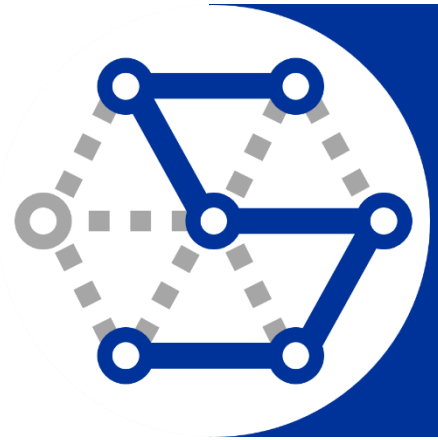

멀티미디어
Multimedia

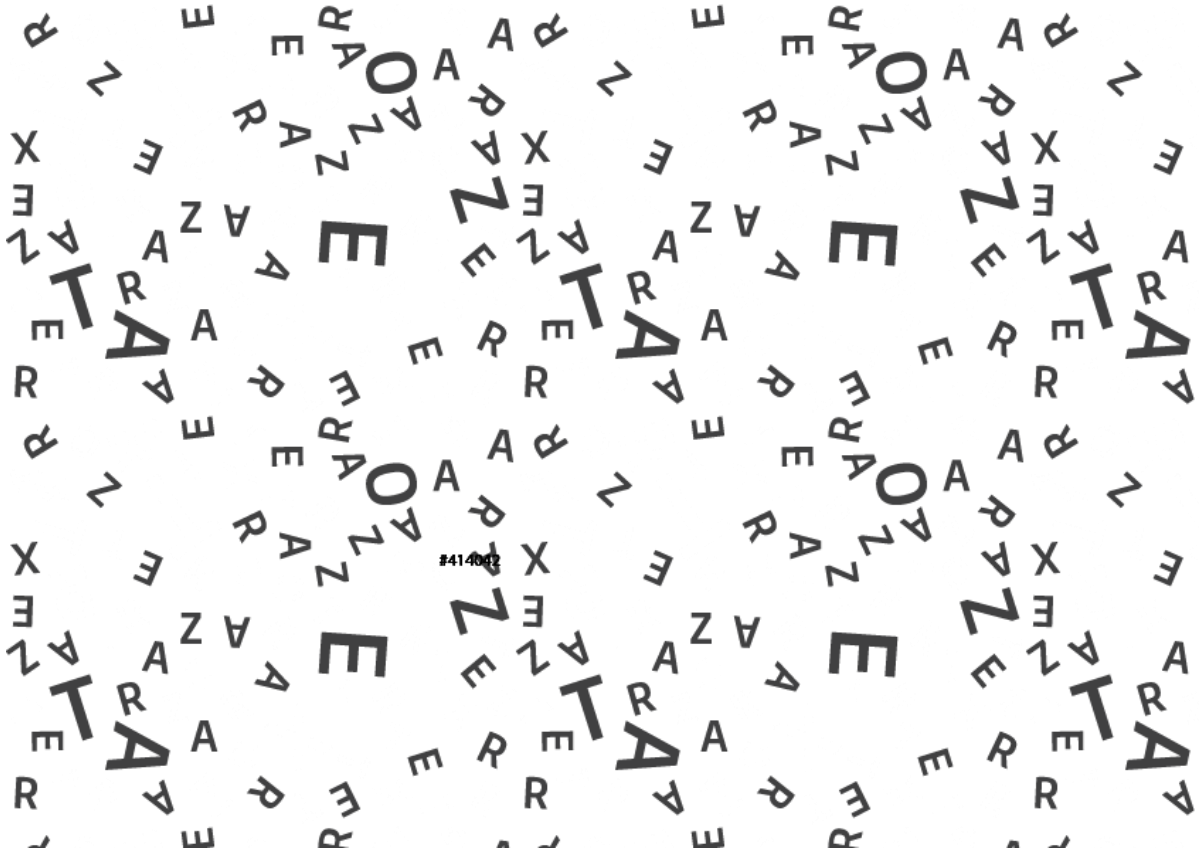

04
텍스트
suanlab
수안 컴퓨터 연구소
suan computer laboratory



1 텍스트의 개요

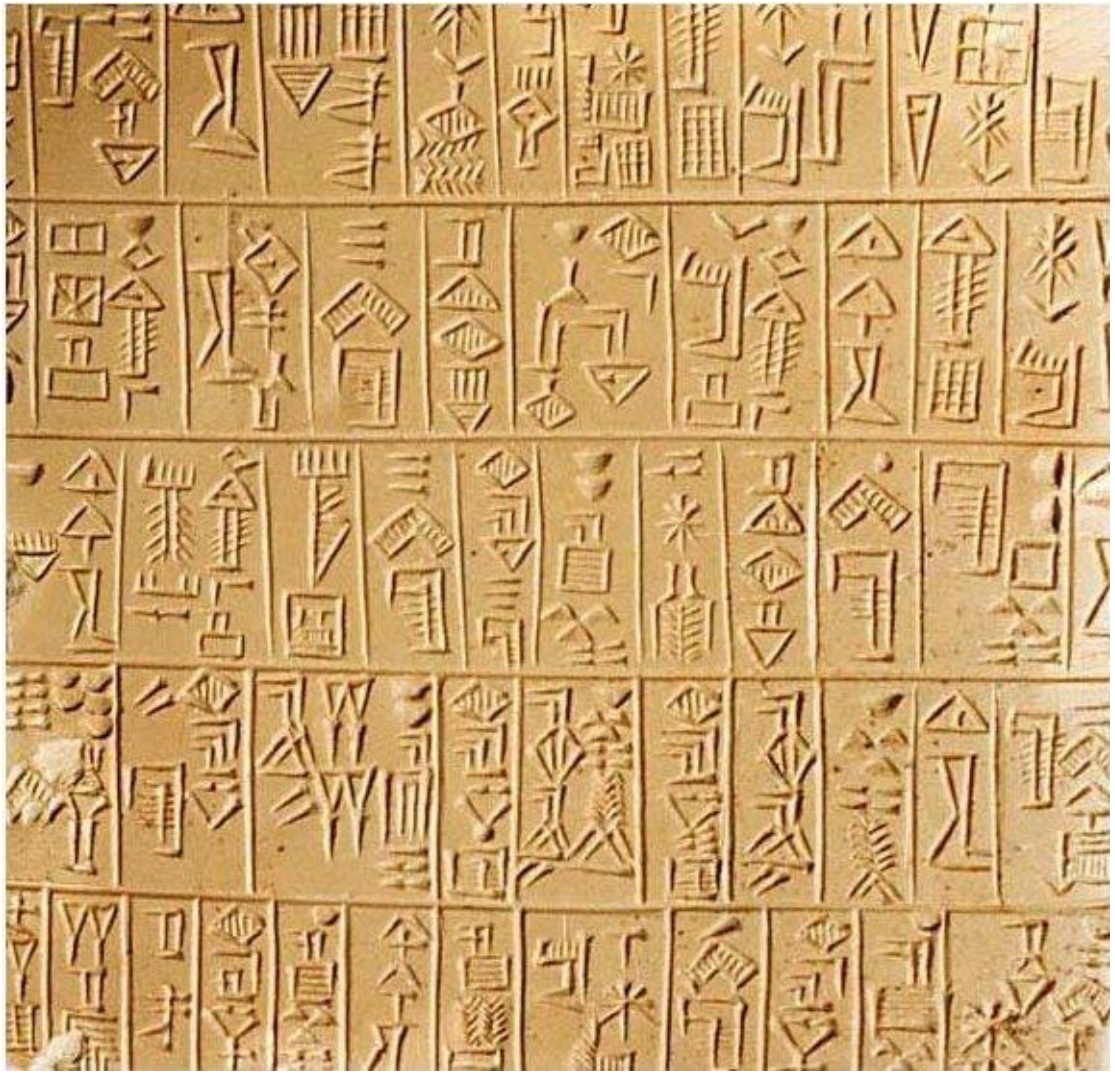
텍스트

- 여러 문장이 모여 만들어진 문장의 집합
- 특정한 의도에 따라 문자를 사용하여 작성된 문서
- 콘텐츠에 포함되는 멀티미디어 데이터 중 가장 많이 사용
- 다른 종류의 멀티미디어 데이터보다 기억 용량을 적게 차지
- 텍스트를 사용하여 문서를 작성하기 위해서는 해당 언어를 지원하는 도구를 사용해야 함



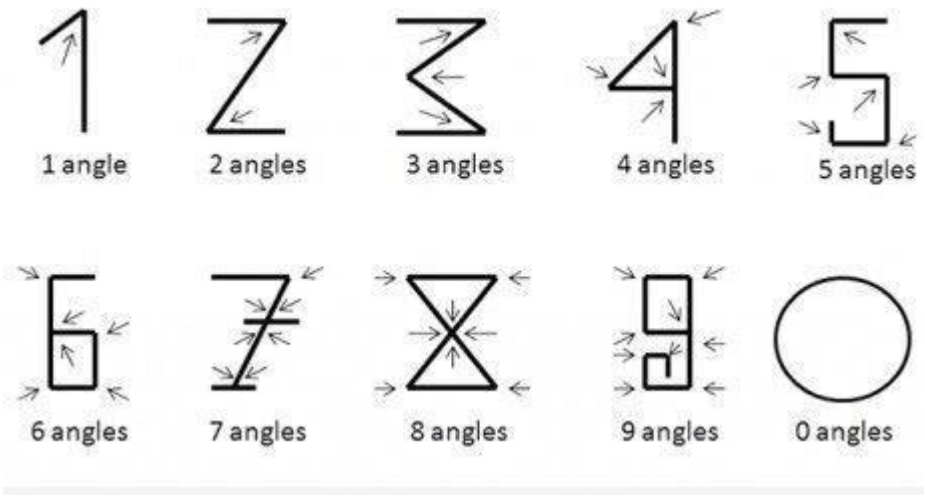
쐐기 문자

- 쐐기 문자(cuneiform) 또는 설형 문자(楔形文字)는 수메르인들이 기원전 3000년경부터 사용했던 상형문자로, 현재 알려진 것 중 가장 최초의 문자
- 시간이 지나면서 상형 문자적인 요소는 줄어들고 점점 추상화됨
- 쐐기문자는 점토판에 썼으며, 철필(스타일러스)이라고 부르는 갈대 가지로 만들었음
- 수메르 문자는 아카드어, 에블라어, 엘람어, 히타이트어(그리고 루비아어), 후르리어에도 쓰였으며, 고대 페르시아어와 우가리트 문자에도 영향을 미침



