

데이터 문해력과 디지털 글쓰기

오픈소스 소프트웨어: 빗에듀(Bit-Eud)

이광춘 / 한국 R 사용자회

11/4/22

목차

서문	4
I 글쓰기	5
1 디지털 글쓰기	6
1.1 디지털 문서 제작 패러다임	6
1.2 문서로 보는 민주주의	7
1.3 디지털 문서 저작	7
1.4 과학문서 저작	8
1.5 사례: 한국 R 컨퍼런스(2021)	8
2 문자와 종이 역사	9
2.1 언어와 문자	9
2.2 매체 역사	9
2.3 문자	11
2.3.1 알파벳	11
2.3.2 문자 시스템	12
2.3.3 글쓰기 방향	12
2.3.4 표의 문자	13
2.4 저작	13
II 디지털 글쓰기	15
3 한글 글꼴	16
3.1 글꼴 분류	16
3.1.1 typeface vs font	16
3.1.2 파일폰트와 웹폰트	18
3.2 출판 문서	18
3.3 css 파일	19
3.4 그래프	20
3.5 표	22
4 한글 PDF	25
4.1 출판 디지털 전환	25

4.2	<i>LaTeX</i> 개요	27
4.3	Quarto PDF	28
4.3.1	<i>LaTeX</i> 엔진	29
4.3.2	YAML 파일	29
4.3.3	저작	30
4.4	한글 글꼴	32
III	빛에듀(Bit-Edu)	34
5	빛에듀	35
5.1	데이터	35
5.2	기술통계	35
	참고문헌	36

서문

Part I

글쓰기

1 디지털 글쓰기

1.1 디지털 문서 제작 패러다임

신속하고 빠르게 누구나 짧은 학습을 통해서 문서를 저작하고 출판할 수 있는 방식으로 인기를 얻은 아래한글 혹은 MS워드 워드프로세서를 사용하는데 이는 위지위그(WYSIWYG: What You See Is What You Get, “보는 대로 얻는다”)에 기초한 것으로 문서 편집 과정에서 화면에 포맷된 낱말, 문장이 출력물과 동일하게 나오는 방식이다. 이전 타자기 대비 경쟁력이 있었고 이를 바탕으로 문서편집기 소프트웨어가 대세로 떠올랐다.

위지위그의 대척점에 있는 것이 위지위임(WYSIWYM, What You See Is What You Mean)으로 대표적인 것인 *LaTeX* 으로 구조화된 방식으로 문서를 작성하면 컴파일을 통해서 최종 문서가 미려한 출판가능한 PDF, PS, DVI 등 확장자를 갖는 출판결과물을 얻을 수 있다.

*LaTeX*이 갖는 장점은 이제 프로그래밍 코드(R / 파이썬 / SQL / 셸스크립트 / 자바스크립트 등)를 문서저작에 포함되며 *literate programming* 패러다임이 대세로 떠올랐고 현존하는 다양한 문서저작 문제에 대한 응답으로 자리를 매김하고 있다. ¹



¹[출처: <http://www.cbinews.co.kr/news/articleView.html?idxno=36859>]

1.2 문서로 보는 민주주의

대한민국은 권력을 입법권, 행정권, 사법권으로 나눠 권력을 나누었다. 세부적으로 보면 다소 차이는 있겠지만 법을 만드는 것은 결국 문서를 저작하는 것이고 대한민국 정부는 법에 담긴 문서를 실행하는 기관이고 사법부는 법 문서를 해석하는 기관으로 볼 수 있다. 결국 민주주의는 문서를 기반으로 존립하는 사회체제라고 볼 수 있다. 최근 들어 인공지능이 부상하면서 문서저작을 보조하던 수단에서 이제는 사람보다 더 나은 통찰력을 보이는 분야도 속속 등장하고 있다. 이런 측면에서 인공지능 기계를 현 민주주의 체계 내에 어떻게 녹여낼 것인지 관심이 높아지고 있다.



1.3 디지털 문서 저작

과거 둘로 쪼개져서 과학기술 문서저작 즉 논문 등은 *LaTeX* 기반 위지윅(WYSIWYM) 패러다임이 확고한 자리를 잡았지만 위지위그(WYSIWYG) 패러다임 기반 아래한글 워드와 같은 문서저작편집기 성능이 높아지면서 사실상 둘 사이 경계도 허물어졌다.

하지만, 과학기술 논문 저작에 진실성, 재현성 등이 강조되고 빅데이터 기반 과학연구가 큰 성과를 보이고 프로그래밍이 과학기술 논문 저작에 주요한 역할을 하면서 이런 추세를 반영한 과학기술 저작 패러다임이 떠오르고 있다.

이러한 추세를 반영한 도구의 발전도 함께 이뤄지면서 과학기술 저작은 물론 거의 모든 문서 저작분야에 큰 변화가 일어나고 있다.

1.4 과학문서 저작

- [심각한 현재상황](#)
- [마크다운 기초](#)
- [고급 마크다운](#)
- [문서저작 실무](#)
- [수학 수식](#)

1.5 사례: 한국 R 컨퍼런스(2021)

- [컨퍼런스 웹사이트](#)
- [컨퍼런스 전 기획](#)
 - [컨퍼런스 포스터](#)
 - [컨퍼런스 후원](#)
- [컨퍼런스 본 행사](#)
 - [발표 슬라이드](#)
 - [발표 템플릿](#)
- [컨퍼런스 후 설문](#)
 - [설문조사 보고서](#)
- [기타](#)
 - [탄소중립 신재생 에너지 캠프](#)

2 문자와 종이 역사

2.1 언어와 문자

전세계인구는 77.5억(2020년 기준)명이며 언어는 대략 7,100개 언어가 존재하고 있는 것으로 파악되고 있으며 영어, 중국어, 힌디어, 스페인어, 프랑스어, 아랍어, 벵골어, 러시아어, 포르투갈어, 인도네시아어 순으로 많은 언어인구를 갖고 있다.

한국어 언어는 한글을 문자로 갖고 있지만 모든 언어가 문자(Writing System)를 갖는 것은 아니다. 다르게 보면 동일한 뿌리를 갖는 문자를 공유하는 언어도 많다. 예를 들어,

- 알파벳: 라틴어(영어, 불어, 독어, 이태리어 등), 쉐틱, 발틱, 슬로박, 등
- 한자: 중국어, 대만어
- 아랍문자: 아랍, 인도-이란 어족(페르시아, 펀잡 등)
- 가나: 일본
- ...

문자가 존재하기 위한 전제조건이 있는데 크게 다음 세가지를 들고 있다.

- 언어 : 한국어, 영어, 중국어, ...
- 아이디어: 생각, 콘텐츠, ...
- 기호: 숫자, 문자, 그림, ...
 - 언어
 - 숫자와 수식
 - 교통표지 등 표식
- 매체: 종이, 파피루스, 디스플레이

2.2 매체 역사

메시지를 전달하기 위해서 의미를 담을 수 있는 상형문자가 전세계 동시 다발적으로 개발되었으나 동시에 추상적인 개념 (예를 들어, 숫자)을 전달하기 위한 문자체계도 필요하여 개발됨과 동시에 문자를 담을 수 있는 매체에 대한 발전도 함께 이뤄졌다.

