

CAU

# 배송기사 GPS 정보를 활용한 파렛트 공급 서비스 센터 위치 선정 최적화

‘아무것도 물류’ 팀  
유희선, 이지민, 이화정, 전주현

# 목차

1

## GPS Data 분석 과정

: 주어진 운송기사 데이터 정보를 분석

2

## 물류센터 위치 선정

: 물류센터의 위치 선정 시 고려 요인 탐색

3

## 물류센터 위치 2차 후보 선정

: 고려 요인을 통해 2차 후보 선정

4

## 물류센터 입지 범위 선정

: 물류센터 후보들의 위치 입지 범위를 선정

5

## 물류센터 부지 최종 선정

: 물류센터 부지 파악 및 최종 선정

6

## 신규 물류센터 이점

: 교통 편의성 및 사업장 안전성

7

## 물류센터 형태 제언

: 신규 물류센터의 형태 제언

8

## 결론



# 1. GPS Data 분석 과정

주어진 운송기사 데이터 정보 분석

# 1 GPS Data 분석 과정 ; Python

```
# 지도 초기화
map = folium.Map(location=[0, 0], zoom_start=2)

# 다양한 랜덤 색상 리스트 생성
colors = ['#%06x' % random.randint(0, 0xFFFFFF) for i in range(len(file_paths))]

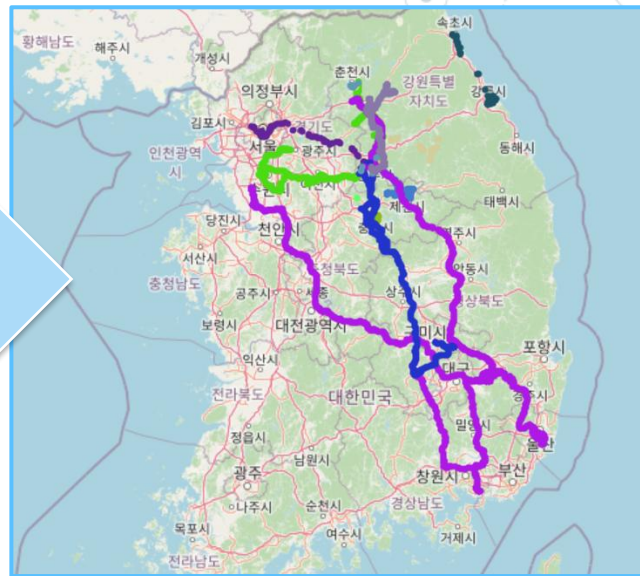
# 역설 파일들을 순회하면서 데이터를 읽어와 지도에 원(점)을 표시
for i, file_path in enumerate(file_paths):
    df = pd.read_excel(file_path)

    latitudes = df['Latitude'].tolist()
    longitudes = df['Longitude'].tolist()

    # 원(점) 추가
    for lat, lon in zip(latitudes, longitudes):
        folium.CircleMarker(location=[lat, lon], radius=1, color=colors[i % len(colors)], fill=True, fill_color=colors[i % len(colors)]).add_to(map)

# 지도를 HTML 파일로 저장
map.save('map_with_circlemarkers.html')

import os
print(os.getcwd())
```



운송기사 A~O까지 각각 운송 경로를 다른 색으로 시각화하기 위해  
랜덤 색상 리스트를 지도에 원(점)을 표시함

# 1 GPS Data 분석 과정 ; R Studio



KCP 물류센터	KPP 물류센터	공동물류센터	권역	본사
항소	구수소		Latitude	Longitude
신원SDC	울진 해구 호안동 1032-2		35.5705434	129.5682775
이태호 라DCC	대구 달서구 호안동 706-2		35.8436701	128.4511271
이태호 공룡센터	광주광역시 서구동 동산로 475-1		35.4768251	126.5148603
황해로 남DCC	경남 김해시 황간로 128-1 (산문동, 황해로물류센터)		35.1885964	128.8215498
지해소 남DCC	경기 용인시 처인구 남사읍 당하로 121 (백석)		37.1486077	127.5143094
KCP(이태호물류센터)	경기 김포시 장안로 44-1		37.4038899	126.7600898
이태호 AIBDCC	경기 시흥시 장왕동 2165-1 KCP-물류센터		37.1004337	126.7147625
홍문로 남DCC	경기 연성시 광곡면 들곡리 941 홍문로물류센터		37.0489264	127.1673205
북한강-연안DCC	충청남도 연기시 백암동 1221-1 (연안동 165-25 (2과반))		37.1473442	127.4526469
기DCC	경기 여주시 가남읍 건창로 142-23		37.1959553	127.5497155
이태호 해수DCC	경기 여주시 용문로 161 (가남읍, 이태호해수물류센터)		37.2649281	127.4540964
남해로 남DCC	경기 오산시 누읍동 141		37.1367608	127.0789901
황해로 남DCC	경기 오산시 경서로 1048-1 (1 (부안동, 황해로산물류센터), 황해로산물류센터 내)		37.1797011	127.0940266
홍문로소하역 오안DCC	경기 오산시 오산동 736 황해로물류센터 8층(동10반도) 2층		37.1486024	127.0673422
황해로 남DCC	경기 용인시 기흥구 하갈로 100-1 (하갈동, 황해로물류센터)		37.255841	127.1012
홍문로소하역 내역	경기 용인시 처인구 남사읍 당하로 121		37.1239944	127.1003103
이태호노르웨이물류센터	경기 용인시 처인구 북곡읍 문현리 136-1		37.1453542	127.4120649
지해소지해로소하역물류센터	경기 용인시 처인구 북곡읍 문현리 136-1		37.2098643	127.2207092
황해로 남DCC	경기 용인시 처인구 북곡읍 문현리 100 (문현역, GS유동)		37.2720232	127.2217793
이태호 AIBDCC	경기 의왕시 마곡동 당일로 462-4 2층		37.2805181	127.365017
지해소 AIBDCC	경기 의왕시 마곡동 당일로 524-2		37.279814	127.391531
이태호 남DCC	경기 의왕시 부림동 마곡로 481		37.2084889	127.4681736
이태호노르웨이물류센터	경기 용인시 기흥구 마곡로 10 (마곡역) (4층 C도크)		37.0000324	127.0787534
이태호노르웨이물류센터	경기 용인시 기흥구 마곡로 10 (마곡역) (8F DRY연하)		37.0000324	127.0787534
기DCC	경기 용인시 기흥구 마곡로 10 (마곡역) (8F DRY연하)		36.9781042	126.845471
KCP(이태호물류센터)	인천 서구 월곶로 556-19		37.59105	126.6323234
홍문로소하역물류센터	경남 함안군 함안읍 관천리 41 (관천리, 홍문로소하역물류센터)		35.367372	128.4761759
지해소 남DCC	경남 함안시 함안읍 11 (관천동, GS(해물물류센터) GS(해물물류) 내)		35.3319344	129.0445529
황해로 남DCC	경기 용인시 용인읍 남사읍 4-1 (백암동) (4층 C도크)		36.8477796	128.4511716
이태호노르웨이물류센터	경기 용인시 용인읍 남사읍 4-1 (백암동) (4층 C도크)		35.9422378	128.8234969
이태호 물류센터	경기 용인시 용인읍 남사읍 4-1 (백암동) (4층 C도크)		35.8775181	129.3159831
남DCC	경기 용인시 용인읍 남사읍 4-1 (백암동) (4층 C도크)		35.868873	129.341549
신원SDC	경기 용인시 용인읍 남사읍 4-1 (백암동) (4층 C도크)		36.871133	129.3684855
예산물류센터	충남 예산군 고대면 대항리 484-22		36.7705394	126.7003036
서DCC	충남 서산시 광안면 광안리 165-10		36.8255479	126.4942927
KCP(이태호물류센터)	충남 서산시 광안면 광안리 165-10		36.8627346	126.5231599
충청남도(이태호물류센터)	충북 충주군 도안면 충주22인로 95 (충주22인로, 황해로산물류센터)		36.815501	127.5698177
KCP 홍문로물류센터	경북 용인구 안성동 1142		35.1172715	126.7581088
KAC(홍문로물류센터)	경북 용인구 안성동 1142		35.1140208	126.8722244
충청남도(이태호물류센터)	전북 완주군 봉동읍 봉산리 926-6		35.9550946	127.1266885