https://ko.wikipedia.org/wiki/쐐기_문자

아라비아 숫자



아라비아 숫자	1	2	3	4	5	6	10	50	100	500	1000
바빌로니아 숫자	𒁹	𒁹𒁹	𒁹𒁹𒁹	𒁹𒁹 𒁹𒁹	𒁹𒁹𒁹 𒁹𒁹	𒁹𒁹𒁹 𒁹𒁹𒁹	𒌋	𒐐	𒐖	𒐖𒁹𒁹	𒐖𒐖𒐖
이집트 숫자	𓏺	𓏺𓏺	𓏺𓏺𓏺	𓏺𓏺𓏺𓏺	𓏺𓏺𓏺𓏺𓏺	𓏺𓏺𓏺𓏺𓏺𓏺	𓎆	𓆐	𓍢	𓍢𓍢𓍢 𓍢𓍢	𓆼
로마 숫자	I	II	III	IV	V	VI	X	L	C	D	M
한자 숫자	一	二	三	四	五	六	十	五十	百	五百	千

https://img.khan.co.kr/news/2008/01/15/8a15k04p.jpg

- 아라비아 숫자는 위치 기수법에 따른 십진법으로 수를 표시하는 인도-아라비아 수체계에서 사용되는 열 개의 숫자 (1, 2, 3, 4, 5, 6, 7, 8, 9, 0)
- 아라비아 숫자는 오늘날 세계에서 가장 널리 사용되는 숫자 표현 기호
- 국제단위계는 아라비아 숫자를 사용
- 0은 고대 인도 수학에서 창안되었고 아라비아 숫자의 정립 과정에서 위치 기수법을 위해 도입

https://ko.wikipedia.org/wiki/아라비아_숫자

종이

- 문자를 글로 남겨 두기 위해 고대 이집트에서는 양피지(羊皮紙)가, 아시아에서는 얇은 죽편(竹片)이 사용
- 고대 이집트(B.C.3000년경)에서는 파피루스(papyrus)라고 하는 풀(草)의 섬유로 종이와 비슷한 것을 만들었는데, 이것이 오늘날 영어의 'paper'의 어원(語源)이 됨
- 현재 사용되고 있는 종이(식물에서 셀룰로오스를 뽑아 내 이것을 체 같은 것으로 걸러서 만든 것)를 처음으로 만든 사람은 중국의 채륜(蔡倫, 105년)인데, 그는 삼·아마(亞麻) 등에서 섬유를 분리하여 이것을 얇은 막상(膜狀)으로 걸러서 떠내어 종이를 만들었음

https://www.thedailypost.kr/news/articleView.html?idxno=70234

https://ko.wikipedia.org/wiki/종이의_역사

인쇄술

http://newsteacher.chosun.com/site/data/img_dir/2017/11/08/2017110800381_1.jpg

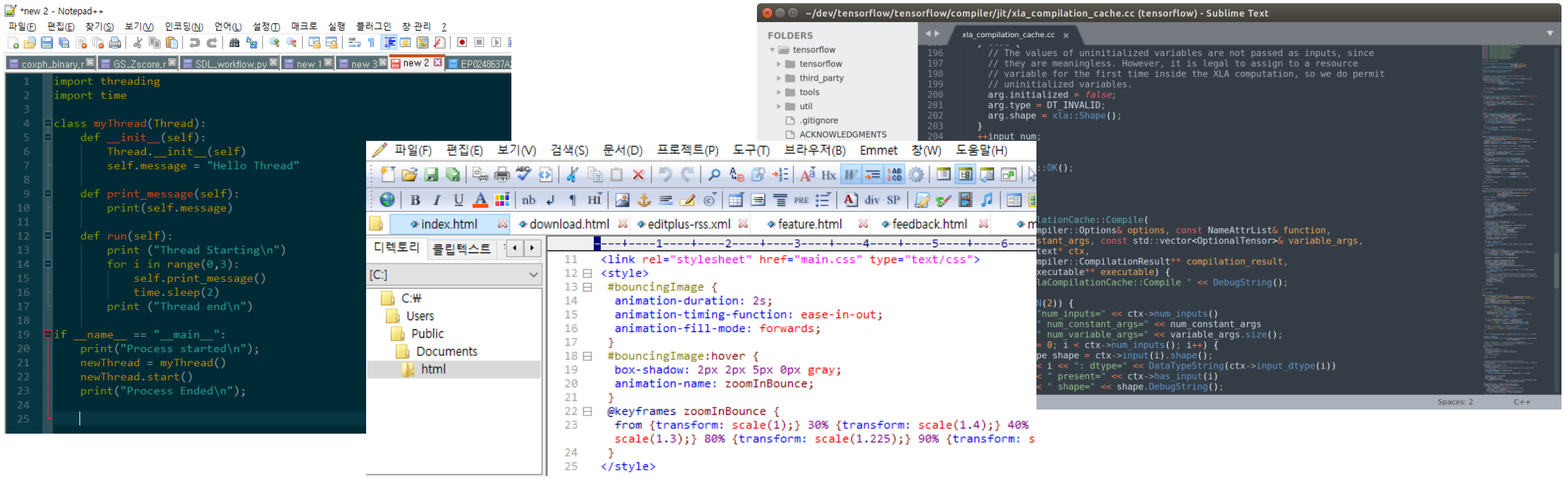
- 요하네스 겐스플라이슈 추르 라덴 춤 구텐베르크(Johannes Gensfleisch zur Laden zum Gutenberg, 1398년경 ~ 1468년 2월 3일)는 약 1440년 경에 금속 활판 인쇄술을 사용한 독일의 금(金) 세공업자
- 본명은 요하네스 겐스플라이슈(Johannes Gensfleisch)이고, 구텐베르크는 통칭
- 구텐베르크의 업적은 활자 설계, 활자 대량 생산 기술을 유럽에 전파한 것
- 그의 진정한 업적은 이런 기술과 유성 잉크, 목판 인쇄기 사용을 결합시켰다는 점
- 그는 활자 제작 재료로 합금을 사용하고, 활자 제작 방식으로 주조를 채용

https://ko.wikipedia.org/wiki/요하네스_구텐베르크

Project Gutenberg: https://www.gutenberg.org/

문서편집기

- 문서를 작성할 때 사용하는 프로그램
- 입력과 수정 기능이 있지만 단순하여 문서 작성에 많은 제한
- 프로그램을 작성하거나 웹 문서 제작에 필요한 소스코드를 입력할 때 주로 사용



워드 프로세서

- 문서의 작성, 편집, 저장 및 인쇄할 때 사용하는 하드웨어, 소프트웨어를 정의하는 용어 (대부분 소프트웨어)
- 문서를 보다 손쉽게 작성할 수 있도록 다양한 편의 기능을 제공
- 문서편집기보다 기능이 다양 (글자의 속성 변경, 이미지나 도표 등을 추가)
- 문서의 호환이 완벽하게 이루어지지 않는다는 단점이 있었으나 개방형 문서 표준 (ODF)의 도입으로 해결됨
- 국내 문서 작성 시장은 MS 워드와 한글이 양분

Q. 내가 사용하는 워드 프로세서는?

VS

2 텍스트의 표현

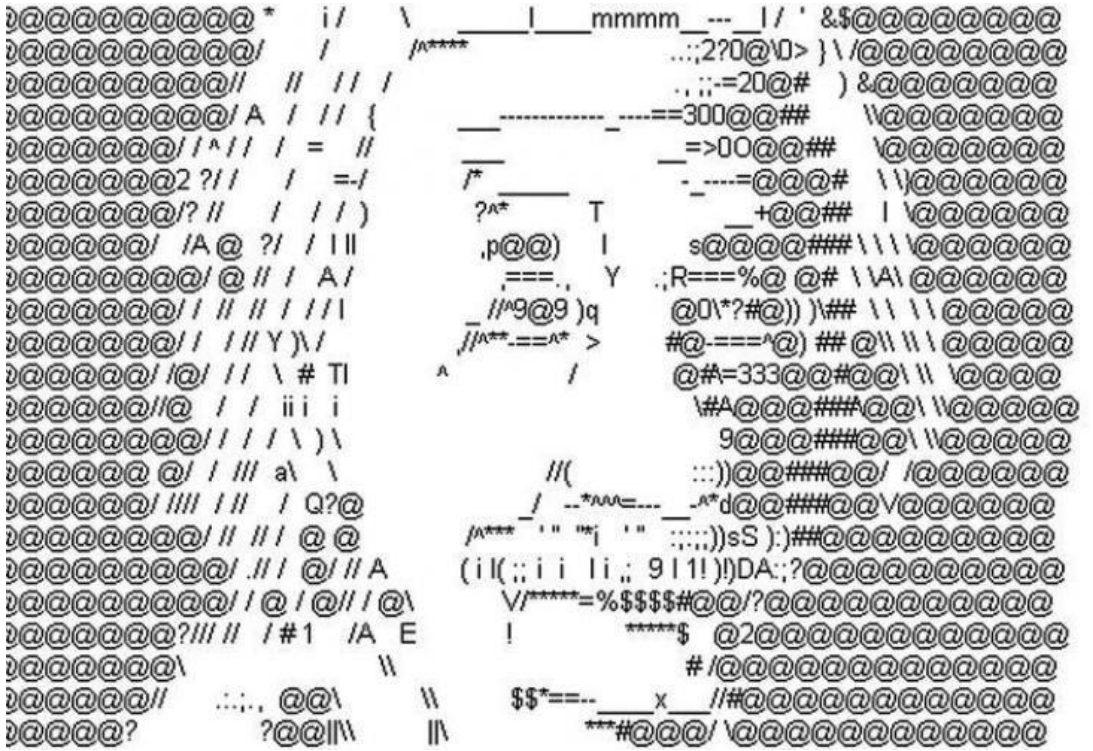
코드시스템

- 컴퓨터에서 문자를 사용하기 위해 약속된 이진코드를 일정한 규칙에 따라 각 문자에 할당하는 것
- 각각의 문자를 컴퓨터에 저장하고 컴퓨터 내부에서 문자들을 구분하여 사용하기 위해 사용
- 컴퓨터 내부에서 모든 문자는 이진코드로 인코딩
- 인코딩: 컴퓨터에 저장하거나 통신에 사용할 목적으로 문자나 기호를 다른 형식으로 변환하는 방식
- 1963년 아스키 코드, 1964년 EBCDIC는 과거에 표준으로 제정되어 사용됨
- 유니코드: 1995년 표준으로 제정, 각 나라의 코드 시스템을 하나로 통합한 것으로, 현재까지 사용

