



# Android Homework 4 (DOM Parsing)

---

경북대학교  
소프트웨어융합과  
배희호 교수



# Report



## ■ XML 문서

내부 문자열

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

```
<students>
```

```
<man part="컴퓨터정보과" bunho = "144456" birth="2004">이대한</man>
```

```
<man part="행정학과" bunho = "321467" birth="2009">한민국</man>
```

```
<man part="간호학과" bunho = "412765" birth="2006">이유림</man>
```

```
<man part="유아교육과" bunho = "543899" birth="2008">나소미</man>
```

```
<man part="건축과" bunho = "178219" birth="1999">정근식</man>
```

```
</students>
```



Futuristic Innovator

京福大學校  
KYUNGBOK UNIVERSITY



# Report



## ■ raw folder에 저장

```
<?xml version="1.0"?>
```

```
<members>
```

```
<member>
```

```
<name>홍길동</name>
```

```
<birth>1972</birth>
```

```
</member>
```

```
<member>
```

```
<name>이대한</name>
```

```
<birth>2003</birth>
```

```
</member>
```

```
<member>
```

```
<name>한민국</name>
```

```
<birth>1997</birth>
```

```
</member>
```

```
<member>
```

```
<name>이미나</name>
```

```
<birth>2002</birth>
```

```
</member>
```

```
</members>
```

member1.xml



# Report



## ■ assets 폴더에 저장

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

```
<users>
```

```
<user>
```

```
<name>홍길동</name>
```

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

```
<location>
```

```
<city>홍천</city>
```

```
<province>강원도</province>
```

```
</location>
```

```
</user>
```

```
<user>
```

```
<name>윤석열</name>
```

```
<designation>경찰총장</designation>
```

```
<location>
```

```
<city>용산구</city>
```

```
<province>서울특별시</province>
```

```
</location>
```

```
</user>
```

users.xml



# Report



## ■ assets 폴더에 저장

users.xml

```
<user>
  <name>이강인</name>
  <designation>후보선수</designation>
  <location>
    <city>발렌시아</city>
    <province>스페인</province>
  </location>
</user>
```

```
<user>
  <name>이대한</name>
  <designation>나의조국</designation>
  <location>
    <city>해운대구</city>
    <province>부산광역시</province>
  </location>
</user>
</users>
```



# Report



## Local Web Server에 저장

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

```
<students>
```

```
<student hakbun = "2201234" grade = "2학년">
```

```
<name> 홍길동 </name>
```

```
</student>
```

```
<student hakbun = "2101342" grade = "3학년">
```

```
<name> 이대한 </name>
```

```
</student>
```

```
<student hakbun = "2301567" grade = "1학년">
```

```
<name> 한민국 </name>
```

```
</student>
```

```
<student hakbun = "21014583">
```

```
<name> 전대진 </name>
```

```
</student>
```

```
</students>
```

student3.xml



Futuristic Innovator

京福大學校  
KYUNGBOK UNIVERSITY



# DOM Parsing



## ■ [assets] 폴더에 저장

student.xml

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

```
<kbu>
```

```
<student>
```

```
<name num = "1701234">홍길동</name>
```

```
<birth>2002</birth>
```

```
<department>스마트IT과</department>
```

```
</student>
```

```
<student>
```

```
<name num = "1704134">이순신</name>
```

```
<birth>2001</birth>
```

```
<department>간호학과</department>
```

```
</student>
```

```
<student>
```

```
<name num = "1501134">강감찬</name>
```

```
<birth>1998</birth>
```

```
<department>컴퓨터정보과</department>
```

```
</student>
```

```
</kbu>
```