초기 동굴벽화처럼 벽면에 그림을 그려서 의사를 전달하는 체계가 출현한 것은 기원전 16,500년 라스코 동굴벽화를 비롯하여 반구암각화에도 그 흔적을 어렵지 않게 찾을 수 있다. 기원전

3,000년 수메르 문명이 출현하면서 수십만의 사람들이 도시에 모여살게 되면서 세금 징수, 계약 등 산적한 정치, 사회, 경제 문제를 해결하는데 필수적으로 문자가 필요하게 되었다. 동시에 세금과 계약관계를 기록하고 보존할 수 있는 방식으로 점토를 사용하였고, 이집트에서도 유사한 목적으로 점토판 대신에 파피루스를 종이처럼 사용하는 방식을 사용했다.



그림 2.1: 문자 매체의 진화

문자를 담을 수 있는 매체도 생산과 유지관리에 비용이 많이 들고, 상형문자가 담을 수 있는 정보에 한계가 보임에 따라 지중해에서 수많은 물품을 거래하던 페니키아인들은 기존 상형문자체계를 벗어나 소리에 의미를 담을 수 있는 방식으로 진화시켜 무한대에 가까운 의미를 담을 수 있는 문자를 만들어냈는데 기원전 1,200년 **페니키아 문자**를 창안하게 되었다. 이후, 그리스 헬레니즘 문화를 거치면서 줄리어스 시저 시대 라틴 알파벳(대문자)이 공표되면서 오늘에 이르게 되었다. 라틴 알파벳을 담을 수 있는 매체로 양피지(Parchment)가 사용되었고 필경사(Scribe)가 한땀 한땀 손으로 필사를 했다.

동굴벽, 점토판, 파피루스 종이, 양피지를 거쳐 드디어 106년 채륜(Cai Lun)이 종이를 발명하면서 문자문화에 큰 변화가 생기게 되었다. 이후 서기 800년경 목판인쇄술이 발명되고 1450년 구텐베르크 인쇄술을 통해 성경이 대량 인쇄되어 르네상스의 초석이 되었으며, 푸어드니어(Fourdrinier) 형제가 1806년 종이를 대량생산하게 되는 초지기(Paper Macine)를 발명하면서 저렴한 비용으로 일반대중도 아이디어를 문자로 종이에 기록하여 널리 알릴 수 있는 토대가 완성되었다. 여기에 더해 1922년 독일에서 ISO 216을 통해 종이에 대한 국제 표준이 자리잡으면서 미국을 비롯한 일부 국가를 제외하고 전세계적으로 문자를 담아낸 매체에 대한 표준이 확립되었다.

2.3 문자

언어를 기록하기 위한 상징체계로 문자(□□, writing system)가 발명되었다. 문자는 언어중에서도 음성언어를 기록하기 위해 생겨난 것으로 알려져 있다. 문자는 정보의 저장과 전달에 신뢰성이 담보된다는 점에서 그 유용성이 크다.

문자는 크게 표음문자(Phonographic)와 표의문자(Logographic)로 나뉜다. 표의문자의 대표적인 문자로 중국 한문과 이집트 상형문자가 꼽힌다. 예를 들어 □(나무 목)은 목이라는 소리를 내지만 뜻은 나무를 갖고 있다. 표음문자는 한글을 비롯한 라틴계열 대부분의 언어가 여기에 포함된다.

2.3.1 알파벳 ¹

알파벳(Alphabet) 문자는 이집트 상형문자에서 시작되었다고 알려져 있다. 이후 상형문자는 페니키아 상인이 소리를 기반으로 의미를 전달하는 방식으로 진화되었고 그리스를 거쳐 알파벳 뒤쪽 문자가 추가되고 로마시대에 오늘날 알려진 영어 대문자가 완성되고 이후 소문자가 손으로 필사되면서 발명되고 다양한 글꼴이 생겨나게 되었다. 구텐베르그가 활자인쇄술을 발명되면서 현재와 같은 알파벳 문자가 완성되었다.

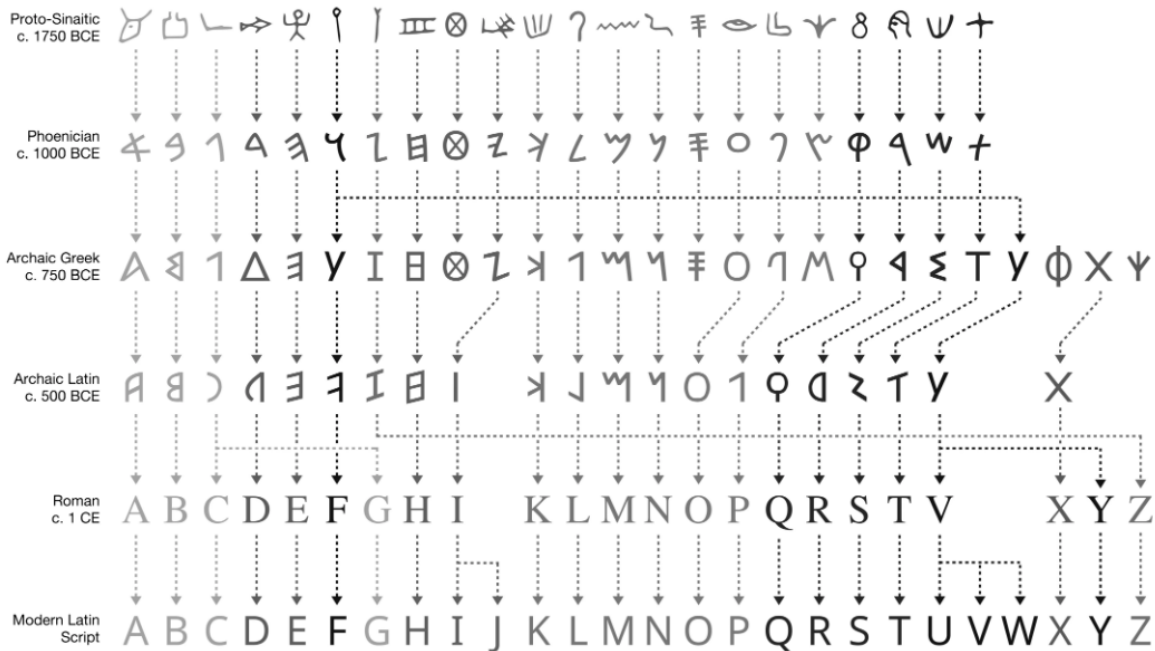


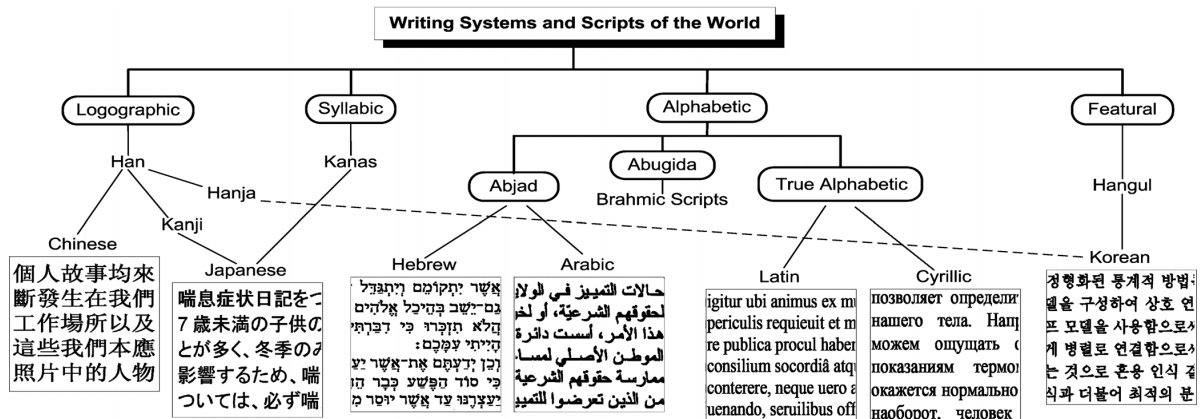
그림 2.2: 알파벳 진화 역사

¹[Evolution of the Alphabet | Earliest Forms to Modern Latin Script [유튜브](#)]

