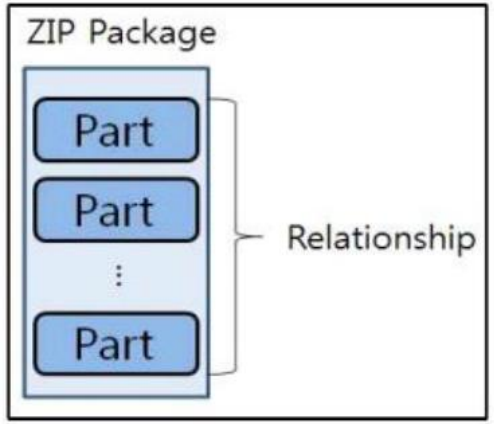
FaS
OOXML 분석 발표

정보보안암호수학과 20182209 김수빈

OOXML이란?

OOXML이란?

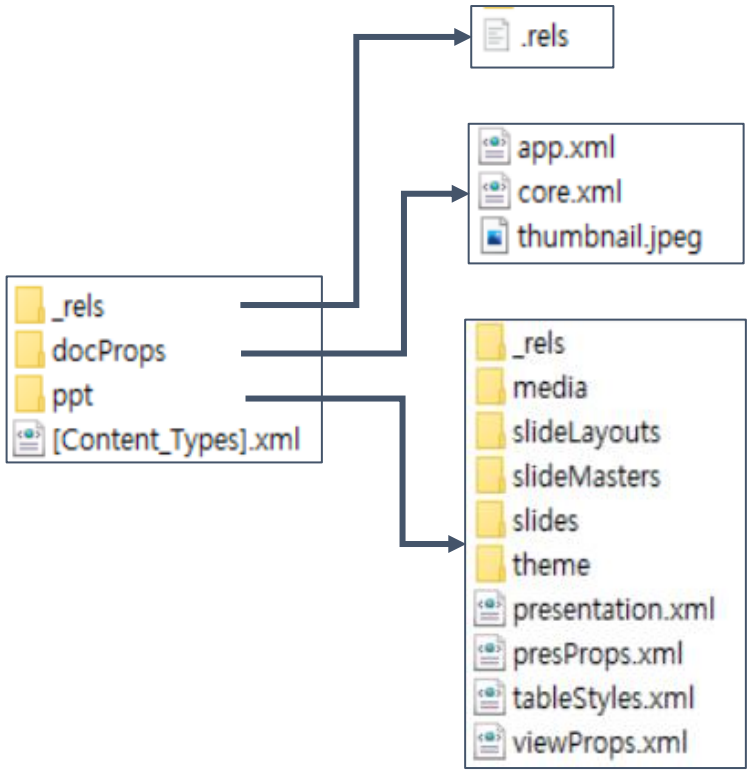
	Previous MS Office 2007	MS Office 2007~2016
File format	Compound File Binary(CFBF)	Office Open XML(OOXML)
Document extension	xls, ppt, doc	xlsx, pptx, docx



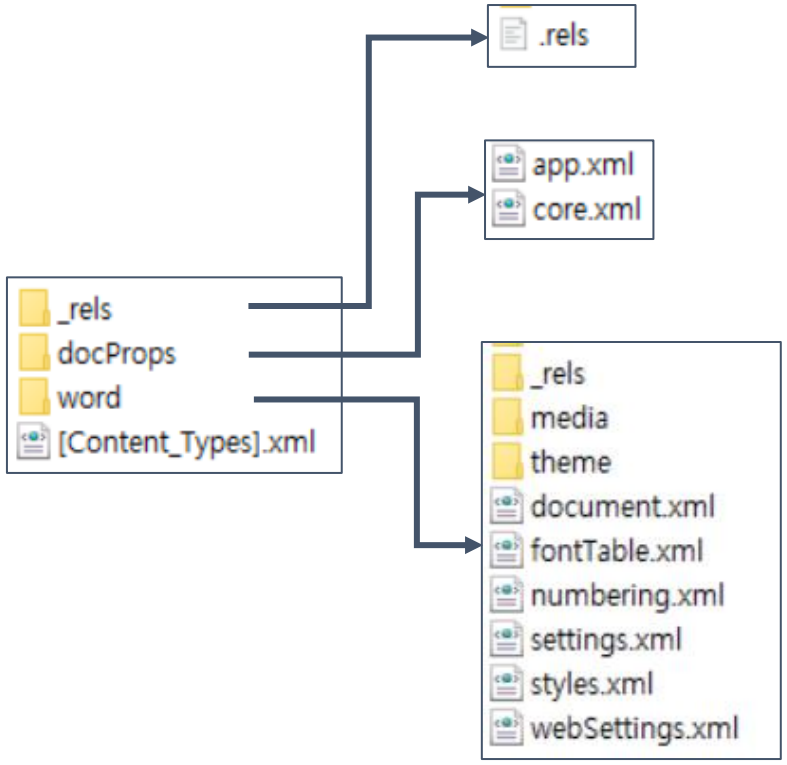
- OOXML(Office Open XML)은 마이크로소프트가 개발하였으며, 국제 표준으로 등록됨
- MS 오피스 2007 이전 버전에서는 CFBF를 사용했지만 문서의 가용성 등 여러 개선 사항을 이유로 OOXML을 사용함
- 표준에는 워드 프로 세서, 파워포인트, 엑셀 문서 형식이 정의되어 있고, 확장자는 기존의 확장자에서 “x”의 postfix가 추가된 형태로 pptx, docx, xlsx로 통일
- OOXML 형식의 문서들은 공통적으로 논리적 개체인 패키지(package)이며 ZIP 아카이브 형식을 가짐
- 패키지는 여러 종류의 파트(part)와 파트 간의 관계를 정의하는 관계 파트(relationship part)로 구성되고 이와 같은 요소들을 통합해 하나의 오브젝트로 표현

OOXML 구조

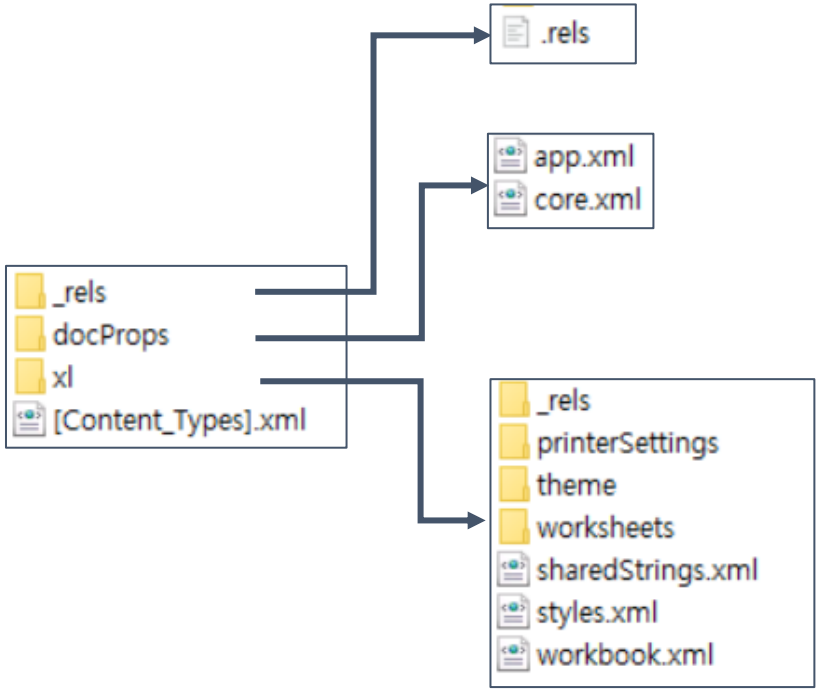
pptx



docx



xlsx



OOXML의 확장자를 ".zip"으로 바꾸어 압축을 해제하면 확인 가능

OOXML 구조 | 공통부분

[content_Types].xml

- 이 파일은 패키지의 모든 part와 타입의 목록이 저장되어 있다.