A	■□■■	■■■□	N	■□■■	□□□■
B	■□■■	■■□■	O	■□■■	□□□□
C	■□■■	■■□□	P	■□■□	■■■■
D	■□■■	■□■■	Q	■□■□	■■■□
E	■□■■	■□■□	R	■□■□	■■□■
F	■□■■	■□□■	S	■□■□	■■□□
G	■□■■	■□□□	T	■□■□	■□■■
H	■□■■	□■■■	U	■□■□	■□■□
I	■□■■	□■■□	V	■□■□	■□□■
J	■□■■	□■□■	W	■□■□	■□□□
K	■□■■	□■□□	X	■□■□	□■■■
L	■□■■	□□■■	Y	■□■□	□■■□
M	■□■■	□□■□	Z	■□■□	□■□■

아스키 코드(ASCII)

- 미국 표준 협회에서 만든 가장 기본적이고 세계적으로 널리 사용되고 있는 코드 체계
- 데이터 처리 및 통신시스템 에서 상호 간의 정보 교환용으로 사용
- 한 문자를 표현하기 위해 7비트를 사용 (문자의 수는 2^7인 128개)



아스키 코드(ASCII)

10진수	16진수	문자	10진수	16진수	문자	10진수	16진수	문자	10진수	16진수	문자	10진수	16진수	문자	10진수	16진수	문자	10진수	16진수	문자	10진수	16진수	문자
0	0x00	NULL	16	0x10	DLE	32	0x20	sp	48	0x30	0	64	0x40	@	80	0x50	P	96	0x60	`	112	0x70	p
1	0x01	SOH	17	0x11	DC1	33	0x21	!	49	0x31	1	65	0x41	A	81	0x51	Q	97	0x61	a	113	0x71	q
2	0x02	STX	18	0x12	DC2	34	0x22	"	50	0x32	2	66	0x42	B	82	0x52	R	98	0x62	b	114	0x72	r
3	0x03	ETX	19	0x13	DC3	35	0x23	#	51	0x33	3	67	0x43	C	83	0x53	S	99	0x63	c	115	0x73	s
4	0x04	EOT	20	0x14	DC4	36	0x24	$	52	0x34	4	68	0x44	D	84	0x54	T	100	0x64	d	116	0x74	t
5	0x05	ENQ	21	0x15	NAK	37	0x25	%	53	0x35	5	69	0x45	E	85	0x55	U	101	0x65	e	117	0x75	u
6	0x06	ACK	22	0x16	SYN	38	0x26	&	54	0x36	6	70	0x46	F	86	0x56	V	102	0x66	f	118	0x76	v
7	0x07	BEL	23	0x17	ETB	39	0x27	'	55	0x37	7	71	0x47	G	87	0x57	W	103	0x67	g	119	0x77	w
8	0x08	BS	24	0x18	CAN	40	0x28	(	56	0x38	8	72	0x48	H	88	0x58	X	104	0x68	h	120	0x78	x
9	0x09	HT	25	0x19	EM	41	0x29	)	57	0x39	9	73	0x49	I	89	0x59	Y	105	0x69	i	121	0x79	y
10	0x0A	₩n	26	0x1A	SUB	42	0x2A	*	58	0x3A	:	74	0x4A	J	90	0x5A	Z	106	0x6A	j	122	0x7A	z
11	0x0B	VT	27	0x1B	ESC	43	0x2B	+	59	0x3B	;	75	0x4B	K	91	0x5B	[	107	0x6B	k	123	0x7B	{
12	0x0C	FF	28	0x1C	FS	44	0x2C	,	60	0x3C	<	76	0x4C	L	92	0x5C	₩	108	0x6C	l	124	0x7C	|
13	0x0D	₩r	29	0x1D	GS	45	0x2D	-	61	0x3D	=	77	0x4D	M	93	0x5D	]	109	0x6D	m	125	0x7D	}
14	0x0E	SO	30	0x1E	RS	46	0x2E	.	62	0x3E	>	78	0x4E	N	94	0x5E	^	110	0x6E	n	126	0x7E	~
15	0x0F	SI	31	0x1F	US	47	0x2F	/	63	0x3F	?	79	0x4F	O	95	0x5F	_	111	0x6F	o	127	0x7F	DEL

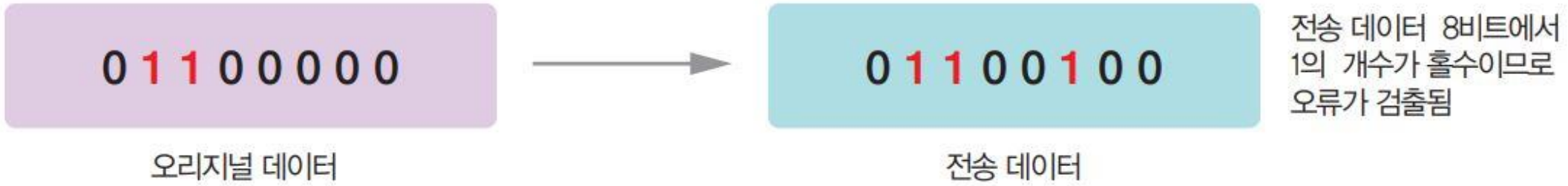
아스키 코드(ASCII)

- 컴퓨터 환경에서 일반적으로 사용하는 데이터 단위는 8비트이므로 아스키 코드를 8비트로 구성하여 사용
- 공백으로 남는 나머지 1비트는 패리티 비트로 활용
- 패리티 비트: 오류 검출을 목적으로 사용하는 비트, 짝수 패리티 비트와 홀수 패리티 비트가 있음

오리지널 데이터	짝수 패리티	홀수 패리티
00000000	0	1
01011011	1	0
01010101	0	1
11111111	0	1
10000000	1	0
01001001	1	0

아스키 코드(ASCII)

- 오류검출 예
  - 짝수 패리티비트에서 아스키 코드가 '1100000'인 경우
  - 7비트 코드 안에 있는 1의 개수가 짝수이므로 패리티 비트는 '0'이 되어 전체 코드는 '01100000'으로 구성
  - 한 비트의 에러가 발생하면 오류 검출이 가능



- 패리티 비트를 이용하면 오류를 쉽게 검출할 수 있지만 해결은 불가능
- 한 비트 오류에 대해서만 검출할 수 있고 두 비트 이상은 해결할 수 없음

EBCDIC

- IBM에서 서버급의 중대형 컴퓨터에 사용하기 위해 개발
- 코드를 구성하기 위하여 8비트를 사용하지만 아스키 코드와는 전혀 다름 (2^8인 256개의 문자)
- 8개의 비트는 2개의 영역으로 나누어 상위 4비트와 하위 4비트로 구분
- 상위 4비트는 존 비트 / 하위 4비트는 디지트 비트 또는 뉴메릭 비트
- 영역에서 읽은 비트는 코드 테이블에 대응시켜 해당 코드가 정의하는 문자를 알아냄
- EBCDIC에서는 실제 256개의 문자를 모두 사용하지 않고 150개 정도의 코드만 사용

EBCDIC

EBCDIC character codes

2nd hex digit \ 1st hex digit	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	DEL	DS		SP	&	•									0
1	SOH	DC1	SOS				/		a	j	0	0	A	J	0	1
2	STX	DC2	FS	SYN					b	k	s	0	B	K	S	2
3	ETX	TM							c	l	t	0	C	L	T	3
4	PF	RES	BYP	PN					d	m	u	0	D	M	U	4
5	HT	NL	LF	RS					e	n	v	0	E	N	V	5
6	LC	BS	ETB	UC					f	o	w	0	F	O	W	6
7	DEL	IL	ESC	EOT					g	p	x	0	G	P	X	7
8		CAN							h	q	y	0	H	Q	Y	8
9		EM							i	r	z	`	I	R	Z	9
A	SMM	CC	SM		C CENT	!		:								
B	VT	CUI	CU2	CU3		$	,	#								
C	FF	IFS		DC4	<	*	%	@								
D	CR	IGS	EMQ	NAK	(	)	_	'								
E	SO	IRS	ACK		+	;	>	=								
F	SI	IUS	BEL	SUB	|	--	?	"								

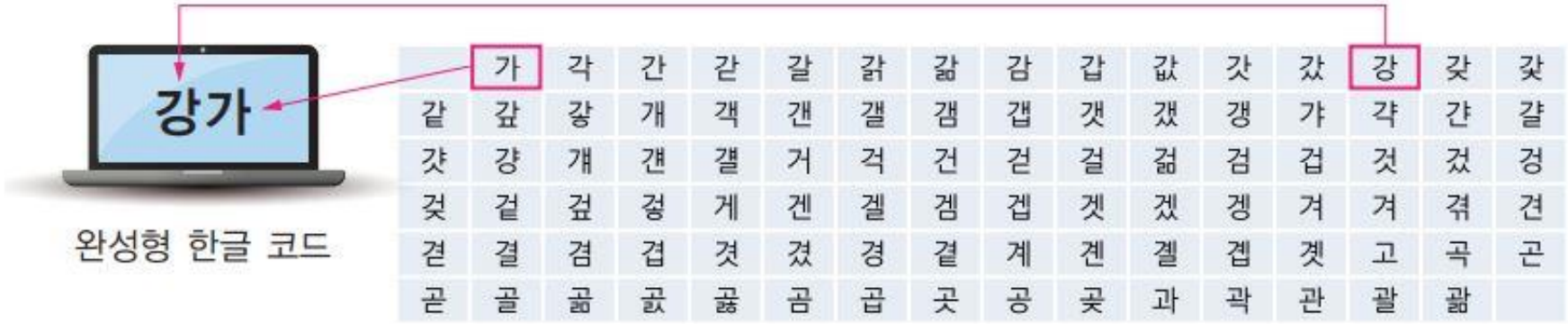
(a) 코드 테이블

(b) 코드 구조

한글 코드

- 한글의 경우 8비트 코드 체계로 표현할 수 없으므로 별도의 코드시스템이 필요
- 글자를 초성, 중성, 종성으로 구분하여 표현하는 방법이 개발

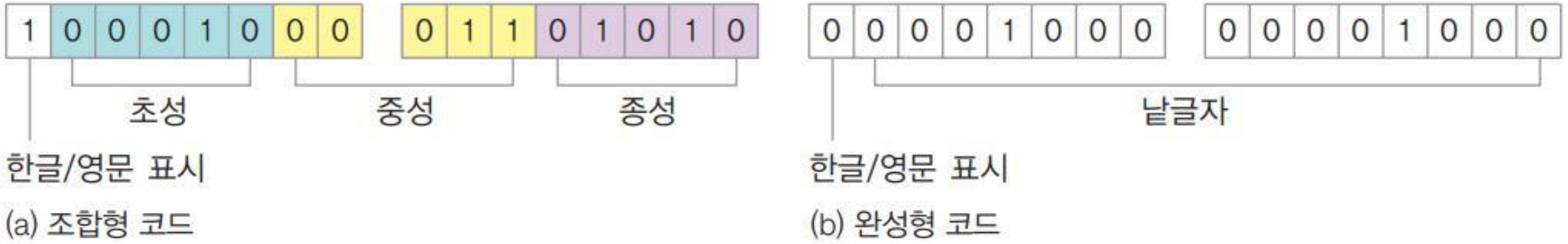
조합형 한글 코드



완성형 한글 코드

한글 코드

- 조합형 코드: 글자를 표현할 때 초성, 중성, 종성에 대한 각각의 코드를 저장하고 실제로 출력할 때 조합
- 완성형 코드: 완성된 한 글자 한 글자에 초점을 두는 방식으로 초성, 중성, 종성의 낱글자들을 미리 조합하여 글자들을 모두 코드로 완성