2.3.2 문자 시스템

앞서 문자를 크게 표음문자와 표의문자로 구분하였다. 이를 좀더 상세히 세분화하면 전세계 문자는 크게 다음과 같이 4개로 구분된다. (Ghosh, Dube, and Adamane 2010) 표어문자의 대표적인 문자가 중국의 한자, 음절문자의 대표적 문자가 일본어, 자질문자의 대표 문자가 한글이다. 그외 음소문자는 로마자로 대표되는 영문자가 있다.

- 표어문자(logographic)
- 음절문자(syllabic)
- 음소문자(alphabetic)
- 자질문자(featural)



종류	기호 표현	예
표어 문자	형태소	중국의 한자
음절 문자	음절 또는 모라	일본의 가나 (문자)
음소 문자	낱소리 (자음 또는 모음)	로마자, 키릴문자, 그리스 문자
아부기다	낱소리 (자음+모음)	데바나가리 문자, 그으즈 문자
아브자드	낱소리 (자음)	아랍 문자, 히브리 문자
자질 문자	음성 자질	한국의 한글

2.3.3 글쓰기 방향

대부분의 문자를 왼쪽에서 오른쪽, 위에서 아래 방향으로 작성하는 방식을 취하고 있으나 이슬람권에서는 오른쪽에서 왼쪽으로 작성을 하고 있고 몽골은 위에서 아래로 작성하고 있다.

2.3.4 표의 문자

한자와 같은 표의 문자가 여러가지 단점이 있는 것처럼 보이기도 하지만 표의문자가 갖는 장점으로 인해 그 중요성이 더욱 부각되고 있다. 표의문자가 갖는 장점은 문자 자체 의미가 부여되기 때문에 전세계 누구도 문자만 보면 그 뜻을 유추할 수 있다는 점이다.

- 그림문자(Pictographic): □, □, □, ...
- 표의문자(Ideographic): □ 금지와 같은 의미를 갖는 문자로 □ □ 교통 표지판, ► 전자제품 사용
- 추상 어표(Abstract Logographic): 한자 사람 인(人), %, &, ...

그림문자와 표의문자를 결합하게 되면 다음과 같이 흡연을 금지하는 의미를 명확히 전달할 수 있다.



그림 2.3: 흡연과 금지가 결합된 표의문자

2.4 저작

점토판에 썰기를 사용해서 문자를 활용하여 갖고 있던 아이디어(세금, 계약, 신화 등)를 작성하던 방식이 중세시대 종이가 발명되며 동양에서는 붓과 머루, 서양에서는 잉크와 새깃털펜으로 저작을 하였다. 이후 타자기가 발명되어 개인 저작이 가능하게 되었으며 이후 전동타자기도 등장하여 아이디어를 문서로 남기는 것이 훨씬 비용도 저렴해지고 품질도 좋아지고 출판까지 과정도 신속해졌다. 제2차 세계대전이 남긴 에니악 컴퓨터는 민수로 전환되어 IBM에서 1964년 워드프로세서(Word Processor)를 시장에 출시하였고 1969년 IBM에서 저장장치도 출시되어 기계식 저작방식이 디지털 저작방식으로 넘어가는 기반이 다져졌다. 1980년대 전후로 컴퓨터를 기반으로 두가지 저작방식 즉, 마이크로소프트 워드 1.0 혹은 아래한글과 같은 위지윅(WYSIWYG) 방식과 문서를 구조적으로 작성하여 컴파일하는 위지윅(WYSIWYM) 방식이 일반화되면서 현재에 이르고 있다.



Part II

디지털 글쓰기

3 한글 글꼴

3.1 글꼴 분류 ¹

- 활자(Type)는 본래는 글자를 만드는 데 사용된 나무 또는 금속 조각을 일컫는 말이었습니다.
- 타이포그래피(Typography): 인쇄된 텍스트의 스타일과 모양을 가리키는 디자인 용어입니다.
- 서체(또는 글꼴 모음, Typeface): 글자 및 기호가 서로 비슷한 특징을 가지도록 디자인된 문자로 타임즈(Times), 에어리얼(Arial) 및 미니언(Minion)을 들 수 있습니다.
- 글꼴(Font): 서체의 특정 스타일을 지칭하는데, 예를 들어, 에어리얼 블랙(Arial Black)은 에어리얼 서체 중 한 글꼴이며, 미니언 프로 이탤릭체(Minion Pro Italic)는 미니언 글꼴 모음에 포함되어 있는 글꼴이다.

글꼴(Font)는 다양하지만 크게 다음 5가지로 영문의 경우 분류된다. 글꼴은 기본적으로 세리프(serif)와 산세리프(sans serif)의 2가지 스타일로 구분되는데 세리프는 글자의 획 끝에 작은 빠침이 있는 글꼴이며 산스(sans)은 라틴어로 없다는 뜻이기 때문에 빠침이 없는 글꼴을 지칭합니다.

- serif
- sans serif
- script
- monospaced
- display

가장 쉽게 분류되는 monospaced 글꼴은 코드를 작성하기 위해 등간격을 갖는 글꼴로 consolas, D2 Coding이 대표적이다. 산세리프(Sans serif)와 세리프(Serif) 글꼴은 본문에 사용하기 적합하지만, 좀더 시선을 끄는 디자인을 위해서는 조금 더 매력적인 디스플레이(Display) 혹은 스크립트(Script)을 선택하고, 제목(Title)에는 산세리프 글꼴을 사용하는 것이 일반적이다.

3.1.1 typeface vs font

typeface는 서체로 번역되고 font는 글꼴로 번역된다. 하나의 typeface에 다수 font가 포함될 수 있다. **팬그램(Pangram)**은 '모든 글자'라는 뜻으로 주어진 모든 문자를 적어도 한 번 이상 사용하여 만든 문장을 뜻한다. 팬그램은 로렘 입숨처럼 글꼴 샘플을 보여주거나 장비를 테스트하는 데 사용된다. 영어 팬그램으로 가장 유명한 'The quick brown fox jumps over the lazy dog'는

¹ Adobe Korea (August 21, 2017), "글꼴의 기능과 디자인에 적합한 글꼴 선택 방법"

GRAVITAS

손글씨



script, display 폰트

corsiva / CRAVITAS
Nanum Pen Script

Title 제목

Title 제목



serif, san serif 폰트

verdana

나눔 명조

대통령의 임기는 5년으로 하며, 중임할 수 없다. 국회는 의장 1인과 부의장 2인을 선출한다. 대통령이 궐위되거나 사고로 인하여 직무를 수행할 수 없을 때에는 국무총리, 법률이 정한 국무위원의 순서로 그 권한을 대행한다.



serif, san serif 폰트

나눔 명조

맑은 고딕

모든 국민은 통신의 비밀을 침해받지 아니한다. 국민경제자문회의의 조직·직무범위 기타 필요한 사항은 법률로 정한다. 위원은 정당에 가입하거나 정치에 관여할 수 없다.