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Types  xmlns="http://schemas.openxmlformats.org/package/2006/content-types">
    <Default  Extension="jpeg" ContentType="image/jpeg"/>
    <Default  Extension="png" ContentType="image/png"/>
    <Default  Extension="rels" ContentType="application/vnd.openxmlformats-package.relationships+xml"/>
    <Default  Extension="xml" ContentType="application/xml"/>
    <Override  PartName="/ppt/presentation.xml" ContentType="application/vnd.openxmlformats-officedocument.presentationml.presentation.main+xml"/>
    <Override  PartName="/ppt/slideMasters/slideMaster1.xml" ContentType="application/vnd.openxmlformats-officedocument.presentationml.slideMaster+xml"/>
    <Override  PartName="/ppt/slides/slide1.xml" ContentType="application/vnd.openxmlformats-officedocument.presentationml.slide+xml"/>
    <Override  PartName="/ppt/presProps.xml" ContentType="application/vnd.openxmlformats-officedocument.presentationml.presProps+xml"/>
    <Override  PartName="/ppt/viewProps.xml" ContentType="application/vnd.openxmlformats-officedocument.presentationml.viewProps+xml"/>
    <Override  PartName="/ppt/theme/theme1.xml" ContentType="application/vnd.openxmlformats-officedocument.theme+xml"/>
    <Override  PartName="/ppt/tableStyles.xml" ContentType="application/vnd.openxmlformats-officedocument.presentationml.tableStyles+xml"/>
    <Override  PartName="/ppt/slideLayouts/slideLayout1.xml" ContentType="application/vnd.openxmlformats-officedocument.presentationml.slideLayout+xml"/>
    <Override  PartName="/ppt/slideLayouts/slideLayout2.xml" ContentType="application/vnd.openxmlformats-officedocument.presentationml.slideLayout+xml"/>
    <Override  PartName="/ppt/slideLayouts/slideLayout3.xml" ContentType="application/vnd.openxmlformats-officedocument.presentationml.slideLayout+xml"/>
    <Override  PartName="/ppt/slideLayouts/slideLayout4.xml" ContentType="application/vnd.openxmlformats-officedocument.presentationml.slideLayout+xml"/>
    <Override  PartName="/ppt/slideLayouts/slideLayout5.xml" ContentType="application/vnd.openxmlformats-officedocument.presentationml.slideLayout+xml"/>
    <Override  PartName="/ppt/slideLayouts/slideLayout6.xml" ContentType="application/vnd.openxmlformats-officedocument.presentationml.slideLayout+xml"/>
    <Override  PartName="/ppt/slideLayouts/slideLayout7.xml" ContentType="application/vnd.openxmlformats-officedocument.presentationml.slideLayout+xml"/>
    <Override  PartName="/ppt/slideLayouts/slideLayout8.xml" ContentType="application/vnd.openxmlformats-officedocument.presentationml.slideLayout+xml"/>
    <Override  PartName="/ppt/slideLayouts/slideLayout9.xml" ContentType="application/vnd.openxmlformats-officedocument.presentationml.slideLayout+xml"/>
    <Override  PartName="/ppt/slideLayouts/slideLayout10.xml" ContentType="application/vnd.openxmlformats-officedocument.presentationml.slideLayout+xml"/>
    <Override  PartName="/ppt/slideLayouts/slideLayout11.xml" ContentType="application/vnd.openxmlformats-officedocument.presentationml.slideLayout+xml"/>
    <Override  PartName="/docProps/core.xml" ContentType="application/vnd.openxmlformats-package.core-properties+xml"/>
    <Override  PartName="/docProps/app.xml" ContentType="application/vnd.openxmlformats-officedocument.extended-properties+xml"/>
</Types>
```

OOXML 구조 | 공통부분

_rels/.rels

- 다른 파트들과 package 밖의 리소스들 사이에 관계를 정의하는 relationships 파트를 포함한다.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Relationships xmlns="http://schemas.openxmlformats.org/package/2006/relationships">
    <Relationship Id="rId3" Type="http://schemas.openxmlformats.org/package/2006/relationships/metadata/core-properties" Target="docProps/core.xml"/>
    <Relationship Id="rId2" Type="http://schemas.openxmlformats.org/package/2006/relationships/metadata/thumbnail" Target="docProps/thumbnail.jpeg"/>
    <Relationship Id="rId1" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/officeDocument" Target="ppt/presentation.xml"/>
    <Relationship Id="rId4" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/extended-properties" Target="docProps/app.xml"/>
</Relationships>
```

/*.xml.rels

- 특정 파일에 대한 관계를 찾으려면 파일에 연관되는 _rels 디렉터리 안에 <파일 이름+.rels> 파일을 찾는다.
- *.xml.rels에는 참조관계가 표현되어 있으며, “Relationship” 속성으로 이를 나타낸다.

OOXML 구조 | 공통부분

docProps/app.xml

app.xml파일은 OOXML 문서에 관련된 특성을 포함한다.
관련된 특성이란 사용된 템플릿, 단어나 페이지 수, 어플리케이션 이름이나 버전 등을 포함한다.

