



Android Homework 4 (DOM Parsing)

경북대학교
소프트웨어융합과
배희호 교수



Report



■ XML 문서

```
<?xml version="1.0" encoding="UTF-8"?>
<persons>
  <person id="2101023">
    <name>한민국</name>
    <age>18</age>
  </person>
  <person id="2301345">
    <name>이대한</name>
    <age>23</age>
  </person>
  <person id="2301421">
    <name>카타르</name>
    <age>20</age>
  </person>
</persons>
```

Raw 폴더
person.xml



Report



```
<?xml version="1.0" encoding="utf-8"?>
```

```
<users>
```

```
<user>
```

```
<name>홍길동</name>
```

```
<designation>Team Leader</designation>
```

```
<location>
```

```
<city>영월</city>
```

```
<province>강원도</province>
```

```
</location>
```

```
</user>
```

```
<user>
```

```
<name>이순신</name>
```

```
<designation>Agricultural Officer</designation>
```

```
<location>
```

```
<city>아산</city>
```

```
<province>충청남도</province>
```

```
</location>
```

```
</user>
```

```
<user>
```

```
<name>박명동</name>
```

```
<designation>Chartered Accountant</designation>
```

```
<location>
```

```
<city>서울</city>
```

```
<province>서울특별시</province>
```

```
</location>
```

```
</user>
```

```
</users>
```

userdetails.xml

assets 폴더에 저장



Report



■ XML 문서 (Local Web Server company.xml)

```
?xml version="1.0"?>
<login>
  <status>OK</status>
  <jobs>
    <job>
      <id>4</id>
      <companyid>4</companyid>
      <company>Android Example</company>
      <address>Parse XML Android</address>
      <city>Tokio</city>
      <state>Xml Parsing Tutorial</state>
      <zipcode>601301</zipcode>
      <country>Japan</country>
      <telephone>928-789-3558</telephone>
      <fax>123-456-7890</fax>
      <date>2012-03-15 12:00:00</date>
    </job>
  </jobs>
</login>
```



Report



■ XML 문서 (Local Web Server company.xml)

```
<job>
  <id>5</id>
  <companyid>6</companyid>
  <company>Xml Parsing In Java</company>
  <address>B-22</address>
  <city>Cantabill</city>
  <state>XML Parsing Basics</state>
  <zipcode>201301</zipcode>
  <country>America</country>
  <telephone>396-878-9355</telephone>
  <fax>673-436-9294</fax>
  <date>2012-05-18 13:00:00</date>
</job>
</jobs>
</login>
```



Report



■ XML 문서 (Local Web Server (laptops.xml))

```
<?xml version="1.0" encoding="UTF-8"?>
<laptops>
  <notebook model = "ThinkPad X1 Carbon 8세대">
    <brand>Lenovo</brand>
    <price>₩2,549,000</price>
    <description>최초의 인텔®모바일 인증 노트북 시리즈 중 하나로 35.56cm (14형)
X1 Carbon 8세대는 오래 지속되는 배터리 수명, 업계 최고의 연결성, 고속 충전 기술까지
지원하는 우수한 모바일 성능을 자랑합니다. </description>
    <technical-details>
      Intel 2nd gen Core i3-2367 1.40 GHz (6MB Cache), 6 GB DIMM, 500GB
      5400 rpm SATA Hard Drive, 13-Inch Screen, Windows 10 Home Premium 64-bit
    </technical-details>
    <image>https://www.computerspace.co.kr/data/pimg/174/174543.jpg</image>
  </notebook>
```



Futuristic Innovator

京福大學校
KYUNGBOK UNIVERSITY



Report



■ XML 문서 (Local Web Server (laptops.xml))

```
<notebook model = "ASUS 크리에이터X – X571">
  <brand>ASUS</brand>
  <price>₩1,199,000</price>
  <description>Strikingly thin with more room to view.</description>
  <technical-details>
    Intel Core i7-2637M (1.70GHz, 4MB Cache), 4 GB DIMM₩n256GB Solid
    State Drive, 13-Inch Screen, Windows 7 Home Premium 64-bit</technical-details>
  <image>https://gdimg.gmarket.co.kr/3954199633/still/280?ver=1719977788</image>
</notebook>
<notebook model = "ChromeBook – C214MA">
  <brand>SAMSUNG</brand>
  <price>₩499,000</price>
  <description>Strikingly thin, with more room to view. </description>
  <technical-details>
    Intel® Celeron® N4000 (1.70GHz, 4MB Cache), 4 GB DIMM, 64G eMMC,
    11.6형 HD 1366×768 안티글레어 터치스크린, Chrome </technical-details>
  <image>https://www.lge.co.kr/kr/images/notebook/md10368828/md10368828-
  280x280.jpg</image>
</notebook>
</laptops>
```