❖로지스올 홈페이지의 국내 네트워크 위치 데이터를 정리한 파일을 활용

❖KCP물류센터/KPP물류센터/공동물류센터의 위치를 분석 결과, 겹치는 데이터가 있음을 파악함

❖해당 엑셀파일 형식을 csv파일 형식으로 변경하여 원활히 분석하도록 만들

# 1 GPS Data 분석 과정 ; R Studio

```
#물류센터 위치 파악하기
KCP물류센터 = read.csv('KCP 물류센터.csv', fileEncoding = 'euc-kr')
KPP물류센터 = read.csv('kpp 물류센터.csv', fileEncoding = 'euc-kr')
공동물류센터 = read.csv('공동물류센터.csv', fileEncoding = 'euc-kr')

#KPP, KCP, 공동 물류 센터 중 겹치는 데이터 정제
물류센터데이터 <- bind_rows(KPP물류센터, KCP물류센터, 공동물류센터)
물류센터데이터 <- distinct(결합된데이터, Latitude, Longitude)
```

```
#운송기사 GPS에 해당하는 범위에 들어있는 물류센터 데이터 필터
물류센터데이터 <- 물류센터 %>%
  filter(Latitude < 38.2 & Latitude > 35.12) %>%
  filter(Longitude < 129.35 & Longitude > 126.8)
```

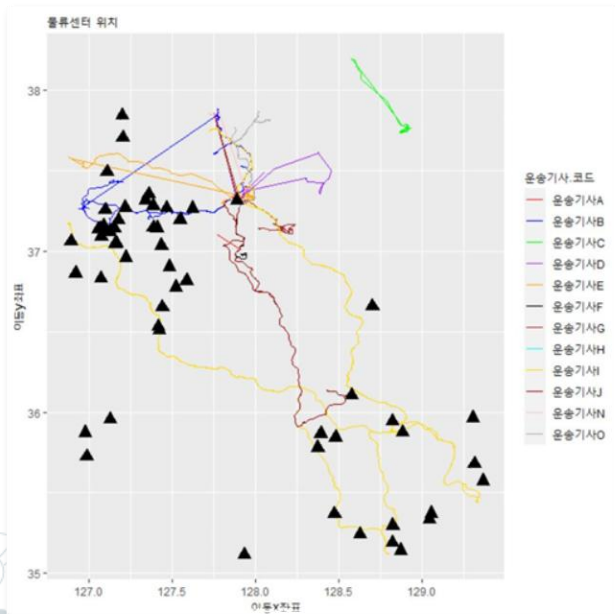
❖ **bind-rows**를 이용해서 물류센터 데이터의 값을 다 합쳤으며 합친 데이터 중 위도와 경도와 겹치는 값을 제거하기 위해 **distinct** 함수를 사용.

❖ 주어진 운송기사 A~ O의 GPS는 위도 35.12~38.2, 경도 126.8~129.35 범위에 해당하기 때문에 물류센터GPS 값도 이에 해당하는 범위를 추출

# 1 GPS Data 분석 과정 ; R Studio

```
#물류 센터 위치 시각화
ggplot() +
  geom_path(data = 전체데이터, aes(x = 이동.x좌표, y = 이동.y좌표, color = 운송기사.코드)) +
  geom_point(data = '물류센터데이터', aes(x = Longitude, y = Latitude), size = 4, shape = 17) +
  ggtitle("'물류센터 위치'") +
  xlab("'이동x좌표'") +
  ylab("'이동y좌표'") +
  scale_color_manual(values = 색갈)
```

## ggplot2 사용



01

- 이는 운송기사 B와 운송기사 I가 물류센터에서 많이 정차하였음을 추측할 수 있음

02

- 정차 시간이 단순히 휴식시간만 존재하는 것이 아닌 물류센터 정차 후 **작업 시간**도 있음을 추측할 수 있음

# 1 GPS Data 분석 과정 ; Excel

## ❖ 엑셀 함수를 이용한 운송기사 A~O의 정차시간 데이터 생성

	A	B	C	D	E	F
1	Time	Longitude	Latitude	Speed	휴식시간	
2	2023-05-03 0:00	127.926	37.395	0		0
3	2023-05-03 0:08	127.926	37.3949	0	0:08	정차
4	2023-05-03 0:11	127.9259	37.3949	0	0:03	0
5				0	0:14	정차
6				2	0:09	정차
7				0	0:06	정차
8				1	0:05	정차
9				10	0:01	0
10				0	0:01	0
11				0	0:01	0
12	2023-05-03 0:50	127.9230	37.3934	0	0:02	0
13	2023-05-03 0:55	127.9236	37.3953	0	0:05	정차
14	2023-05-03 1:59	127.9235	37.3952	8	1:04	정차
15	2023-05-03 2:00	127.9235	37.3963	2	0:01	0
16	2023-05-03 2:01	127.9239	37.3965	0	0:01	정차
17	2023-05-03 2:49	127.9239	37.3965	7	0:48	정차
18	2023-05-03 2:50	127.9238	37.396	0	0:01	0

=IF(OR(AND(위도A=위도B, 경도A=경도B, 속도A=0), 시간차>5분), " 정차 ",0)

## 1 정차시간 구하기

- 1) 특정 Time의 GPS 위도, 경도가 직후 Time의 GPS 위도, 경도와 같고 Speed=0
- 2) 특정 Time과 바로 직후 Time의 차이가 5분 이상