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Properties  xmlns="http://schemas.openxmlformats.org/officeDocument/2006/extended-properties"
    <TotalTime>2</TotalTime>
    <Words>8</Words>
    <Application>Microsoft Office PowerPoint</Application>
    <PresentationFormat>와이드스크린</PresentationFormat>
    <Paragraphs>4</Paragraphs>
    <Slides>1</Slides>
    <Notes>0</Notes>
    <HiddenSlides>0</HiddenSlides>
    <MMClips>0</MMClips>
    <ScaleCrop>false</ScaleCrop>
```

```xml
    <LinksUpToDate>false</LinksUpToDate>
    <SharedDoc>false</SharedDoc>
    <HyperlinksChanged>false</HyperlinksChanged>
    <AppVersion>16.0000</AppVersion>
```

ooxml	app.xml 주요요소	설명
common	<Application>	Application 종류
	<AppVersion>	Application 버전
pptx	<Slides>	슬라이드 개수
	<HiddenSlides>	숨겨진 슬라이드 수
	<notes>	노트 개수
docx	<pages>	페이지 개수
xlsx	<TitlesOfPart>	시트 개수
		시트 타이틀

OOXML 구조 | 공통부분

docProps/core.xml

core.xml파일은 사용자가 package 내에 생성자 이름, 생성 일자, 제목 등을 찾아서 설정할 수 있게 해준다.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<cp:coreProperties xmlns:cp="http://schemas.openxmlformats.org/package/2006/metadata/core-properties" xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:dcterms=
    <dc:title>PowerPoint 프레젠테이션</dc:title>
    <dc:creator>김기윤(학생-정보융합보안전공)</dc:creator>
    <cp:lastModifiedBy>김기윤(학생-정보융합보안전공)</cp:lastModifiedBy>
    <cp:revision>2</cp:revision>
    <dcterms:created xsi:type="dcterms:W3CDTF">2021-01-29T03:06:59Z</dcterms:created>
    <dcterms:modified xsi:type="dcterms:W3CDTF">2021-01-29T03:09:30Z</dcterms:modified>
</cp:coreProperties>
```

core.xml 주요요소	설명	ppt1.pptx 특징
<dc:creator>	생성자	김기윤(학생-정보융합보안전공)
<cp:lastModifiedBy>	마지막 수정자	김기윤(학생-정보융합보안전공)
<dcterms:created ~>	파일 생성 시간	2021-01-29T03:06:59Z
<dcterms:modified ~>	마지막으로 수정된 시간	2021-01-29T03:09:30Z

OOXML 구조 | PowerPoint

ppt/presentation.xml

이 파일은 pptx 전체 문서의 주요 부분으로서 이를 오피스 문서 내에서 표현하기 위한 각 본문 속성에 맞는 XML과 Object 참조에 대한 내용을 담고 있다

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<p:presentation xmlns:a="http://schemas.openxmlformats.org/drawingml/2006/main" xmlns:r="http://schemas.openxmlfor
    <p:sldMasterIdLst>
        <p:sldMasterId id="2147483648" r:id="rId1"/>
    </p:sldMasterIdLst>
    <p:sldIdLst>
        <p:sldId id="256" r:id="rId2"/>
    </p:sldIdLst>
    <p:sldSz cx="12192000" cy="6858000"/>
    <p:notesSz cx="6858000" cy="9144000"/>
    <p:defaultTextStyle>
        <a:defPPr>
            <a:defRPr lang="ko-KR"/>
        </a:defPPr>
        <a:lvl1pPr marL="0" algn="l" defTabSz="914400" rtl="0" eaLnBrk="1" latinLnBrk="1" hangingPunct="1">
            <a:defRPr sz="1800" kern="1200">
                <a:solidFill>
                    <a:schemeClr val="tx1"/>
                </a:solidFill>
                <a:latin typeface="+mn-lt"/>
                <a:ea typeface="+mn-ea"/>
                <a:cs typeface="+mn-cs"/>
            </a:defRPr>
        </a:lvl1pPr>
```

presentation.xml 주요요소	설명	pptx 특징
<p:sldMasterId>	슬라이드 마스터와의 관계	rId1 참조
<p:sldId>	슬라이드와의 관계	rId2 참조
<p:defaultTextSize>	프리젠테이션의 기본 텍스트 스타일이 요소로 저장	

r:id는 Relationship ID를 의미
id는 파트 자체에 해당하는 고유 값

OOXML 구조 | PowerPoint

ppt/_rels/presentation.xml.rels

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Relationships xmlns="http://schemas.openxmlformats.org/package/2006/relationships">
    <Relationship Id="rId3" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/presProps" Target="presProps.xml"/>
    <Relationship Id="rId2" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/slide" Target="slides/slide1.xml"/>
    <Relationship Id="rId1" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/slideMaster" Target="slideMasters/slideMaster1.xml"/>
    <Relationship Id="rId6" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/tableStyles" Target="tableStyles.xml"/>
    <Relationship Id="rId5" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/theme" Target="theme/theme1.xml"/>
    <Relationship Id="rId4" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/viewProps" Target="viewProps.xml"/>
</Relationships>
```

rId1 → slideMasters/slideMaster1.xml 참조
rId2 → slide/slide1.xml 참조

OOXML 구조 | PowerPoint

ppt/slideMaster/slideMaster1.xml

슬라이드 마스터는 슬라이드가 작성되는 기본 또는 템플릿으로, 사용할 테마와 사용 가능한 레이아웃 등을 지정
슬라이드 마스터에 지정된 내용과 서식은 slideLayout과 slides에 의해 변경 될 수 있음

ppt/slideLayout/slideLayout#.xml

슬라이드 레이아웃은 슬라이드 마스터에 제공된 정보를 변경하거나 보충하는 재정의 역할을 함.

OOXML 구조 | PowerPoint

ppt/slides/slide#.xml

slide#.xml에는 텍스트, 도형, 차트, 다이어그램 등이 포함

slide#.xml 주요요소	설명
<p:spTree>	슬라이드에 있는 텍스트, 표, 도형 또는 기타 내용으로 구성됨 개체가 나타나는 순서에 따라 개체가 겹쳐졌을 때 순서가 결정
<p:cNvPr>	사용한 개체 이름 같은 개체면 이름 뒤에 숫자가 1씩 늘어남
<a:t>	개체가 TextBox일 경우, 텍스트 내용이 표시됨
<a:blip r:embed=~>	개체가 이미지일 경우, media와의 관계 표시

```
<p:spTree>
    <p:nvGrpSpPr>
        <p:cNvPr id="1" name=""/>
        <p:cNvGrpSpPr/>
        <p:nvPr/>
    </p:nvGrpSpPr>
    <p:grpSpPr>
        <a:xfrm>
            <a:off x="0" y="0"/>
            <a:ext cx="0" cy="0"/>
            <a:chOff x="0" y="0"/>
            <a:chExt cx="0" cy="0"/>
        </a:xfrm>
    </p:grpSpPr>
    <p:sp>
        <p:nvSpPr>
            <p:cNvPr id="8" name="Rectangle 7">
                <a:extLst>
                    <a:ext uri="{FF2B5EF4-FFF2-40B4-BE49-F238E2
                        <a16:creationId xmlns:a16="http://schema
                    </a:ext>
                    <a:ext uri="{C183D7F6-B498-43B3-948B-1728B5
                        <adec:decorative xmlns:adec="http://sch
                    </a:ext>
                </a:extLst>
            </p:cNvPr>
```