Report



```
<?xml version="1.0" encoding="utf-8"?>
```

university.xml

```
<students>
```

```
  <student>
```

```
    <name>홍길동</name>
```

```
    <item studentid="20181579">컴퓨터공학과</item>
```

```
  </student>
```

```
  <student>
```

```
    <name>김현아</name>
```

```
    <item studentid="20181573">간호학과</item>
```

```
  </student>
```

```
  <student>
```

```
    <name>제갈민제</name>
```

```
    <item studentid="20181611">건축공학과</item>
```

```
  </student>
```

```
  <student>
```

```
    <name>배동혁</name>
```

```
    <item studentid="20171512">컴퓨터공학과</item>
```

```
  </student>
```

```
</students>
```

Local Web Server에 저장



Report



■ 화면 구성

■ 학번 짝수

■ TabLayout + ViewPager

■ 학번 홀수

■ TabLayout + ViewPager2

■ 출력 방법

■ 학번 짝수

■ Source 보기는 TextView를 Fragment로 구현

■ DOM Parsing 결과는 ListView를 Fragment로 구현

■ 학번 홀수

■ Source 보기는 TextView를 Fragment로 구현

■ DOM Parsing 결과는 ListView를 Fragment로 구현



Report

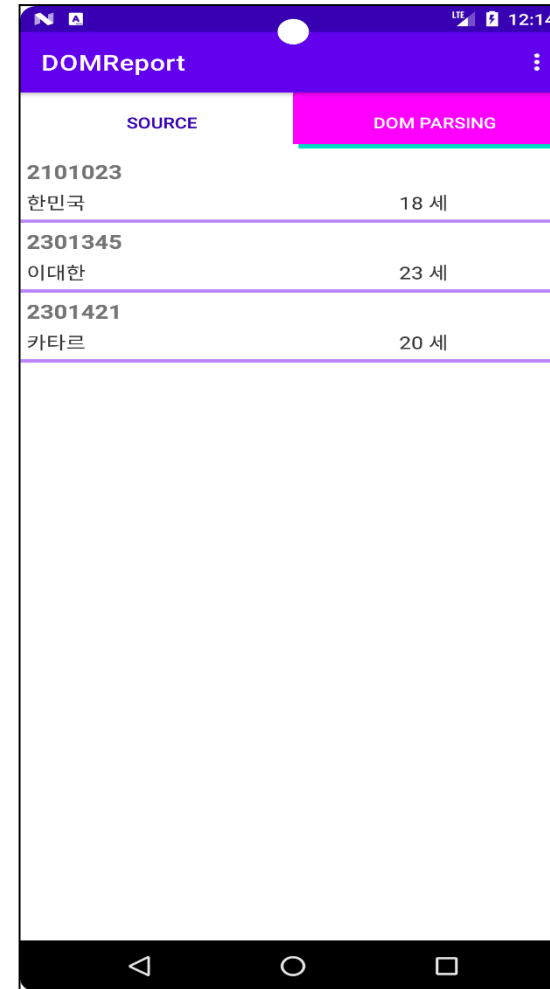


- Internet을 통해 XML File Download 받는 방법
 - company.XML 파일 : Thread로 구현
 - laptop.XML 파일 : AsyncTask로 구현
 - laptop,XML 파일 내의 image Tag는 Glide Library 사용
 - university.xml 파일 : Runnable로 구현
- Parsing 방법은 Down 받은 후 Parsing하는 방법을 선택할 것



