

#2022 서울시립대학교 AICOSS

기업 연계형 실습 중심의 딥러닝 부트캠프

D조

김다은, 김민주, 방승현, 조형진, 최도운

[Rexsoft]



목차

A table of Contents

- #1 문제 정의
- #2 데이터셋 소개 및 사용방안
- #3 선택한 모형
- #4 결과

[문제 정의]

문제 정의

건강성적표

“건강성적표로 건강 상태를 한눈에 살펴볼 수 있습니다.”

건강성적표는 여러 기관의 건강 검진 데이터, 건강보험공단 등의 자료를 활용하여 고객의 건강한 정도, 기대 수명, 노화 정도, 질환 발생 위험도를 고객의 성별 및 연령에 따라 맞춤형으로 평가해주는 서비스입니다.

‘건강성적표’는 검진 결과를 성적표 형식으로 구성하고 분석하여, 의학 전문 지식이 없는 일반 소비자도 쉽게 이해할 수 있습니다.

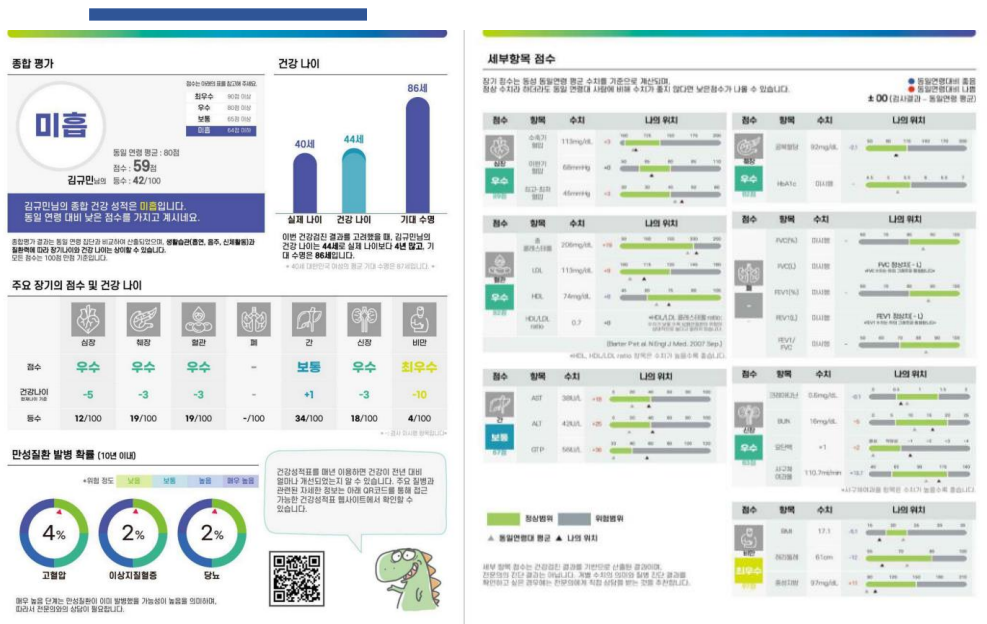


Rexsoft 미연계 기관에서 받은 검진 결과로도
건강성적표를 출력해보면 어떨까?

[데이터셋 소개 및 사용방안]