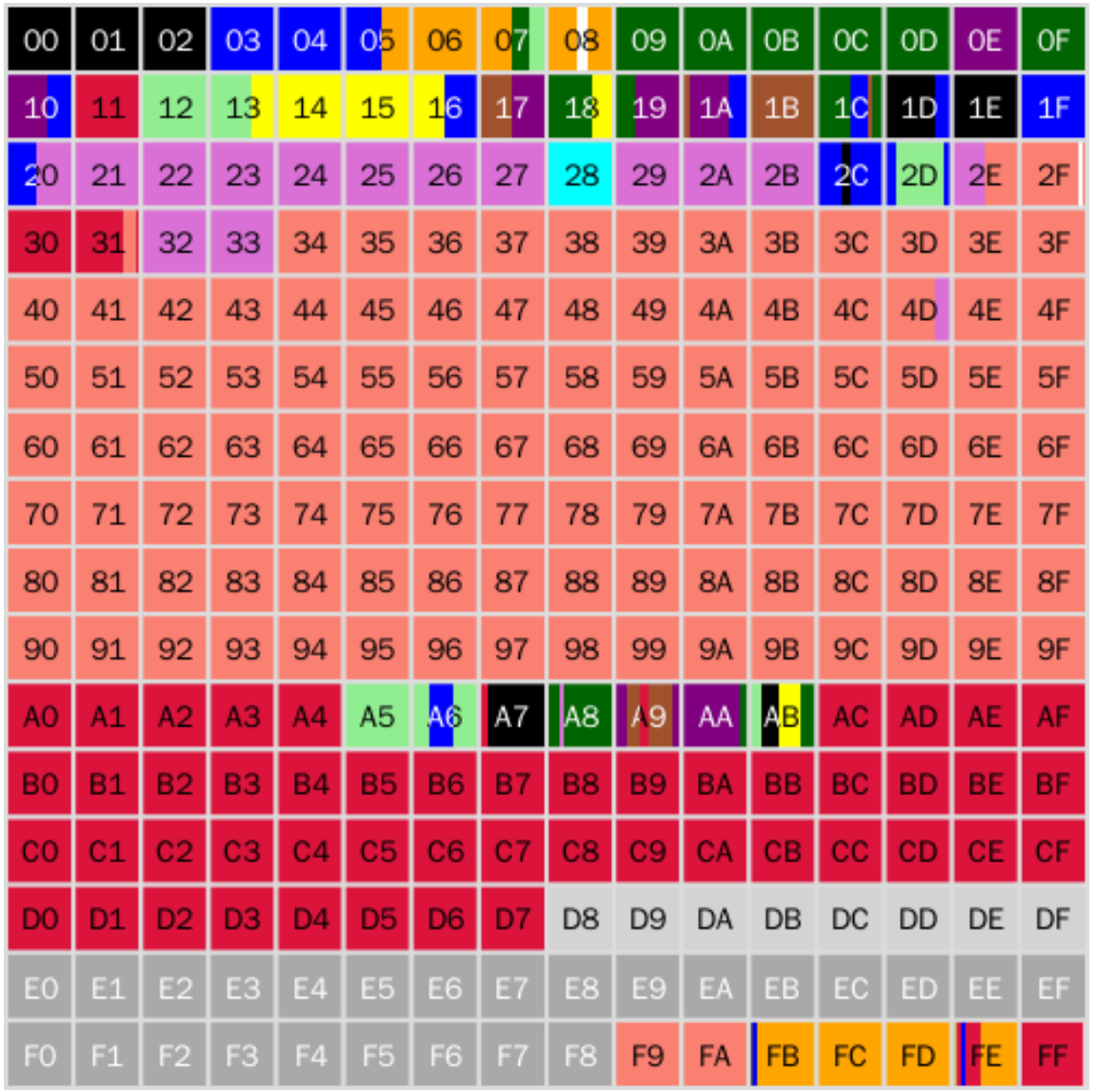


- 최상위비트 (MSB)가 1인 경우는 한글 코드이므로 2바이트를 가지고 해석
- MSB가 0인 경우는 아스키 코드이므로 1바이트씩 해석

유니코드

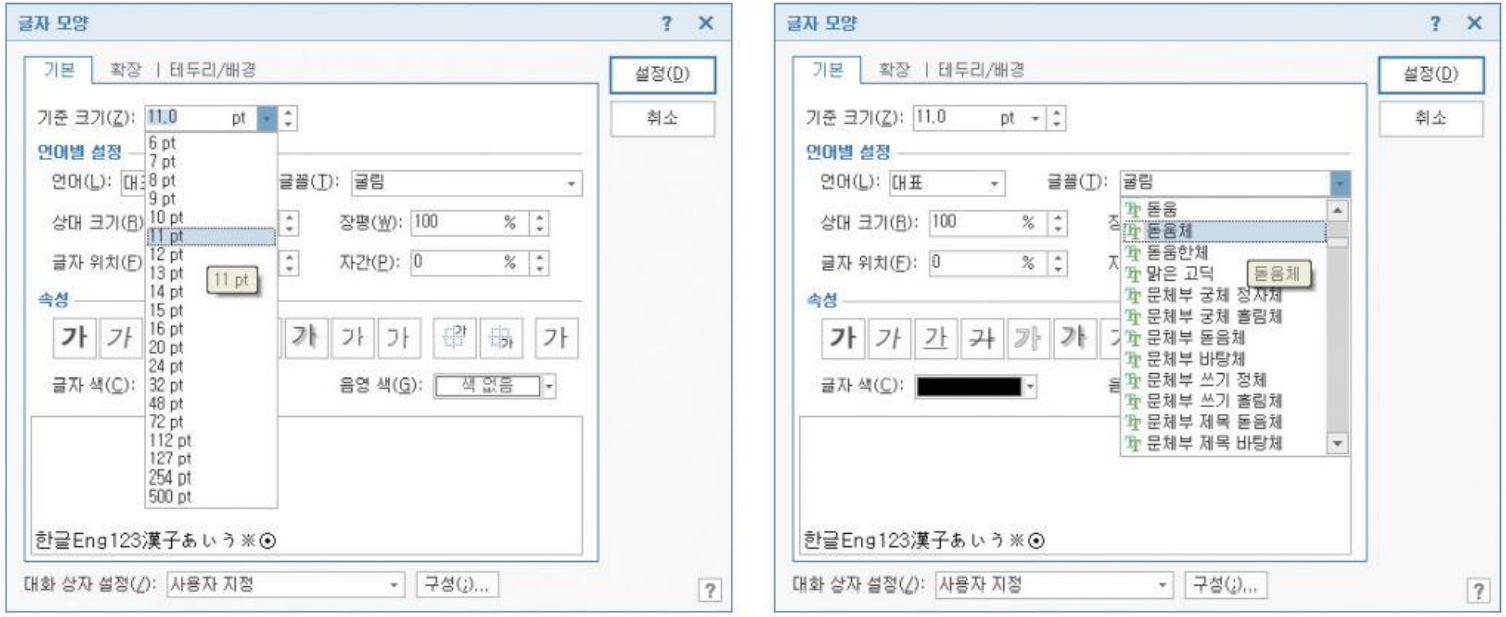
- 모든 나라에서 공통으로 사용할 수 있는 코드 체계가 필요함에 따라 개발된 코드
- 1990년에 제록스 외 몇 개 업체의 연구자들이 모여서 개발한 산업체 표준
- 1991년에 유니코드 버전 1.0이 발표되었고 세계 표준으로 채택
- 컴퓨터에서 다양한 언어 표현이 가능해짐에 따라 소프트웨어의 국제화에 기여
- 전 세계의 모든 문자를 8비트 단위인 옥텟으로 표현
- 데이터 용량을 많이 차지하기 때문에 문서 표현이나 처리보다는 문서 교환이나 통신 분야에 알맞음
- 최대 수용 문자 수는 $2^{16}=65,536$

유니코드


00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F
20 21 22 23 24 25 26 27 28 29 2A 2B 2C 2D 2E 2F
30 31 32 33 34 35 36 37 38 39 3A 3B 3C 3D 3E 3F
40 41 42 43 44 45 46 47 48 49 4A 4B 4C 4D 4E 4F
50 51 52 53 54 55 56 57 58 59 5A 5B 5C 5D 5E 5F
60 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F
70 71 72 73 74 75 76 77 78 79 7A 7B 7C 7D 7E 7F
80 81 82 83 84 85 86 87 88 89 8A 8B 8C 8D 8E 8F
90 91 92 93 94 95 96 97 98 99 9A 9B 9C 9D 9E 9F
A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 AA AB AC AD AE AF
B0 B1 B2 B3 B4 B5 B6 B7 B8 B9 BA BB BC BD BE BF
C0 C1 C2 C3 C4 C5 C6 C7 C8 C9 CA CB CC CD CE CF
D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 DA DB DC DD DE DF
E0 E1 E2 E3 E4 E5 E6 E7 E8 E9 EA EB EC ED EE EF
F0 F1 F2 F3 F4 F5 F6 F7 F8 F9 FA FB FC FD FE FF
로마자, 로마자권 기호
기타 유럽 문자
아프리카 문자
중동·서남아시아 문자
남부와 중앙 아시아 문자
동남아시아 문자
동아시아 문자
CJK 문자
인도네시아, 오세아니아 문자
북미 및 남미 문자
Notational systems
기호
사용자 정의 영역
UTF-16 상·하위 대체 영역
쓰이지 않음
유니 코드 버전 13.0

폰트

- 인쇄 환경에서 사용하던 용어로 글자의 모양
- 크게 세리프 (명조체)와 산세리프 (고딕체) 두 가지로 분류
- 하나의 서체에 스타일과 크기를 고정시키면 하나의 폰트가 됨
- 문자의 서체와 글자의 크기를 비롯해 위첨자, 아래첨자, 강등과 같은 글자의 속성이 포함
- 구성 방법에 따라 비트맵 폰트와 벡터 폰트로 구분