코딩 폰트는 달라요

```
library(tidyverse)
```

```
novel_dat %>%  
  ggplot(aes(x= `나이`)) +  
    geom_freqpoly(breaks= seq(10, 80, by=10)) +  
    geom_point(stat="bin", aes(y=..count..), breaks= seq(10,  
80, by=10), size = 3, color = "midnightblue") +  
    scale_x_continuous(breaks = seq(10, 80, by=10)) +  
    labs(x = "연령대",  
         y = "노벨평화상 수상자수",  
         title = "2001년부터 2015년까지 노벨 평화상을 받은 개인  
수상자들의 수상 당시 나이 분포") +  
    theme_light()
```



monospaced 폰트

consolas

D2 Coding

그림 3.1: 출판문서 글꼴 분류

19세기부터 사용되어 왔다. 유사하게 한글에서는 “다람쥐 헌 쳇바퀴에 타고파”가 사용되고 있다.

3.1.2 파일폰트와 웹폰트

데이터 사이언스 출판을 위해 필요한 글꼴을 반영하는 방식은 크게 두가지로 나뉜다. 하나는 전통적인 방식으로 운영체제에 내장된 글꼴을 문서제작에 글꼴을 반영하는 방식이고, 다른 하나는 웹폰트를 사용하는 방식이다. 웹폰트를 사용하게 될 경우 장점은 출판문서가 동작되는 컴퓨터에 해당 글꼴이 없어도 인터넷을 통해 다운로드하여 글꼴이 반영되기 때문에 글꼴이 없어 생기는 다양한 문제를 해결하는 장점이 있다. 웹폰트로 구글폰트와 눈누가 유명하다. 저작권을 살펴보고 문서제작에 필요한 글꼴을 사용하면 된다.

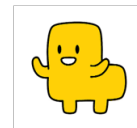
- 눈누
- 구글 폰트



C:\Windows\Fonts

MaruBuri-Regular.ttf

다람쥐 헌 쳇바퀴에 타고파. 1234567890
다람쥐 헌 쳇바퀴에 타고파. 1234567890
다람쥐 헌 쳇바퀴에 타고파. 1234567890
다람쥐 헌 쳇바퀴에 타고파. 1234567890
다람쥐 헌 쳇바퀴에 타고파. 1234567890



눈누: <https://noonnu.cc/>

구글폰트: <https://fonts.google.com/>

```
@font-face {  
  font-family: 'MaruBuri-Regular';  
  src: url('https://.../MaruBuri-Regular.woff') format('woff');  
  font-weight: normal;  
  font-style: normal; }  
  
@import url('https://fonts.googleapis.com/css2?family=Noto+Sans+KR:wght@500');
```

3.2 출판 문서

데이터 사이언스 한글 출판 문서 제작을 위해서 다음 글꼴을 선택한다. 제목으로 산스 세리프 폰트로 구글 폰트에서 “노토 산스 세리프”를 선정하고, 본문은 눈누 웹사이트에서 “마루부리”를 선정하고 코딩 글꼴 폰트로 눈누에서 D2Coding 폰트를 지정한다.

- 산스 세리프(제목): Noto Sans Serif
- 세리프(본문): 마루부리
- 등간격(코딩): D2Coding

library(tidyverse)

```

mass_flipper <- ggplot(data = penguins,
                      aes(x = flipper_length_mm,
                          y = body_mass_g)) +
  geom_point(size = 3,
             alpha = 0.8) +
  theme_minimal() +
  labs(title = "팔머 관측소 LTER 서식 펭귄 크기",
       subtitle = "Adelie, Chinstrap, Gentoo 펭귄에 대한 물갈퀴 길이와 체질량",
       x = "물갈퀴 길이(Flipper length) (mm)",
       y = "체질량(Body mass) (g)") +
  theme_minimal()

```

3.3 css 파일

출판문서 제작에 사용된 글꼴 관련 사항을 css/quarto-fonts.css 파일에 지정해둔다.

```

/* 글꼴 */
@import url('https://fonts.googleapis.com/css2?family=Noto+Sans+KR');

@import url('https://fonts.googleapis.com/css2?family=Noto+Serif+KR');

@import url('https://fonts.googleapis.com/css2?family=Nanum+Myeongjo&display=swap');

@font-face {
  font-family: 'MaruBuri-Regular';
  src: url('https://cdn.jsdelivr.net/gh/projectnoonnu/noonfonts_20-10-21@1.0/MaruBuri-Regular.woff') format('woff');
  font-weight: normal;
  font-style: normal;
}

@font-face {
  font-family: 'D2Coding';
  src: url('https://cdn.jsdelivr.net/gh/projectnoonnu/noonfonts_three@1.0/D2Coding.woff') format('woff');
  font-weight: normal;
  font-style: normal;
}

/* HTML */

h1, h2, h3, h4, h5, h6 {

```

```

font-family: "Noto Sans KR", sans-serif;
}

body {
  font-family: "Nanum Myeongjo", "MaruBuri-Regular", "Noto Serif KR", serif;
  font-size: 15px;
}

tt, code, pre {
  font-family: "D2Coding", Consolas, monospace;
  font-size: 15px;
}

```

3.4 그래프

그래프(ggplot)에 사용되는 글꼴도 마찬가지로 showtext 패키지 웹폰트를 사용해서 다양한 한글도 사용할 수 있다. showtext 패키지는 font_add_google() 함수를 통해서 구글 한글폰트를 지정하여 이를 ggplot 에 적용시킬 수 있다.

```

library(tidyverse)
library(showtext)
library(palmerpenguins)

font_add_google("Nanum Pen Script", "nanum_pen_script")
font_add_google("Jua", "Jua")
showtext_auto()

theme_quarto <- theme(
  text = element_text(family = 'Jua', size = 25),
  plot.title.position = 'plot',
  plot.title = element_text(
    family = 'nanum_pen_script', size = 55,
    face = 'bold',
    colour = thematic::okabe_ito(8)[3],
    margin = margin(t = 2, r = 0, b = 3, l = 0, unit = "mm")
  ),
  plot.subtitle = element_text(
    family = 'Jua', size = 25,
    face = 'bold',
    colour = thematic::okabe_ito(8)[5],