OOXML 구조 | PowerPoint

ppt/slides/slide1.xml

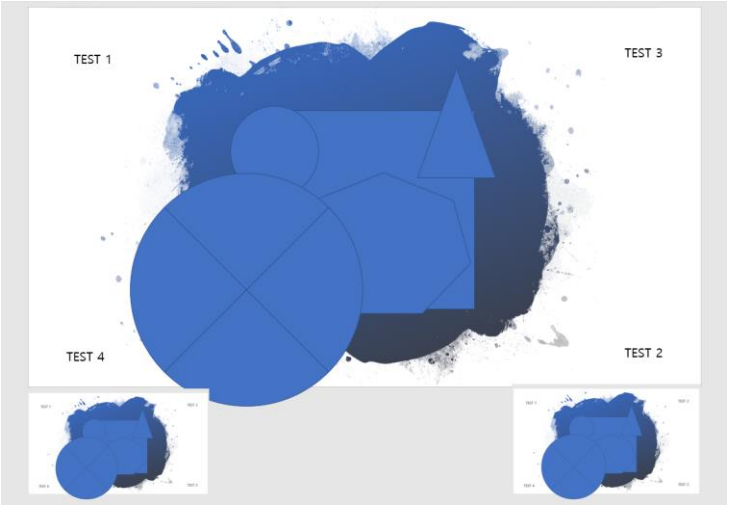
```
<p:cNvPr id="10" name="Picture 9">
<a:blip r:embed="rId2">
```

```
<p:cNvPr id="4" name="직사각형 3">
<p:cNvPr id="5" name="직사각형 4">
<p:cNvPr id="6" name="직사각형 5">
<p:cNvPr id="7" name="직사각형 6">
<p:cNvPr id="9" name="직사각형 8">
<p:cNvPr id="11" name="직사각형 10">
<p:cNvPr id="12" name="직사각형 11">
<p:cNvPr id="13" name="직사각형 12">
<p:cNvPr id="14" name="직사각형 13">
<p:cNvPr id="15" name="타원 14">
<p:cNvPr id="16" name="이등변 삼각형 15">
<p:cNvPr id="17" name="칠각형 16">
```

```
<p:cNvPr id="18" name="TextBox 17">
<a:t>TEST 1</a:t>

<p:cNvPr id="19" name="TextBox 18">
<a:t>TEST 2</a:t>

<p:cNvPr id="20" name="TextBox 19">
<a:t>TEST 3</a:t>

<p:cNvPr id="21" name="TextBox 20">
<a:t>TEST 4</a:t>
```

```
<p:cNvPr id="22" name="순서도: 가산 접합 21">
```

```
<p:cNvPr id="23" name="그림 22">
<a:blip r:embed="rId3"/>

<p:cNvPr id="24" name="그림 23">
<a:blip r:embed="rId3"/>
```



- 도형 13개
  - 직사각형 : 9개
  - 타원 : 1개
  - 이등변 삼각형 : 1개
  - 칠각형 : 1개
  - 순서도 가산 접합 : 1개
- Textbox 4개
  - TEST 1, TEST 2, TEST 3, TSET 4
- 그림 3개
  - Picture9 : 1개, rId2에 참조
  - 그림 22 : 2개, rId3에 참조

<p : spTree>의 일부 (개체의 이름 부분만)

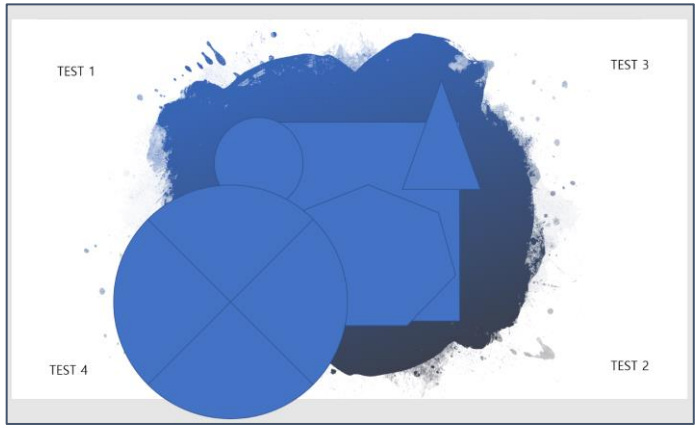
OOXML 구조 | PowerPoint

ppt/slides/_rels/slide1.xml.rels

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Relationships xmlns="http://schemas.openxmlformats.org/package/2006/relationships">
    <Relationship Id="rId3" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/image" Target="../media/image2.png"/>
    <Relationship Id="rId2" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/image" Target="../media/image1.png"/>
    <Relationship Id="rId1" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/slideLayout" Target="../slideLayouts/slideLayout1.xml"/>
</Relationships>
```

rId2 → media/image1.png 참조
rId3 → media/image2.png 참조

ppt/media/image1,2.png



OOXML 구조 | Word

word/document.xml

전체 문서의 주요 부분으로서 문서의 본문을 구성하는 텍스트와
이를 오피스 문서 내에서 표현하기 위한 각 본문 속성에 맞는 XML과 Object 참조에 대한 내용을 담고 있다.

```
<w:p w14:paraId="2C09605C" w14:textId="77777777" w:rsidR="00A92FE1" w:rsidRPr="00A92FE1" w:rsidRDefault="00A92FE1" w:rsidP
    <w:pPr>
        <w:rPr>
            <w:rFonts w:ascii="나눔스퀘어" w:eastAsia="나눔스퀘어" w:hAnsi="나눔스퀘어"/>
        </w:rPr>
    </w:pPr>
    <w:r w:rsidRPr="00A92FE1">
        <w:rPr>
            <w:rFonts w:ascii="나눔스퀘어" w:eastAsia="나눔스퀘어" w:hAnsi="나눔스퀘어" w:hint="eastAsia"/>
        </w:rPr>
        <w:t>표준에는</w:t>
    </w:r>
    <w:r w:rsidRPr="00A92FE1">
        <w:rPr>
            <w:rFonts w:ascii="나눔스퀘어" w:eastAsia="나눔스퀘어" w:hAnsi="나눔스퀘어"/>
        </w:rPr>
        <w:t xml:space="preserve"> 워드 프로 세서, 파워포인트, 엑셀 문서 형식이 정의되어 있고, 확장자는 기존의 확장자에서
    </w:r>
</w:p>
```

document.xml 주요요소	설명
<w:p w14:parald=~>	문서 본문을 구성하는 텍스트 한 줄에 대한 폰트와 내용을 나타냄
<w:t> or <w:t xml:space="preserve“>	텍스트 내용

OOXML 구조 | Excel

xl/workbook.xml

workbook에는 실제 내용이 없지만 데이터가 포함된 별도의 worksheet 부분에 대한 참조와 함께 스프레드 시트의 일부 속성만 포함됨.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<workbook  xmlns="http://schemas.openxmlformats.org/spreadshe
    <fileVersion  appName="xl" lastEdited="7" lowestEdited="7
    <workbookPr  defaultThemeVersion="166925"/>
    <mc:AlternateContent xmlns:mc="http://schemas.openxmlform
        <mc:Choice Requires="x15">
            <x15ac:absPath xmlns:x15ac="http://schemas.micros
        </mc:Choice>
    </mc:AlternateContent>
    <xr:revisionPtr revIDLastSave="0" documentId="13_ncr:1_{6
    <bookViews>
        <workbookView  xWindow="0" yWindow="0" windowWidth="2
    </bookViews>
    <sheets>
        <sheet  name="년도,논문지" sheetId="1" r:id="rId1"/>
        <sheet  name="논문명" sheetId="2" r:id="rId2"/>
    </sheets>
```

workbook.xml 주요요소	설명	pptx 특징
<sheet ~>	시트 이름 시트와의 관계	시트 이름 : “년도, 논문지“, “논문명” 시트 관계 : rId1, rId2 참조

