

Харківський національний університет імені В. Н. Каразіна
Факультет комп'ютерних наук

ЗВІТ
З ЛАБОРАТОРНОЇ РОБОТИ №3

ДИСЦИПЛІНА: «КРОС-ПЛАТФОРМНЕ ПРОГРАМУВАННЯ»

Виконав: студентка групи КС22
Мазуренко Анжеліки
Перевірив: Споров Олександр
Євгенович

Харків
2024

Лабораторна робота No3 Java & XML

Git Repository: https://github.com/miorezu/LB3_JavaXML

Завдання №0

Із сайту з відкритими даними (<https://catalog.data.gov/dataset/popular-baby-names>) було отримано свіжий (від 3 березня, 2023), великий за розміром датасет в XML форматі з інформацією про популярні імена дітей у місті Нью-Йорк. Цей датасет складений за офіційною інформацією із служби реєстрації актів цивільного стану міста Нью-Йорка. Архів з цим датасетом має назву **Popular_Baby_Names_NY.zip** та розміщений в лекційному Гугл-класі в розділі **Методичні вказівки з виконання лабораторних робіт**.

Кожен запис цього датасету представляє інформацію про дитину: вказано дату народження, гендер, етнічну приналежність мами, власне ім'я дитини, кількість (count) дітей з цим іменем та рейтинг (rating) імені у відповідній групі. Потрібно провести попередній аналіз цих даних та вибрати з них лише потрібну для подальшої роботи інформацію.

Виконати наступні завдання:

1. Написати програму для виведення на екран частини XML документу за допомогою SAX парсеру без валідації для вивчення його структури та вмісту; програмно отримати перелік всіх тегів, імена яких присутні в документі.

Для виконання першого завдання було створено 2 класи: SAXParser та SAXHandler, які знаходяться у пакеті Task1.

Перший клас є точкою запуску обробки xml документа, а другий обробляє та виводить інформацію у консоль. Щоб не виводити весь документ ми у SAXParser ми задаємо максимальну кількість тегів row, що будуть виведені.

```
Start Document Processing
  Start Element (response) processing
    Start Element (row) processing
      Start Element (row) processing
        Attributes:
          _id = row-v4f5~xz3v-vr86
          _uuid = 00000000-0000-0000-E0ED-52E8592E0A7A
          _position = 0
          _address = https://data.cityofnewyork.us/resource/\_25th-nuif/row-v4f5~xz3v-vr86
        Start Element (brth_yr) processing
          2011
        Stop Element (brth_yr) processing
        Start Element (gndr) processing
          FEMALE
        Stop Element (gndr) processing
        Start Element (ethcty) processing
          HISPANIC
        Stop Element (ethcty) processing
        Start Element (nm) processing
          GERALDINE
        Stop Element (nm) processing
        Start Element (cnt) processing
          13
        Stop Element (cnt) processing
        Start Element (rnk) processing
          75
        Stop Element (rnk) processing
      Stop Element (row) processing
    Start Element (row) processing
      Attributes:
        _id = row-gdep~mr7x-dj3u
        _uuid = 00000000-0000-0000-68AD-7B741D1DF31B
        _position = 0
        _address = https://data.cityofnewyork.us/resource/\_25th-nuif/row-gdep~mr7x-dj3u
      Start Element (brth_yr) processing
        2011
      Stop Element (brth_yr) processing
      Start Element (gndr) processing
        FEMALE
      Stop Element (gndr) processing
      Start Element (ethcty) processing
        HISPANIC
      Stop Element (ethcty) processing
      Start Element (nm) processing
        GIA
      Stop Element (nm) processing
      Start Element (cnt) processing
        21
      Stop Element (cnt) processing
      Start Element (rnk) processing
        67
      Stop Element (rnk) processing
    Stop Element (row) processing
```

```

Start Element (row) processing
  Attributes:
    _id = row-4f22~xggt-ah4b
    _uuid = 00000000-0000-0000-B276-36740F71A706
    _position = 0
    _address = https://data.cityofnewyork.us/resource/\_25th-nujf/row-4f22~xggt-ah4b
  Start Element (brth_yr) processing
    2011
  Stop Element (brth_yr) processing
  Start Element (gndr) processing
    FEMALE
  Stop Element (gndr) processing
  Start Element (ethcty) processing
    HISPANIC
  Stop Element (ethcty) processing
  Start Element (nm) processing
    GIANNA
  Stop Element (nm) processing
  Start Element (cnt) processing
    49
  Stop Element (cnt) processing
  Start Element (rnk) processing
    42
  Stop Element (rnk) processing
  Stop Element (row) processing
List of all tags:[response, row, brth_yr, gndr, ethcty, nm, cnt, rnk]

Process finished with exit code 0