Report

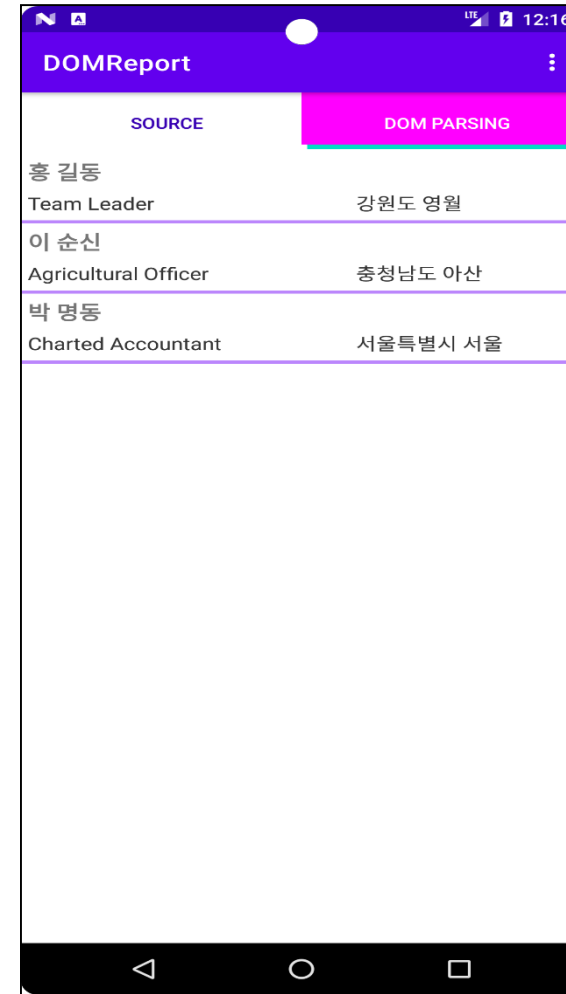
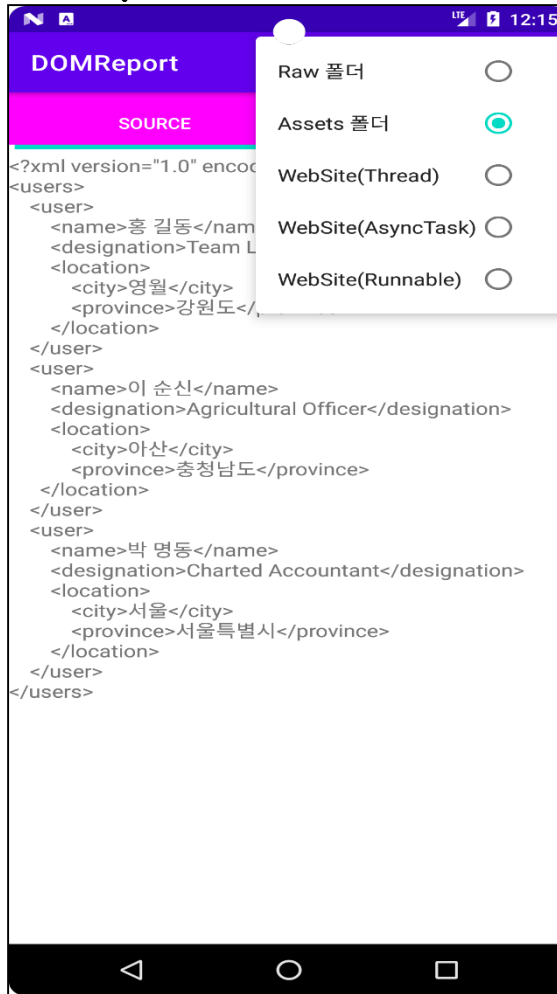
■ 실행 결과(person.xml)





