

데이터베이스기초 실습 과제 (9 주차)

<주의사항>

- 각각의 문제 바로 아래에 답을 작성 후 제출해 주세요.
- 제출 기한: 2023. 04. 26 (수요일) ~ 05. 02 (화요일) 23:55 까지
 - 부정행위 적발 시, 원본(보여준 사람)과 복사본(베낀 사람) 모두 0 점 처리함
- SmartLEAD 에 아래의 파일을 제출해 주세요
 - 보고서 (PDF 파일로 변환 후 제출)
 - 보고서 파일명에 이름과 학번을 입력 해 주세요.
 - 보고서에, 과제에 사용한 코드 및 실행 결과를 캡처하고, pdf 파일로 변환하여 제출.
 - 위 사항 누락 시 감점

이름	김영식
학번	20175119
소속 학과/대학	빅데이터학과

문제 1.

위 xml 데이터를 보고, 당신은 도메인 통계를 위해, email 중 linkedin 으로 가입되어진 이메일만 출력하고자 합니다.

이에 맞는 xslt 문을 작성하세요.

< 출력 형태 >

bag0437@naver.combag0437@hallym.ac.kr...

.

.

**** 반드시 위 출력 형태 처럼 출력되도록 하세요. ****

답변 :

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform" xmlns:fo="http://www.w3.org/1999/XSL/Format">
  <xsl:template match="dataset">
    <xsl:apply-templates select="//email[contains(., 'linkedin.')]"/>
  </xsl:template>
</xsl:stylesheet>
```

pyon3@linkedin.comgallaway1w@linkedin.commpolk5e@linkedin.com

문제 2.

당신은 xml 파일에 존재하는 사람들의 성씨 통계를 위해, 사람들의 last name 요소만 출력하고자 합니다.

위 출력 문을 xquery로 작성하세요.

< 출력 형태 >

```
<people_list>
  <last_name>baek</last_name>
  <last_name>baek</last_name>
  <last_name>baek</last_name>
  .
  .
  .
```

```
</people_list>
```

**** 반드시 위 출력형태 처럼 출력되도록 하세요. ****

답변 :

```
xquery version "1.0";

declare namespace ext="http://www.altova.com/xslt-extensions";

<people_list>
{
  for $name in /dataset/record
  |   return ($name/last_name)
}
</people_list>
```

```

<people_list>
  <last_name>Larmet</last_name>
  <last_name>O'Kenny</last_name>
  <last_name>Lorenzetto</last_name>
  <last_name>Yon</last_name>
  <last_name>Brookwood</last_name>
  <last_name>Seebright</last_name>
  <last_name>Renn</last_name>
  <last_name>Korda</last_name>
  <last_name>Baudacci</last_name>
  <last_name>Riggott</last_name>
  <last_name>Callway</last_name>
  <last_name>Gittings</last_name>
  <last_name>Bromfield</last_name>
  <last_name>Burns</last_name>
  <last_name>Silcocks</last_name>
  <last_name>Zebedee</last_name>
  <last_name>Bollom</last_name>
  <last_name>Goodinson</last_name>
  <last_name>McLucky</last_name>
  <last_name>Veale</last_name>
  <last_name>Biasini</last_name>
  <last_name>Kleinplac</last_name>
  <last_name>Caulfield</last_name>
  <last_name>Danbye</last_name>
  <last_name>Senechault</last_name>
  <last_name>Seebright</last_name>
  <last_name>McHarry</last_name>
  <last_name>Lampens</last_name>
  <last_name>Kitt</last_name>
  <last_name>Doncom</last_name>
  <last_name>Georghiou</last_name>
  <last_name>Earley</last_name>
  <last_name>Sweepstone</last_name>
  <last_name>Robshaw</last_name>
  <last_name>Basini-Gazzi</last_name>
  <last_name>Harling</last_name>
  <last_name>MacMearty</last_name>
  <last_name>Shobbrook</last_name>
  <last_name>Philot</last_name>
  <last_name>Tonry</last_name>
  <last_name>Fernan</last_name>
  <last_name>Doblin</last_name>
  .....
  <last_name>Hurt</last_name>
  <last_name>Mathouse</last_name>
  <last_name>Francois</last_name>
  <last_name>Van den Broek</last_name>
  <last_name>Caffery</last_name>
  <last_name>Kendall</last_name>
  <last_name>Picott</last_name>
  <last_name>Tison</last_name>
  <last_name>Stroband</last_name>
  <last_name>Glenister</last_name>
  <last_name>Rehn</last_name>
  <last_name>Botten</last_name>
  <last_name>Leggett</last_name>
  <last_name>Willison</last_name>
  <last_name>Coldbathe</last_name>
  <last_name>Bertin</last_name>
  <last_name>Melley</last_name>
  <last_name>Cyseley</last_name>
  <last_name>O'Shavlan</last_name>
  <last_name>Ferenc</last_name>
  <last_name>Tatem</last_name>
  <last_name>Ray</last_name>
  <last_name>Ropert</last_name>
  <last_name>Gilderoy</last_name>
  <last_name>Baybutt</last_name>
  <last_name>Dilgarno</last_name>
  <last_name>Cruddace</last_name>
  <last_name>Verty</last_name>
  <last_name>Bedow</last_name>
  <last_name>Conisbee</last_name>
  <last_name>Demetr</last_name>
  <last_name>Gilderoy</last_name>
  <last_name>Haile</last_name>
  <last_name>Ashbee</last_name>
  <last_name>McMylor</last_name>
  <last_name>Pirot</last_name>
  <last_name>Vinau</last_name>
  <last_name>Blakeman</last_name>
  <last_name>Antonucci</last_name>
  <last_name>Trathen</last_name>
  <last_name>Bodycombe</last_name>
  <last_name>Pallent</last_name>
</people_list>

```

문제 3.

여러분은 위 xml에서, 비교적 최근에가입한 사람의 이름을 찾고자 합니다.

id 요소의 값이 990 초과이고, gender 요소가 Male 인 사람의 first_name을 출력하세요.

조건에 해당하는 값이 하나 이상인 경우, 모두 출력하세요.

<출력 예시>

```
<first_name>kyeong1</first_name>
```

```
<first_name>kyeong2</first_name>
```

```
<first_name>kyeong3</first_name>
```

```
.
```

```
.
```

```
.
```

**** 반드시 위 출력형태처럼 되도록 출력하세요 ****

답변 :

```
xquery version "1.0";
```

```
declare namespace ext="http://www.altova.com/xslt-extensions";
```

```
for $person in /dataset/record
  where $person/id > 990 and $person/gender = 'Male'
  return ($person/first_name, '&#xA;')
```

```
<first_name>Derward</first_name>
```

```
<first_name>Geordie</first_name>
```

```
<first_name>Orran</first_name>
```

```
<first_name>Steward</first_name>
```

```
<first_name>Rolf</first_name>
```

```
|
```