데이터셋 소개 및 사용방안

<PDF template>



<가정>

실제 수검자는 이 PDF를 A4 용지에 출력한 결과지를 소지하고 있다고 가정

결과지를 핸드폰 카메라로 찍어서, 웹 또는 어플에 업로드하여 전달하는 것을 가정

미연계 기관의 경우, 해당 기관에서 제공하는 서식에 맞춘 결과지가 제공되나, 현재는 예시로 제공된 하나의 서식만 있다고 가정

선택한 모형

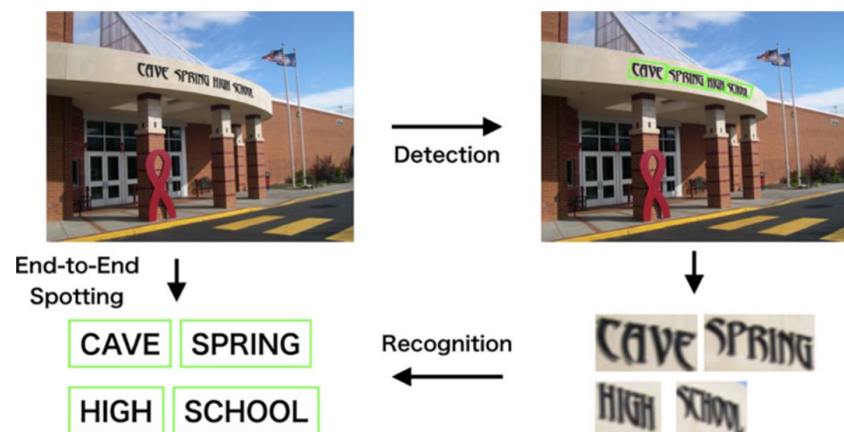
-Easy OCR-

EasyOCR

OCR

광학 문자 인식(OCR)이란 무엇인가?

광학 문자 인식(OCR)은 텍스트 이미지를 기계가 읽을 수 있는 텍스트 포맷을 변환하는 과정
OCR을 사용하면 이미지를 텍스트 데이터로 저장 가능



• <https://link.springer.com/article/10.1007/s11263-020-01369-0>

전처리

- 컬러 이미지를 먼저 회색조로 변환
- 밝기와 명암 대비

글자 검출

- 딥러닝 시스템을 이용하여 전체 이미지에서 텍스트인 영역을 골라내는 작업을 수행

글자 인식

- 글자를 구분하는 여러 특징을 대량의 데이터를 통해 컴퓨터가 학습 및 추출

후처리

- 출력된 텍스트에 부자연스러운 단어나 문자가 포함되면 이를 수정하여 정확도 향상

EasyOCR

데이터 전처리

<alignment process>



image



template



after alignment

```
import numpy as np
import imutils
import cv2

def align_images(image, template, maxFeatures=500, keepPercent=0.2,
                 debug=False):
    imageGray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
    templateGray = cv2.cvtColor(template, cv2.COLOR_BGR2GRAY)
    orb = cv2.ORB_create(maxFeatures)
    (kpsA, descA) = orb.detectAndCompute(imageGray, None)
    (kpsB, descB) = orb.detectAndCompute(templateGray, None)

    method = cv2.DESCRIPTOR_MATCHER_BRUTEFORCE_HAMMING
    matcher = cv2.DescriptorMatcher_create(method)
    matches = matcher.match(descA, descB, None)

    matches = sorted(matches, key=lambda x: x.distance)

    keep = int(len(matches) * keepPercent)
    matches = matches[:keep]
```

Homography

3차원 공간상의 평면을 서로 다른 시점에서 바라봤을 때 획득되는 영상 사이의 관계를 나타내는 용어

하나의 평면을 다른 평면으로 투시변환 하는 것과 같은 맥락

```
if debug:
    matchedVis = cv2.drawMatches(image, kpsA, template, kpsB, matches, None)
    matchedVis = imutils.resize(matchedVis, width=1000)
    cv2.imshow("Matched Keypoints", matchedVis)
    cv2.waitKey(0)
ptsA = np.zeros((len(matches), 2), dtype="float")
ptsB = np.zeros((len(matches), 2), dtype="float")

for (i, m) in enumerate(matches):
    ptsA[i] = kpsA[m.queryIdx].pt
    ptsB[i] = kpsB[m.trainIdx].pt

(H, mask) = cv2.findHomography(ptsA, ptsB, method=cv2.RANSAC)
(h, w) = template.shape[:2]
aligned = cv2.warpPerspective(image, H, (w, h))

return aligned
```

Keypoint matching

<keypoint matching 예시>



• <https://records.sigmm.org/keypoint-match-example/>

EasyOCR

데이터 전처리



왜 “ROI(Region Of Interest)”를 사용해야 하는가?

Easyocr을 활용하여 건강성적표 전체를 읽어내는 것은 어려움. 따라서 cv2.selectROI 를 사용하여 엑셀 값에 정리할 숫자들만 추출하여 ocr을 진행하는 것이 효과적

##영역구하기

#마우스 드래그로 영역구하는 함수

```
def get_pos(img, start, end):
    idx = list(range(start, end+1))
    positions = []
    for i in range(start, end+1):
        x1, y1, width, height = cv2.selectROI("location", img, False)
        x2 = x1+width
        y2 = y1+height
        positions.append([x1, y1, x2, y2])
```

```
cv2.destroyAllWindows()
return idx, positions
```