만족 → 정차

만족 X → 0



# 1 GPS Data 분석 과정 ; Excel

❖ 엑셀 함수를 이용한 운송기사 A~O의 정차시간 데이터 생성

	A	B	C	D	E	F
1	Time	Longitude	Latitude	Speed	휴식시간	
2	2023-05-03 0:00	127.926	37.395	0		0
3	2023-05-03 0:08	127.926	37.3949	0	0:08	정차
4	2023-05-03 0:11	127.9259	37.3949	0	0:03	0
5	2023-05-03 0:25	127.9258	37.3949	0	0:14	정차
6	2023-05-03 0:34	127.9258	37.3949	2	0:09	정차
7	2023-05-03 0:40	127.9257	37.3949	0	0:06	정차
8	2023-05-03 0:45	127.9259	37.3949	1	0:05	정차
9	2023-05-03 0:46	127.9242	37.3957	10	0:01	0
10	2023-05-03 0:47	127.9234	37.3956	0	0:01	0
11	2023-05-03 0:48	127.9235	37.3953	0	0:01	0
12	2023-05-03 0:50	127.9236	37.3954	0	0:02	0
13	2023-05-03 0:55	127.9236	37.3953	0	0:05	정차
14	2023-05-03 1:59	127.9235	37.3952	8	1:04	정차
15	2023-05-03 2:00	127.9235	37.3963	2	0:01	0
16	2023-05-03 2:01	127.9239	37.3965	0	0:01	정차
17	2023-05-03 2:49	127.9239	37.3965	7	0:48	정차
18	2023-05-03 2:50	127.9238	37.396	0	0:01	0

## 2 정차시간이 5분 이하인 데이터를 삭제

5분 이하의 정차시간이면서 속도가 0인  
데이터



해당 구간에서 신호에 걸렸거나  
교통체증이 있었음을 추측할 수 있음

# 1 GPS Data 분석 과정 ; R Studio

## ❖ K-mean clustering을 통한 물류센터 위치 추정

- 사용목적 - 데이터를 유사한 패턴, 그룹으로 묶어서 그룹 간의 차이를 이해하기 위함

```
#k-mean을 통해 물류센터 위치 추정
```

```
#NA 값 제거
```

```
휴식시간 <- na.omit(휴식시간)
```

```
# 시간을 분 단위로 변환
```

```
휴식시간$휴식시간 <- as.POSIXct(휴식시간$휴식시간, format = "%H:%M")
```

```
휴식시간$휴식시간 <- hour(휴식시간$휴식시간) * 60 + minute(휴식시간$휴식시간)
```

```
#운송기사코드를 숫자로 표시
```

```
휴식시간 <- 휴식시간 %>%
```

```
  mutate(Category_numeric = as.numeric(factor(운송기사.코드, levels = unique(운송기사.코드))))
```

```
# 숫자형 열만 선택
```

```
numeric_columns <- sapply(휴식시간, is.numeric)
```

```
numeric_GPS <- 휴식시간[, numeric_columns]
```

```
numeric_GPS <- na.omit(numeric_GPS)
```

# 1 GPS Data 분석 과정 ; R Studio

❖ 1차적으로 최적의 물류센터 후보 10개를 선정하기 위해, 클러스터 수를 10개로 설정

```
# 클러스터 수 설정
num_clusters <- 10

# kmeans 클러스터링
kmeans_result <- kmeans(numeric_GPS, centers = num_clusters)

# 클러스터 중심 좌표
cluster_centers <- as.data.frame(kmeans_result$centers)

# 시각화
ggplot() +
  geom_point(data = numeric_GPS, aes(x = Longitude, y = Latitude),
            color = "blue") +
  geom_point(data = cluster_centers, aes(x = Longitude, y = Latitude),
            color = "red", size = 3, shape = 4) +
  ggtitle("운송 기사 위치 및 물류 센터 예상 위치") +
  xlab("Longitude") +
  ylab("Latitude") +
  theme_minimal()
```

## centers :

- 클러스터의 개수를 나타내는 매개변수로서 클러스터의 개수 지정에 사용

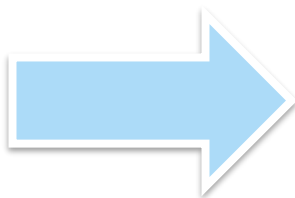
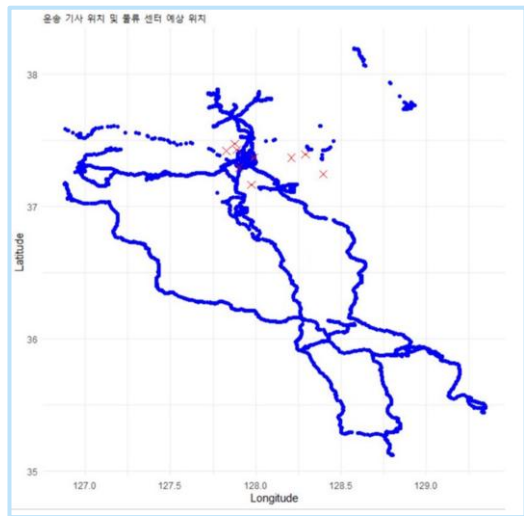
## kmean\_result :

- K-mean의 함수를 클러스터링 수행 후 그 결과를 나타낸 리스트

○ **Kmean\_result**로 클러스터의 각 특성의 평균값을 나타내어 **클러스터의 중심좌표 표현**

# 1 GPS Data 분석 과정 ; R Studio

## ❖ 클러스터링 수행 후 중심좌표 표현 결과



	Latitude	Longitude
1	37.47599	127.8771
2	37.39880	127.9497
3	37.29447	127.9018
4	37.39644	128.2913
5	37.43049	127.8914
6	37.42215	127.8281
7	37.16545	127.9740
8	37.36962	128.2095
9	37.24565	128.3956
10	37.37383	127.9914

해당 클러스터들의 각 특성의 평균값을 나타내어 표현한 클러스터의 중심좌표들을 운송기사의 운송경로와 함께 시각화하고 해당 좌표는 다음과 같음

A decorative network diagram in the top-left corner, featuring a complex web of interconnected nodes and lines. The nodes are represented by circles of varying sizes, some solid and some hollow, connected by thin lines. The overall structure is organic and branching, resembling a molecular or biological network.

2.

# 물류센터 위치 선정

물류센터의 위치 선정 시 고려 요인 탐색

A decorative network diagram in the bottom-right corner, similar to the one in the top-left. It shows a complex web of interconnected nodes and lines, with nodes represented by circles of varying sizes, some solid and some hollow, connected by thin lines. The structure is organic and branching.

## 2 물류센터 위치 선정



### <입지가능 지역>

- 1) 교통의 편의성  
-> 고속도로 IC로부터 5km이내 지역
- 2) 사업장의 안전성  
-> 좌표상 가장 가까운 소방시설과의  
거리 고려



### <입지제한 지역>

- 1) 주거, 상업, 학교건물 반경 200m이내
- 2) 산악 지역
- 3) 도시지역
- 4) 문화재 보호구역
- 5) 개발제한구역
- 6) 하천, 호수, 저수지(지반 경도 문제)

유한솔, 김남석. 2015. GIS 및 하이브리드 입지선정문제를 이용한 공동물류센터 입지선정. 대한교통학회

A decorative network diagram in the top-left corner, featuring a complex web of interconnected nodes and lines. The nodes are represented by circles of varying sizes, some with concentric rings, and the lines are thin and grey. The diagram is partially cut off by the top and left edges of the frame.

