

Харківський національний університет імені В. Н. Каразіна  
Факультет комп'ютерних наук  
Кафедра штучного інтелекту та програмного забезпечення

ЗВІТ  
З ЛАБОРАТОРНОЇ РОБОТИ №3  
«Java & XML»  
дисципліна: «Крос-платформне програмування»

Виконала: студентка групи КС21

Пушкіна Олеся

Перевірив: доцент кафедри ШПЗ

Споров Олександр Євгенович

Харків

2024

## Основні завдання

### Завдання

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:

- 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.
- За невеликим характерним фрагментом скласти *xsd* схему документу, створити валідатор та перевірити, чи правильно було зрозуміло структуру документу.
- Написати програмне рішення, що за допомогою *SAX* парсеру без валідації отримає назви всіх національних груп, що представлені в документі.
- 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.

При виконанні завдань потрібно уважно визначити структуру тегів документу — там є тег, що йде два рази поспіль.

Для виконання завдання було розроблено кілька Java програм, кожна з яких відповідає певній частині завдання.

Програма `DisplayXMLStructure.java` використовує SAX парсер для виведення на екран частини XML документу без валідації, що дозволяє вивчити його структуру та вміст. Вона виводить всі початкові та кінцеві елементи документу, а також їх атрибути.

Програма `ListXMLTags.java` також використовує SAX парсер, але її завданням є отримання переліку всіх тегів, імена яких присутні в документі. Вона виводить список всіх унікальних тегів.

Програма `ListEthnicGroups.java` використовує SAX парсер для отримання назв всіх етнічних груп, представлених у документі. Вона збирає та виводить список унікальних етнічних груп.

Програма `PopularNames.java` використовує DOM парсер для вибору 5 жіночих та 5 чоловічих найбільш популярних імен у заданій етнічній групі. Вона створює відповідні Java об'єкти для зберігання інформації про ім'я, гендер, кількість і рейтинг, сортує їх по збільшенню номера в рейтингу та зберігає вибрану та відсортовану інформацію у новий XML файл.

Програма ReadSortedXML.java також використовує DOM парсер для читання нового XML файлу та виведення інформації, що в ньому зберігається, на екран.

Програма XMLValidator.java створена для валідації XML документу проти XSD схеми. Вона перевіряє, чи відповідає структура XML документу XSD схемі, що було складено на основі характерного фрагменту.

## Результати виконання завдань:

### Завдання №1:

```
rt Element: row
Attribute: _id = row-8iz8_dxjs~ypp9
Attribute: _uuid = 00000000-0000-0000-4E7C-ADC5180AB204
Attribute: _position = 0
Attribute: _address = https://data.cityofnewyork.us/resource/\_25th-nuif/row-8iz8\_dxjs~ypp9
Start Element: brth_yr
Characters: 2018
End Element: brth_yr
Start Element: gndr
Characters: MALE
End Element: gndr
Start Element: ethcty
Characters: HISPANIC
End Element: ethcty
Start Element: nm
Characters: Emanuel
End Element: nm
Start Element: cnt
Characters: 17
End Element: cnt
Start Element: rnk
Characters: 82
End Element: rnk
End Element: row
Start Element: row
Attribute: _id = row-s5qu_3fvn~5buw
Attribute: _uuid = 00000000-0000-0000-61EF-3C993515C4E8
Attribute: _position = 0
Attribute: _address = https://data.cityofnewyork.us/resource/\_25th-nuif/row-s5qu\_3fvn~5buw
Start Element: brth_yr
Characters: 2018
End Element: brth_yr
Start Element: gndr
Characters: MALE
End Element: gndr
Start Element: ethcty
Characters: HISPANIC
End Element: ethcty
Start Element: nm
Characters: Isaias
End Element: nm
Start Element: cnt
Characters: 17
```

Рис. 1 — скріншот роботи консольної програми DisplayXMLStructure

```
C:\Users\pushk\.jdk\openjdk-21.0.2\bin\java.exe "-javaagent:C:\Program Files\Jet
Tags found in the document: [ethcty, gndr, response, cnt, brth_yr, row, rnk, nm]
|
Process finished with exit code 0
```

Рис. 2 — скріншот роботи програми ListXMLTags.

Перелік всіх тегів, імена яких присутні в документі

## Завдання №2

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:element name="response">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="row" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="row" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="brth_yr" type="xs:int"/>
                    <xs:element name="gndr" type="xs:string"/>
                    <xs:element name="ethcty" type="xs:string"/>
                    <xs:element name="nm" type="xs:string"/>
                    <xs:element name="cnt" type="xs:int"/>
                    <xs:element name="rnk" type="xs:int"/>
                  </xs:sequence>
                  <xs:attribute name="_id" type="xs:string" use="required"/>
                  <xs:attribute name="_uuid" type="xs:string" use="required"/>
                  <xs:attribute name="_position" type="xs:string" use="required"/>
                  <xs:attribute name="_address" type="xs:string" use="required"/>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Рис. 3 — xsd схема документу

```
C:\Users\pushk\.jdk\openjdk-21.0.2\bin\java.exe
Validation is successful

Process finished with exit code 0
```

Рис. 4 — скріншот повідомлення про успішну валідацію

### Завдання №3

```
Ethnic groups found in the document:
- WHITE NON HISPANIC
- BLACK NON HISP
- WHITE NON HISP
- HISPANIC
- BLACK NON HISPANIC
- ASIAN AND PACI
- ASIAN AND PACIFIC ISLANDER

Process finished with exit code 0
```

Рис. 5 — скріншот роботи програми ListEthnicGroups.

Назви всіх національних груп, що представлені в документі

### Завдання №4

```
Enter the ethnicity:
WHITE NON HISP
Popular Names for ethnicity: WHITE NON HISP
Name      Gender  Count  Rank
JOSEPH    MALE    300    1
EMMA      FEMALE  228    1
DAVID     MALE    289    2
LEAH      FEMALE  219    2
MICHAEL   MALE    245    3
SARAH     FEMALE  209    3
JACOB     MALE    242    4
OLIVIA    FEMALE  198    4
SOPHIA    FEMALE  198    4
MOSHE     MALE    238    5

Process finished with exit code 0
```

Рис. 6 — скріншот роботи програми PopularNames. (Топ 5 жіночих та чоловічих імен)

## Завдання №5

Popular Names from Sorted XML:

Name	Gender	Count	Rank
JOSEPH	MALE	300	1
EMMA	FEMALE	228	1
DAVID	MALE	289	2
LEAH	FEMALE	219	2
MICHAEL	MALE	245	3
SARAH	FEMALE	209	3
JACOB	MALE	242	4
OLIVIA	FEMALE	198	4
SOPHIA	FEMALE	198	4
MOSHE	MALE	238	5

Process finished with exit code 0

Рис. 7 — скріншот роботи програми ReadSortedXML