OOXML 구조 | Excel

xl/shareStrings.xml

Excel의 대부분의 실제 콘텐츠는 worksheets 부분과 sharedStrings 부분에 있음

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<sst  xmlns="http://schemas.openxmlformats.org/spreadsheetml/2006/main" cou
    <si>
        <t>년도</t>
            <phoneticPr  fontId="1" type="noConversion"/>
        </si>
        <si>
            <t>논문지</t>
                <phoneticPr  fontId="1" type="noConversion"/>
            </si>
            <si>
                <t>논문명</t>
                    <phoneticPr  fontId="1" type="noConversion"/>
                </si>
                <si>
                    <t>DOL</t>
                        <phoneticPr  fontId="1" type="noConversion"/>
                    </si>
                    <si>
                        <t>한국교육학술정보원</t>
                            <phoneticPr  fontId="1" type="noConversion"/>
                        </si>
```

sharedStrings.xml 주요요소	설명
<t>	텍스트 내용을 나타냄

OOXML 파싱

파싱할 정보 정리

OOXML 종류	xml 파일	파싱할 정보	xml 요소
common	docProps/app.xml	OOXML 종류	<Application>
		version	<AppVersion>
	docProps/core.xml	생성자	<dc:creator>
		수정자	<cp:lastModifiedBy>
		생성 날짜	<dcterms:created ~>
		수정 날짜	<dcterms:modified ~>
powerpoint	docProps/app.xml	슬라이드 개수	<Slides>
		숨겨진 슬라이드 개수	<HiddenSlides>
	ppt/slides/slide#.xml	사용한 개체(도형) 이름	<p:cNvPr>
		텍스트 내용	<a:t>
word	docProps/app.xml	페이지 개수	<pages>
	word/document.xml	텍스트 내용	<w:t> or <w:t xml:space="preserve“>
excel	docProps/app.xml	시트 개수, 타이틀	<TitlesOfPart>
	xl/sharedStrings.xml	텍스트 내용	<t>

OOXML 파싱

file name, data offset, ooxml 여부 출력

```python
# little endian, 10진수로 변환
def little2(hex):
    return struct.unpack('<H', hex)[0] # 2byte

zipfilename = 'ooxml/ppt1.zip'
f = open(zipfilename, 'rb+')

# local file signature offset 찾기
LF_sig = b'\x50\x4B\x03\x04'
CF_sig = b'\x50\x4B\x01\x02'
LF_sig_offset = []
offset = 0

while True:
    f.seek(offset)
    fr =  f.read(4)

    if fr == LF_sig:
        LF_sig_offset.append(offset)
    elif fr == CF_sig:
        CF_sig_start_offset = offset
        break
    offset += 1
```

```python
# 파일 이름과 data offset 찾기
name = []
data_offset = []
ooxml = 'No'

for i in range(len(LF_sig_offset)):
    # Name Length
    nameLen_offset = LF_sig_offset[i] + 26
    f.seek(nameLen_offset)
    nameLen_hex = f.read(2)
    nameLen = little2(nameLen_hex)

    # Extra Length
    extraLen_offset = nameLen_offset + 2
    f.seek(extraLen_offset)
    extraLen_hex = f.read(2)
    extraLen = little2(extraLen_hex)

    # Name
    name_offset = extraLen_offset + 2
    f.seek(name_offset)
    name_hex = f.read(nameLen)
    name.append(name_hex.decode())

    # data
    dataOffset = name_offset + nameLen + extraLen
    data_offset.append(dataOffset)

    if name_hex.decode() == '[Content_Types].xml':
        ooxml = 'Yes'
```

```python
# data 길이 구하기
dataLen = []
for i in range(len(data_offset)):
    if i == len(data_offset)-1:
        dataLen.append(CF_sig_start_offset - data_offset[i])
    else:
        dataLen.append(LF_sig_offset[i+1] - data_offset[i])

print('File Name :', name)
print('\nData Offset :', data_offset)
print('\nOOXML인가? :', ooxml)
```

저번에 구현했던 zip 파싱 프로그램을 사용해
File name, data offset, OOXML 여부를 출력

→ OOXML 여부는 [Content_Types].xml 파일이 존재하면 OOXML이 맞다고 판단

OOXML 파싱

pptx

```
File Name : ['[Content_Types].xml', '_rels/.rels', 'ppt/slides/_rels/slide1.xml.rels', 'ppt/slides/slide1.xml', 'ppt/presentation.xml', 'ppt/_rels/presentation.xml.rels', 'ppt/slideMasters
/slideMaster1.xml', 'ppt/slideLayouts/_rels/slideLayout2.xml.rels', 'ppt/slideLayouts/_rels/slideLayout1.xml.rels', 'ppt/slideLayouts/slideLayout11.xml', 'ppt/slideMasters/_rels/slideMaste
r1.xml.rels', 'ppt/slideLayouts/slideLayout1.xml', 'ppt/slideLayouts/slideLayout2.xml', 'ppt/slideLayouts/slideLayout3.xml', 'ppt/slideLayouts/slideLayout4.xml', 'ppt/slideLayouts/slideLay
out5.xml', 'ppt/slideLayouts/slideLayout6.xml', 'ppt/slideLayouts/slideLayout7.xml', 'ppt/slideLayouts/slideLayout8.xml', 'ppt/slideLayouts/slideLayout9.xml', 'ppt/slideLayouts/slideLayout
10.xml', 'ppt/slideLayouts/_rels/slideLayout3.xml.rels', 'ppt/slideLayouts/_rels/slideLayout4.xml.rels', 'ppt/slideLayouts/_rels/slideLayout5.xml.rels', 'ppt/slideLayouts/_rels/slideLayout
6.xml.rels', 'ppt/slideLayouts/_rels/slideLayout7.xml.rels', 'ppt/slideLayouts/_rels/slideLayout8.xml.rels', 'ppt/slideLayouts/_rels/slideLayout9.xml.rels', 'ppt/slideLayouts/_rels/slideLa
yout10.xml.rels', 'ppt/slideLayouts/_rels/slideLayout11.xml.rels', 'ppt/media/image2.png', 'ppt/media/image1.png', 'docProps/thumbnail.jpeg', 'ppt/theme/theme1.xml', 'ppt/viewProps.xml', '
ppt/presProps.xml', 'ppt/tableStyles.xml', 'docProps/core.xml', 'docProps/app.xml']

