



*AI 그냥 쓰면 2류!*

내 데이터로  
차별화된  
AI 만드는 법

정케빈



# 강사 소개


## 00 정케빈

### 주요 이력

- ✓ 현) 금융 IT 기업 데이터 분석 담당
- ✓ 전) SK 대형 포털 회사 데이터 분석
- ✓ 전) AI 전문 회사 연구원
  
- ✓ 전산학 박사(AI전공)
- ✓ 5년 차 금융 종사자 대상 AI 강사
- ✓ TensorFlow 자격증 보유
- ✓ Elastic Engineering 1,2 과정 수료
- ✓ 지피터스 9기 재테크 서포터, 10·11기 내집마련 파트너
  
- ✓ <https://www.threads.net/@humanist96>

매주 목요일 (주)상상력집단  
**무료 AI 웨비나**  
2월 20일 19시(목)

**LIVE**



**AI, 그냥 쓰면 2류! 내 데이터로  
차별화된 AI 만드는 법**

정개빈  
AI 데이터 분석 분야 20년 경력

# 강의 소개

## 00 목차와 수업 의도 설명

### AI 도구를 활용한 데이터 분석 & 자동화 실습

#### ◆ Part 1: 나만의 데이터 분석 도구 만들기

##### ✦ Day 1: AI와 데이터 학습의 이해

- 데이터 학습이란? 핵심 개념과 원리
- AI의 필요성과 데이터 분석이 중요한 이유
- AI, 머신러닝, 딥러닝, 생성형 AI의 차이점과 특징

##### ✦ Day 2: 데이터 파이프라인 구축 & 시각화

- 데이터 분석 vs 검색 vs AI - 비교와 관련성에 대하여
- Elastic 개념 및 Elastic Cloud 활용법
- 부동산 대시보드 구성 및 실전 제작

#### ◆ Part 2: 나만의 데이터 수집 & 정리 자동화

##### ✦ Day 1: Apps Script을 활용한 데이터 자동 수집 & 정리

- 뉴스레터 전송 자동화
- 블로그 검색 및 요약 자동화
- 유튜브 검색 및 요약 자동화
- 부동산 대시보드 커스터마이징 (과제)와 피드백

##### ✦ Day 2: 데이터 정리 & 분석 자동화

- 오피시디언을 활용한 요약 보고서 자동 작성
- Make를 이용한 실시간 증권 뉴스 분석
- Make & 오피시디언을 활용한 멀티에이전트 요약 분석

##### ✦ 💡 강의 핵심 포인트

- ✓ AI를 활용한 데이터 분석 & 자동화 핵심 개념 완벽 이해
- ✓ 맞춤형 데이터 도구 구축 & 실전 활용 스킬 습득
- ✓ 부동산, 금융 등 실무 데이터 활용 사례 실습

# 강의 소개

## 00 수강생에게 드리는 혜택

- 실전 데이터 분석 도구를 이용한 나만의 아파트 매물정보 대시보드 구축
- 2024년 12월 개편한 네이버 아파트 매물 수집 코드 제공
- 뉴스레터 자동화를 위한 Apps Script 소스와 템플릿 제공
- 오피시디언 환경파일과 아파트 단지 분석 보고서 프롬프트 템플릿 제공





# 데이터 학습 이해

# 💡 데이터 학습 이해하기

01 가위바위보 게임을 컴퓨터가 이해할 수 있도록 해볼까요?



- 우리의 뇌는 가위바위보에서 누가 이겼는지 어떻게 판단할까?
- 판단을 어떻게 학습했고, 결정을 내렸는지 설명하기 어려움

# 💡 데이터 학습 이해하기

## 01 익숙한 룰기반 프로그래밍으로 가능할 것 같아요.

- 0:가위, 1:바위, 2:보
- My=1, computer = 2



- If my == 0 and computer == 0 then 비김
- If my == 0 and computer == 1 then 짐
- If my == 0 and computer == 2 then 이김
- If my == 1 and computer == 0 then 이김
- If my == 1 and computer == 1 then 비김
- If my == 1 and computer == 2 then 짐
- If my == 2 and computer == 0 then 짐
- If my == 2 and computer == 1 then 이김
- If my == 2 and computer == 2 then 비김

- 프로그래밍 언어가 지원하는 if 문장을 사용해 구현 가능

# 💡 데이터 학습 이해하기

01 익숙한 룰기반 프로그래밍으로 가능할 것 같아요.



- 모든 경우의 수를 기억하고 있으면 상황에 맞는 **지식을 쉽게 추론할 수 있음**
- **학습은 모든 경우의 수를 머릿속에 저장하는 작업**



# 데이터 학습 이해하기

01 좀 더 효과적인 방법도 생각해 볼 수 있어요.

## ● 0: 가위, 1:바위, 2:보

- my : 0    computer : 0     $\rightarrow 0 - 0 = 0$      $\rightarrow$  비김
- my : 0    computer : 1     $\rightarrow 0 - 1 = -1$      $\rightarrow$  짐
- my : 0    computer : 2     $\rightarrow 0 - 2 = -2$      $\rightarrow$  이김
  
- my : 1    computer : 0     $\rightarrow 1 - 0 = 1$      $\rightarrow$  이김
- my : 1    computer : 1     $\rightarrow 1 - 1 = 0$      $\rightarrow$  비김
- my : 1    computer : 2     $\rightarrow 1 - 2 = -1$      $\rightarrow$  짐
  
- my : 2    computer : 0     $\rightarrow 2 - 0 = 2$      $\rightarrow$  짐
- my : 2    computer : 1     $\rightarrow 2 - 1 = 1$      $\rightarrow$  이김
- my : 2    computer : 2     $\rightarrow 2 - 2 = 0$      $\rightarrow$  비김

비김 경우 : 0  
진 경우 : -1, 2  
이김 경우 : 2, 1

# 데이터 학습 이해하기

## 01 어떤 공식과 패턴을 찾아내면 효과적인 프로그래밍도 가능

- 0: 가위, 1: 바위, 2: 보
- 판단 공식 =  $(my - computer + 3) \% 3$
- 0 : 비김, 1 : 이김, 2 : 짐

		computer		
		0:가위	1:바위	2:보
my	0:가위	비김 $(0-0+3)\%3=0$	짐 $(0-1+3)\%3=2$	이김 $(0-2+3)\%3=1$
	1:바위	이김 $(1-0+3)\%3=1$	비김 $(1-1+3)\%3=0$	짐 $(1-2+3)\%3=2$
	2:보	짐 $(2-0+3)\%3=2$	이김 $(2-1+3)\%3=1$	비김 $(2-2+3)\%3=0$

- 학습은 계산 가능한 수학 공식을 만드는 것
- 공식을 만들 수 있다고 해도 변수들의 범위 등이 조금만 변해도 다른 결과가 나옴

# 💡 데이터 학습 이해하기

## 01 그런데 이런 패턴 찾기위해 컴퓨터의 학습하는 것을 일반화 → 기계학습

- (my, computer) → label
- (0,0) → 0(비김)
- (0,1) → 1(짐)
- (0,2) → 2(이김)

학습 데이터

0(가위)  
1(바위)  
1(짐)