Report

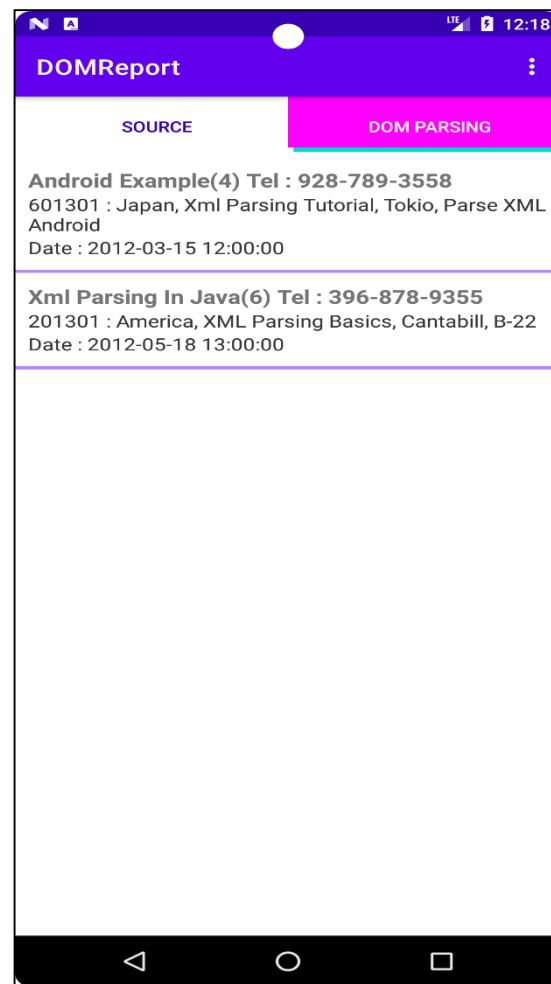
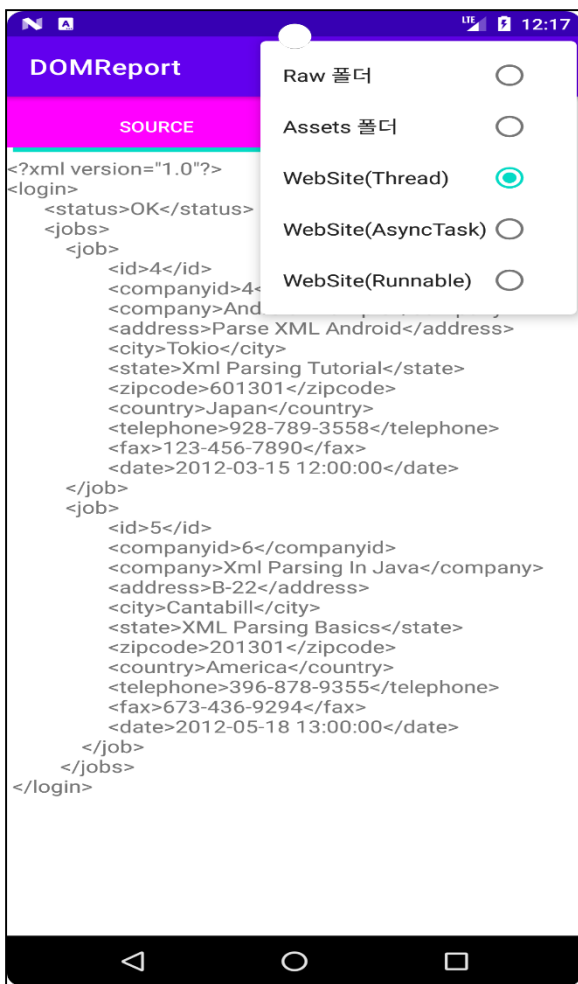
■ 실행 결과(userdetails.xml)





Report

■ 실행 결과 (company.xml)



Futuristic Innovator

京福大學校
KYUNGBOK UNIVERSITY



Report

■ 실행 결과 (laptos.xml)

DOMReport

SOURCE


Raw 폴더 ☐
Assets 폴더 ☐
WebSite(Thread) ☐
WebSite(AsyncTask) ☒
WebSite(Runnable) ☐


```
<?xml version="1.0" encoding="UTF-8" ?>
<laptops>
  <notebook model = "ThinkPad X1 Carbon 8세대"
    <brand>Lenovo</brand>
    <price>₩2,549,000</price>
    <description>최초의
하나로 35.56cm (14형) X1
터리 수명, 업계 최고의 연결
우수한 모바일 성능을 자랑합니다. </description>
    <technical-details>
      Intel 2nd gen Core i3-2367 1.40 GHz (6MB Cache),
      6 GB DIMM, 500GB 5400 rpm SATA Hard Drive, 13-Inch
      Screen, Windows 10 Home Premium 64-bit </technical-
      details>
    <image>https://www.computerspace.co.kr/data
    /pimg/174/174543.jpg</image>
  </notebook>
  <notebook model = "ASUS 크리에이터X - X571">
    <brand>ASUS</brand>
    <price>₩1,199,000</price>
    <description>Strikingly thin with more room to view.</
    description>
    <technical-details>
      Intel Core i7-2637M (1.70GHz, 4MB Cache), 4 GB
      DIMM\n256GB Solid State Drive, 13-Inch Screen, Windows
      7 Home Premium 64-bit</technical-details>
    <image>https://gdimg.gmarket.co.kr/3954199633
    /still/280?ver=1719977788</image>
  </notebook>
  <notebook model = "ChromeBook - C214MA">
    <brand>SAMSUNG</brand>
    <price>₩499,000</price>
    <description>Strikingly thin, with more room to view.
    </description>
    <technical-details>
      Intel® Celeron® N4000 (1.70GHz, 4MB Cache), 4
      GB DIMM, 64G eMMC, 11.6형 HD 1366×768 안티글레어 터
      치스크린, Chrome </technical-details>
    <image>https://www.lge.co.kr/kr/images/notebook
```