# Report

## Local Web Server에 저장

sawon.xml

파일 편집 보기

sawon.xml

```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
  <Employee>
    <id>101</id>
    <name>홍길동</name>
    <gender>남</gender>
    <salary>450000</salary>
    <image>https://search.pstatic.net/common?type=a&size=120x150&quality=95&direct=true&src=http%3A%2F%2Fsstatic.naver.net%2Fpeople%2F148%2F201601141124027901.jpg</image>
  </Employee>
  <Employee>
    <id>102</id>
    <name>이대환</name>
    <gender>남</gender>
    <salary>560000</salary>
    <image>https://search.pstatic.net/common?type=a&size=120x150&quality=95&direct=true&src=http%3A%2F%2Fsstatic.naver.net%2Fpeople%2Fportrait%2F201705%2F2017051003420102-7278827.jpg</image>
  </Employee>
  <Employee>
    <id>103</id>
    <name>나소미</name>
    <gender>여</gender>
    <salary>350000</salary>
    <image>https://search.pstatic.net/common?type=a&size=120x150&quality=95&direct=true&src=http%3A%2F%2Fsstatic.naver.net%2Fpeople%2Fportrait%2F202007%2F20200722102146456.jpg</image>
  </Employee>
  <Employee>
    <id>104</id>
    <name>정대준</name>
    <gender>남</gender>
    <salary>250000</salary>
    <image>https://search.pstatic.net/common?type=a&size=120x150&quality=95&direct=true&src=http%3A%2F%2Fsstatic.naver.net%2Fpeople%2Fportrait%2F202002%2F20200218142933542.png</image>
  </Employee>
  <Employee>
    <id>105</id>
    <name>고주봉</name>
    <gender>남</gender>
  </Employee>
</root>
```

줄 18, 열 17 | 2,898자

100% Windows (CRLF) UTF-8







# Report



## ■ Local Web Server에 저장

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

university1.xml

```
<students>
```

```
  <student studentid="20181579" name="홍길동">컴퓨터공학과</student>
```

```
  <student studentid="20181537" name="김현아">간호학과</student>
```

```
  <student studentid="20181611" name="제갈민재">건축공학과</student>
```

```
  <student studentid="20171512" name="배동혁">컴퓨터공학과</student>
```

```
</students>
```



Futuristic Innovator

京福大學校  
KYUNGBOK UNIVERSITY



# 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



Futuristic Innovator

京福大學校  
KYUNGBOK UNIVERSITY



# Report

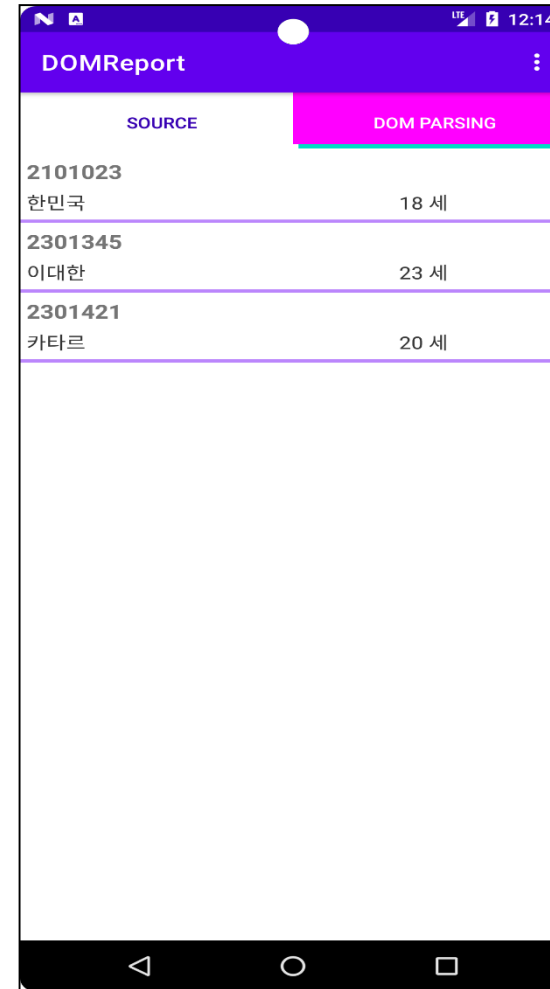


- Internet을 통해 XML File Download 받는 방법
  - 사원명부.XML 파일 : Thread로 구현
  - laptop.XML 파일 : AsyncTask로 구현
    - laptop,XML 파일 내의 image Tag는 Glide Library 사용
  - university.xml 파일 : Runnable로 구현
- Parsing 방법은 Down 받은 후 Parsing하는 방법을 선택할 것