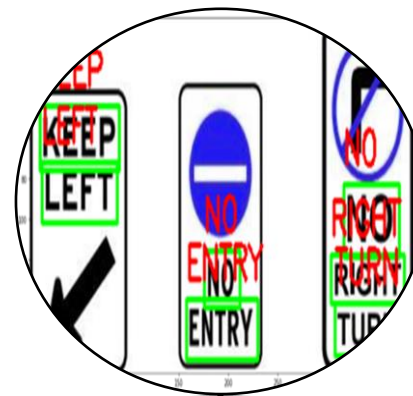
EasyOCR

OCR



문자 “검출”

- 글자들이 존재하는 위치를 찾아내고 이를 bounding box로 묶음
- bounding box로 묶인 글자들은 인식 모델의 입력으로 들어감
- Text Detection 은 주로 CNN 계열의 모델을 사용

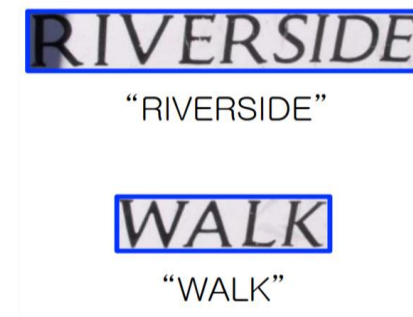


문자 “인식”

- bounding box 안 글자가 어떤 내용인지 알아냄
- Text Recognition은 주로 RNN 계열의 모델을 사용



〈글자 검출〉



〈글자 인식〉

EasyOCR

후처리

1

```
for i in range(len(km)):
    if km[i] == '':
        km[i] = 'nan'
```

km이라는 리스트 안에 있는 원소가 ''이면 'nan'으로 대체

2

```
for i in range(4, len(km)):
    t = km[i]
    if type(t) == str:
        if ('미' in t) or ('시' in t) or ('행' in t) is True:
            km[i] = '미시행'
```

```
bmi = km[16]
rate = str(km[4])
km = [str(km[i])[0:3] for i in range(len(km))]
km[16] = bmi[0:4]
km[4] = rate[0:5]
```

이름, 성별, 나이를 제외한 모든 항목에서 미 / 시 / 행 중 하나라도 검출되면 그 항목값을 미시행으로 바꿔줌

3

```
for i in range(len(km)):
```

```
    if type(km[i]) == str:
        km[i] = km[i].replace('g', '9')
        km[i] = km[i].replace('o', '0')
        km[i] = km[i].replace('o', '0')
        km[i] = km[i].replace('0', '0')
        km[i] = km[i].replace('G', '6')
        km[i] = km[i].replace('b', '6')
        km[i] = km[i].replace(',', ',')
        km[i] = km[i].replace('l', '7')
```

오인식되는 패턴을 찾아 직접 바꾸어줌

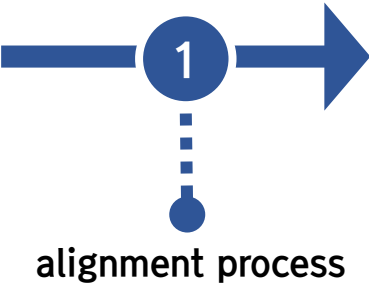
[결과]