예측값

2(이김)



판단할 데이터 (0, 2)

- 기계학습 및 딥러닝에서 학습은 학습 데이터를 가장 잘 설명할 수 있는 모델을 만들어가는 과정

# 불가지론

(不可知論, agnosticism)

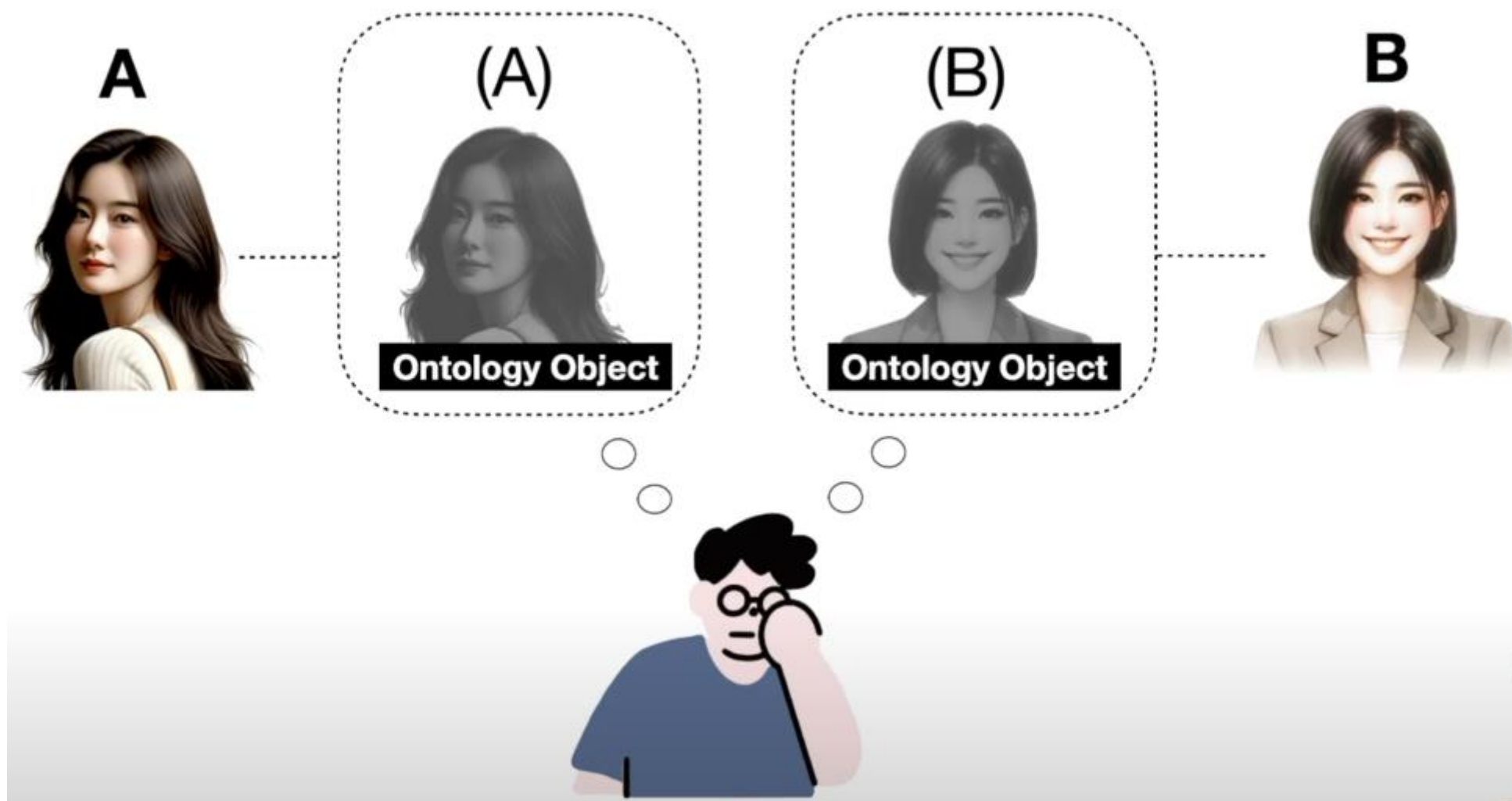
인간의 인지 방식으로 사물의 본질을 파악할 수 없다

The essence of things cannot be understood through human cognition.