# Report

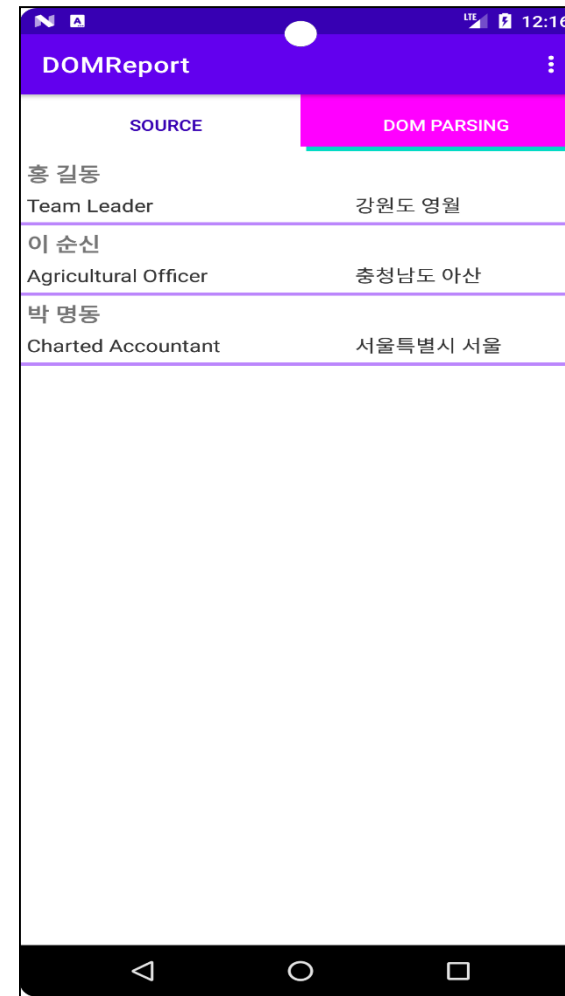
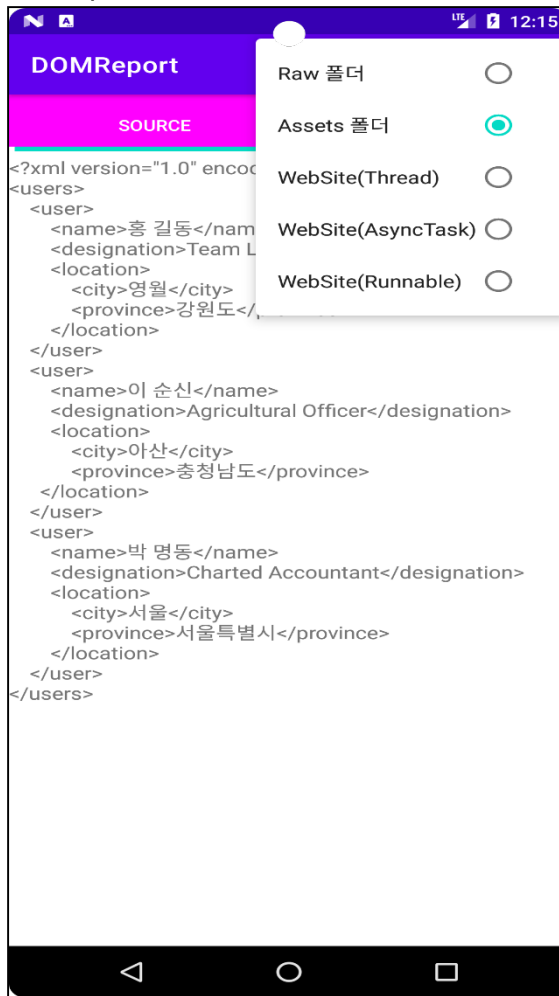
## ■ 실행 결과