Data Offset : [569, 1567, 1888, 2160, 4989, 5866, 6197, 8457, 8719, 8971, 10343, 10683, 12126, 13400, 14913, 16292, 18019, 19144, 20180, 21830, 23415, 24734, 24996, 25258, 25520, 25782, 26
044, 26306, 26569, 26832, 27070, 447206, 543080, 549817, 551598, 552054, 552500, 552983, 553709]

OOXML인가? : Yes
```

docx

```
File Name : ['[Content_Types].xml', '_rels/.rels', 'word/document.xml', 'word/_rels/document.xml.rels', 'word/media/image1.png', 'word/theme/theme1.xml', 'word/settings.xml', 'word/numberi
ng.xml', 'word/styles.xml', 'word/webSettings.xml', 'word/fontTable.xml', 'docProps/core.xml', 'docProps/app.xml']

Data Offset : [569, 1498, 1784, 5085, 5418, 78439, 80235, 81491, 82499, 85528, 86112, 87004, 87726]

OOXML인가? : Yes
```

xlsx

```
File Name : ['[Content_Types].xml', '_rels/.rels', 'xl/workbook.xml', 'xl/_rels/workbook.xml.rels', 'xl/worksheets/sheet1.xml', 'xl/worksheets/sheet2.xml', 'xl/theme/theme1.xml', 'xl/style
s.xml', 'xl/sharedStrings.xml', 'xl/worksheets/_rels/sheet1.xml.rels', 'xl/worksheets/_rels/sheet2.xml.rels', 'xl/printerSettings/printerSettings1.bin', 'xl/printerSettings/printerSettings
2.bin', 'docProps/core.xml', 'docProps/app.xml']

Data Offset : [569, 1501, 1790, 3035, 3339, 4444, 5624, 7547, 8496, 9220, 9478, 9923, 10435, 11189, 11825]

OOXML인가? : Yes
```

OOXML 파싱

xml 파일의 data offset, data len 구하기

```
# data offset, data len 구하기
media_name, media_data_offset, media_dataLen = [], [], []
slide_data_offset, slide_dataLen = [], []
for i in range(len(name)):
    if '/media/image' in name[i]:
        media_name.append(name[i])
        media_data_offset.append(data_offset[i])
        media_dataLen.append(dataLen[i])
    elif name[i] == 'docProps/app.xml':
        app_data_offset = data_offset[i]
        app_dataLen = dataLen[i]
    elif name[i] == 'docProps/core.xml':
        core_data_offset = data_offset[i]
        core_dataLen = dataLen[i]
    elif 'ppt/slides/slide' in name[i]:
        slide_data_offset.append(data_offset[i])
        slide_dataLen.append(dataLen[i])
    elif name[i] == 'xl/sharedStrings.xml':
        sheet_data_offset = data_offset[i]
        sheet_dataLen = dataLen[i]
    elif name[i] == 'word/document.xml':
        pages_data_offset = data_offset[i]
        pages_dataLen = dataLen[i]
```

[ppt or word]/media/image.png
docProps/app.xml
docProps/core.xml
ppt/slides/slide#.xml
xl/sharedStrings.xml
word/document.xml

이 파일들의 data offset과 data len 구하기

OOXML 파싱

media/image.png 파일 따로 저장

```python
# media data
for i in range(len(media_name)):
    f.seek(media_data_offset[i])
    media_data = f.read(media_dataLen[i])
    name = media_name[i].split('/')[-1]
    zipname = zipfilename.split('/')[-1][:-4]
    os.makedirs('ooxml/media/'+zipname, exist_ok=True)
    fw = open('ooxml/media/'+zipname+'/'+name, 'wb+')
    fw.write(media_data)
    fw.close()
```

[ppt or word]/media/image.png
파일의 데이터 영역은
파일을 만들어 따로 저장하기

pptx

docx

OOXML 파싱

xml 파일의 data 구하는 함수 구현

```
def xmldata(dataoffset, datalen):
    f.seek(dataoffset)
    data = f.read(datalen)
    data = zlib.decompress(data, -zlib.MAX_WBITS)
    data = data.decode()
    return data
```

Data offset과 data len을 이용해 데이터 영역 구하기
데이터 영역을 압축 해제하고 디코딩 해서 xml 파일의 data 구하기

OOXML 파싱

xml 파일의 필요한 요소 추출 함수 구현

```python
def xml(data, words):
    word = words.split()
    if len(word) == 1:
        word_s = word_e = word[0]
    else:
        word_s, word_e = words, word[0]

    word_result = []
    while(1):
        word_start = data.find("<"+word_s+">")
        word_end = data.find("</"+word_e+">", word_start)

        if word_start == -1 or word_end == -1:
            if len(word_result) == 0:
                word_result.append('')
            break
        else:
            wordsLen = len(words) + 2
            word = data[word_start+wordsLen:word_end]
            data = data[word_end+11:]
            if word == '': continue
            else:
                word_result.append(word)

    if len(word_result) == 1:
        return word_result[0]
    else:
        return word_result
```

xml 파일에서 필요한 데이터 뽑아내는 함수 구현

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Properties  xmlns="http://schemas.openxmlformats.org/officeDocument/2006/extended-properties"
    <TotalTime>2</TotalTime>
    <Words>8</Words>
    <Application>Microsoft Office PowerPoint</Application>
    <PresentationFormat>와이드스크린</PresentationFormat>
    <Paragraphs>4</Paragraphs>
    <Slides>1</Slides>
    <Notes>0</Notes>
    <HiddenSlides>0</HiddenSlides>
    <MMClips>0</MMClips>
    <ScaleCrop>false</ScaleCrop>
```

예를 들어, ooxml type을 구하기 위해
<Application>과 </Application> 사이의 값을 추출

OOXML 파싱

app.xml 데이터 추출하기

```python
# app.xml data
app_data = xmldata(app_data_offset, app_dataLen)

ooxml_type = xml(app_data, "Application")
print('\nOOXML Type :', ooxml_type)
print('Version :', xml(app_data, "AppVersion"))

if ooxml_type == 'Microsoft Office Word':
    print('Pages 개수 :', xml(app_data, "Pages"))
elif ooxml_type == 'Microsoft Office PowerPoint':
    print('Slides 개수 :', xml(app_data, "Slides"))
    print('숨겨진 Slides 개수 :', xml(app_data, "HiddenSlides"))
elif ooxml_type == 'Microsoft Excel':
    TitlesOfParts = xml(app_data, "TitlesOfParts")
    TitlesOfParts_num = int(TitlesOfParts.split()[1][6:-1])
    print('Sheets 개수 :', TitlesOfParts_num)
    print('Sheets 타이틀 :', xml(TitlesOfParts, "vt:lpstr"))
```