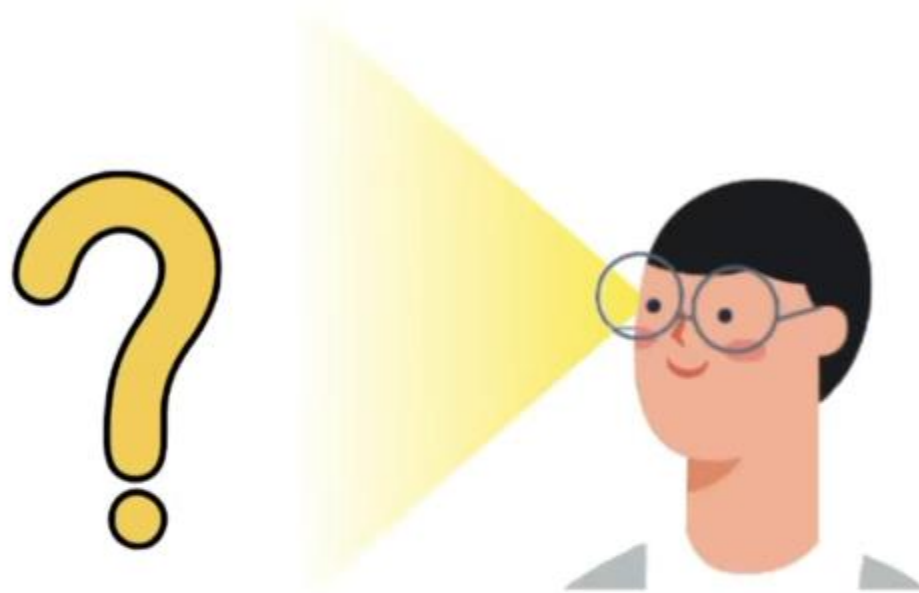
# 💡 데이터 학습 이해하기

## 02 불가지론



# 데이터 학습 이해하기

## 02 불가지론

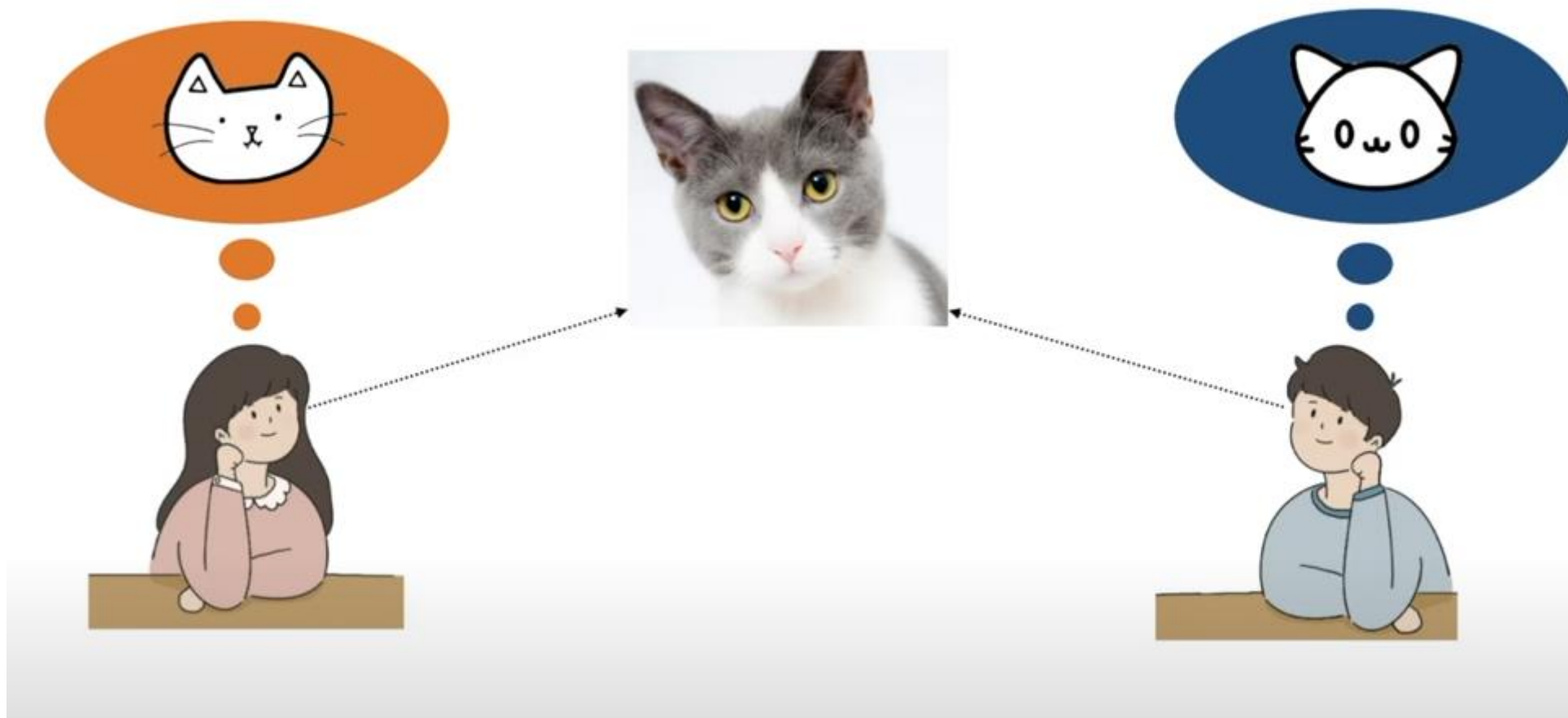


어떻게 해야 그 사물의 본질을 더 제대로 파악할 수 있을까?

How can we better and more accurately understand the essence of the things?

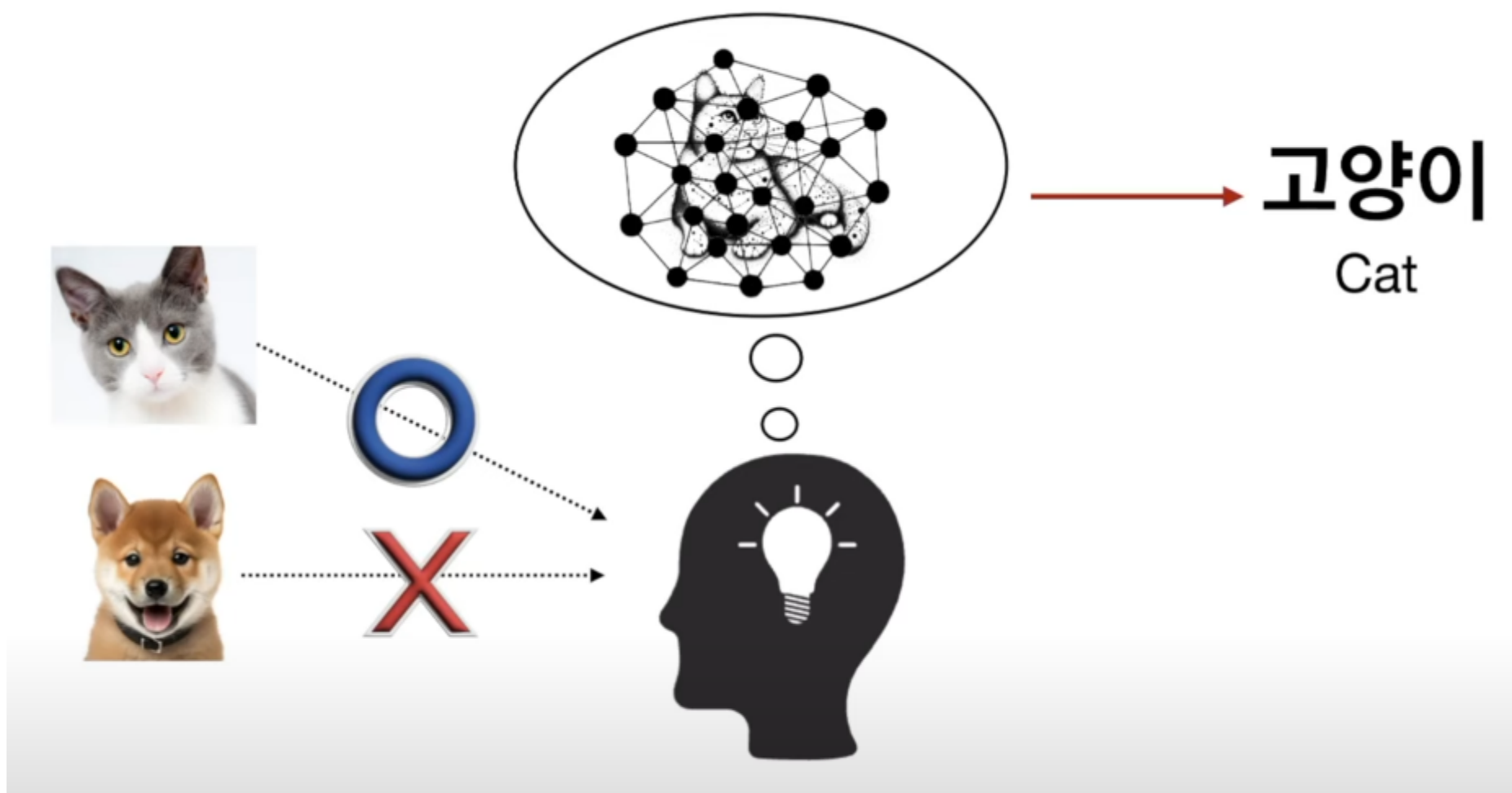
# 💡 데이터 학습 이해하기

## 02 불가지론



# 💡 데이터 학습 이해하기

## 02 불가지론





# 데이터 학습 이해하기

## 02 불가지론

귀가 2개, 눈이 2개 기타 등등...이런 게 고양이야.

A cat has two ears, two eyes, and so on.

This is what makes a cat.