# Report

## ■ 실행 결과



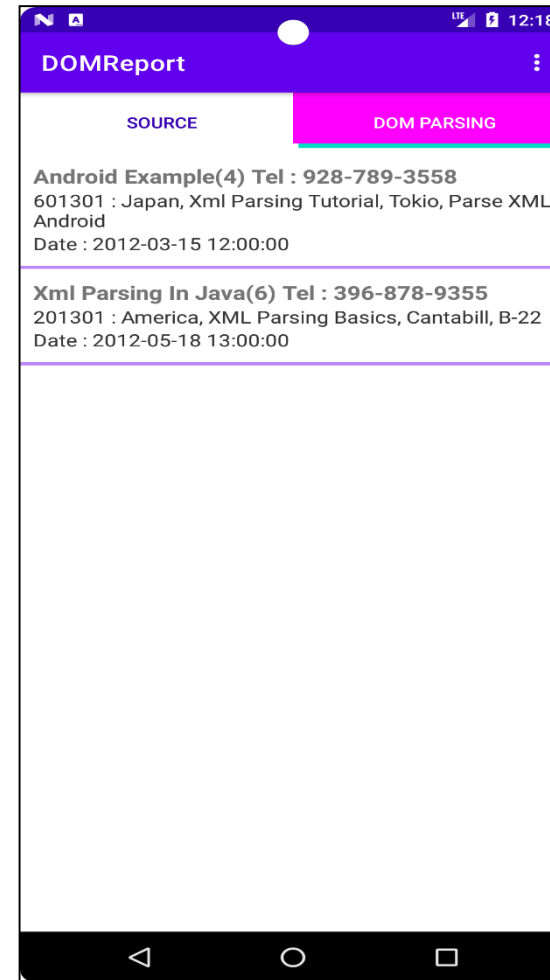
Futuristic Innovator

京福大學校  
KYUNGBOK UNIVERSITY



# Report

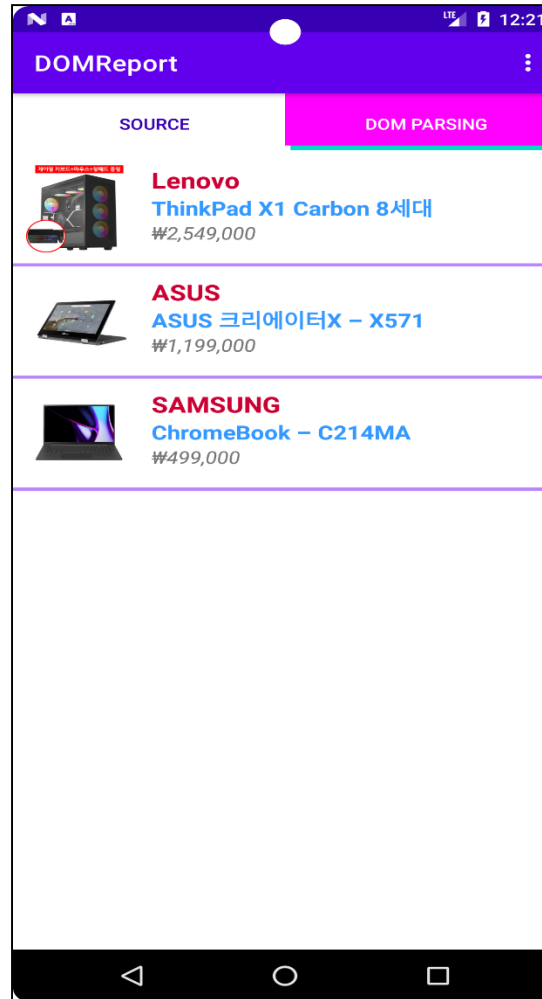
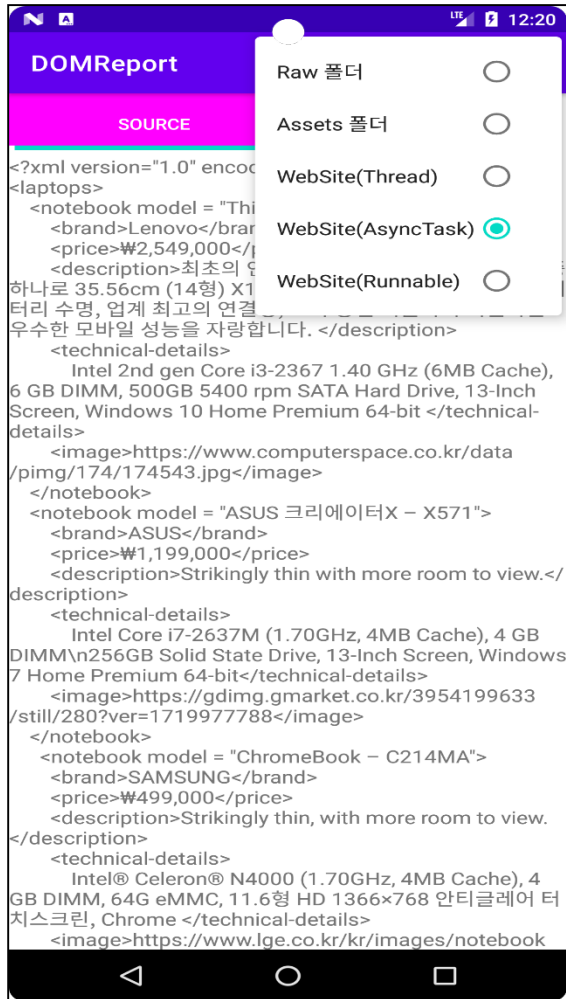
## ■ 실행 결과