3.

# 물류센터 위치 2차 후보

고려 요인을 통해 2차 후보 선정

A decorative network diagram in the bottom-right corner, similar to the one in the top-left. It shows a cluster of nodes and connecting lines, with some nodes highlighted by concentric circles. The diagram is partially cut off by the bottom and right edges of the frame.

### 3 물류센터 위치 2차 후보 선정

1차 후보	좌표	산악지역	주거지역, 학교, 상업지역 200m 이내	고속도로 IC가 5km 이내
1	37.47599, 127.8771	O	O	X
2	37.39880, 127.9497	X	X	O
3	37.29447, 127.9018	O	O	터널 한복판
4	37.39644, 128.2913	O	캠핑장에 위치	X
5	37.43049, 127.8914	X	X	O
6	37.42215, 127.8281	O	O	O
7	37.16545, 127.9740	X	X	X
8	37.36962, 128.2095	O	X	X
9	37.24565, 128.3956	X	O	X
10	37.37383, 127.9914	X	X	O

- 1차 후보 10개 지역을 [입지제한 지역]의 고려요소에 따라 입지제한 요소가 존재하는지 여부를 O,X로 표시

→ 입지제한 요소 'O' 지역 제외

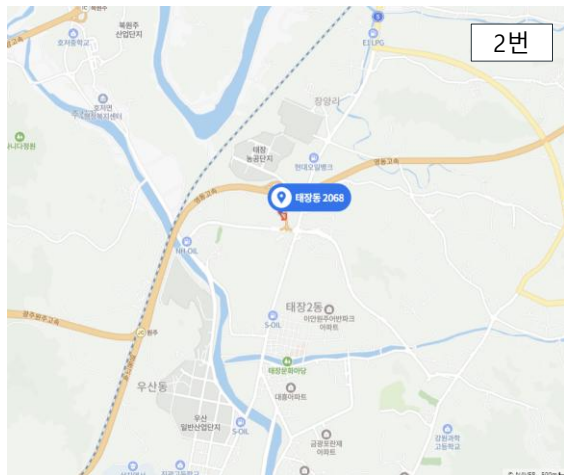
- 1차 후보 10개 지역 중 [입지가능 지역] 요소인 '고속도로 IC로부터 5km 이내 지역' 인지 여부를 O,X로 표시

→ 입지가능 요소 'X' 지역들을 제외

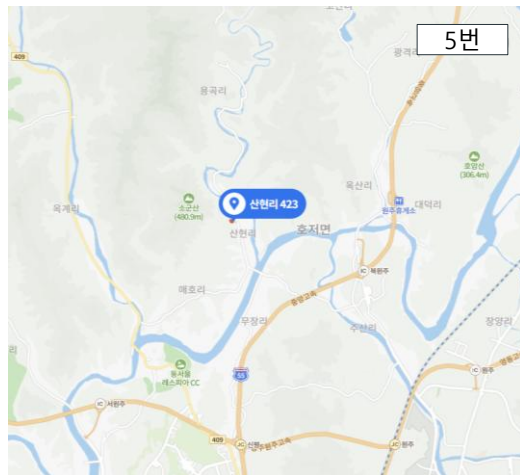


### 3 물류센터 위치 2차 후보 선정

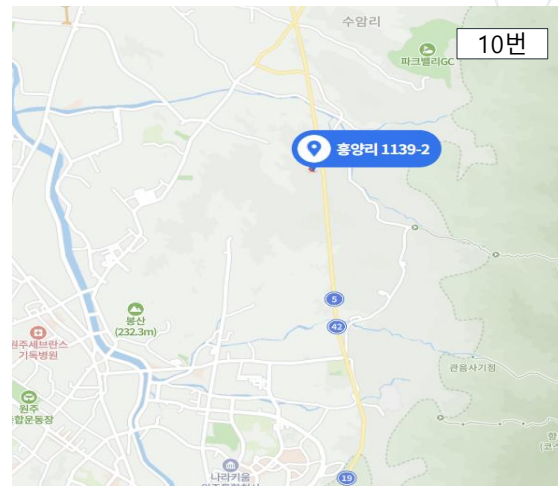
- 위 슬라이드에서 설명된 과정을 통해 뽑은 후보 3개 지역은 다음과 같음



37.39880, 127.9497  
강원특별자치도 원주시 태장동 2068



37.43049, 127.8914  
강원특별자치도 원주시 호저면 산현리 423



37.37383, 127.9914  
강원특별자치도 원주시 소초면 홍양리 1139-2

A decorative network diagram in the top-left corner, featuring a complex web of interconnected nodes and lines, with some nodes highlighted in grey and others in white.

4.

# 물류센터 입지 범위 선정

물류센터 후보들의 위치 입지 범위를 선정

A decorative network diagram in the bottom-right corner, featuring a complex web of interconnected nodes and lines, with some nodes highlighted in grey and others in white.

