

Adatkezelés XML környezetbe – 10. Practice

Témakör: **xPath** program készítése, módosítása

JSON dokumentum készítése (read, write) Java és JavaScript segítségével

A mappa neve: **NEPTUNKOD_1119**

Töltse fel a GitHub rendszer mappába a forrás fájlokat!

Határidő: 2024.11.19.

A feladatokat fejlesztőkörnyezettel készítse el!!!

1. Task

Java nyelven készítse el a *xPathNEPTUNKOD* feladatot, amely egy *studentNeptunkod.xml* dokumentumot olvas be, majd a megadott lekérdezéseket kiírja a konzolra.

Projekt name: **xPathNEPTUNKOD**

Package: **xpathneptunkod1119**

XPath class name: **xPathNEPTUNKOD.java**

a) Készítse el az XML dokumentumot a következő séma alapján – neve: **studentNEPTUNKOD.xml**

Root element: **class**

Gyerekelemei: **student**

student id (keresztnev, vezeteknev, becenev, kor) – ebből legalább 3 példány legyen.

1b) Készítse el a *studentNeptunkod.xml* dokumentumhoz illeszkedő *studentNeptunkod.xsd* sémát.

1c) Java nyelven készítse el a *studentNeptunkod.xml* alapján a **xPath dokumentumot**, melynek neve: **xPathNEPTUNKOD.java**.

Az utasítás sorokat futtatás után tegye megjegyzésbe.

Végezze el a következő lekérdezéseket:

- 1) Válassza ki az összes *student* element, amely a *class* gyermekei!
- 2) Válassza ki azt a *student* elemet, amely rendelkezik "*id*" attribútummal és értéke "02"!
- 3) Kiválasztja az összes *student* elemet, függetlenül attól, hogy hol vannak a *dokumentumban*!
- 4) Válassza ki a *második student element*, amely a *class root element* gyermeke!
- 5) Válassza ki az *utolsó student* elemet, amely a *class root element* gyermeke!
- 6) Válassza ki a *utolsó előtti student* elemet, amely a *class root element* gyermeke!
- 7) Válassza ki az *első két student* elemet, amelyek a *root element* gyermekei!

- 8) Válassza ki class root element összes gyermek elemét!
- 9) Válassza ki az összes student elemet, amely rendelkezik legalább egy bármilyen attribútummal!
- 10) Válassza ki a dokumentum összes elemét!
- 11) Válassza ki a class root element összes student elemét, amelynél a kor > 20!
- 12) Válassza ki az összes student elem összes keresztnev or vezeteknev csomópontot!

2. task

Java nyelven készítse el a xPathNeptunkod (xpath kifejezéseket) feladatot, amely az *orarendNeptunkod.xml* dokumentumot felhasználva *legalább három módosítást* végezzen el, majd a módosítással együtt kiírja a konzolra és egy fájlba strukturált formában.

- 1 módosítás: szak nevének módosítása
- 2 módosítás: a tárgy nevéhez a monogram hozzáfűzése: pl. Adatkezelés XML környezetben (BL)
- 3 módosítás: id=3 példány - helyszín: XXXVII módosítása

Mentés: *orarendNeptunkod1.xml*

Projekt name: xPathNeptunkod

Package: xpathneptunkod1119

DOM class name: XPathModify

DOM file name: xPathNeptunkod.java

XML name: orarendNeptunkod1.xml

3. Task

Adott a következő JSON dokumentum: *orarendNeptunkod.json*!

Az *orarendNeptunkod.json* kódot írja meg JavaScript-be (kimeneti minta), melynek kiterjesztése: *JScriptNeptunkod.html*.

A futtatás eredménye: **Mező:** értékformában írja ki a kimenetre **minden objektumot**.

Például:

Neptunkod Órarend 2024

Tárgy: Adatkezelés XML-ben

Helyszín: Inf/103

Oktato: Dr. Bednarik László

Szak: MI, PTI

4. Task

Adott a következő JSON dokumentum: *orarendNeptunkod.json!*

a.) Készítsen hozzá JSON schemát a tanultak alapján.

Mentés: *orarendNeptunkodSchema.json!*

b.) Készítsen Java nyelven programot, amely ellenőrzi a sémát és kiírja – *Validation*.

Javaslom Maven projekt használatát:

```
pom.xml tartalma:  
  
<dependencies>  
  <dependency>  
    <groupId>org.everit.json</groupId>  
    <artifactId>org.everit.json.schema</artifactId>  
    <version>1.14.1</version>  
  </dependency>  
  <dependency>  
    <groupId>org.json</groupId>  
    <artifactId>json</artifactId>  
    <version>20230227</version>  
  </dependency>  
</dependencies>
```

Projekt nev: JSONParseNeptunkod

Class name: JSONValidationNeptunkod.java

JSON file name: orarendNeptunkod.json

JSON schema name: orarendJSONSchemaNeptunkod.json