DOMReport

SOURCE **DOM PARSING**

 **Lenovo**
ThinkPad X1 Carbon 8세대
₩2,549,000

 **ASUS**
ASUS 크리에이터X - X571
₩1,199,000

 **SAMSUNG**
ChromeBook - C214MA
₩499,000

DOMReport

ThinkPad X1 Carbon 8세대

Lenovo
게이밍 키보드+마우스+장패드 증정

 ₩2,549,000

Intel 2nd gen Core i3-2367 1.40 GHz (6MB Cache)
6 GB DIMM
500GB 5400 rpm SATA Hard Drive
13-Inch Screen
Windows 10 Home Premium 64-bit
최초의 인텔®모바일 인증 노트북 시리즈 중 하나
로 35.56cm (14형) X1 Carbon 8세대는 오래 지
속되는 배터리 수명, 업계 최고의 연결성, 고속 충
전 기술까지 지원하는 우수한 모바일 성능을 자랑
합니다.

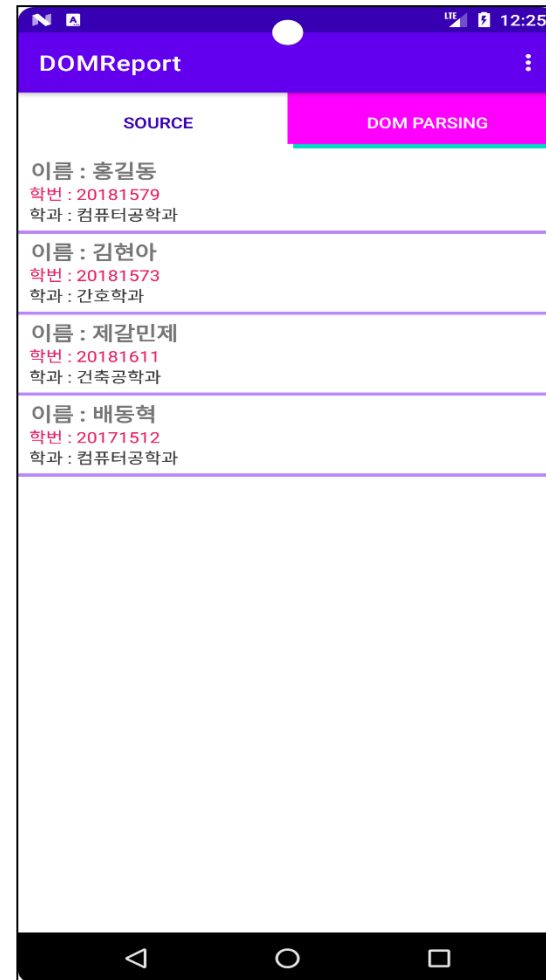
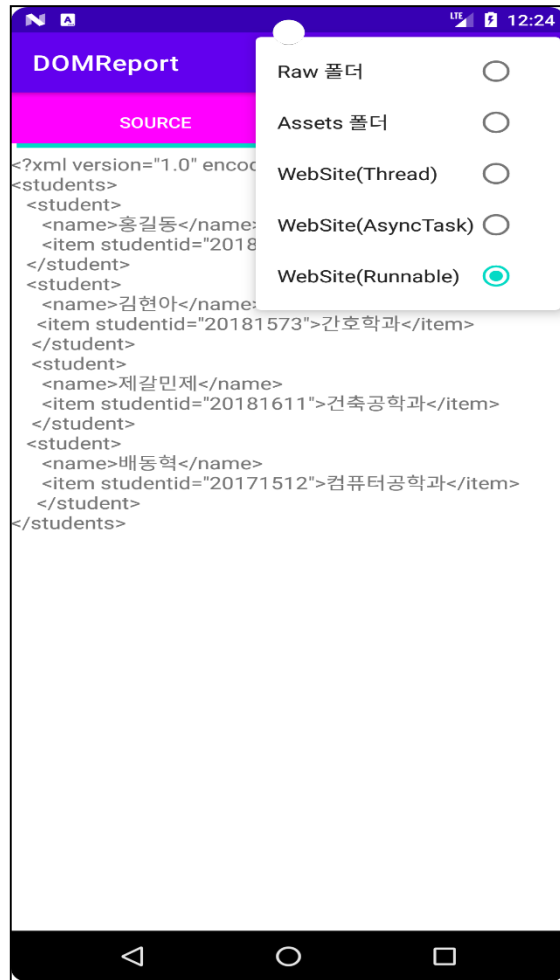
종료