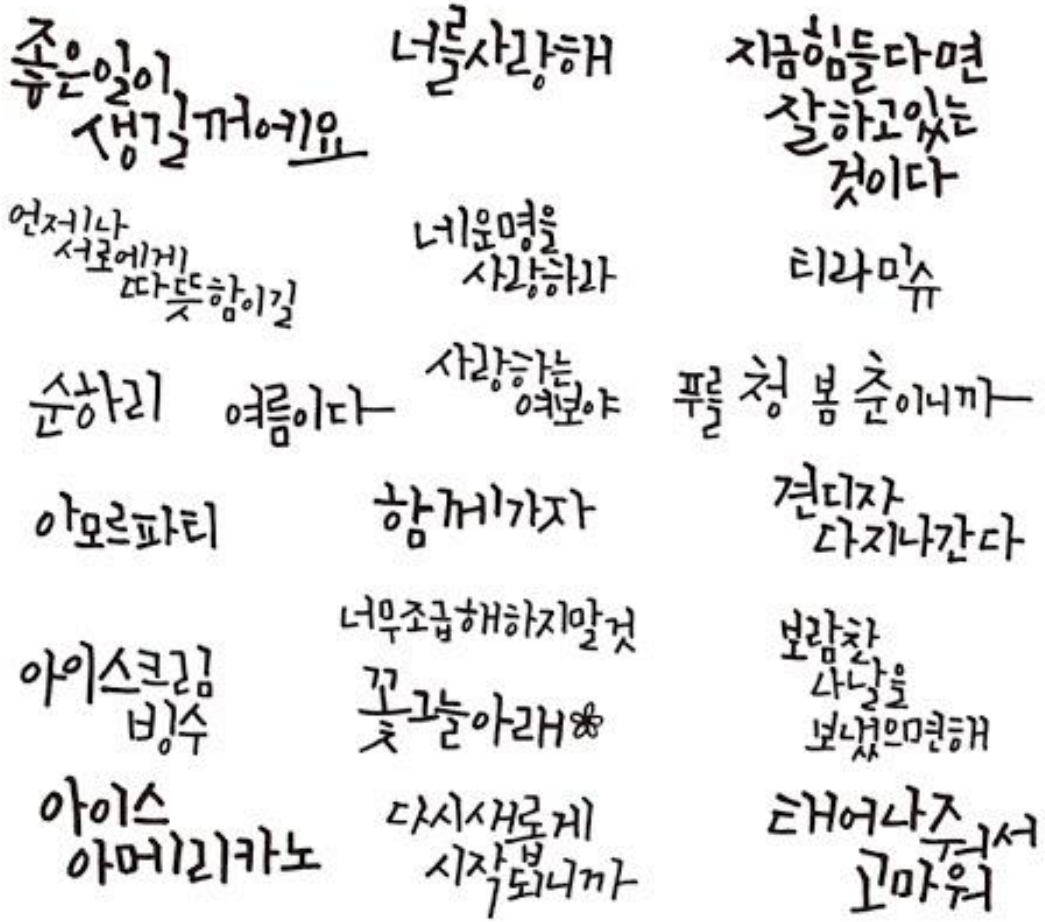
한글 폰트의 생성과 진화

- 최정순, 최정호라는 글씨 장인 두 명의 평생에 걸친 노력에 힘입어 현재의 한글 폰트가 등장
- 한글은 디자이너 관점에서 글자의 제작 방법과 글자 모양의 변화에 연계돼 정리
- 굴림체 폰트: 1970년대에 일본의 인기 글꼴인 나루체(둥근고딕)를 모방하여 급히 제작
- 맑은고딕 폰트: 한국적인 아름다움과 조형미를 현대적으로 표현하고, 가독성을 극대화한 폰트로 제작
- 윈도우에 탑재된 한글 시스템의 폰트의 용량이 큰 경우 기본 서체를 변경하여 사용할 수 있음

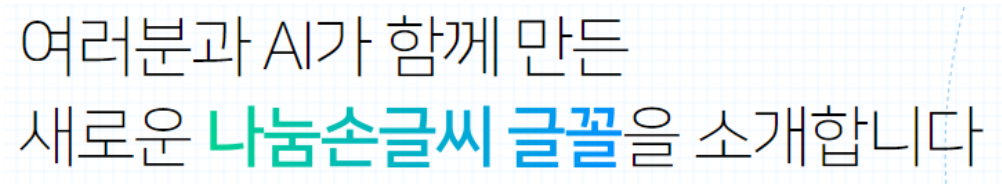
Windows 기본 글꼴				
바탕체	굴림체	돋움체	궁서체	맑은 고딕

캘리그라피

- 글씨를 쓰는 사람의 의도를 표현하기 위해 마치 그림을 그리듯이 글씨를 쓰는 것
- 불규칙함, 동적인 선, 조형적인 효과, 독창성을 특징으로 하는 아름다운 손글씨



폰트 생성



https://clova.ai/handwriting/

글꼴 소개말 | 기본 문구

나눔손글씨 가람연꽃

취업 준비생입니다. 글씨에 성격이 담긴다지만, 저는 이상을 담으려 해요. 불투명한 미래에도 희망이 있다는 믿음을 가지려 합니다.

나눔손글씨 갈맷글

저희 동네 예쁜 바닷길 이름이 갈맷길입니다. 그 이름에서 따왔습니다. 구불구불 편안한 길 같은 글로 쓰였으면 합니다

나눔손글씨 강부장님체

오래전부터 글씨를 많이 쓰는 사무실에서 일해온 강 부장입니다. 사무실에서 흔히 볼 수 있는 이 글씨로 타이핑해보면 어떨까요?

나눔손글씨 강인한 위로

고3학생 만학도입니다. 글을 통해 답답한 마음을 풀고 힘든 시기를 버틸 수 있었습니다. 제 글씨로 위로가 되고 싶습니다.

나눔손글씨 고딕 아니고 고딩

저는 고등학생이고, 친구들과 함께 있는 게 너무 즐거운데요. 제 글씨로 학교 친구들과의 추억을 떠올리셨으면 좋겠습니다.

나눔손글씨 고려글꼴

고려인으로서 모든 고려인들이 한글이라는 모국어를 더 많이 배우고 익히는 데 도움이 되는 고려 글꼴이 되었으면 합니다.

디지털 시대의 텍스트 위기

텍스트에 대한 이해력 저하

- 젊은 세대는 텍스트를 훑듯이 읽고 지나가면서 자신은 이해한 것으로 착각
- 요즘 학생들은 과제를 해결하기 위하여 검색엔진 대신 유튜브에 접속해 과제를 해결
- 짧은 문장에 익숙하고 긴 문장을 읽기 힘든 현상은 텍스트에 대한 이해력 저하로 이어짐

위기의 한글 사용

- 인터넷의 발달로 언어 파괴 현상이 심각해짐
- 파괴 현상은 PC 통신이 보급된 시기부터 나타남
- 젊은 세대를 중심으로 축약된 말이나 다른 세대와 소통 되지 않는 외래어 같은 말을 쓰는 부정적인 측면

디지털 시대와 손글씨의 가치

- 지금은 손글씨보다 키보드와 스마트폰 자판이 더 익숙한 시대이지만 손글씨는 여전히 가치가 있음
- 손글씨로 문장을 작성하면 자판을 사용할 때보다 사용하는 단어와 어휘가 더 풍부해 짐
- 손글씨를 예쁘게 쓰기 위해 집중하는 힘도 길러짐
- 수업 시간에 필기를 병행하면 내용이 오랫동안 기억되고, 학습 내용을 재구성하는 능력도 향상

3 텍스트 관련 기술의 변화

문서 표준

- 예전엔 문서편집기가 MS워드, 아래한글, 엑셀, 파워포인트 등을 각각의 문서 파일로 인식하여 여러 문제 발생
- 각각의 프로그램에서 향상된 편집 기능을 제공하기 위하여 지속적으로 새로운 버전의 소프트웨어를 제공해야 함 → 새로운 버전과 이전 버전 사이에 작성된 문서의 호환성 문제가 발생
- 일반적으로 많이 사용되는 HWP, DOC, XLS, PPT 도구는 특정 기업에 종속된 폐쇄형 전자문서임 → 새로운 버전의 문서 개발도 이들 기업에 종속

ODF

- 폐쇄형 전자문서 문제를 해결하기 위해 개방형 문서 표준인 ODF가 탄생
- ODF 전자문서가 도입되면서 그동안 사회·경제적으로 비효율적으로 인식된 종이 문서 사용량이 감소
- 개방형 표준 전자문서에서는 대표적으로 개방형 문서 표준 형식(ODF), XML(HTML), PDF 등을 사용
- 개발 주체가 불명확하여 주기적으로 소프트웨어의 성능을 개선하는 폐쇄형 전자문서보다 품질이 떨어지는 문제

클라우드 환경의 오피스 서비스

- 최근의 오피스 소프트웨어는 PC 중심의 설치 방식에서 모바일과 클라우드 환경으로 이동
- 별도의 프로그램 설치 없이 PC, 웹, 모바일 환경에서 서로 연동해 사용
- 클라우드 서비스 사용자는 제작한 문서를 클라우드 저장소에 저장하고 모든 플랫폼에서 공유,편집이 가능

폴라리스 오피스

- 클라우드 환경에서 문서를 작성하고, 공유를 통해 협업하고, 결과물을 생산하는 최근 트렌드를 반영한 서비스
- 워드, 엑셀, 파워포인트 등 다양한 형태의 문서를 PDF 파일로 제작하고 공유
- 반대로 PDF 문서를 워드, 엑셀, 파워포인트 문서로 변경·편집한 뒤 다시 PDF로 저장하는 재편집도 가능

PDF의 개념

- 개방형 문서 표준인 ODF기반의 전자문서
- 원래 어도비의 소유였으나 월드와이드 웹 컨소시엄(W3C)에서 누구나 이용할 수 있는 개방형 문서 표준으로 공개
- 문서 위·변조 방지 기능이 있기 때문에 여러 개방형 표준 가운데서도 열람, 보관용으로 가장 적합한 문서
- PDF 작성 프로그램인 아크로뱃은 유료이나 뷰어 프로그램인 아크로뱃 리더는 무료
- 전자책 시장에서도 EPUB과 더불어 주요 전자책 포맷으로 각광 받음

PDF의 장점

- 문서의 호환성이 높고 파일의 무결성을 가짐
- PDF 파일은 대부분의 컴퓨터에서 읽기와 인쇄가 가능하여 인터넷, 인트라넷에서 정보를 공유할 때 적합한 형식
- 파일의 무결성: 온·오프라인으로 전송된 파일이 원본 문서와 동일하다는 의미
- 어떤 프로그램으로 PDF를 작성하더라도 텍스트, 도면, 이미지, 그래픽 등 소스파일 정보가 그대로 유지