## 4 물류센터 입지 범위 선정

```
import folium
from geopy.distance import geodesic

# 주어진 3개의 GPS 좌표
gps_coordinates = [(37.37383, 127.9914), (37.39880, 127.9497), (37.43049, 127.8914)]

# 목표 지점 좌표
targets = {
    "원주IC": (37.400893, 127.950230),
    "원주JC": (37.385957, 127.930703),
    "북원주IC": (37.419664, 127.920615)
}

# 초기값 설정
min_distance = float('inf')
closest_coordinate = None
closest_target = None

# 각 GPS 좌표와 목표 지점 간의 거리 계산 및 가장 가까운 좌표 찾기
for coordinate in gps_coordinates:
    for target_name, target_coordinate in targets.items():
        distance = geodesic([coordinate, target_coordinate]).meters
        if distance < min_distance:
            min_distance = distance
            closest_coordinate = coordinate
            closest_target = target_name

# 결과 출력
print(f"가장 가까운 목표 지점: {closest_target}")
print(f"가장 가까운 좌표: {closest_coordinate}")
```



```
map_object = folium.Map(location=map_center, zoom_start=12)

# GPS 좌표 마커 추가 (파란색 원)
for coordinate in gps_coordinates:
    folium.CircleMarker(location=coordinate, radius=5, color='blue', fill=True, fill_color='blue',
                        fill_opacity=0.7, popup="GPS 좌표").add_to(map_object)

# 목표 지점 마커 추가 (파란색 원)
for target_name, target_coordinate in targets.items():
    # 가장 가까운 목표 지점은 파란색 원
    folium.CircleMarker(location=target_coordinate, radius=5, color='blue', fill=True, fill_color='blue',
                        fill_opacity=0.7, popup=target_name).add_to(map_object)

# 가장 가까운 좌표는 빨간색 원
folium.CircleMarker(location=closest_coordinate, radius=5, color='red', fill=True, fill_color='red',
                    fill_opacity=0.7, popup="가장 가까운 좌표").add_to(map_object)

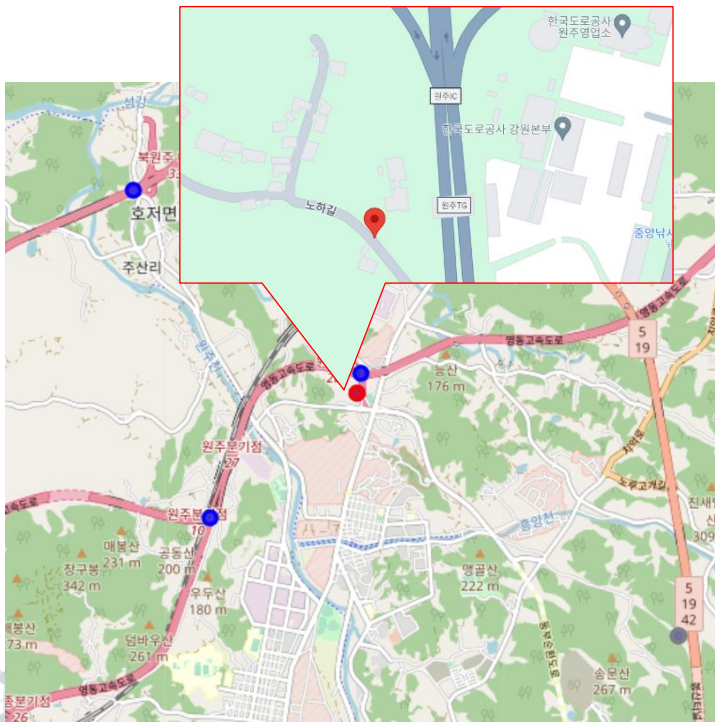
# 나머지 좌표는 회색 원
for coordinate in gps_coordinates:
    if coordinate != closest_coordinate:
        folium.CircleMarker(location=coordinate, radius=5, color='gray', fill=True, fill_color='gray',
                            fill_opacity=0.7, popup="다른 좌표").add_to(map_object)

# 결과 지도 저장
map_object.save("result_map2.html")
```

Python을 이용해 이전 슬라이드의 3개 물류센터 위치 후보들 중, 고속도로IC와 가장 가까운 물류센터 위치를 선정함

## 4 물류센터 입지 범위 선정

- 최종 선정 입지 범위의 좌표: (37.3988, 127.9497)



**파란색 점:** 주변 고속도로 IC 좌표

**회색 점:** 직전 슬라이드의 3개 후보군  
중 최종 선정되지 않은 나머지 2곳의  
좌표

**빨간색 점** : 표시된 위치가 가장 고속도로  
IC와 가까운 좌표로 최종 선정된 위치 의미

A decorative network diagram in the top-left corner, featuring a complex web of interconnected nodes and lines, with some nodes highlighted in grey and others in white.

5.

# 물류센터 부지 최종 선정

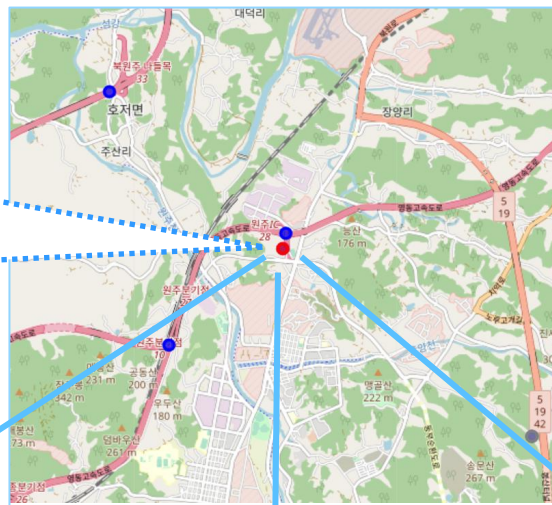
물류센터 부지 파악 및 최종 선정

A decorative network diagram in the bottom-right corner, featuring a complex web of interconnected nodes and lines, with some nodes highlighted in grey and others in white.

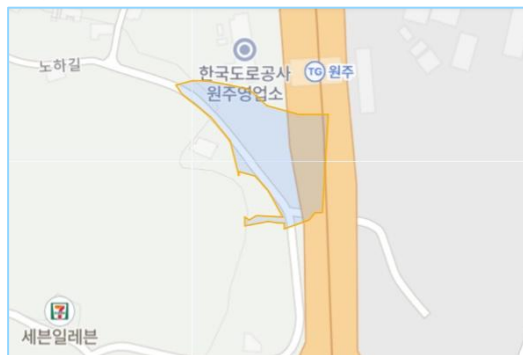
## 5

## 물류센터 부지 최종 선정

최종 선정 입지 범위의 좌표:  
(37.3988, 127.9497)의 인접  
지역 중 세 곳을 임의로 선정



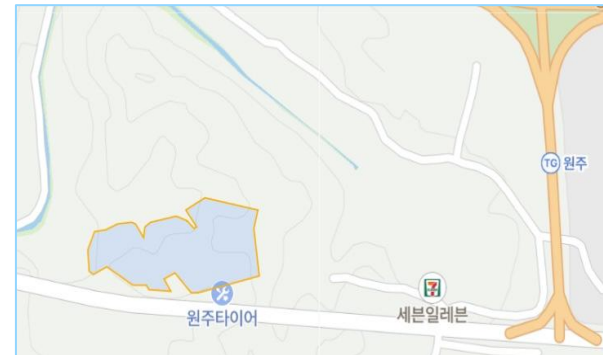
(a)

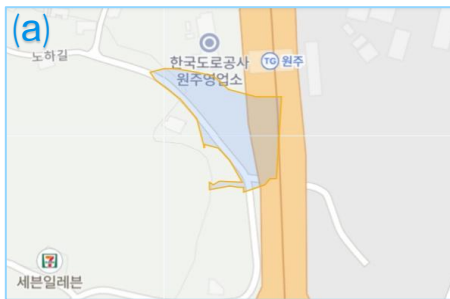


(b)



(c)





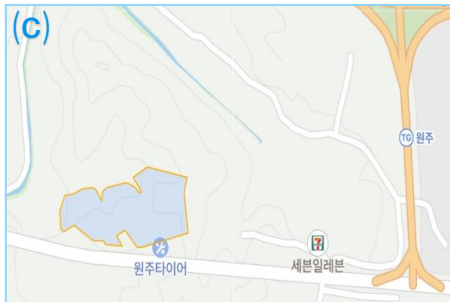
부지 넓이:  $4,573\text{m}^2 / 1,3833.33\text{평}$   
 → (c)와 비교하여 물류센터 건설 상 제약  
 판단

➡ 최종 부지에서  
제외



한국도로공사 건물과 인접  
 → 물류센터 건설에 제약 발생 가능성

➡ 최종 부지에서  
제외



부지 넓이:  $19,127\text{m}^2 / 5,785.92\text{평}$

➡ 최종 선택

A decorative network diagram in the top-left corner, featuring a complex web of interconnected nodes and lines, with some nodes highlighted in grey and others in white.