```

Рисунок 1 – 3 – результати виконання першого завдання з maxRows = 3

2. За невеликим характерним фрагментом скласти xsd схему документу, створити валідатор та перевірити, чи правильно було зрозуміло структуру документу.

Згенерували схему - Popular_Baby_Names_NY.xsd. Для її перевірки був створений валідатор – ValidatorXSD. А для обробки його помилок – MyErrorHandler.

Передаємо у наш валідатор файли Popular_Baby_Names_NY.xml та Popular_Baby_Names_NY.xsd та перевіряємо.

```

C:\Users\User\.jdk\openjdk-19.0.2\bin\java.exe
The file Popular_Baby_Names_NY.xml is valid

Process finished with exit code 0

```

Рисунок 4 – результати виконання другого завдання

3. Написати програмне рішення, що за допомогою SAX парсеру без валідації отримає назви всіх національних груп, що представлені в документі.

ParserEthnicity по прикладу першого завдання з точкою запуску обробки xml документа, а HandlerEthnicity обробляє документ, додає етнічність у ArrayList, якщо вона ще в ньому не знаходиться. ArrayList ethnicities зберігає у собі назви етносу без повторювань. Виводимо інформацію.

```
Start Document Processing
Finish Document Processing

All ethnicities:
[HISPANIC, WHITE NON HISPANIC, ASIAN AND PACIFIC ISLANDER, BLACK NON HISPANIC, ASIAN AND PACI, BLACK NON HISP, WHITE NON HISP]

Process finished with exit code 0
```

Рисунок 5 – результати виконання 3 завдання

4. Написати додаток, що з всього XML документу вибирає задану кількість найбільш популярних імен в заданій етнічній групі із зберіганням інформації про: ім'я, гендер, кількість імен та рейтинг імен, а також створює відповідні Java об'єкти для зберігання цієї інформації та сортує інформацію по збільшенню номеру в рейтингу. Зберегти вибрану та відсортовану інформацію до нового XML файлу за допомогою DOM парсеру.

Прочитати цей новий документ за допомогою DOM парсеру та вивести інформацію, що в ньому зберігається, на екран.

SearchPopularParser є компонентом, який використовується для парсингу XML-документа та повертає список об'єктів.

SearchPopularHandler є обробником подій для парсингу XML, витягує інформацію про рік народження, поле, етнічну приналежність, ім'я, кількість і ранг.

Зберігає всі імена з рангом 1 за допомогою BabyAList, де і сортує їх за значенням count. BabyAList є підкласом ArrayList, в ньому список об'єктів Baby з додатковими можливостями, такими як сортування та обмеження розміру, гетери та сетери.

Клас Baby інкапсулює дані про кожну дитину та надає методи для керування цими даними.

Клас SearchPopularParser повертав нам список об'єктів, щоб ми його зберігли й зручно передали у клас DOM, який відповідає за генерацію xml документа. Метод XMLCompose перебирає список об'єктів BabyAList і для кожного BabyAList створює елемент <ethnicity> з атрибутом GroupName, який представляє етнічну приналежність. Потім для кожного Baby в

BabyAList він створює елемент <Baby> з атрибутами, що становлять ім'я дитини, дату народження, стать, етнічну приналежність, кількість і ранг.