EasyOCR

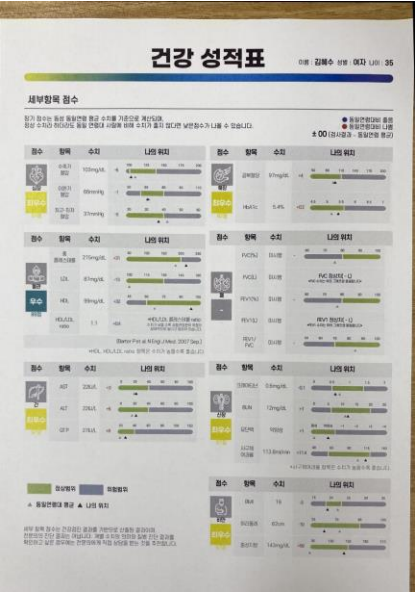
OCR



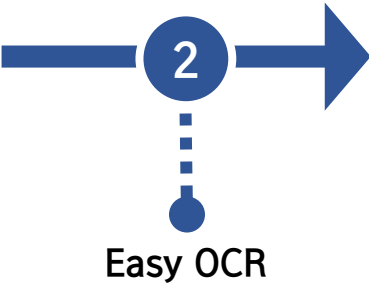
〈원본 사진〉



alignment process



〈이미지 변환 후〉



Easy OCR

	A	B	C
1		항목	텍스트
2	0	이름	김혜수
3	1	성별	여자
4	2	나이	35
5	3	공복혈당	97mg/dL
6	4	사구체여과율	113 8mVmin
7	5	요단백(정상/약양성/+1/+2/+3/+4)	의양성
8	6	중성지방	43mg/dL
9	7	총콜레스테롤	215mg/dL
10	8	최고-최저 혈압	37mmia
11	9	수축기혈압	io3mg/dL
12	10	이완기혈압	66mmkg
13	11	크레아티닌	O.bmg/dL
14	12	허리둘레	62cmn
15	13	HbA1c	5.49a
16	14	ALT	22U/L
17	15	AST	22U/
18	16	BMI	19
19	17	FEV1/FVC	미시행
20	18	FEV1(L)	미시행
21	19	FEV1(%)	미시행
22	20	FVC(%)	미시행
23	21	FVC(L)	미시행

〈OCR 적용 후〉

EasyOCR

OCR

	A	B	C
1		항목	텍스트
2	0	이름	김혜수
3	1	성별	여자
4	2	나이	35
5	3	공복혈당	97mg/dL
6	4	사구체여과율	113 8mVmin
7	5	요단백(정상/약양성/+1/+2/+3/+4)	의양성
8	6	중성지방	43mg/dL
9	7	총콜레스테롤	215mg/dL
10	8	최고-최저 혈압	37mmia
11	9	수축기혈압	io3mg/dL
12	10	이완기혈압	66mmkg
13	11	크레아티닌	0.bmg/dL
14	12	허리둘레	62cmn
15	13	HbA1c	5.49a
16	14	ALT	22IJNL
17	15	AST	22U/
18	16	BMI	19
19	17	FEV1/FVC	미시행
20	18	FEV1(L)	미시행
21	19	FEV1(%)	미시행
22	20	FVC(%)	미시행
23	21	FVC(L)	미시행