# Report

## ■ 실행 결과



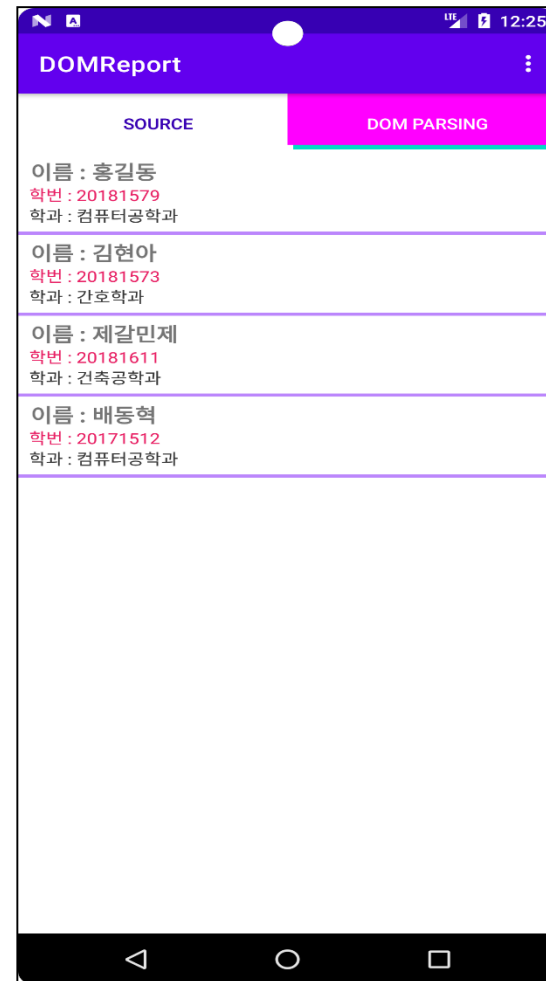
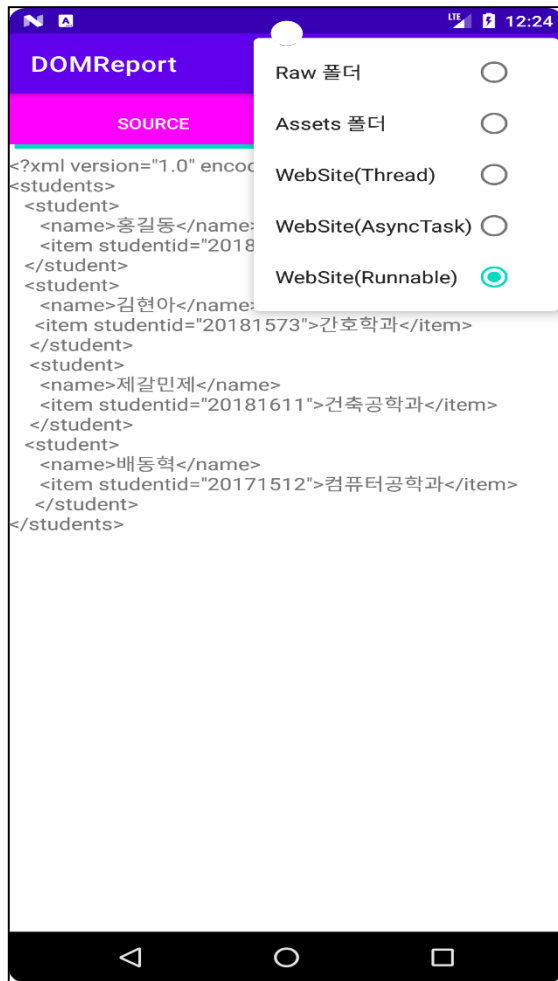
Futuristic Innovator

京福大學校  
KYUNGBOK UNIVERSITY



# Report

## ■ 실행 결과







# 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