6.

# 신규 물류센터의 이점

교통 편의성 및 사업장 안전성

A decorative network diagram in the bottom-right corner, featuring a complex web of interconnected nodes and lines, with some nodes highlighted in grey and others in white.



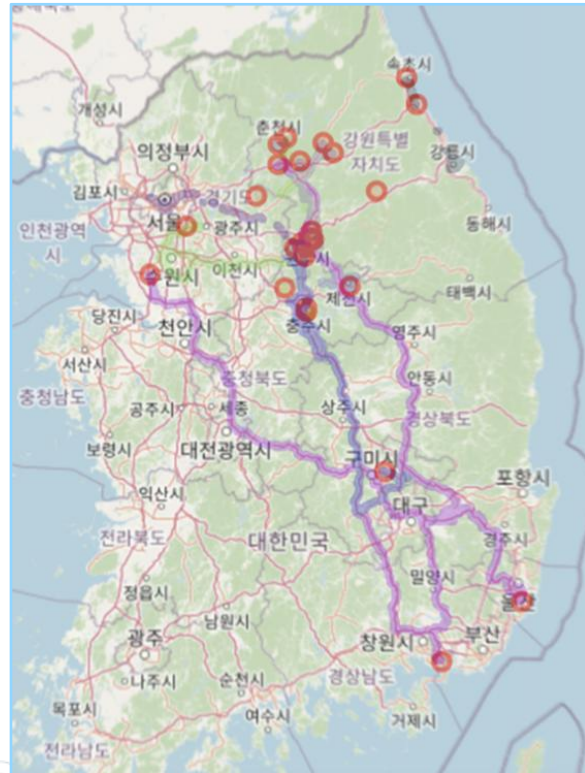
```

1 import folium
2 from folium import plugins
3 import shutil
4
5 # 기존 지도 파일 로드
6 existing_map_filepath = "C:\\pystudy\\pythonstudy\\map_with_circlemarkers_realpath.html"
7 new_map_filepath = "combined_map_with_existing.html"
8
9 # 기존 지도 파일을 복사본으로 새로운 맵 생성
10 shutil.copyfile(existing_map_filepath, new_map_filepath)
11
12 # 새로운 맵에 기존 지도 데이터 로드
13 new_map = folium.Map(location=[37.5, 127.5], zoom_start=7)
14
15 # 새로운 좌표 리스트
16 new_coordinates = [
17     (37.083, 127.91797),
18     (37.12849, 127.778102),
19     (37.88807, 127.77075),
20     (37.431823, 127.12692),
21     (38.28198, 128.57387),
22     (38.06023, 128.64148),
23     (37.3731, 127.9414),
24     (37.61881, 128.37833),
25     (37.48918, 127.93219),
26     (37.58816, 127.58816),
27     (38.9813, 127.9129),
28     (37.31358, 127.88919),
29     (37.1197, 128.1988),
30     (37.31466, 127.88853),
31     (37.36165, 127.95944),
32     (37.36165, 127.95944),
33     (37.17477, 126.88601),
34     (37.75884, 127.72980),
35     (35.11993, 128.80148),
36     (35.43989, 129.33277),
37     (37.34804, 127.88801),
38     (36.13828, 128.42892),
39     (37.3121, 127.8377),
40     (37.271882, 127.90588),
41     (37.8573, 127.7281),
42     (37.271882, 127.90588),
43     (37.77366, 127.87410),
44     (37.81555, 128.88864),
45     (37.88851, 128.01962),
46     (37.31466, 127.88853),
47 ]
48
49 # 새로운 좌표를 맵에 원자로 추가
50 for coord in new_coordinates:
51     folium.CircleMarker(location=coord, radius=5, color='red', fill=True, fill_color='red').add_to(new_map)
52
53 # 새로운 맵을 HTML 파일로 저장
54 new_map.save(new_map_filepath)

```

운송기사 GPS 지도에서  
출발지 좌표 추출

운송기사(A~O)의  
출발지 좌표 및  
경로를 동시에  
시각화



네이버지도, 4층 이상 특수화물차, 평일 오후 3시 기준

운송 기사명	출발지 좌표	출발지 주소	로지스올 원주물류센터 (KPP파렛트 강원영업소)	신규 물류센터	로지스올 근접	신규 물류센터 근접
A	37.003, 127.91757	충청북도 충주시 충원대로 862	0:37	0:45	0:08	-
	37.105491, 127.775102	충청북도 충주시 양성면 사미품무골길 66-7	0:42	0:50	0:08	-
B	37.88867, 127.77675	강원도 춘천시 동면 만천리 340-36	1:22	0:56	-	0:26
	37.431823, 127.12693	경기도 성남시 중원구 성남동 1668	1:13	0:54	-	0:19
C	38.20190, 128.57387	강원도 속초시 교동 737-6	2:06	1:50	-	0:16
	38.06623, 128.64148	강원도 양양군 손양면 간리 1-4	1:56	1:32	-	0:24
D	37.3731, 127.9414	강원도 원주시 우산공단길 23	0:20	0:06	-	0:14
	37.61081, 128.37833	강원도 평창군 봉평면 기풍로 92	1:11	0:53	-	0:18
E	37.40918, 127.95219	강원도 원주시 소초면 복원로 2955	-	-	-	-
	37.58816, 37.58816	서울특별시 마포구 상암동 1756-3	2:17	1:58	-	0:19
F	36.9813, 127.9129	충청북도 충주시 칠금동 41-7	0:40	0:48	0:08	-
	37.313505, 127.889191	강원도 원주시 흥업면 대안리 400	0:11	0:20	0:09	-
G	37.1197, 128.1988	충청북도 제천시 청풍호로10길 31	0:44	0:34	-	0:10
	37.31466, 127.88853	강원도 원주시 흥업면 사제리 911-2	-	-	-	-
H	37.36165, 127.95944	강원도 원주시 치악로 2033	0:23	0:10	-	0:13
	-	-	-	-	-	-
I	37.17477, 126.88601	경기도 화성시 팔탄면 울암리 98-31	1:23	1:17	-	0:06
	37.75084, 127.72980	강원도 춘천시 동산면 군자리 1011-4	0:55	0:47	-	0:08
	35.11993, 128.80148	경상남도 창원시 진해구 용동 1동 623	3:55	3:30	-	0:25
	35.43989, 129.33277	울산광역시 울주군 온산읍 화산리 1006-5	3:26	3:12	-	0:14
J	37.34064, 127.88061	강원도 원주시 흥업면 사제리 1330-4	0:14	0:18	0:04	-
	36.13828, 128.426928	경상북도 구미시 옥계동 807	2:19	2:03	-	0:16
L	37.3221, 127.8377	강원도 원주시 문막읍 동건길 9-7	0:12	0:15	0:03	-
	37.271882, 127.905586	강원도 원주시 흥업면 매지리 1202-2	0:11	0:18	0:07	-
N	37.8573, 127.7281	강원도 춘천시 춘주로 174	1:02	0:53	-	0:09
	37.271882, 127.905586	강원도 원주시 흥업면 매지리 1202-2	0:11	0:18	0:07	-
O	37.77166, 127.87410	강원도 홍천군 북방면 성동로 879-5	0:57	0:48	-	0:09
	37.81555, 128.08864	강원도 홍천군 내촌면 도관리 465-1	1:22	1:00	-	0:22
	37.86851, 128.01962	강원도 홍천군 두촌면 자문리 816-4	1:03	0:55	-	0:08
	37.31466, 127.88853	강원도 원주시 흥업면 사제리 911-2	-	-	-	-

