct code="QE091">
11
             <name>Nabijecka PENTAGRAM Pocket USB
12
13
             <price>608</price>
14
         </product>
15
         cproduct code="QU286">
             <name>RAMECEK ViPower VYMENNY PLAST ATA
16
17
             <price>572</price>
         </product>
19
         cproduct code="PU441">
20
             <name>RAZER LYCOSA Gaming Keyboard
21
             <price>160</price>
22
         </product>
23
         cproduct code="PI287"> <!-- CHYBNA CENA -->
24
             <name>VoIP Perfect GW</name>
             <price>-999</price>
25
26
         < code="XX449"> <!-- CHYBNY KOD -->
27
28
             <name>LOGO SAFE obal na Notebook neopren pro
             <price>328</price>
30
         </product>
     </products>
32
```

Produkty jsou popisovány jednotlivými elementy **product**. Každý produkt má svůj kód (atribut **code** elementu product), jméno (podelement **name**) a cenu (podelement **price**).

Na data máme následující omezení:

- · Kód produktu musí být unikátní.
- Kód produktu musí mít následující tvar. Začíná velkým písmenem z množiny {P, Q, R, S, T}, pokračuje velkým písmenem z množiny {A, E, I, O, U} a končí číslem v rozsahu 001-450.
- Cena musí být kladné číslo.

Údaje o jakémkoliv produktu porušující některé z těchto pravidel považujeme za neplatné.

Stav skladu

Ukázka:

Produkty jsou opět popisovány elementy **product**. Atribut **code** se odkazuje na kód produktu z databáze produktů. Atribut **amount** určuje aktuální množství produktu na skladě.

Na data máme následující omezení:

- Pro daný produkt musí existovat (platný) záznam v databázi produktů.
- Počet kusů musí být kladné číslo (skutečně kladné číslo, nulu nedovolujeme).

Údaje o jakémkoliv produktu porušující některé z těchto pravidel považujeme za neplatné. Situace, kdy pro nějaký produkt vyskytující se v databázi nemáme ve stavu skladu žádný záznam, je zcela korektní: vyjadřuje, že produkt v současné době není dostupný.

Objednávky

Ukázka:

```
<?xml version="1.0"?>
     <orders>
         <order number="99503" date="2012-10-18T15:29:37">
 3
4
             <address type="shipping">
5
                 <name>Ellen Adams</name>
6
                 <street>123 Maple street</street>
7
                 <city>Mill Valley</city>
8
                 <state>CA</state>
9
                 <zip>10999</zip>
10
                 <country>USA</country>
11
             </address>
12
             <address type="billing">
13
                 <name>Tai Yee</name>
14
                 <street>8 Oak Avenue</street>
15
                 <city>Old Town</city>
                 <state>PA</state>
17
                 <zip>95819</zip>
18
                 <country>USA</country>
19
             </address>
20
             <notes>Please leave packages in shed by driveway.
21
             <items>
                 <item code="PE316">
23
                     <name>Zvukotesna vlozka SPIRE pro redukci hluku</name>
```

```
24
                      <quantity>1</quantity>
25
                      <price>104</price>
26
                  </item>
                  <item code="RI080">
27
                      <name>FAN COOLER PASIVNI</name>
28
                      <quantity>2</quantity>
30
                      <price>491</price>
                  </item>
32
              </items>
          </order>
35
     </orders>
```

Každou objednávku reprezentuje element **order**. Objednávka obsahuje jednu nebo dvě adresy (adresa typu billing je přítomna vždy, adresa typu shipping může chybět). Položky objednávky popisují elementy **item** (vnořené v elementu **items**). Kód produktu je hodnotou atributu **code**, název, množství a cena jsou dány po řadě obsahem elementů **name**, **quantity** a **price**.

Číslo objednávky nás nebude příliš zajímat, za to pro nás bude důležité její datum (atribut **date**). To je uvedeno ve tvaru, kdy jednotlivé části data jsou odděleny spojovníky, jednotlivé části času dvojtečkami a čas samotný je od data oddělen písmenem T. Datum v tomto tvaru můžete na odpovídající objekt v Ruby buď převést ručně "vypreparováním" jednotlivých položek, nebo můžete použít metodu Time.xmlschema.

Cíl a průběh programu

Úkolem programu je především pokusit se vyřídit jednotlivé objednávky. Zároveň ale chceme využít seznamu objednávek k tomu, abychom si vybudovali databázi zákazníků, která nám dosud chybí. Třetí funkcí programu bude informovat nás o všech chybách v datech, na které narazí. Program tedy bude mít následující výstupy (které budou popsány dále):

- Sklad. Jedná se o stav skladu po vyřízení všech objednávek, které bylo možné zpracovat.
- Databáze zákazníků. Seznam zákazníků (a údajů o nich) získaný ze seznamu objednávek. Zajímají nás pouze osoby uvedené v adrese typu "billing" (jak můžete vidět v ukázce),
- Upozornění na chyby. Seznam všech chyb nebo problémů, ke kterým při vyřizování objednávek došlo.

Průběh programu bude zhruba následující:

- 1. Program načte databázi produktů. Chybné položky neuvažuje; na jejich výskyt upozorní.
- 2. Program načte stav skladu. Chybné položky neuvažuje; na jejich výskyt upozorní.
- 3. Program projde jednotlivé objednávky v pořadí dle data (a času) objednávky (toto pořadí nemusí odpovídat pořadí objednávek ve vstupním souboru) a pro každou vykoná následující kroky.
 - a. Zapamatuje si údaje o zákazníkovi (z adresy typu "billing").
 - b. Pro každou položku objednávky vykoná následující kroky.
 - i. Program srovná název a cenu produktu z objednávky s názvem a cenou produktu v databázi produktů (srovnání děláme na základě kódu produktu). Pokud se názvy nebo ceny neshodují nebo pokud produkt s daným kódem v databázi produktů neexistuje, upozorní program na tuto situaci a považuje položku objednávky za neplatnou a dále se jí nezabývá.
 - ii. Program podle kódu produktu a požadovaného množství zjistí, zda máme produktu dostatečné množství na skladu. Pokud ano, odečte požadované množství od stavu skladu (tím považujeme položku objednávky za vyřízenou). Pokud ne, program na situaci nedostatečného množství produktu upozorní.
- 4. Program uloží výsledný stav skladu.
- 5. Program uloží databázi zákazníků.

Výstupní data

Výsledný stav skladu bude ve shodném XML formátu, jako byl vstupní stav. Databáze zákazníků bude následujícího tvaru:

O detekovaných chybách ve vstupních datech uživatele informujte vhodnými chybovými hlášeními zapisovanými na standardní výstup (do konzole).

Argumenty programu

Program bude očekávat pět argumentů z příkazové řádky, a to (po řadě) s následujícím významem:

- 1. soubor se vstupní databází produktů,
- 2. soubor se stavem skladu,
- 3. soubor se seznamem objednávek,
- 4. soubor s výslednou databází produktů (= výsledný stav skladu),
- 5. soubor s výslednou databází zákazníků.

Například tedy program může být spuštěn takto:

1 | ruby main.rb products.xml store.xml orders.xml store_output.xml customers.xml

Struktura programu

Předpřipravili jsme několik tříd, které program bude využívat. Jedná se ovšem pouze o jakousi kostru skládající se především z definice rozhraní některých metod. Předpokládané použití metod přímo plyne ze zadání celé úlohy. Jednotlivé okomentované třídy naleznete v příloze artefaktu.

Řešení

Řešení posílejte tlačítkem níže:

Odeslat řešení domácího úk...