PDF의 장점

- 문서 사용과 관리가 편리
- PDF 파일은 자체에 압축 기능이 들어 있어 다른 파일에 비해서 상대적으로 용량이 적음
- 문서의 깨짐을 방지하기 위해 폰트, 이미지, 그래픽, 표 등과 같은 정보를 하나의 파일에 자유롭게 포함(임베딩)하여 저장
- PDF를 사용하여 책 한 권을 하나의 파일로 만들 수 있으며 책갈피 및 링크 기능을 첨가하여 원하는 부분을 쉽게 찾을 수 있음

PDF의 장점

- 문서 보안이 뛰어남
- PDF 파일은 문서에 암호를 설정하는 기능을 제공하기 때문에 보안이 뛰어남
- 단순히 파일을 읽는 경우에만 보안 기능이 수행되는 것이 아니라 인쇄, 복사, 편집 등 각각의 과정에 대하여 제한을 설정할 수 있음

- 쌍방향으로 인터페이스 삽입이 가능
- 일반 워드프로세서에는 없는 쌍방향 인터페이스(체크박스, 글상자, 멀티미디어 등)를 삽입하여 효율적으로 공동 작업을 진행할 수 있게 함

PDF의 단점

- 문서의 제작 방식이 동일하지 않고, 다양한 파일 형식이 존재함
- PDF 문서는 제작 방식에 따라 크게 텍스트 PDF 파일과 이미지형 PDF 파일이 있음
- 텍스트 PDF 파일은 대부분의 텍스트와 이미지를 수정하거나 편집할 수 있음
- 이미지형 PDF 파일은 스캐너로 문서를 입력하기 때 문에 편집 범위가 제한

PDF의 단점

- 편집이 불편함
- PDF의 장점 중 하나인 '보안성'은 편집이 불가능 하거나 힘들다는 데에서 나온 것
- PDF 파일은 하나의 커다란 이미지 형태이기 때문에 워 드 파일처럼 일부를 수정하는 데 제한이 따름
- PDF 파일을 편집하려면 별도의 프로 그램을 사용해 MS 워드나 엑셀 같은 다른 문서 포맷으로 변환해야 함

PDF의 단점

- 모니터에서 가독성이 낮음
- 대부분의 PC용 모니터는 4:3 또는 16:9의 비율로 제작
- PDF 파일은 인쇄하기 쉽 게 A4나 A3 규격에 맞춘 것이 대부분
- 모니터로 PDF 파일을 열람할 경우 화면에 꽉 차지 않기 때문에 스크롤 기능을 사용해야 함

DRM(Digital Rights Management) 기술

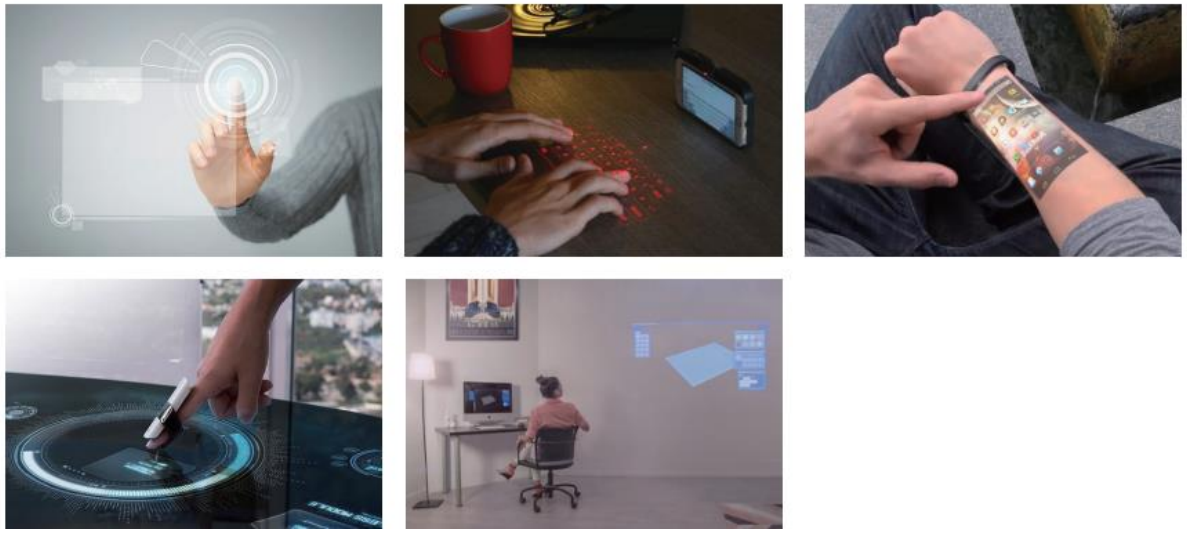
- PDF는 문서에 대한 권한이 없는 사용자가 문서를 열람하고 읽는 것이 가능하기 때문에 개인 정보 유출이 불가피
- 최근에는 PDF 문서에 디지털 저작물 보호 및 관리를 의미하는 DRM 기술이 도입
- DRM 기술은 디지털 콘텐츠의 무단 사용을 방지하는 기술
- 콘텐츠에 특정 인물만 접근하거나 정해진 시간 동안만 접근할 수 있게 제약하는 기술
- 내용 복사나 화면 캡처도 불가능
- DRM이 설정된 콘텐츠는 지정된 PC와 스마트폰 등에서만 제한적으로 사용할 수 있음
- PDF 문서에 DRM 기술이 도입됨에 따라 기밀 서류를 보관하거나 인터넷에 데이터베이스를 구축 하는데 PDF 문서가 많이 이용

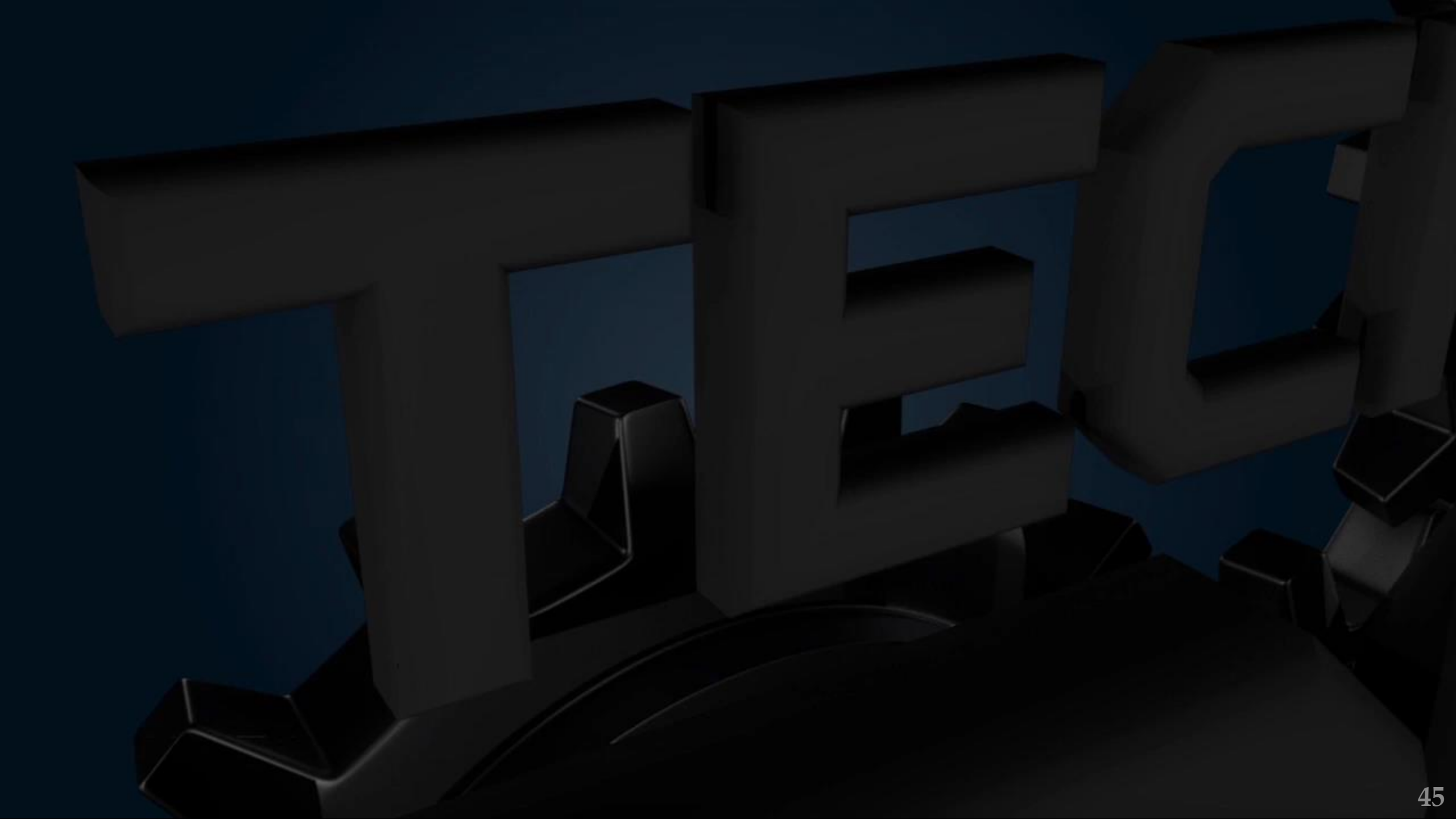
동작 인식, 음성인식 방식 글자 입력 기술

- 최근 키보드 대신 사용자의 동작이나 음성으로 입력하는 방식이 차세대 제어 수단으로 주목
- '갤럭시노트10' S펜의 '에어모션': 화면을 터치하지 않아도 S펜의 움직임으로 여러 기능이 가능
- 픽셀4'의 모션 인식 기능: 사용자의 손짓에 따라 음악 재생, 알람 중지 기능을 제공
- 애플의 모션 인식: 손목의 움직임으로 메시지에 간단하게 답함