기존 물류센터와 신규 위치(원주IC)를  
목적지로 선정 후  
소요시간 산정

신규 물류센터(원주IC)와  
근접하여 소요시간이 적은  
곳을  
파란색으로 표시

기존 물류센터와 근접하여  
소요시간이 적은 곳을  
노란색으로 표시

출발지 좌표가 기존물류센터, 원주IC인 곳은 제외



### 안전한 사업장 구현

근로자의 안전보건활동에 대한 참여를 유도하고,  
안전 및 보건에 관한 임직원 의견을 청취하고 개선조치하여  
안전한 사업장을 만든다.  
내외부 위험요인으로부터 임직원 및 지역주민 보호를 위해  
재해 대비 및 대응체계를 구축하여 운영한다.

▼국가화재정보시스템, 2018/12/01~2023/12/01 창고시설 화재 발생

구분		화재건수	사망	부상	인명피해 계	부동산피해(천원)	동산피해(천원)	재산피해(천원)	재산피해/건당(천원)
합계		6965	68	253	321	372009273	673285576	1045294849	150078
창고시설	소계	6965	68	253	321	372009273	673285576	1045294849	150078
	냉장 냉동창고	440	46	39	85	41382970	60220524	101603494	230917
	창고 물품저장소	4354	13	159	172	310954650	591008967	901963617	207157
	하역장	18	0	3	3	633569	1766853	2400422	133357
	기타 창고	2153	9	52	61	19038084	20289232	39327316	18266



포쓰저널 · 2021.06.17.

### 쿠팡 덕평물류센터 화재 악화 건물 전소...소방관 1명 실종

포쓰저널 문기수 기자 17일 오후 경기도 이천시 마장면 쿠팡 덕평물류센터 모습./연합17일 오전 1차 화재 발생 직후 덕평물류센터 모습./연합 17일 밤 경기도 이천시 마장면 쿠팡 덕평물류센터 화재 현장에서 소방관...



◆ 안전한 사업장을 운영하고자 하는 로지스올의 **경영이념** 고려

- ◆ 통계 및 실제사례 → 물류현장의 빈번한 화재 발생
- ◆ 화재의 초기진압은 막대한 물적/인적 피해를 위해 매우 중요
- ◆ **소방시설**이 물류센터 인근에 있으면 **사업장의 안전한 관리**에 이점 ↑

따라서, 기존 물류센터와 신규 물류센터 중 소방시설과 더 가까운 곳을 Python을 통해 구하고자 함

기관명	구주소	Latitude	Longitude
학성119안전센터	강원도 원주시 학성동 학성길 122	37.3536864	127.9448591
우산119안전센터	강원도 원주시 우산동 우산공단길 9	37.3741155	127.9418168
원주소방서-흥업-119 안전센터	강원도 원주시 귀래면 북원로 53-5	37.1660345	127.884201
원주소방서-문막-119 안전센터	강원도 원주시 문막읍 문막시장1길 90	37.3034438	127.8144975
원주소방서-원주_신림-119 안전센터	강원도 원주시 신림면 치악로 27	37.2297912	128.0789601
원주소방서-현장대응과(명륜)-119 안	강원도 원주시 남원로 487	37.3284147	127.9442919
원주소방서-단구-119 안전센터	강원도 원주시 강변로 255 단구119안전센터	37.2297349	128.0790113
원주소방서	강원도 원주시 남원로 487	37.3284147	127.9442919
태장119안전센터	강원도 원주시 태장동 태장공단길 4 원주소방서태장11	37.4014894	127.9525462

1. 원주시 내 소방시설 9개의 좌표(위도, 경도)를 파악한다.

2. 기존 물류센터와 신규 물류센터 각각에 대해 가장 가까운 소방시설과의 거리를 계산한다.

```
from math import radians, sin, cos, sqrt, atan2

def haversine(lat1, lon1, lat2, lon2):
    R = 6371.0 # 지구 반지름 (단위: km)

    lat1, lon1, lat2, lon2 = map(radians, [lat1, lon1, lat2, lon2])

    dlat = lat2 - lat1
    dlon = lon2 - lon1

    a = sin(dlat / 2)**2 + cos(lat1) * cos(lat2) * sin(dlon / 2)**2
    c = 2 * atan2(sqrt(a), sqrt(1 - a))

    distance = R * c # 거리 (단위: km)
    return distance

def find_closest_fire_station(logistics_center, fire_station_locations):
    closest_location = None
    min_distance = float('inf')

    for location in fire_station_locations:
        lat, lon = location
        distance = haversine(logistics_center[0], logistics_center[1], lat, lon)

        if distance < min_distance:
            min_distance = distance
            closest_location = location

    return closest_location, min_distance

# 물류센터와 소방시설 좌표
logistics_center = (37.3988, 127.9497)
existing_logistics_center = (37.313272, 127.890398)

fire_station_locations = [
    (37.3536864, 127.9448591),
    (37.3741155, 127.9418168),
    (37.1660345, 127.884201),
    (37.3034438, 127.8144975),
    (37.2297912, 128.0789601),
    (37.3284147, 127.9442919),
    (37.2297349, 128.0790113),
    (37.3284147, 127.9442919),
    (37.4014894, 127.9525462)
]
```

```
# 중복된 좌표 제거
fire_station_locations = list(set(fire_station_locations))

# 가장 가까운 소방시설 찾기
closest_existing_fire_station, distance_existing = find_closest_fire_station(existing_logistics_center, fire_station_locations)
closest_new_fire_station, distance_new = find_closest_fire_station(logistics_center, fire_station_locations)

# 결과 출력
print("기존 물류센터와 가장 가까운 소방시설:", closest_existing_fire_station)
print("거리:", distance_existing, "km")

print("\n새로운 물류센터와 가장 가까운 소방시설:", closest_new_fire_station)
print("거리:", distance_new, "km")

# 비교
if distance_existing < distance_new:
    print("\n기존 물류센터가 더 가까워요.")
elif distance_existing > distance_new:
    print("\n새로운 물류센터가 더 가까워요.")
else:
    print("\n두 물류센터와의 거리가 같아요.")
```

3. 두 거리를 비교하여, 기존과 신규 중 어느 쪽이 더 소방시설과 가까운 지 결과값을 도출한다.