```

```

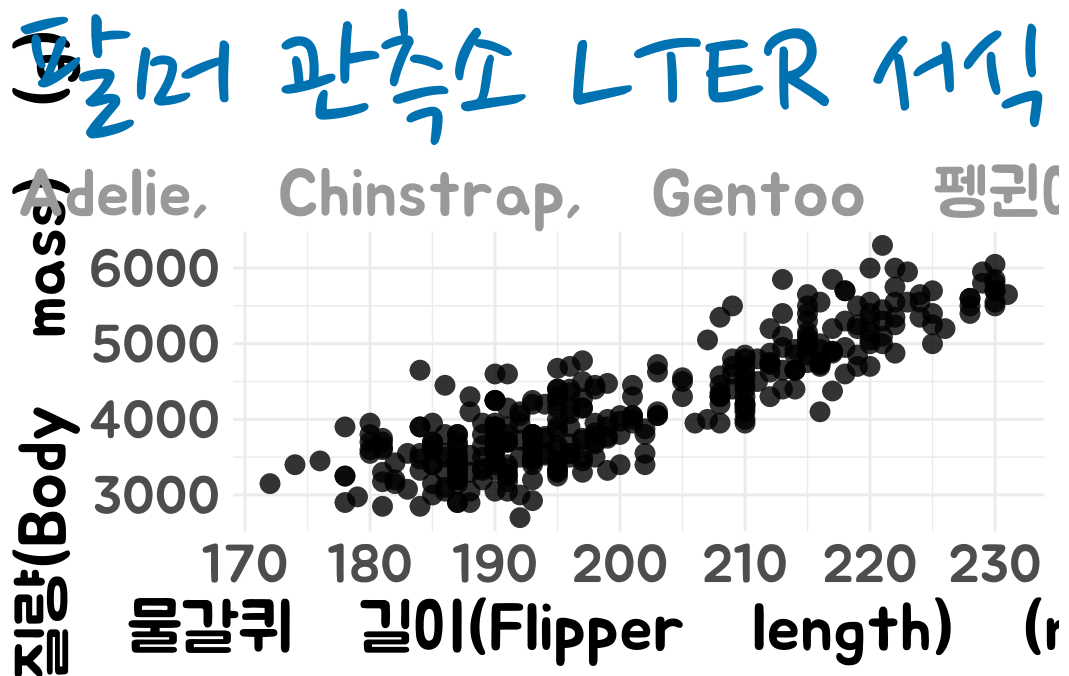
    margin = margin(t = 0, r = 0, b = 0, l = 0, unit = "mm")
  )
)

theme_set(theme_minimal() + theme_quarto)

mass_flipper <- ggplot(data = penguins,
                      aes(x = flipper_length_mm,
                          y = body_mass_g)) +
  geom_point(size = 3,
             alpha = 0.8) +
  labs(title = "팔머 관측소 LTER 서식 펭귄 크기",
       subtitle = "Adelie, Chinstrap, Gentoo 펭귄에 대한 물갈퀴 길이와 체질량",
       x = "물갈퀴 길이(Flipper length) (mm)",
       y = "체질량(Body mass) (g)")

mass_flipper

```



3.5 표

구글 폰트에서 다양한 글꼴을 표 제목, 표 칼럼명, 표 셀값에 다르게 적용하며 크기는 물론 색상과 굵기도 적용이 가능하다.

```
library(gt)

basic_theme <- function(data, ...){
  data %>%
    tab_options(
      column_labels.background.color = "blue",
      table.font.size = px(12),
      column_labels.font.size = px(20),
      row.stripping.background_color = "#9678b6",
      heading.align = "left",
      heading.title.font.size = px(30)
    )
}

palmerpenguins::penguins %>%
  select( 펭귄종=species, 섬=island, bill_length_mm, body_mass_g) %>%
  slice_sample(n = 10) %>%
  gt() %>%
  tab_header(title = md("팔머 펭귄 표본 **10 마리** 측정 정보")) %>%
  tab_source_note(source_note = "데이터: Kristen Gorman 박사와 남극 팔머 관측소(Palmer Station, Antarctic Peninsula)") %>%
  cols_label(bill_length_mm = "부리 길이 (mm)",
             body_mass_g = "체질량 (g)") %>%
  tab_style(
    style = cell_text(
      font = c(
        google_font(name = "Nanum Pen Script"),
        default_fonts()
      ),
      size = "xx-large",
      color = "gray35",
      weight = "bolder"
    ),
    locations = cells_body(columns = bill_length_mm)
  ) %>%
  # 체질량 칼럼 -----
  tab_style(
```

```

style = cell_text(
  font = c(
    google_font(name = "Black Han Sans"),
    default_fonts()
  ),
  size = "large",
  color = "blue",
  weight = "bolder"
),
locations = cells_body(columns = body_mass_g)
) %>%
# 표 Header 와 첫번째 칼럼 -----
tab_style(
  style = cell_text(
    font = google_font("Do Hyeon"),
    size = "large"
  ),
  locations = list(
    cells_column_labels(everything()),
    cells_body(columns = 1)
  )
) %>%
# 표 제목 -----
tab_style(
  style = cell_text(
    font = google_font("Dokdo"),
    align = "left",
    size = "xx-large"
  ),
  locations = cells_title("title")
) %>%
basic_theme()

```

팔머 펭귄 표본 10 마리 측정 정보

펭귄종	섬	부리 길이 (mm)	체질량 (g)
Gentoo	Biscoe	45.5	4200
Adelie	Torgersen	39.2	4675
Chinstrap	Dream	50.2	3800
Gentoo	Biscoe	49.8	5950
Adelie	Dream	40.3	4350

Adelie	Dream	39.5	3250
Gentoo	Biscoe	55.1	5850
Adelie	Dream	37.6	3300
Chinstrap	Dream	50.9	3550
Adelie	Biscoe	40.5	3200

데이터: Kristen Gorman 박사와 남극 팔머 관측소(Palmer Station, Antarctica LTER)에서 수집

R 마크다운 문서 작성에 필요한 글꼴 관련 정보

- [글꼴\(font\)](#)
- [구글 노토 글꼴\(Noto Fonts\)](#)
- [R 마크다운 폰트](#)

4 한글 PDF

4.1 출판 디지털 전환

200자 원고지 4장은 통상 A4 용지 한장에 해당한다. 만약 A4 용지 10장 보고서나 논문을 작성한다고 하면, 종이 200자 원고지 40장에 연필이나 볼펜을 가지고 작성하게 된다. 최종본을 얻기까지 빨간색으로 선생님이나 주변의 첨삭지도를 받아 최종본을 만들기까지 여러번의 반복과정을 거치게 된다.

최종 원고가 200자 원고지 40장에 담겨 탈고를 마치게 되면, 독자에게 다가갈 준비를 한다. 그림을 넣거나 아름다운 글씨체를 적용해 보고 종이 재질을 바꿔보기도 하고 영혼이 담긴 원고를 독자가 보기 좋고, 이해하기 쉽고, 때로는 감동과 재미를 줄 수 있도록 출판 단계에서 많은 사람들의 노력이 들어간다.

탈고를 마친 원고에 글씨체를 비롯한 디자인 작업이 완료되면 출판장비를 갖춘 출판소에서 인쇄장비를 거쳐 종이책 혹은 보고서가 완성되고, 이렇게 대량으로 출력되면 각 서점으로 배송되어 독자가 서점에서 구독을 하게 된다.