# 💡 데이터 학습 이해하기

## 02 불가지론



### 고양이 특징 Cat Characteristics

1. 귀가 2개 Two ears
2. 눈이 2개 Two eyes
3. 꼬리 1개 One tail
4. 수염
5. ~~~~~  
.  
.  
.

# 💡 데이터 학습 이해하기

## 02 불가지론



특징 Characteristics

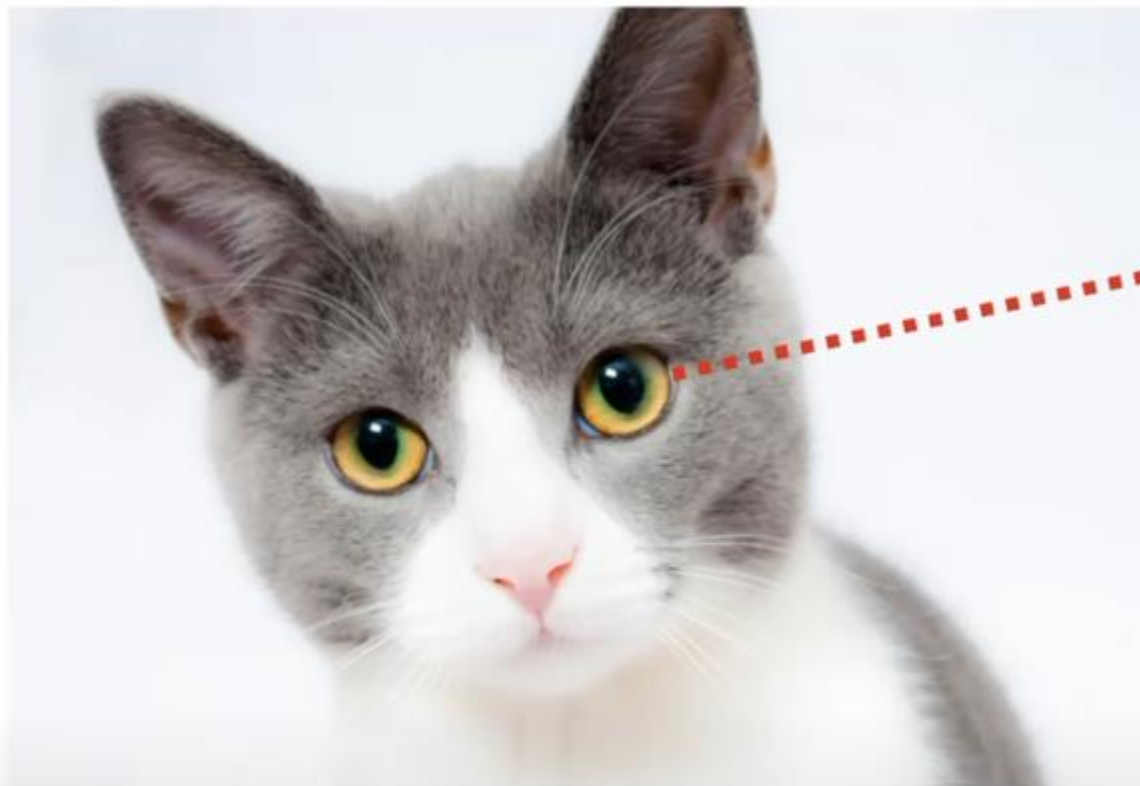


본질 Essence?

1. 귀가 2개 Two ears
2. 눈이 2개 Two eyes
3. 꼬리 1개 One tail
4. ~~~~~
5. ~~~~~
- .
- .
- .

# 데이터 학습 이해하기

## 02 불가지론

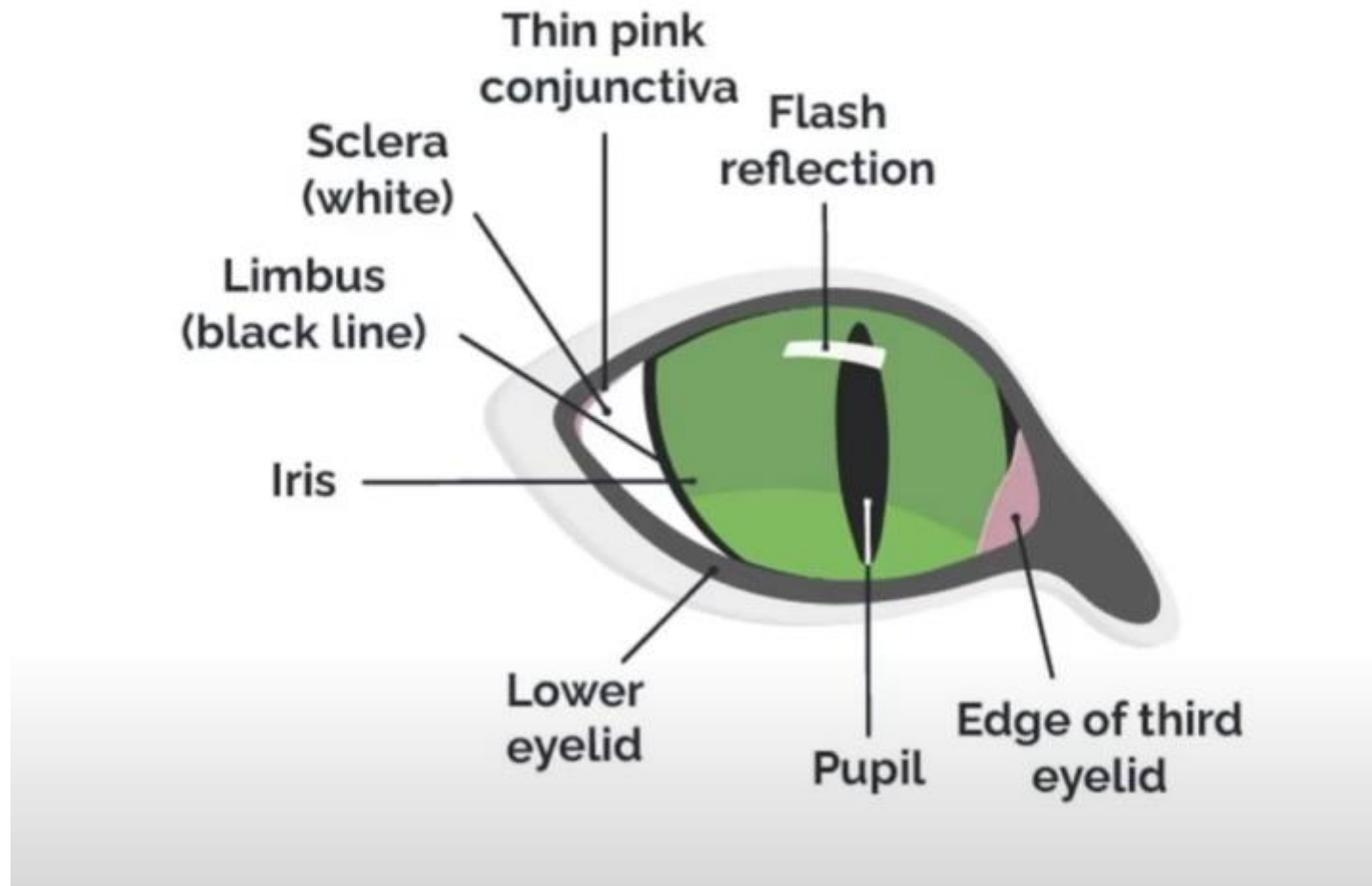


눈?  
Eye?