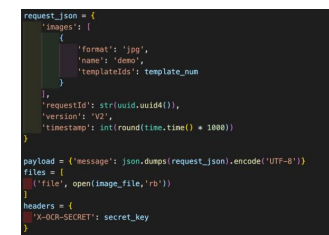
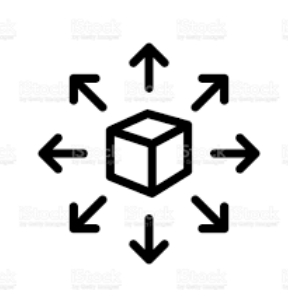
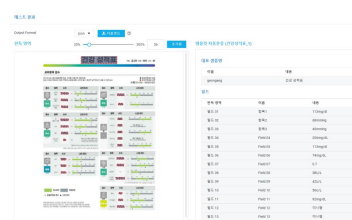
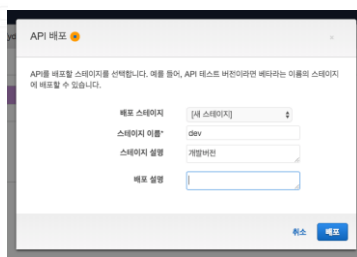
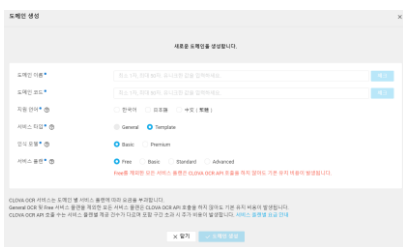
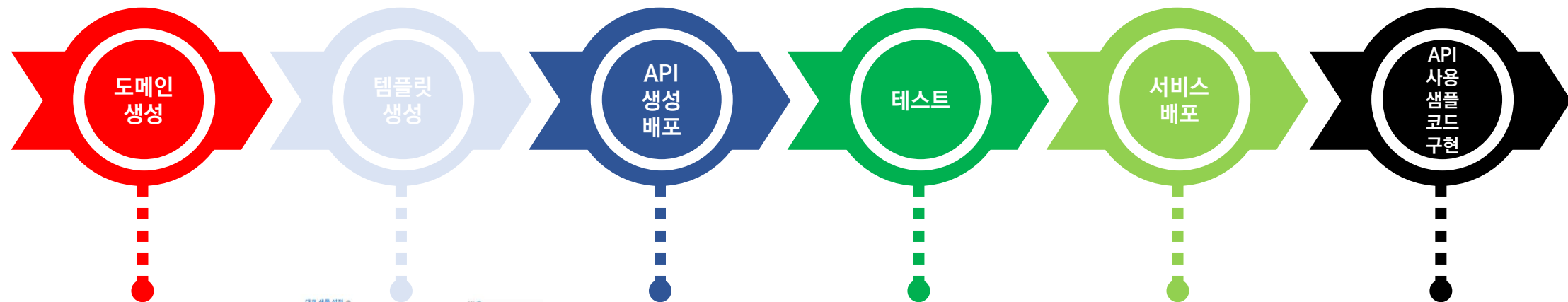
	A	B	C	D	E	F	G
1	순번	항목	01 김규민 1/25	02 김은경 6/25	09 한정연 5/25	04 안수진 7/25	03 김혜수 4/25
2	1	이름	o	o	o	o	o
3	2	성별	o	o		o	o
4	3	나이	o	o	o	o	o
14	13	공복혈당	o		o	o	o
15	14	사구체여과율	o		o		o
16	15	요단백 (정상/약양성/+1/+2/+3/+4)					
17	16	중성지방	o	o	o		
18	17	총콜레스테롤	o	o	o	o	o
19	18	최고-최저 혈압	o	o	o	o	o
20	19	수축기혈압	o	o	o	o	o
21	20	이완기혈압	o	o	o	o	o
22	21	크레아티닌	o	o	o		o
23	22	허리둘레	o		o		o
24	23	HbA1c	o	o	o	o	o
25	24	ALT	o			o	o
26	25	AST	o	o		o	o
27	26	BMI	o		o		o
28	27	FEV1 / FVC	o	o	o	o	o
29	28	FEV1 (L)	o	o	o	o	o
30	29	FEV1 (%)	o	o	o	o	o
31	30	FVC (%)	o	o	o	o	o
32	31	FVC (L)	o	o	o	o	o
33	32	GTP	o	o			o
34	33	HDL	o	o	o	o	o

선택한 모형

-Clova AI-

Clova AI

How to install



- <https://www.istockphoto.com/kr/%EB%B2%A1ED%84%B0/%EB%B0%B0ED%8F%AC-%EC%83%81EC%9E%90-%EC%95%84EC%9D%B4EC%BD%98-gm961052532-262434147>

Clova AI

```
request_json = {
    'images': [
        {
            'format': 'jpg',
            'name': 'demo',
            'templateIds': template_num
        }
    ],
    'requestId': str(uuid.uuid4()),
    'version': 'V2',
    'timestamp': int(round(time.time() * 1000))
}

payload = {'message': json.dumps(request_json).encode('UTF-8')}
files = [
    ('file', open(image_file, 'rb'))
]
headers = {
    'X-OCR-SECRET': secret_key
}
```

```
response = requests.request("POST", api_url, headers=headers, data = payload, files = files)

res = json.loads(response.text.encode('utf8'))
df = pd.json_normalize(res)
df = df['images'][0][0]['fields']

data = list()
column_name = list()
for i in range(len(df)):
    data.append(df[i]['inferText'])
    column_name.append(df[i]['name'])

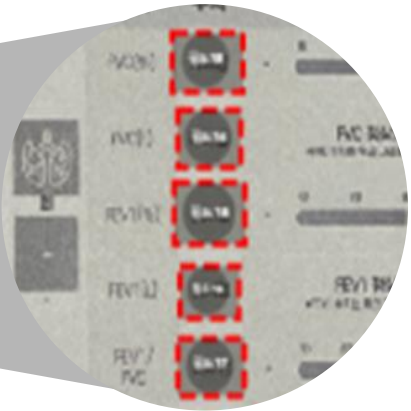
df = pd.DataFrame(data)
df = df.transpose()
df.columns = column_name

if not os.path.exists('/Users/doyunchoi/Desktop/test_6.csv'):
    df.to_csv('/Users/doyunchoi/Desktop/test_6.csv', mode='w', encoding='utf-8-sig')
else:
    df.to_csv('/Users/doyunchoi/Desktop/test_6.csv', mode='a', encoding='utf-8-sig', header=False)
```