저자의 생각을 글로 표현하면 현재도 대부분 편집 및 디자인, 출판, 배송, 서점 등 각 단계에 사람이 관여하고 필요에 따라서는 컴퓨터가 작업을 지원하는데 사용되어 최종 독자에게 전달된다.

책과 보고서 제작하는 과정은 워드 프로세스를 기본으로 두고, 숫자 계산이 필요하면 엑셀, 이미지가 필요하면 그래픽 전문 소프트웨어, 정보 검색도 웹 브라우저를 사용한다. 글자, 단어, 문장, 문단, 장/절/항으로 생각의 단위를 서문, 본문, 결어 및 참고문헌, 주석, 그림, 그래프, 도표 등도 함께 체계적으로 작성해 나간다.

다양한 소프트웨어를 통해서 원고를 탈고하게 되면 서식, 문장 문단 모양, 색인, 참조문헌 등 보기 좋고 가독성 높은 형태로 외양을 입히는 과정을 거치게 되는데 이 과정도 소프트웨어가 핵심적인 역할을 하게 된다. 책과 보고서에 대한 디자인 작업이 완료되면 다양한 형태로 배포하기 위한 프로그래밍 과정을 거쳐 PDF, 전자책(ePUB), 웹(HTML), 출판책 형태로 독자에게 전달된다.

글을 써서 이를 독자에게 전달되는 과정은 과거에 전문작가, 디자인 전문가, 출판전문가, 서점운영자 등 수많은 전문가가 연관되어 있고 타자기, 디자인 기계, 인쇄기 등 수많은 기계가 필요했으며 이를 사업적으로 기술적으로 연결할 수 있는 다양한 통신 방법이 필요했다. 비전문가가 밖에서 보면 복잡하고 어려워 보이지만, 컴퓨팅 사고의 시각으로 보면 단순할 수 있다.

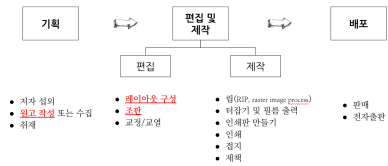


그림 4.2: 아날로그 출판

그림 4.1: PDF 작업흐름

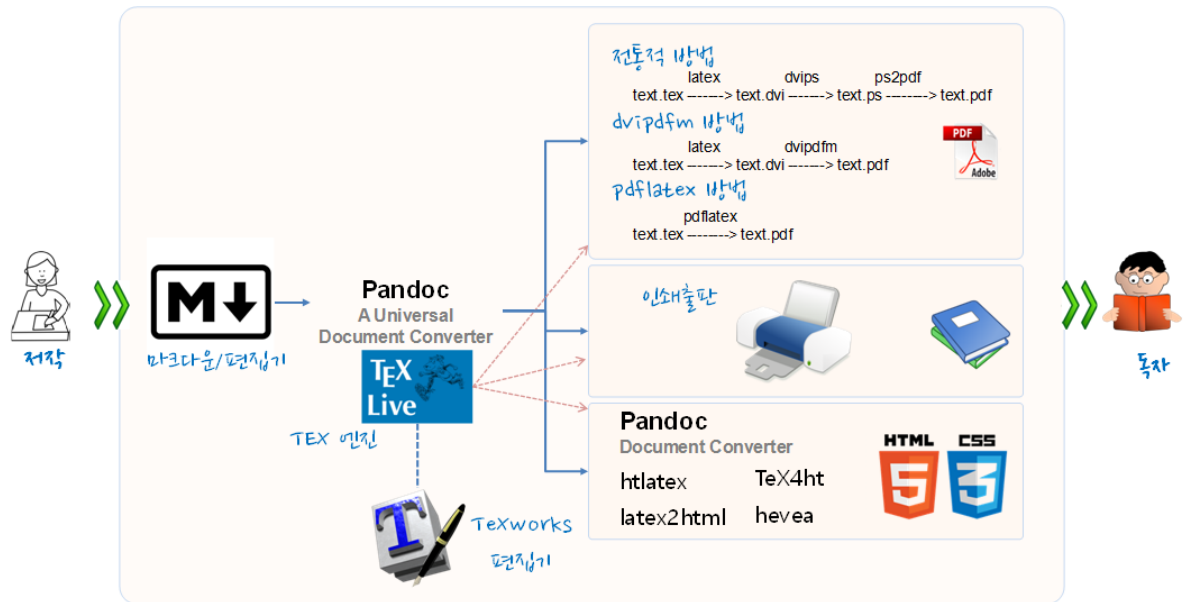


그림 4.3:

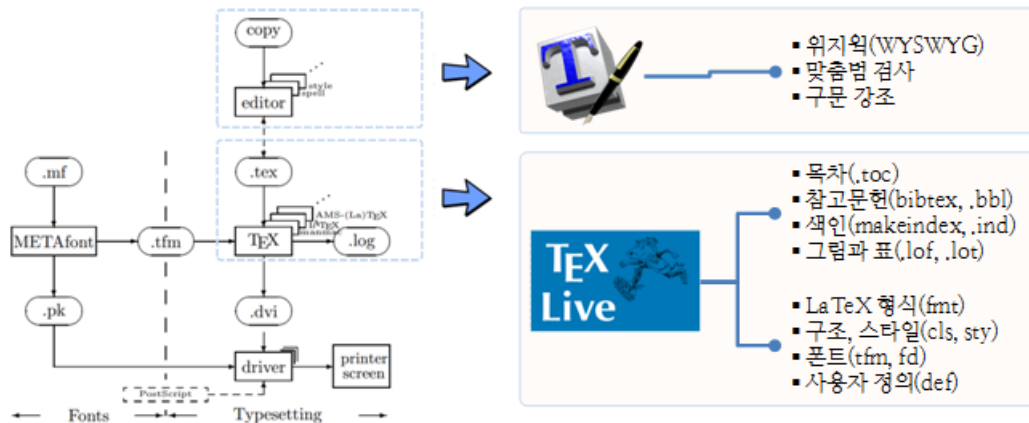
4.2 *LaTeX* 개요

미국 스탠포드 대학 크누스(Donald Knuth) 교수가 1977년에 만든 문서조판 프로그램을 *TeX*(“텍”이라고 읽음)이라고 하고, Leslie Lamport 교수가 만든 *TeX* 매크로 패키지를 *LaTeX*(“레이텍”이라고 읽음)이라고 한다. 수학과 교수였던 크누스 교수는 문서 식자체계를 통해 수식도 사용하기 쉽게 개발했고 METAFONT 폰트도 개발하여 양질의 글자체를 제작했다. 이후 레슬리 램포트가 *LaTeX*을 쉽게 쓸 수 있도록 다양한 매크로를 작성하여 공개했다. 이를 바탕으로 일반인도 고품질의 문서를 *LaTeX*으로 작성할 수 있게 되었다. 버클리 마이클 스피박(Michael Spivak) 교수는 수식을 쉽게 문서화시킬 수 있는 AMS-*TeX*을 매크로로 작성하여 공개하여 “The Joy of *TeX*” 책을 출간하기도 했다.