Report

■ 실행 결과 (university.xml)



Futuristic Innovator

京福大學校
KYUNGBOK UNIVERSITY



Report



- XML 문서를 다운 받은 후 Parsing 과정
 - HTTP 요청을 통해 문서를 Download하고, 이를 Memory에 저장하거나 File로 저장
 - 저장된 Data를 Load한 뒤 Parsing을 실행
 - 단점
 - Memory/Storage 사용
 - 큰 문서를 처리할 경우 저장 공간과 Memory가 더 많이 필요
 - 추가 I/O 비용
 - File 쓰기 및 읽기 작업이 필요
 - 실시간 처리 어려움
 - Data를 Download하는 데 시간이 걸리므로 실시간 Data Parsing에는 적합하지 않음



Report



- XML 문서를 다운 받은 후 Parsing 과정
 - 장점
 - 재사용 가능성
 - Data를 저장해 두면 나중에 다시 열람하거나 Parsing을 반복 실행 가능
 - Debugging 용이성
 - 저장된 Data를 활용해 Parsing을 조정하거나 Test 할 수 있음
 - 성능 최적화 가능성
 - Data를 미리 Load한 후 Parsing하면 Network Delay와 무관하게 Local에서 빠르게 처리 가능
 - 단순화
 - Data와 Parsing 과정이 분리되므로 Code 설계가 단순화될 수 있음



Report



- Stream 자체를 Parsing하는 방법
 - Data가 Network에서 Stream 형태로 전송되는 동안 직접 Parsing하는 방법
 - Data를 한 번에 처리하지 않고, Streaming 방식으로 처리
 - 장점
 - 실시간 처리
 - Data가 수신되는 즉시 처리 가능
 - Memory 절약
 - 전체 Data를 Memory에 Load하지 않고 Stream을 통해 부분적으로 처리
 - 빠른 응답성
 - 대규모 Data를 Download할 필요 없이 처리 과정을 시작 가능
 - 저장 공간 불필요
 - Data를 Disk에 저장하지 않아도 됨





Report



- Stream 자체를 Parsing하는 방법

- 단점

- 복잡한 구현

- Stream Parsing은 상태를 유지하며 Data를 처리해야 하므로 구현 난이도가 높을 수 있음

- 재사용 어려움

- Data가 Stream으로 한 번만 처리되므로, 같은 Data를 반복적으로 Parsing하려면 다시 Streaming해야 함

- Error 처리 어려움

- Streaming 도중 Error가 발생하면 전체 Data를 다시 수신해야 하는 경우가 있을 수 있음



Report



- 사용 사례에 따른 선택 기준
 - 문서를 Download한 후 Parsing이 적합한 경우
 - Data 크기가 작거나, 한 번만 Download 후 여러 번 처리해야 하는 경우
 - Network 연결이 불안정한 환경에서 안정적으로 Data를 저장하고 싶을 때
 - Debugging 및 Parsing Logic을 Test해야 하는 경우
 - Stream Parsing이 적합한 경우
 - 실시간 Data 처리(예: Log Monitoring, Web Socket Data) 시
 - 매우 큰 Data를 처리할 때 Memory 사용량을 최소화하고 싶은 경우
 - Network 연결이 안정적이고, Data가 연속적으로 제공되는 경우



Report 제출 방법(PPT 작성)

- 각종 XML File과 JAVA File의 Source를 Text 형태로 PPT에 복사하여 제출 (수업자료 처럼 할 것)
 - 실행 화면은 반드시 capture해서 추가할 것
 - Program Source는 반드시 Text로 할 것 (capture한 Image File은 안됨)
 - 마지막 장에는 Report를 하면서 느낀점이나 질문 등을 반드시 작성할 것
- Report File 이름 작성 방법
 - 본인의 이름과 학번, 주차를 이용하여 작성
 - 파일명: 4주-홍길동-1401234.ppt
- 자신의 실제 Device(스마트 폰)에 넣어가지고 학교에 와서 동작 상황을 언제든지 시연할 수 있어야 함