# 💡 데이터 학습 이해하기

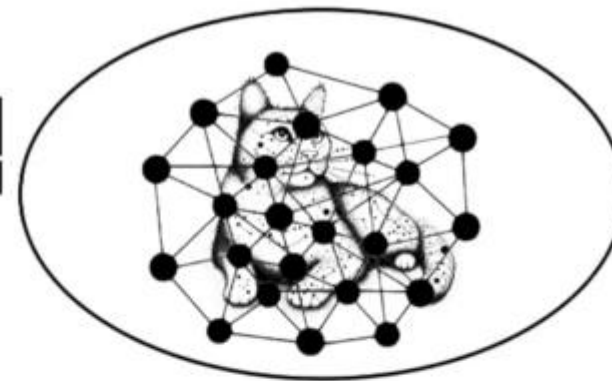
## 02 불가지론



# 데이터 학습 이해하기

## 02 불가지론

규칙  
Rule



완벽하게 정의된 지식?

Perfectly defined Knowledge?



# 💡 데이터 학습 이해하기

## 02 불가지론

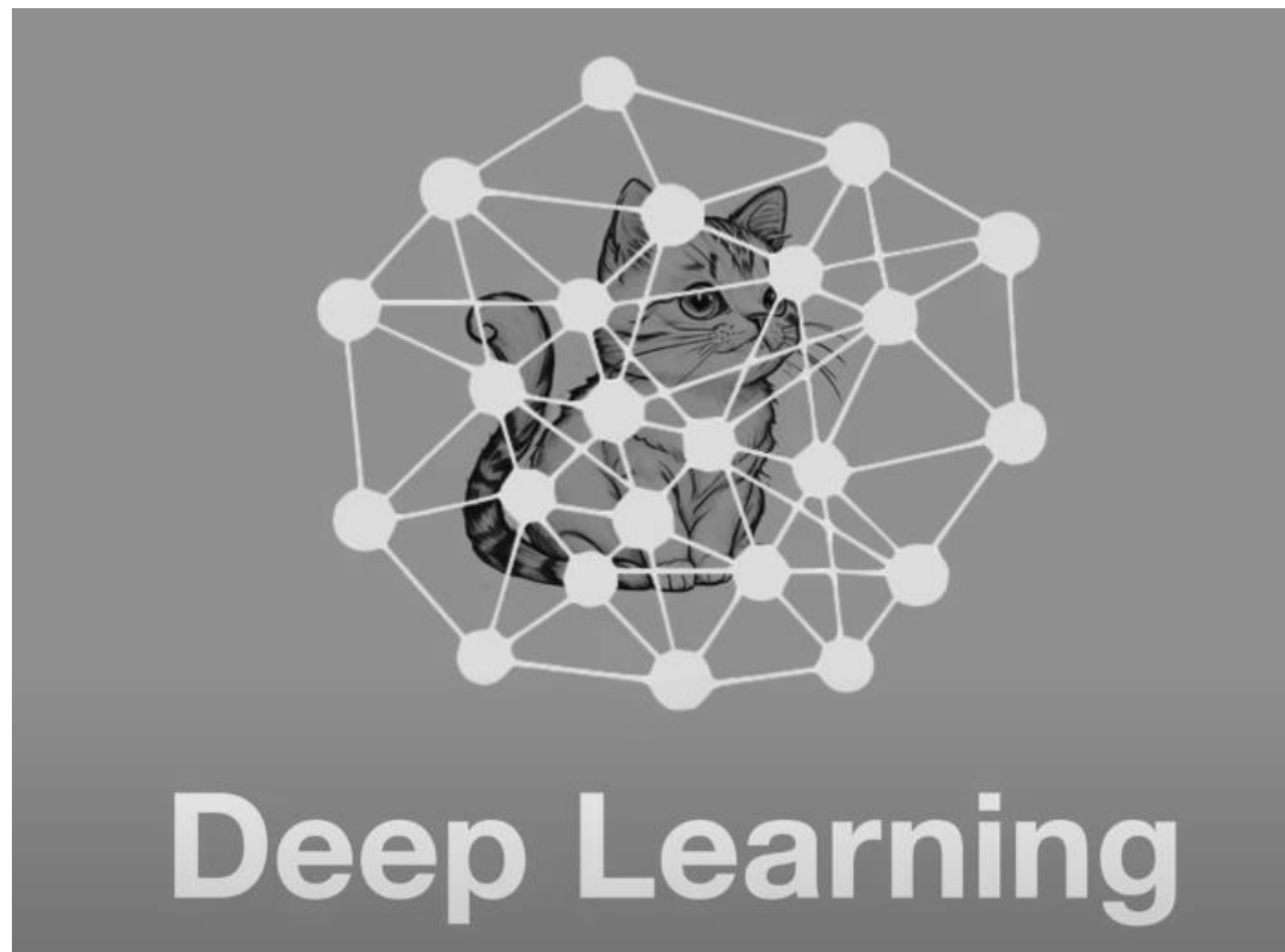
개념을 완벽하게 정의할 수 있다?

A concept can be defined perfectly?