다양한 터치스크린 방식 글자 입력 기술

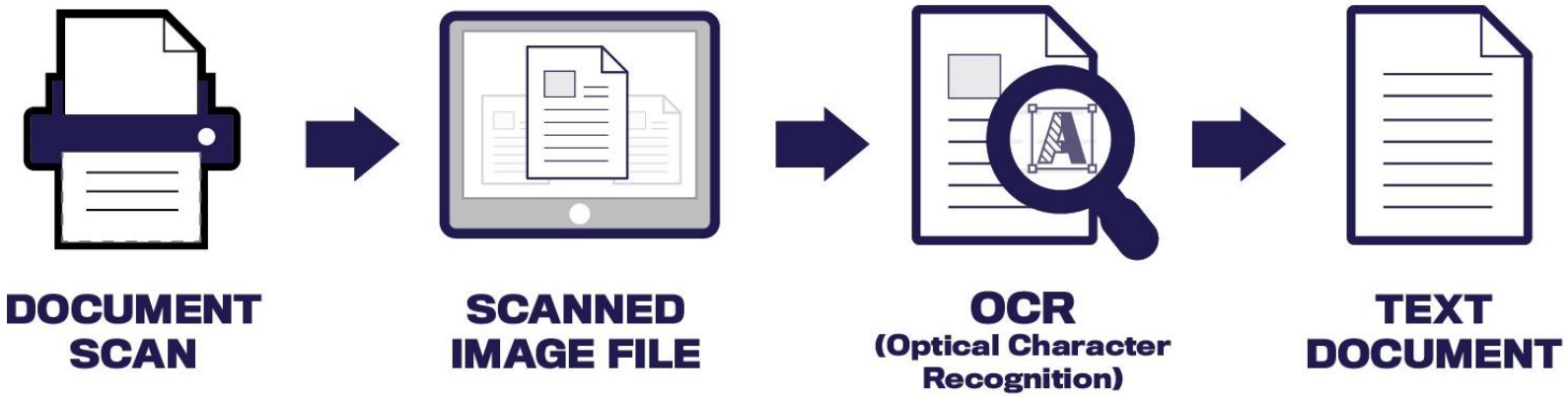
- 빅(ViKC): 스마트폰에 케이스를 씌워 빛 반사가 없는 평평한 표면에 놓고 사용
- 시크릿 팔찌: 빔을 활용한 소형 피코프로젝터를 통해 팔목에 자판 영상을 투영 센서는 손가락이 터치한 부분과 손가락의 움직임을 감지하여 사용자의 팔목을 스마트폰처럼 사용할 수 있게 함
- 버드: 손가락에 끼우면 어떤 사물이든 터치할 수 있으며 동작 인식, 음성 인식도 가능




THE

문자인식 기술

- 책·잡지·신문 등 기존의 인쇄 자료, 손으로 기록한 문자·기호·마크 등을 컴퓨터가 자동으로 인식하는 기술
- 종이에 기록된 문자를 광학적인 장치인 스캐너를 사용하여 인식하기 때문에 광학문자 인식(OCR)이라고 부름
- 1970년대부터 상업적 용도로 널리 사용되기 시작
- 오늘날에는 여권 처리, 보안 문서 처리(수표, 재무 문서, 청구서), 우편물 추적, 출판 등에서 자동화 작업에 사용



문자인식 방법

패턴 정합

- 입력된 문자와 컴퓨터에 기억된 문자의 유사성, 정합도에 의해 문자를 식별하는 방식
- 주로 인쇄 문자를 인식하는 데 사용
- 문자의 인식 능력 난이도는 숫자, 영문자, 기호, 한글, 한자 순으로 높아짐

구조 분석

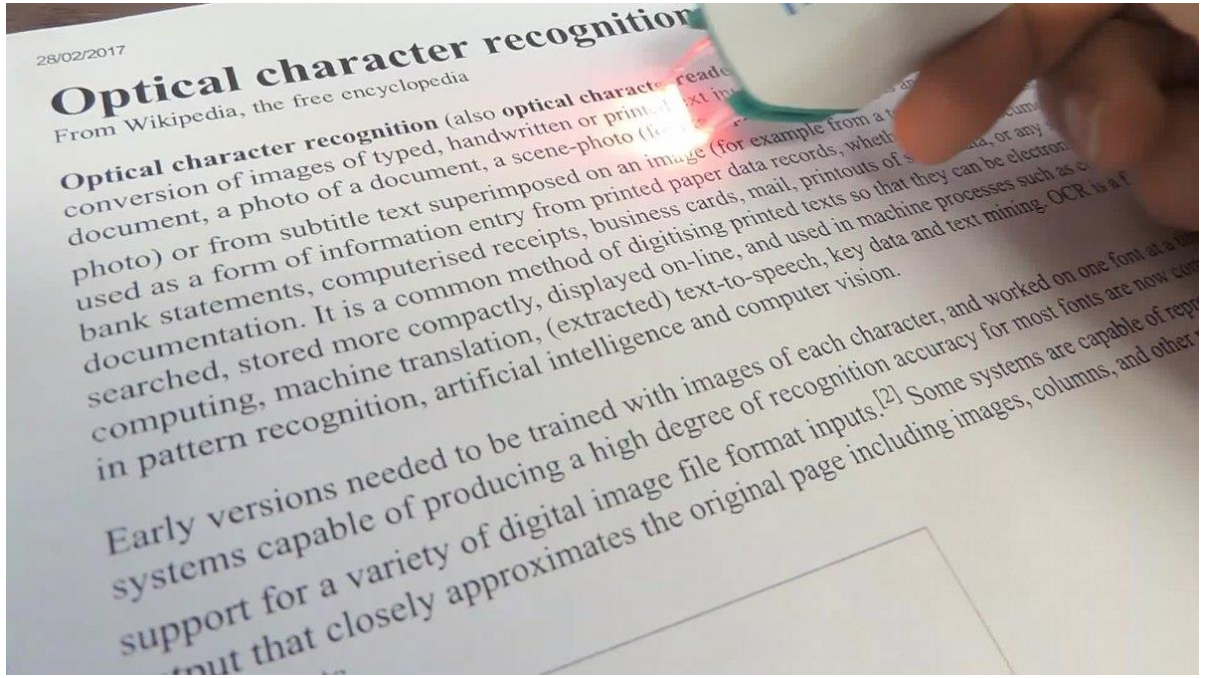
- 입력된 문자를 구성하는 고유의 특징적인 선의 형태와 특성에 의해 문자를 식별하는 방식
- 주로 필기 문자를 인식하는 데 사용

문자인식 순서

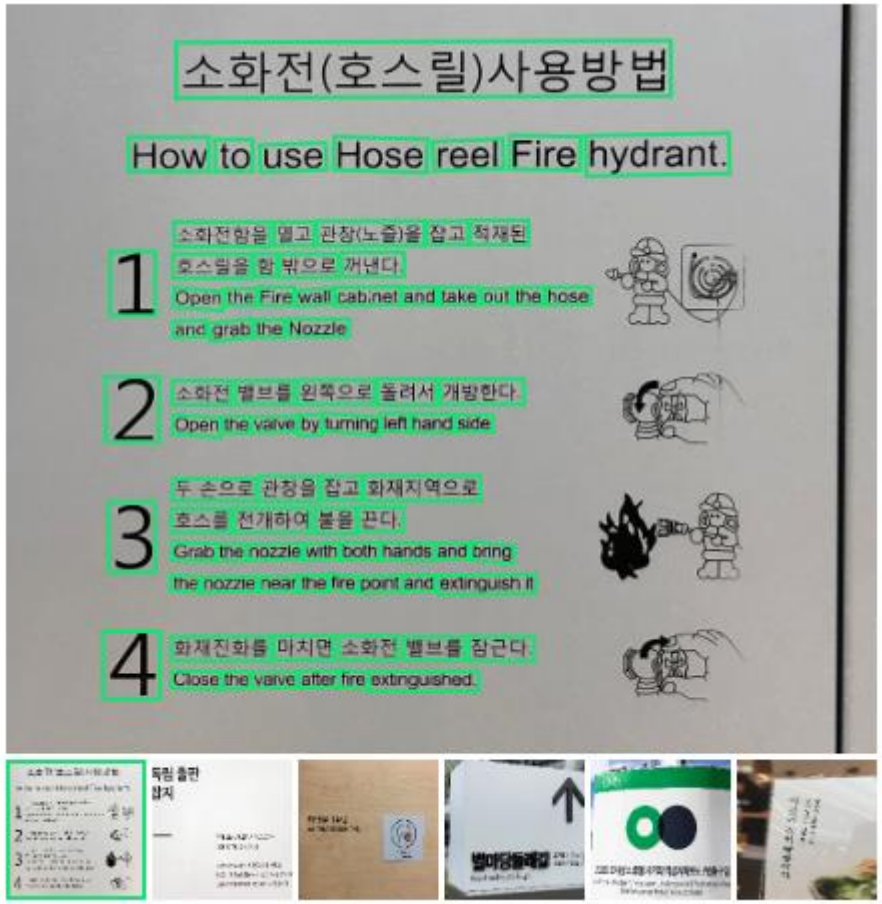
- 기존 인쇄 자료를 스캐너·카메라를 통하여 이미지 형태로 읽음
- 데이터 내용을 분석한 후 그림 영역과 글자 영역으로 구분
- 글자 영역의 문자들을 편집기에서 수정이 가능하도록 텍스트로 변환

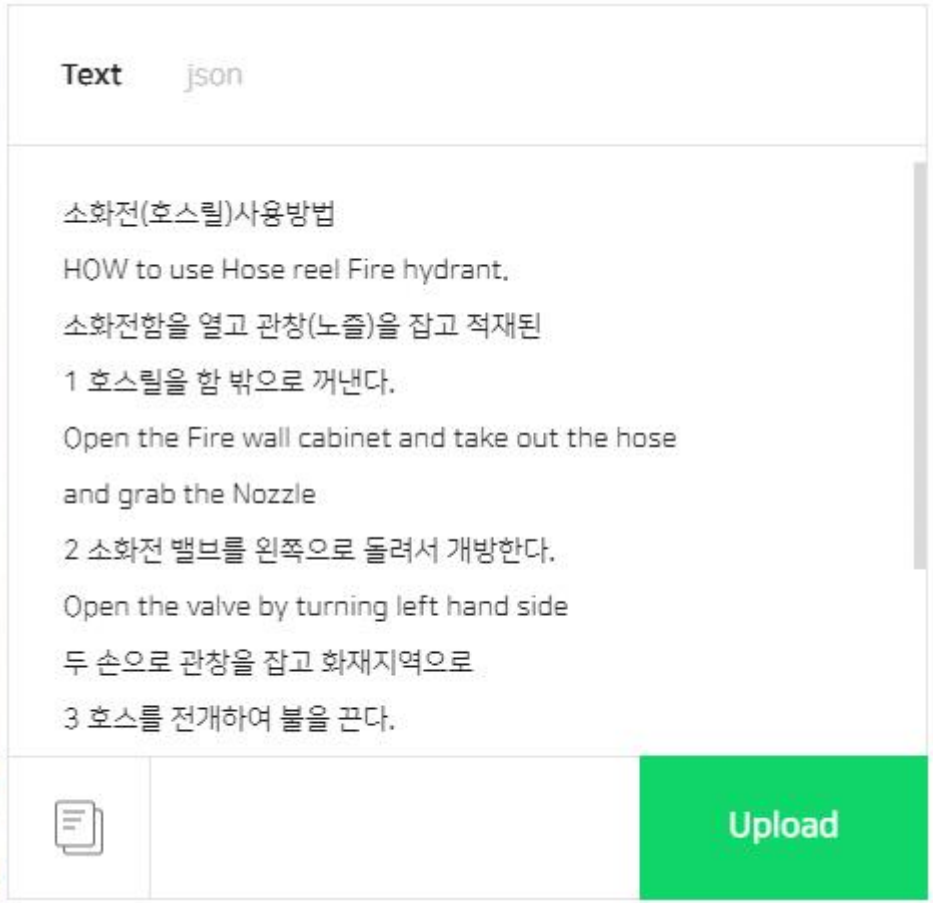
OCR 기술

- 이미지에 포함된 문자를 인식해서 기계가 읽고 활용할 수 있는 텍스트로 변환하는 기술
- 예전에는 손으로 쓴 글씨나 인쇄된 문자를 디지털로 변환하려면 일일이 컴퓨터에 입력해야 했음
- OCR 기술의 등장으로 각종 서류의 글자, 이미지를 인식해 데이터로 자동 변환할 수 있게 되면서 문제가 해결
- OCR 기술과 인공지능을 접목하여 AI의 핵심 기능인 기계학습(머신러닝)을 위한 데이터를 확보