Метод XMLSave налаштовує Transformer формат виводу, а потім перетворює DOM-представлення XML-документа на текстовий формат, який записується у вказаний файл за допомогою StreamResult. Отримали - Babies_DOM.xml.

Тепер для виводу інформації на екран з цього документа використовуємо клас DOMAnalyser.

```
Start Document Processing
Finish Document Processing

Popular names:
[

BabyAList{babies=[
Baby{name='JAYDEN', birthdayDate='2011', gender='MALE', ethnicity='HISPANIC', count='426', rank='1'},
Baby{name='JAYDEN', birthdayDate='2011', gender='MALE', ethnicity='HISPANIC', count='426', rank='1'},
Baby{name='JAYDEN', birthdayDate='2011', gender='MALE', ethnicity='HISPANIC', count='426', rank='1'},
Baby{name='JAYDEN', birthdayDate='2011', gender='MALE', ethnicity='HISPANIC', count='426', rank='1'},
Baby{name='Liam', birthdayDate='2019', gender='MALE', ethnicity='HISPANIC', count='423', rank='1'},
Baby{name='Liam', birthdayDate='2017', gender='MALE', ethnicity='HISPANIC', count='407', rank='1'},
Baby{name='Liam', birthdayDate='2018', gender='MALE', ethnicity='HISPANIC', count='399', rank='1'}]
},

BabyAList{babies=[
Baby{name='JAYDEN', birthdayDate='2011', gender='MALE', ethnicity='HISPANIC', count='426', rank='1'},
Baby{name='JAYDEN', birthdayDate='2011', gender='MALE', ethnicity='HISPANIC', count='426', rank='1'},
Baby{name='JAYDEN', birthdayDate='2011', gender='MALE', ethnicity='HISPANIC', count='426', rank='1'},
Baby{name='JAYDEN', birthdayDate='2011', gender='MALE', ethnicity='HISPANIC', count='426', rank='1'},
Baby{name='Liam', birthdayDate='2019', gender='MALE', ethnicity='HISPANIC', count='423', rank='1'},
Baby{name='Liam', birthdayDate='2017', gender='MALE', ethnicity='HISPANIC', count='407', rank='1'},
Baby{name='Liam', birthdayDate='2018', gender='MALE', ethnicity='HISPANIC', count='399', rank='1'}]
},
```



```

Name: JAYDEN    birthdayDate: 2011    count: 426    ethnicity: HISPANIC    gender: MALE    rank: 1
Name: JAYDEN    birthdayDate: 2011    count: 426    ethnicity: HISPANIC    gender: MALE    rank: 1
Name: JAYDEN    birthdayDate: 2011    count: 426    ethnicity: HISPANIC    gender: MALE    rank: 1
Name: JAYDEN    birthdayDate: 2011    count: 426    ethnicity: HISPANIC    gender: MALE    rank: 1
Name: Liam     birthdayDate: 2019    count: 423    ethnicity: HISPANIC    gender: MALE    rank: 1
Name: Liam     birthdayDate: 2017    count: 407    ethnicity: HISPANIC    gender: MALE    rank: 1
Name: Liam     birthdayDate: 2018    count: 399    ethnicity: HISPANIC    gender: MALE    rank: 1
Process finished with exit code 0

```

Рисунок 9 – 11 – завдання 4. вивід інформації з нашого нового xml документа за допомогою domparser

Висновки.

У ході виконання лабораторної роботи було проведено аналіз XML-документу, який містить інформацію про популярні імена дітей у місті Нью-Йорк. Робота складалася з кількох завдань, що передбачали роботу з парсингом XML-документів за допомогою SAX та DOM методів, знаходження вибраних об'єктів та створення XSD-схеми. валідація структури документу.