1. app.xml의 data 구하기
2. OOXML Type, Version 구하기
3. OOXML Type에 따라
   - Word면 Pages 개수 구하기
   - Powerpoint면 Slides 개수와 숨겨진 Slides 개수 구하기
   - Excel이면 Sheets 개수와 타이틀 구하기

pptx

```
OOXML Type : Microsoft Office PowerPoint
Version : 16.0000
Slides 개수 : 1
숨겨진 Slides 개수 : 0
```

xlsx

```
OOXML Type : Microsoft Excel
Version : 16.0300
Sheets 개수 : 2
Sheets 타이틀 : ['년도,논문지', '논문명']
```

docx

```
OOXML Type : Microsoft Office Word
Version : 16.0000
Pages 개수 : 2
```

OOXML 파싱

core.xml 데이터 추출하기

```
# core.xml data
core_data = xmldata(core_data_offset, core_dataLen)

print('\n생성한 사람 :', xml(core_data, "dc:creator"))
print('수정한 사람 :', xml(core_data, "cp:lastModifiedBy"))
print('생성한 날짜 :', xml(core_data, 'dcterms:created xsi:type="dcterms:W3CDTF"'))
print('수정한 날짜 :', xml(core_data, 'dcterms:modified xsi:type="dcterms:W3CDTF"'))
```

1. core.xml의 data 구하기
2. 생성한 사람 구하기
3. 수정한 사람 구하기
4. 생성한 날짜 구하기
5. 수정한 날짜 구하기

pptx

```
생성한 사람 : 김기윤(학생-정보융합보안전공)
수정한 사람 : 김기윤(학생-정보융합보안전공)
생성한 날짜 : 2021-01-29T03:06:59Z
수정한 날짜 : 2021-01-29T03:09:30Z
```

docx

```
생성한 사람 : (정보보안암호수학과)김수빈
수정한 사람 : (정보보안암호수학과)김수빈
생성한 날짜 : 2021-02-10T11:01:00Z
수정한 날짜 : 2021-02-10T11:03:00Z
```

xlsx

```
생성한 사람 : kimsu
수정한 사람 : kimsu
생성한 날짜 : 2020-08-04T09:00:19Z
수정한 날짜 : 2021-02-12T10:28:55Z
```

OOXML 파싱

pptx의 slide.xml에서 도형 이름과 text 구하기

```python
# ppt slide figure, text data
if ooxml_type == 'Microsoft Office PowerPoint':
    for i in range(len(slide_data_offset)):
        slide_data = xmldata(slide_data_offset[i], slide_dataLen[i])

        # figure 구하기
        nvSpPr = xml(slide_data, "p:nvSpPr")
        fig, fig_dic = [], {}
        for j in range(len(nvSpPr)):
            cNvPr_e = nvSpPr[j].find(">")
            cNvPr = nvSpPr[j][0:cNvPr_e+1]
            fn_s = cNvPr.find("name=")
            figs = cNvPr[fn_s+6:-2]
            fig = figs.split()
            fig_name = ' '.join(fig[0:-1])
            if fig_name not in fig_dic:
                fig_dic[fig_name] = 1
            else:
                fig_dic[fig_name] += 1
        print('\nSlide'+str(i+1)+' Figure :', fig_dic)

        # text 구하기
        txBody = xml(slide_data, 'p:txBody')
        text = []
        for k in range(len(txBody)):
            at = xml(txBody[k], 'a:t')
            if isinstance(at, list) == True:
                at = ''.join(at)
            if at != '':
                text.append(at)
        print('Slide'+str(i+1)+' Text :', text)
```

1. slide.xml의 data 구하기
2. ppt/slides/slide.xml 파일에서
   ppt slide당 사용한 도형과 text 추출하기

```
Slide1 Figure : {'Rectangle': 1, '직사각형': 9, '타원': 1, '이등변 삼각형': 1, '칠각형': 1, 'TextBox': 4, '순서도: 가산 접합': 1}
Slide1 Text : ['TEST 1', 'TEST 2', 'TEST 3', 'TEST 4']
```

OOXML 파싱

docx의 document.xml에서 text 구하기

```
# word pages text
elif ooxml_type == 'Microsoft Office Word':
    f.seek(pages_data_offset)
    pages_data = f.read(pages_dataLen)
    pages_data = zlib.decompress(pages_data, -zlib.MAX_WBITS)
    pages_data = pages_data.decode()

    p_text = []
    while(1):
        p_s = pages_data.find('<w:p w14:paraId=')
        p_e = pages_data.find('>', p_s)
        if p_s == -1 or p_e == -1:
            break
        wp = xml(pages_data, pages_data[p_s+1:p_e])
        wt = xml(wp, 'w:t')
        wts = xml(wp, 'w:t xml:space="preserve"')
        p_t = ''.join(wt) + ''.join(wts)
        if p_t != '': p_text.append(p_t)
        pages_data = pages_data[p_e:]

    print('\nPages text :', p_text)
```

1. document.xml의 data 구하기
2. word/document.xml 파일에서 사용한 text 추출하기

```
Pages text : ['OOXML이란', 'Previous MS Office 2007', 'MS Office 2007~2016', 'File format', 'Compound File Binary(CFBF)', 'Office Open XML(OOXML)', 'Document extension', 'xls, ppt, doc', '
xlsx, pptx, docx', 'OOXML(Office Open XML)은 마이크로소프트(MS)가 개발하였으며, 국제 표준 ECMA-376과 ISO/IEC 29500으로 등록되어 있다. ', 'MS 오피스 2007 이전 버전에서는 CFBF(Compound File
Binary Format)을 사용했지만 문서의 가용성 등 여러 개선 사항을 이유로 MS 오피스 2007부터 OOXML(Office Open XML)을 사용하고 있으며, 포맷이 변경됨에 따라 오피스 파일의 구성이 완전히 바뀌었다.
', '표준에는 워드 프로 세서, 파워포인트, 엑셀 문서 형식이 정의되어 있고, 확장자는 기존의 확장자에서 “x”의 postfix가 추가된 형태로 .docx, pptx, xlsx로 통일되어 있다. ', 'OOXML 형식의 문서들
은 공통적으로 논리적 개체인 패키지(package)이며 ZIP 아카이브 형식을 갖는다. ', '패키지는 여러 종류의 파트(part)와 파트 간의 관계를 정의하는 관계 파트(relationship part)로 구성되고 이와 같
은 요소들을 통합해 하나의 오브젝트로 표현해 준다.']
```

OOXML 파싱

xlsx의 sharedStrings.xml에서 text 구하기

```
# excel sheet text
elif ooxml_type == 'Microsoft Excel':
    f.seek(sheet_data_offset)
    sheet_data = f.read(sheet_dataLen)
    sheet_data = zlib.decompress(sheet_data, -zlib.MAX_WBITS)
    sheet_data = sheet_data.decode()

    print('\nSheet text :', xml(sheet_data, 't'))
```

1. sharedStrings.xml의
   data offset과 data len으로 데이터 영역 구하기
2. xl/sharedStrings.xml 파일에서
   사용한 text 추출하기