*pdf_{tex}*과 *e-TeX*이 합쳐져서 사실상(de factor) 표준 *TeX*이 되었고, *XeTeX*과 *LuaTeX*이 기존 *TeX*을 대체하면서 발전하고 특히, 한글 폰트 때문에 *XeTeX* [“지텍”이라고 읽음]을 한글처리를 위해서 많이 사용하고 있다.¹

*LaTeX*의 동작원리는 Salomon이 도식화한 것에 영감을 받아, Kees van der Laan²이 1994년 정리한 논문에 잘 정리되어 있다. 기본적으로 고품질 전자출판을 위해서 폰트와 더불어 *TeX*이 필요하고 이를 하나로 묶어서 고품질 출력물을 얻게된다. 저자가 편집기(editor)로 작업한 *.tex* 파일이 폰트(Fonts) 작업과정과 조판(Typesetting)과정이 하나로 합쳐져 출력물이 산출된다. *TeX* 기본 엔진을 지원하는 다양한 *TeX* 패키지, AMS-(La)*TeX*, *LaTeX*, *manmac*가 눈에 띈다.

문서에 들어가는 다양한 구성요소, 목차, 색인, 참고문헌, 그림과 표 등은 모듈화되어 관리되고, 폰트나 구조 스타일과 관련된 것도 별도로 구분되어 관리되고 있으며, *TeXWorks*같이 위치역을 지원하는 편집기는 구문강조 및 맞춤법 검사등 고품질 *LaTeX* 저작물 작성을 위해 저작자를 지원하고 있다.



기본적인 *LaTeX* 작동방식 및 주요 모듈에 대해 살펴본 후에 효과적인 *LaTeX* 문서를 위한 구성요소를 세부적으로 살펴보자.

¹Modern *LaTeX*

²What is *TeX* and *Metafont* all about?

1. **배포판**: 과거 다양한 배포판이 윈도, 리눅스, 맥에 사용되었으나 현재는 [TeX Live](#) 가 유일의 LaTeX 작업환경이 되었다. 물론 [MikTeX](#)이 독자적으로 유지되고 있으나 별 차이가 없다.
2. **폰트**: 과거 폰트 메트릭(tfm), 메타폰트(MetaFont)를 사용하였으나, Type 1 을 걸쳐 트루타입과 오픈타입을 사용하도록 확대되었다.
 - 폰트를 식품으로 표현하면 밥과 같이 거의 항상 평타를 칠 수 있는 폰트를 선택하는 것이 중요하다. 물론 특식 필요한 경우 그에 맞게 적절한 폰트를 선택하는 것이 중요하다.
 - 폰트는 Serif와 San Serif를 한쪽으로 삼아 한글로 보면 명조체(삐침이 있는 글꼴)과 고딕체를 동일한 폰트로 선택하는게 전체적으로 균형이 맞는다. 예를 들어, 나눔고딕-나눔명조, KoPub돋움-KoPub바탕, 함초롬돋움-함초롬바탕 조합을 선택한다.
 - 글자 크기는 10~12 포인트를 선택하는데 이유는 장시간 글을 읽게 되도 피로감을 줄일 수 있다.
3. **그래픽**: TeX/LaTeX은 기본적으로 그림은 TeX/LaTeX의 영역이 아닌 것으로 간주하고 있다. dvips를 사용하여 과거 EPS그림을 주로 사용했으나, 현재는 .jpg, .png, .pdf도 잘 처리한다.
4. **문자와 입력**: 컴퓨터가 미국에서 개발되어 발전되어 주로 ASCII 코드가 주를 이루었으나 2바이트 문자권인 한중일도 컴퓨터에서 자유로이 문자를 표현하고 입력할 수 있는 방식으로 UTF-8을 권장하고 있다. 특히, `\usepackage[utf8x]{inputenc}` 사용이 필요한 경우가 많이 있다.
5. **색인과 참고문헌**: `makeindex`와 `bibtex`가 색인과 참고문헌 처리를 위한 표준이지만, 유니코드를 좀더 잘 처리하고자 `xindy`와 `biber`도 주목받고 있다.
6. **수식**: 수학표현관련 미국수학회에서 발전시킨 `amsmath`가 표준이다.
7. **출력**: 과거 .dvi가 기본이었으나, 현재는 .pdf가 기본이고, 웹을 위한 다양한 .html 출력 생성도 지원한다. 즉, 과거에는 책 형태가 최종문서 산출물의 기본이었지만, 현재는 그리고 앞으로는 웹이 과거 책 형태의 위치를 차지하여 기본이 되고 책은 보조적인 역할로 남을 것으로 보인다.

4.3 Quarto PDF

PDF는 기술적, 학술적, 사업적으로 혁신적인 문서 양식이다. PDF가 개발된 것은 오랜 역사를 자랑하지만 아래한글, 워드퍼펙트, MS 워드, 리브레 워드 등에 가려 그 진가를 발휘하지 못했다. 하지만 이제 PDF는 문서 출판의 최고봉이라는데 이의를 제기하는 사람은 없을 것이다.

PDF를 통해 아도비(Adobe) 회사는 엄청난 성공을 이어나가고 있고, 과학기술 저작에도 필수적인 파일 형태라 PDF에 대한 기본소양은 인공지능 시대를 살고 있는 시민으로 누구나 갖추어야 하는 소양이다.

4.3.1 *LaTeX* 엔진

보통 TexLive에 기반을 둔 `tinytex` 를 사용하는데 한글을 사용하는 입장에서는 `kotex`를 설치하는 방법도 모색이 필요하다. 유니코드 문자를 처리하기 위해서 `xelatex`이 Quarto *LaTeX* 기본엔진이다. 달리 특정하지 않게 되면 `xelatex`을 통해 CJK 문자가 포함된 PDF 파일 제작이 가능하다.

```
quarto install tool tinytex
```

`kotex` 이 갖는 장점은 한글관련 이슈가 될만한 거의 모든 *LaTeX* 패키지와 기본 한글글꼴도 내장하고 있다는 점이다. 하지만, `tidytex`를 기본 *LaTeX* 엔진으로 사용하는 경우 빠진 패키지로 인해 후속 작업이 어려운 경우 `tinytex::parse_install()` 함수를 사용하게 되면 다음과 같이 오류난 텍스트를 붙여넣게 되면 자동으로 설치해준다.

```
tinytex::parse_install(  
  text = "! LaTeX Error: File `titlepic.sty' not found."  
)
```

4.3.2 YAML 파일

문서를 구성하는 기본사항을 적어둔다. PDF 문서를 저작하는 것이라 *LaTeX* 에 대한 기본 지식은 Quarto PDF 문서를 생성하는데 많은 도움이 된다. 글꼴 지정관련하여 로컬 컴퓨터에 폰트 디렉토리에 관련 글꼴 정보가 운영체제와 연결되어 있어야 한다.

Pandoc 문서에 메타데이터 [variables for LaTeX](#)을 참조하여 필요한 상세설정작업을 수행한다.

```
---  
title: "한글 PDF"  
format:  
  pdf:  
    documentclass: report  
    papersize: a4  
    toc: true  
    toc-depth: 2  
    number-sections: true  
    colorlinks: true  
    include-in-header:  
      text: |  
        \usepackage{pdfpages}  
  bibliography: bibliography.bib  
  link-citations: yes
```

```

csl: apa-single-spaced.csl
mainfont: "NanumMyeongjo"
sansfont: "NanumGothic"
monofont: "D2Coding"
---
```

Quarto 가 문서 변환 핵심 엔진으로 사용하고 있는 Pandoc 은 *LaTeX* 외에도 ConTeXt, roff ms, HTML (via wkhtmltopdf) 을 사용해서 PDF 파일을 제작할 수 있다. 기존 Pandoc을 이런 방식으로 PDF 를 제작하신 분은 다음 내용을 건너뛰는 것도 좋을 듯 싶다.