〈소스 코드〉



〈테스트 샘플〉



수축기혈압	이완기혈압	최고-최저혈압	총콜레스테롤			LDL	HDL	AST	ALT
113mg/dL	68mmHg	45mmHg	206mg/dL			113mg/dL	74mg/dL	38IU/L	42IU/L
113mg/dL	68mmHg	45mmHg	206mg/dL			113mg/dL	74mg/dL	38IU/L	42IU/L
GTP	공복혈당	HbA1c	PVC	PVC_L	PEV	FEV	FEV1_PVC	크레아티닌	BUN
56IU/L	92mg/dL	미시행	미시행	미시행	미시행	미시행	미시행	0.6mg/dL	16mg/dL
56IU/L	92mg/dL	미시행	미시행	미시행	미시행	미시행	미시행	0.6mg/dL	16mg/dL

〈결과〉

테스트 결과

Output Format: json 다운로드

관독 영역: 20% 300% 56 초기화

행문 및 자동분류 (건강성적표_1)

대표 샘플명

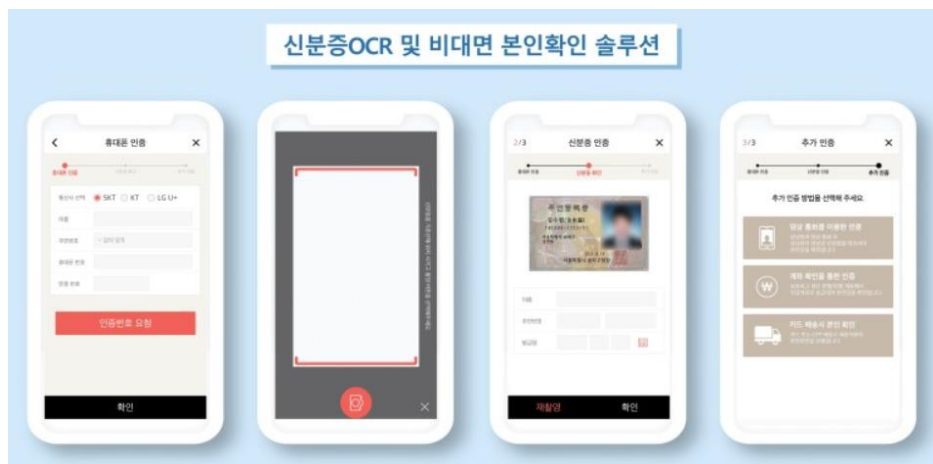
이름	내용
geongang	건강 성적표

필드

관독 영역	이름	내용
필드 01	항목 1	113mg/dl
필드 02	항목 2	68mmHg
필드 03	항목 3	45mmHg
필드 04	Field 04	206mg/dL
필드 05	Field 05	113mg/dl
필드 06	Field 06	74mg/dL
필드 07	Field 07	0.7
필드 08	Field 08	38IU/L
필드 09	Field 09	42IU/L
필드 10	Field 10	56IU/L
필드 11	Field 11	92mg/dL
필드 12	Field 12	미시행
필드 13	Field 13	미시행

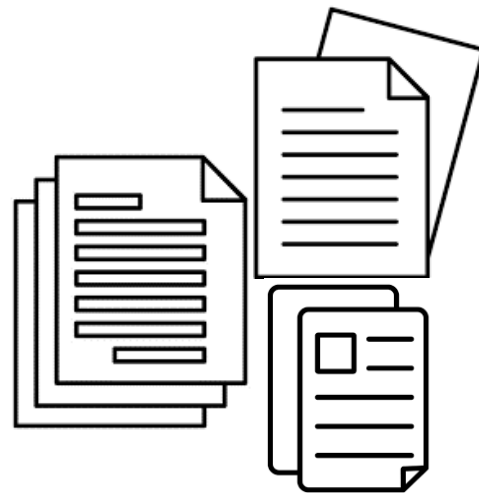
[추가 내용]

추가 내용



<https://blog.naver.com/PostView.nhn?blogId=inzisoft&logNo=222356106818>

〈주민등록증 OCR〉



https://kor.pngtree.com/freepng/document-icon-for-your-project_4812896.html

〈다른 템플릿〉



경청해주셔서 감사합니다.

Q&A