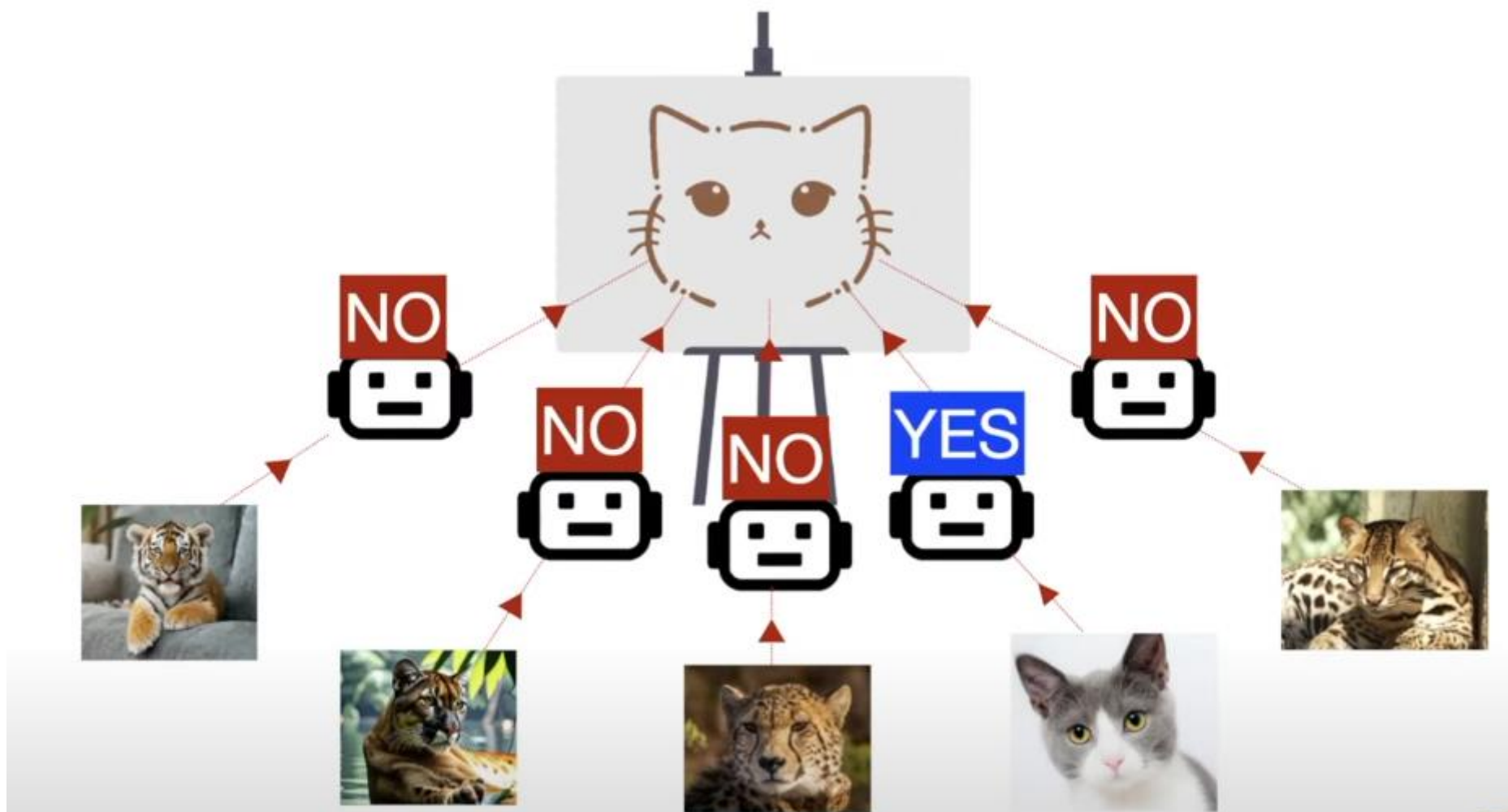
# 데이터 학습 이해하기

## 02 불가지론



# 💡 데이터 학습 이해하기

## 02 불가지론

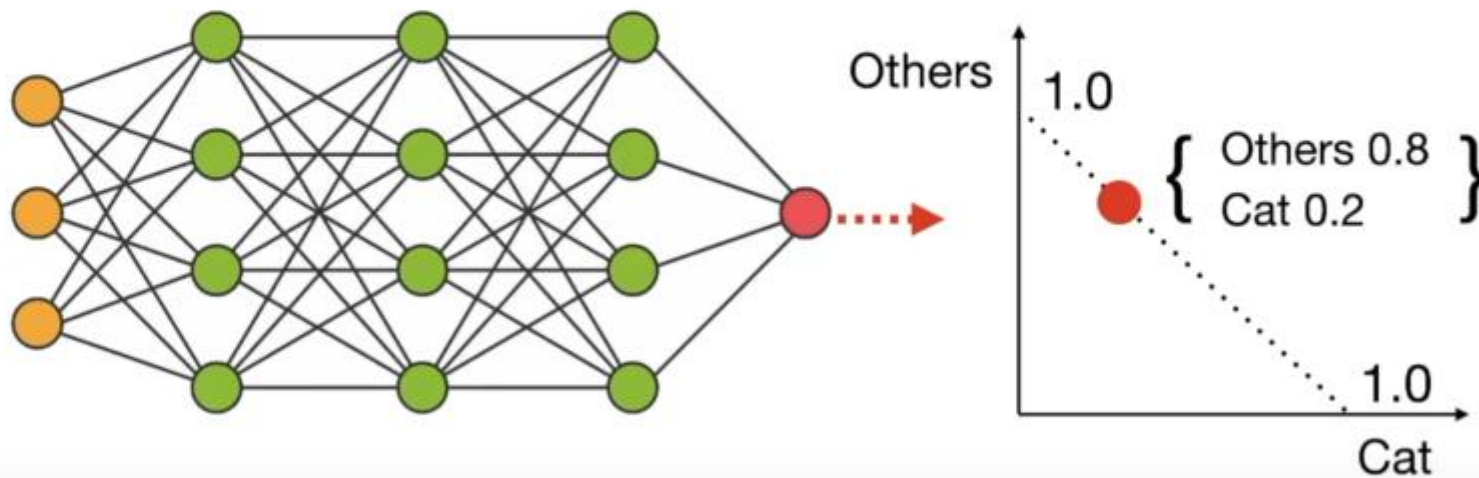




# 💡 데이터 학습 이해하기

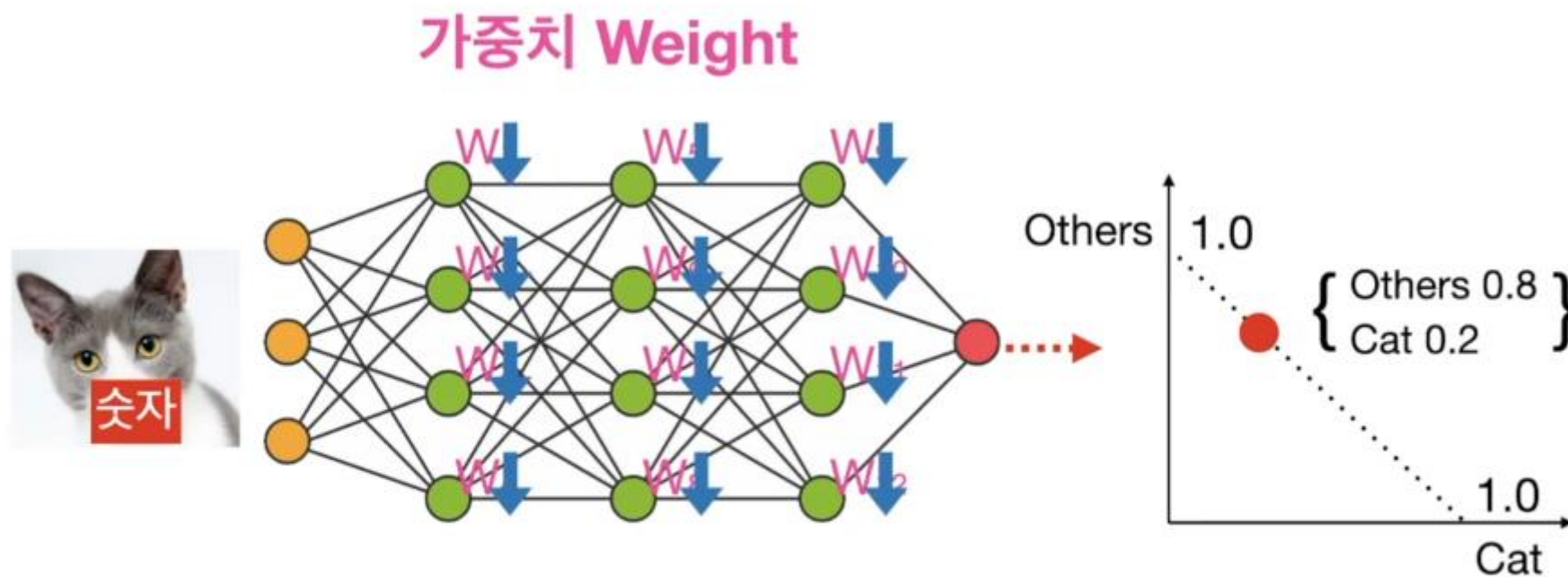