4.3.3 저작

마크다운으로 이미지, 표, 텍스트, 서식, 수식 등 다양한 저작사항을 .qmd 파일에 담아내면 *LaTeX* 엔진이 PDF로 변환작업을 수행해준다.

4.3.3.1 이미지

*LaTeX*에 이미지도 삽입가능하다.

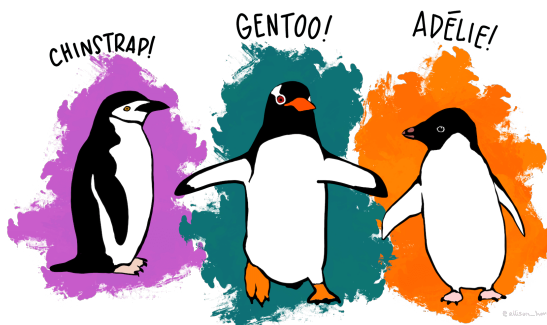


그림 4.4: Allison Horst 박사가 그린 펭귄 3종 도식화

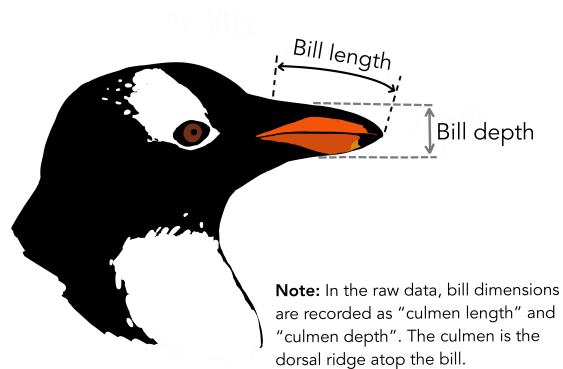


그림 4.5: 펭귄 부리(bill) 변수 설명

4.3.3.2 R 코드

*LaTeX*에 R 파이썬 코드를 통해 계산된 결과를 그래프, 표, 요약통계량 등으로 뽑아내 삽입가능하다.

```

mass_flipper <- ggplot(data = penguins,
  aes(x = flipper_length_mm,
```

```

y = body_mass_g)) +
geom_point(size = 3,
            alpha = 0.8) +
theme_minimal() +
labs(title = "팔머 관측소 LTER 서식 펭귄 크기",
      subtitle = "Adelie, Chinstrap, Gentoo 펭귄에 대한 물갈퀴 길이와 체질량",
      x = "물갈퀴 길이(Flipper length) (mm)",
      y = "체질량(Body mass) (g)") +
theme_minimal()

mass_flipper

```

4.3.3.3 표

*LaTeX*에 *LaTeX* 표를 넣을 수도 있고 R 혹은 파이썬에서 나온 표를 PDF 파일에 넣을 수도 있다.

4.3.3.3.1 LaTeX 표

연령	빈도수
18-25	15
26-35	33
36-45	22

4.3.3.3.2 gt 표

```

knitr::kable(
  mtcars[1:6, 1:10]
)

```

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear
Mazda RX4	21.0	6	160	110	3.90	2.620	16.46	0	1	4
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875	17.02	0	1	4
Datsun 710	22.8	4	108	93	3.85	2.320	18.61	1	1	4
Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3
Hornet Sportabout	18.7	8	360	175	3.15	3.440	17.02	0	0	3
Valiant	18.1	6	225	105	2.76	3.460	20.22	1	0	3

4.3.3.4 데이터

🔥 데이터 수집에 대해 궁금하신 분은 클릭하여 추가 학습하세요

팔머펭귄 데이터는 Kristen Gorman 박사와 남극 팔머 관측소(Palmer Station, Antarctica LTER)에서 수집되어 공공에 이용가능하게 공개되었습니다.

{palmerpenguins} 패키지 (Horst, Hill, and Gorman 2020)를 통해 R에서 손쉽게 이용가능하게 되었고 원데이터는 Gorman, Williams, and Fraser (2014) 에서 최초 공개되었다.

4.3.3.5 PDF 겉면

fitpaper *LaTeX* 패키지를 사용해서 책 첫장(\frontmatter)과 마지막장(\backmatter) 겉면을 장식할 수 있다.

```
\includepdf[fitpaper]{book/canva-backmatter}
```

4.4 한글 글꼴

Quarto PDF Engines에서 *LaTeX* 엔진을 별도 다룰 정도로 *LaTeX* 엔진은 중요하다. Quarto에서는 TinyTeX를 기반엔진으로 사용되는데 TeX Live 배포판에 기반하고 Pandoc 에서 가장 많이 사용되는 대략 200 여개 패키지를 포함시켜 사용성은 좋게하면서 크기는 대폭 줄였다.

.qmd 파일 YAML 헤더에 로컬 컴퓨터에 등록된 글꼴을 지정하는 방식도 있다.

```
mainfont: "NanumMyeongjo"  
sansfont: "NanumGothic"  
monofont: "D2Coding"  
CJKmainfont: "NanumGothic"
```

다른 방식은 글꼴 관련 사항은 packages.tex *TeX* 파일에 지정하는 것이다.

```
format:  
  pdf:  
    include-in-header:  
      - packages.tex  
      - macros.tex  
    include-before-body: before.tex
```

예를 들어 packages.tex 에 한글 글꼴을 다음과 같이 지정한다.


```
% 한글 폰트 -----  
\setmainhangulfont{NanumMyeongjo} % 나눔명조  
\setsanshangulfont{NanumGothic}   % 나눔고딕  
\setmonohangulfont{D2Coding}      % D2코딩
```

i 노트

이광춘, “글을 쓰는 것은 소프트웨어 개발이다!”, 강서양천신문 특집기고(’15년 9월) 와 LaTeX, PDF, 문서, 그리고 사고에서 관련 내용을 발췌한 것임을 밝혀둡니다. 그리고, [Quarto](#) PDF 문서를 많이 참고 했습니다.

Part III

빛에듀(Bit-Edu)

5 빛에듀

5.1 데이터

5.2 기술통계

참고문헌

- Ghosh, Debashis, Tulika Dube, and Shivaprasad Adamane. 2010. “Script Recognition—a Review.” *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32 (December): 2142–61. <https://doi.org/10.1109/TPAMI.2010.30>.
- Gorman, Kristen B., Tony D. Williams, and William R. Fraser. 2014. “Ecological Sexual Dimorphism and Environmental Variability Within a Community of Antarctic Penguins (Genus *Pygoscelis*).” *PloS One* 9 (3): e90081. <https://doi.org/10.1371/journal.pone.0090081>.
- Horst, Allison Marie, Alison Presmanes Hill, and Kristen B Gorman. 2020. Palmerpenguins: Palmer Archipelago (Antarctica) Penguin Data. <https://allisonhorst.github.io/palmerpenguins/>.