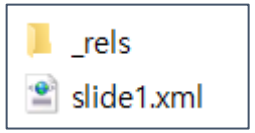
```
Sheet text : ['년도', '논문지', '논문명', 'DOL', '한국교육학술정보원', 'http://www.riss.kr/link?id=T15530427&amp;outLink=K', '한국정보보호학회', '한국디지털포렌식학회', 'LG 갤러리 애플리케이션 잠금 파일 복호화 연구', 'http://www.dbpia.co.kr/journal/articleDetail?nodeId=NODE08736974', 'http://www.dbpia.co.kr/journal/articleDetail?nodeId=NODE07298527', 'http://dx.doi.org/10.13089/JKIISC.2018.28.2.407', 'macOS Mojave에서의 포렌식 분석 절차 연구', '디지털 포렌식에서의 인스타그램 사용자 행위 분석', '포렌식 도구 기반 스마트폰 디지털 증거 수집 및 분석 절차', 'http://www.dbpia.co.kr/journal/articleDetail?nodeId=NODE09329596', 'macOS 포렌식 분석 기법']
```

ppt1.pptx와 ppt2.pptx의 차이점

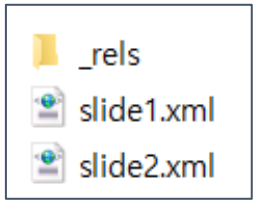
ppt1.pptx와 ppt2.pptx의 차이점

ppt1.pptx와 ppt2.pptx는 직접 열어서 보면 슬라이드 1개로 똑같이 생겼다.
하지만 ppt/slides/ 를 보면 ppt2.pptx는 slide가 하나 더 있는 것을 알 수 있다.

ppt1/ppt/slides/



ppt2/ppt/slides/



ppt1.pptx와 ppt2.pptx의 차이점

ppt2/ppt/presentation.xml.rels

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Relationships xmlns="http://schemas.openxmlformats.org/package/2006/relationships">
    <Relationship Id="rId3" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/slide" Target="slides/slide2.xml"/>
    <Relationship Id="rId7" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/tableStyles" Target="tableStyles.xml"/>
    <Relationship Id="rId2" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/slide" Target="slides/slide1.xml"/>
    <Relationship Id="rId1" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/slideMaster" Target="slideMasters/slideMaster1.xml"/>
    <Relationship Id="rId6" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/theme" Target="theme/theme1.xml"/>
    <Relationship Id="rId5" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/viewProps" Target="viewProps.xml"/>
    <Relationship Id="rId4" Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/presProps" Target="presProps.xml"/>
</Relationships>
```

ppt2/ppt/presentation.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<p:presentation xmlns:a="http://schemas.openxmlformats.
    <p:sldMasterIdLst>
        <p:sldMasterId id="2147483648" r:id="rId1"/>
    </p:sldMasterIdLst>
    <p:sldIdLst>
        <p:sldId id="256" r:id="rId2"/>
    </p:sldIdLst>
```

presentation.xml.rels 파일을 보면 slide#.xml을 참조하고 있는 것은 rId2, rId3 두 개인데 presentation.xml에서는 rId2만 참조하고 있다.

presentation.xml 파일에서 rId3를 참조하는 부분을 삭제해서 은닉한 것으로 보인다.

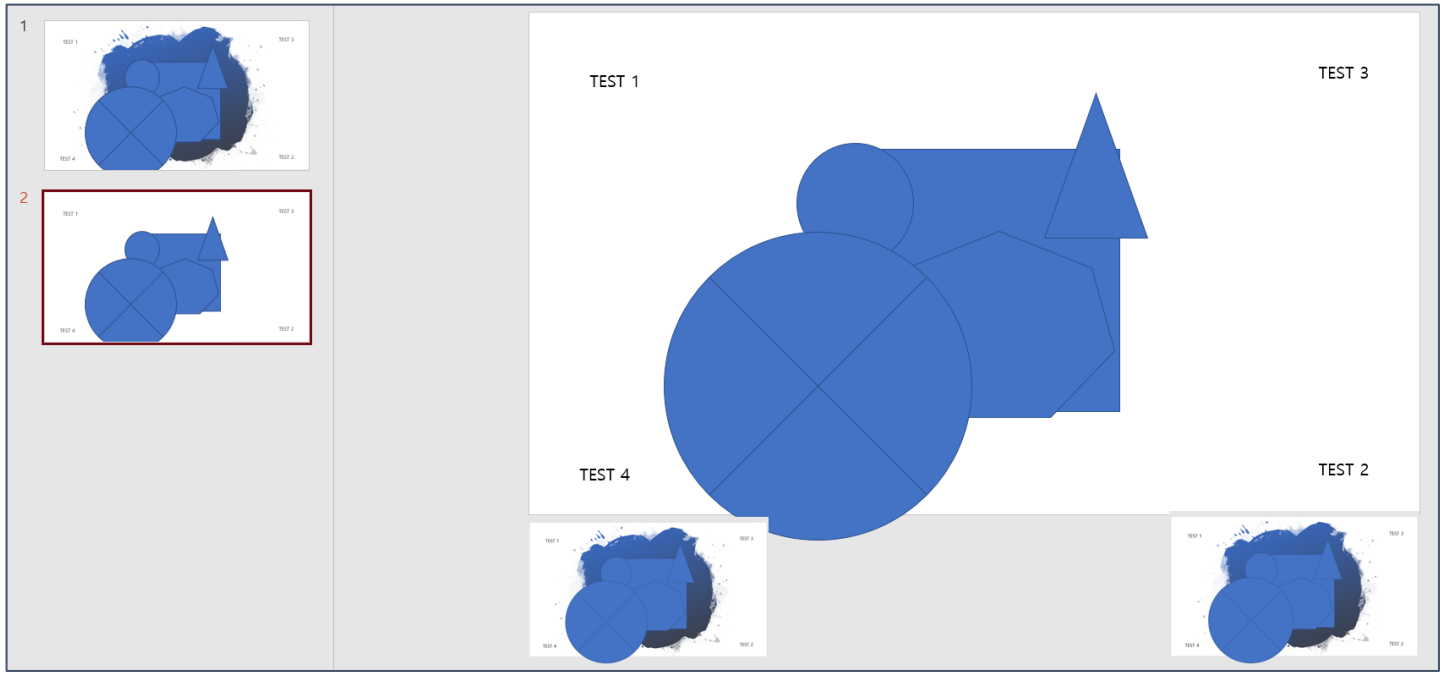
ppt1.pptx와 ppt2.pptx의 차이점

ppt2/ppt/presentation.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?
<p:presentation xmlns:a="http://schemas.openxmlformats
    <p:sldMasterIdLst>
        <p:sldMasterId id="2147483648" r:id="rId1"/>
    </p:sldMasterIdLst>
    <p:sldIdLst>
        <p:sldId id="256" r:id="rId2"/>
        <p:sldId id="257" r:id="rId3"/>
    </p:sldIdLst>
```

은닉된 데이터를 다시 보이도록 하기 위해서 rId3를 참조하는 부분을 추가해줍니다.

ppt2.pptx



수정한 내용을 저장한 뒤 다시 압축하여 확장자를 .pptx로 변경하면, 은닉된 슬라이드를 볼 수 있습니다.