## 02 불가지론

### 순전파 Forward Propagation



# 💡 데이터 학습 이해하기

## 02 불가지론

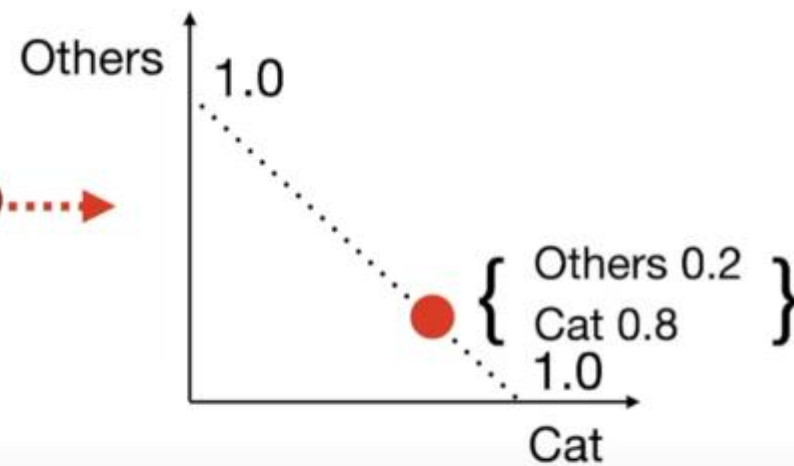
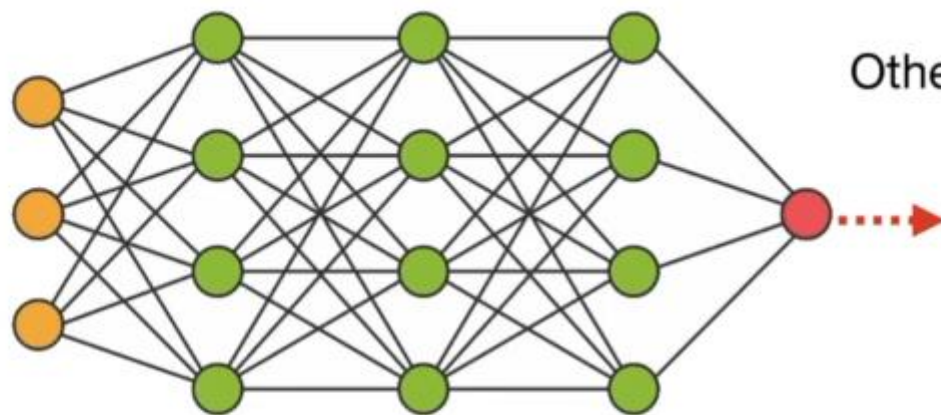