#### 4. 결과를 확인한다.

```
Python 3.10.4 (tags/v3.10.4:9d38120, Mar 23 2022, 23:13:41) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

===== RESTART: C:\Users\W82109\Downloads\가장 가까운 소방시설과 물류센터 간의 거리.py =====
기존 물류센터와 가장 가까운 소방시설: (37.3284147, 127.9442919)
거리: 5.054441869666674 km

새로운 물류센터와 가장 가까운 소방시설: (37.4014894, 127.9525462)
거리: 0.3906924856170558 km

새로운 물류센터가 더 가까워요.
```

```
Python 3.10.4 (tags/v3.10.4:9d38120, Mar 23 2022, 23:13:41) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

===== RESTART: C:\Users\W82109\Downloads\가장 가까운 소방시설과 물류센터 간의 거리.py =====
기존 물류센터와 가장 가까운 소방시설: (37.3284147, 127.9442919)
거리: 5.054441869666674 km

새로운 물류센터와 가장 가까운 소방시설: (37.4014894, 127.9525462)
거리: 0.3906924856170558 km

새로운 물류센터가 더 가까워요.
```

### [결과값 해석]

\* 기존 물류센터와 가장 가까운 소방시설 [거리] : 원주소방서-현장대응과(명륜)-119안전센터 [5.054441869666674 km]

\* 신규 물류센터와 가장 가까운 소방시설[거리] : 태장 119안전센터 [거리: 0.3906924856170558 km]

→ 신규 물류센터는 기존 물류센터보다 소방시설과의 거리가 4.663749384 km 더 가까워,  
화재 발생 시 빠른 초기 진압으로 피해 규모 최소화 가능

A decorative network diagram in the top-left corner, featuring a complex web of interconnected nodes and lines. The nodes are represented by small circles, some of which are larger and have concentric circles, suggesting a hierarchical or central structure. The lines are thin and gray, connecting the nodes in a non-linear fashion.

7.

# 물류센터 형태 제언

신규 물류센터의 형태 제언

A decorative network diagram in the bottom-right corner, similar to the one in the top-left. It shows a cluster of nodes connected by lines, with some nodes being larger and more prominent than others, indicating a focal point or a specific type of node within the network.



### ▷ 경기 이천시 부발읍에 위치한 L사 통합물류센터

- 경사지에 지어 지하층을 효율적으로 활용
- 화주맞춤형 설계방식으로 비용과 공사기간도 크게 줄임

물류센터를 지을때, 원주시의 도시계획 조례에 따라 '4층 이하의 창고 또는 집배송시설 건설 가능.

#### 원주시 도시계획 조례 별표16

(자연녹지지역 안에서 건축할 수 있는 건축물)

2. 우리 시 조례에 의하여 건축할 수 있는 건축물(4층 이하의 건축물에 한한다)

가. 「건축법 시행령」 별표 1 제18호가목의 창고(농업·임업·축산업·수산업용으로 쓰는 것은 제외한다) 및 같은 호 라목의 집배송 시설

「건축법 시행령」 별표 1 제18호. 창고시설(위험물 저장 및 처리 시설 또는 그 부속용도에 해당하는 것은 제외한다)

가. 창고(물품저장시설로서 「물류정책기본법」에 따른 일반창고와 냉장 및 냉동 창고를 포함한다) 라. 집배송 시설

- ▶ 부지 개발 시 **경사지를 이용하면** 평지일 때보다 더 효율적이며, 경사지를 이용하여 지하층 건설 가능
- ▶ 지하층으로 인정되면 건폐율과 용적률에 산입되지 않음 (대한민국 건축관계법)
- ▶ 따라서, 최대 건폐율과 용적률 이상의 실제 연면적을 만드는 물류센터가 개발 가능함

건폐율: 대지 면적에 대한 건물의 바닥 면적 비율, 건축 밀도를 나타낸다  
 용적률: 전체 대지면적에서 건물 각층의 면적을 합한 연면적이 차지하는 비율, 건축물에 의한 토지의 이용도를 보여준다.



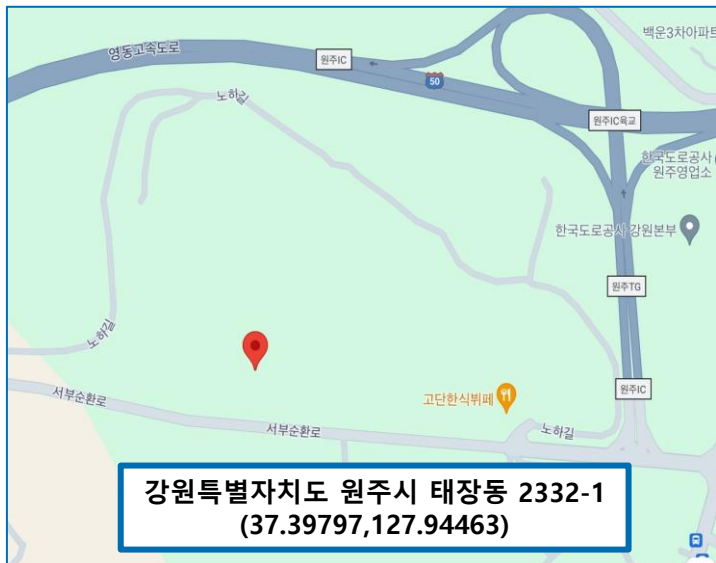
A decorative network diagram in the top-left corner, featuring a complex web of interconnected nodes and lines. The nodes are represented by circles of varying sizes, some with solid outlines and others with dashed outlines, connected by thin, light gray lines. The overall structure is organic and branching, resembling a molecular or biological network.

8.

# 최종 결론

A decorative network diagram in the bottom-right corner, similar to the one in the top-left. It consists of a cluster of interconnected nodes and lines, with nodes represented by circles of different sizes and line styles (solid and dashed). The diagram is positioned in the lower right area of the slide, mirroring the top-left design element.

## 8 최종 결론



### [위치 선정 이유]

- ▶ 제한 조건(산악 지역, 주거&상업 지역/도심/학교와의 근접성, 고속도로 IC 근접성)에 영향을 받지 않음
- ▶ 주변 도로와의 접근성, 교통의 용이성, 사업장 안전성 그리고 입지의 가용성이 가장 우수함
- ▶ 경사지를 이용하여 지하층을 건설하여 경제성을 높이는 방안을 구상함

### [물류센터 형태 제언]

- ▶ 물류센터 건설 시 경제성을 위해 부지의 경사지를 이용할 것을 제언함.
- ▶ 약 5700평 규모의 지상층과 지하층을 가진 건물로 짓도록 제언함.



**Thanks!**

**Any questions?**