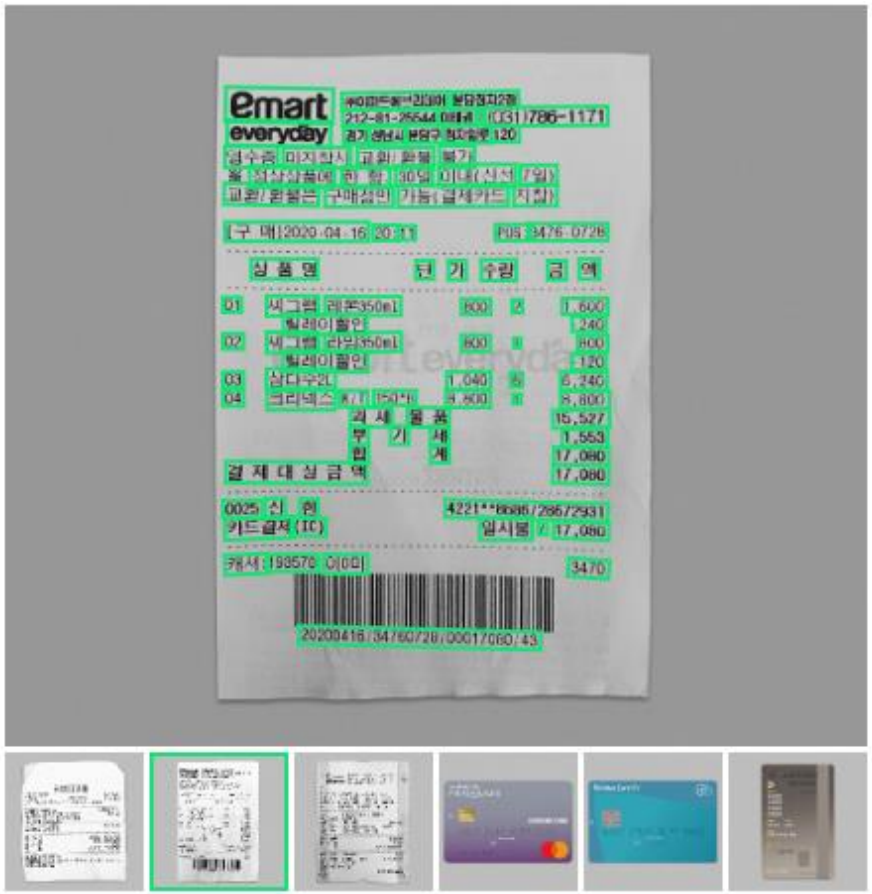
CLOVA OCR





https://clova.ai/ocr/

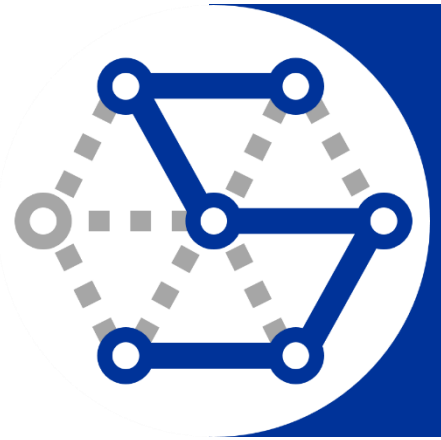
CLOVA OCR



https://clova.ai/ocr/

OCR 기술의 활용

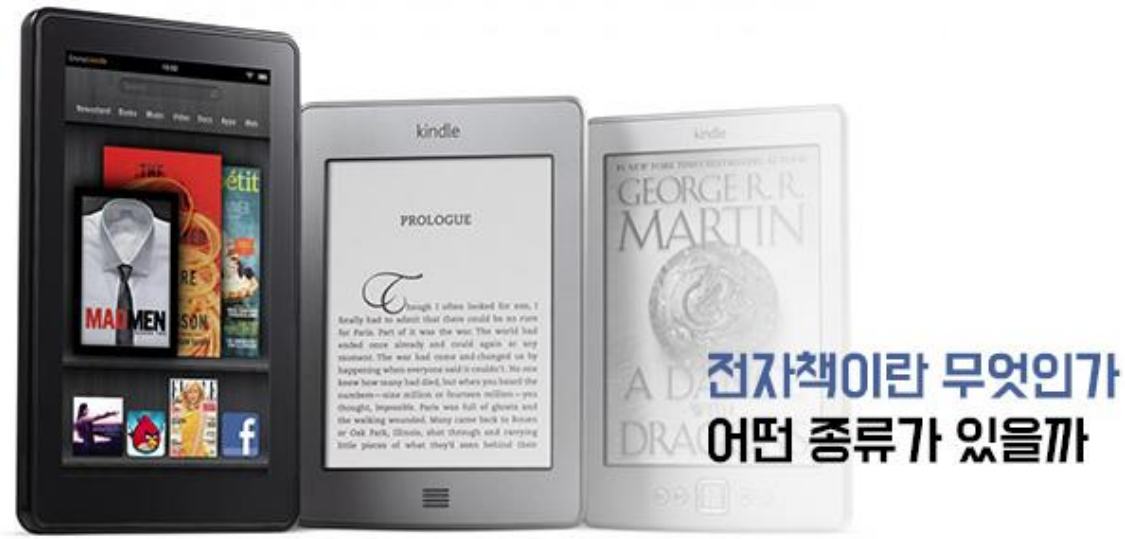
- 문서 자동 분류 기능을 활용하여 반복되는 검증 업무를 줄여줌
- 음원 플랫폼을 변경할 때 재생 목록을 쉽게 옮길 수 있게 함
- 번역, 이미지 검색, 텍스트 분석, 챗봇 등에 사용
- 이미지에 포함된 문자의 추출·인식·번역 기능을 활용하는 이미지 번역으로 진화



4 텍스트 기반 멀티미디어 서비스

전자책

- 디지털 콘텐츠로 제작된 파일 형식의 책을 의미
- 스마트폰과 태블릿이 본격적으로 대중화된 시점에서 종이 책의 대안으로 등장
- 읽기 위해선 전자책 콘텐츠, 전자책을 보기 위한 앱·뷰어, 전자책을 담는 디바이스가 필요
- 초기의 텍스트 위주의 책, 종이 책을 스캔한 수준에서는 벗어났지만 아직도 빠르게 성장하지 못함



전자책 시장의 현 상황과 전망

- 업계 전문가들은 스마트 디바이스의 보급으로 종이 출판 산업은 사양산업이 될 것으로 예견했으나 그렇지 않음
- 단말기 성능과 품질이 소비자 원하는 수준을 충족하지 못하기 때문에 성장 둔화
- DRM 호환성 문제가 가장 큼
- 사람들의 관심이 동영상·음성 콘텐츠에 집중되어 있고, 전자책에 특화한 콘텐츠가 부족
- 미국은 이전부터 아마존을 중심으로 저렴한 가격의 전자책 시장이 활성화된 상태
- 국내에도 디지털 시대에 맞춘 콘텐츠인 웹툰·웹 소설에 대한 수요가 증가하면서 조금씩 변화

오디오북

- 스마트 디바이스의 사용 증가에 따른 디지털 피로도가 높아지면서 오디오북이 주목 받음
- 시간과 장소에 관계없이 편리하게 휴대할 수 있고, 책을 들으면서 다른 일을 할 수 있음
- 장애인, 노약자 등 정보 취약층에게도 유용하게 활용
- 책을 읽어주는 방식이 다양하기 때문에 2~3시간 만에 책 한 권을 읽을 수 있음
- 현재 어느 정도를 들었는지 확인하기 어려운 단점

오디오북

- 최근에 스마트폰, AI 스피커를 통한 글로벌 오디오북 시장이 급성장
- 구글은 구글플레이에 오디오북을 출시
- 네이버는 오디오북을 제작하기 위하여 텍스트를 목소리로 바꿔주는 TTS 엔진을 개발
- 오디오북의 특성상 기계음보다 감정을 살려서 읽는 낭독자에 대한 수요는 줄지 않을 것으로 예상

밀리의 서재

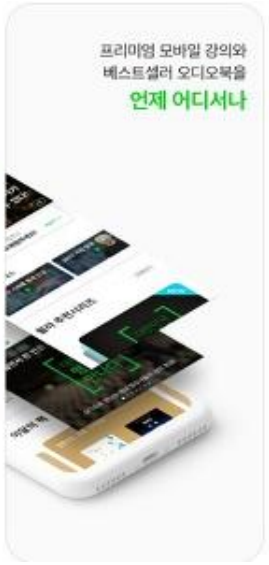
- 월정액으로 도서를 대여해 읽을 수 있는 전자책 서비스
- 2016년에 서영택 전 웅진씽크빅 대표이사가 설립
- 2020년 12월 현재 국내 월정액 도서 서비스 중 최고 수준인 약 10만 권 정도의 책을 읽을 수 있음

리디북스

- 리디북스는 리디 주식회사에서 2009년 11월 16일에 서비스를 시작한 대한민국의 전자책 서점
- 리디 주식회사는 서울대학교 전기공학부를 졸업한 배기식 대표가 설립한 회사
- 국내 최초의 스마트폰 기반의 전자책 서비스로 사용자는 온라인 서점에서 구매한 도서를 무료로 제공되는 애플리케이션 및 전자책 전용 뷰어를 통해 읽을 수 있음

윌라

- 오디오북 및 모바일 강의 서비스로 월정액 구독형 콘텐츠 스트리밍 제공
- 국내 서비스되고 있는 오디오북 플랫폼 중 가장 많은 수의 오디오북 제공




suanlab
이수안 컴퓨터 연구소
suan computer laboratory