← 역전파 Back Propagation

# 💡 데이터 학습 이해하기

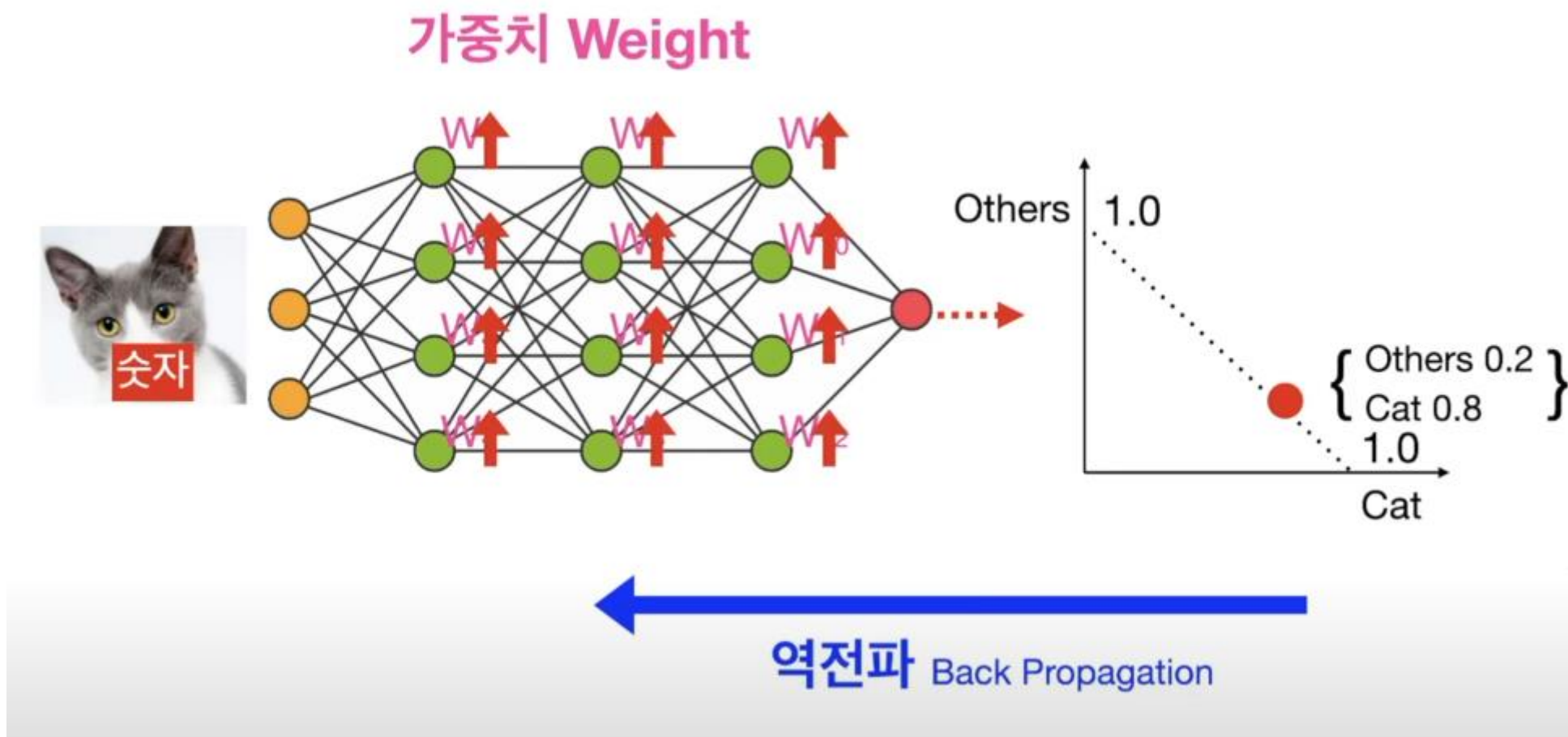
## 02 불가지론

순전파 Forward Propagation



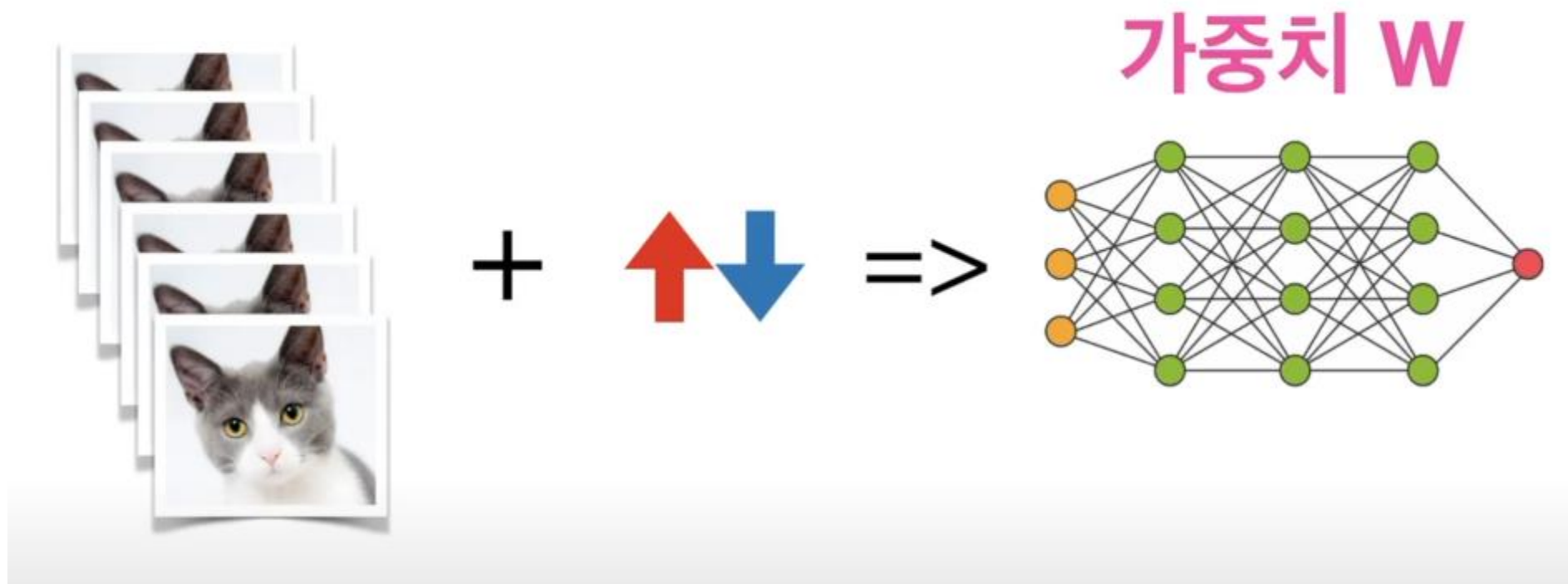
# 💡 데이터 학습 이해하기

## 02 불가지론



# 💡 데이터 학습 이해하기

## 02 불가지론



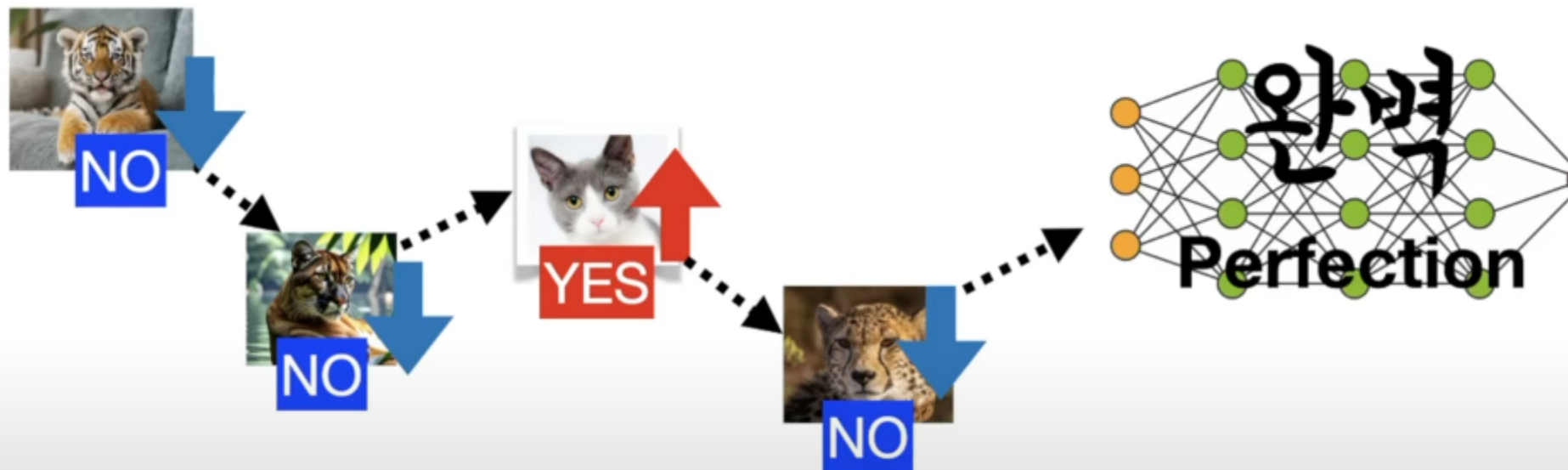


# 💡 데이터 학습 이해하기

## 02 불가지론

개념을 완벽하게 정의할 수 없다?

A concept can not be defined perfectly?



# 💡 데이터 학습 이해하기

## 02 불가지론



### 여러분의 이상형은 무엇인가요?

What kind of person is your ideal type as a romantic partner?



# 💡 데이터 학습 이해하기

## 02 불가지론



**A**

VS



**B**

# 데이터 학습 이해하기

## 02 불가지론



**C**

**VS**



**D**



# 데이터 학습 이해하기

## 02 불가지론



**E**

**VS**



**F**

# 데이터 학습 이해하기

## 02 불가지론

### 중요한 진실

Important truth

끊임없는 비교를 통해 개념의 본질에 다가갈 수 있다.

Constant comparison is the key method to approach the essence of a concept.

## 02 이미지 인식 경진대회 우리 모두 참여해 봅시다

# 딥러닝기반 강아지 고양이 이미지인식 경진대회

2017. 8. 4(금) 오전 10시  
DGIST 연구행정동 국제회의장



### 지원자격

- 전국대학생  
(석박사과정포함)
- 타지역 학생 교통비 지원
- 대구 - DGIST 셔틀버스 운행

### 지원방법

dgist.imagechallenge.kr 접속  
참가신청 및 데이터 다운로드,  
딥러닝기반으로 알고리즘 작성  
알고리즘 학습 후 행사 참여  
\*신청마감 7/30(일)

### 행사진행

- 오전 10시 검증데이터 50% 공개
- 오후 4시 나머지 데이터 공개
- 정확도에 따라 순위결정

### 특 전

- 대상(1명) NVIDIA Titan Xp
- 우승(2명) NVIDIA GTX 1080Ti

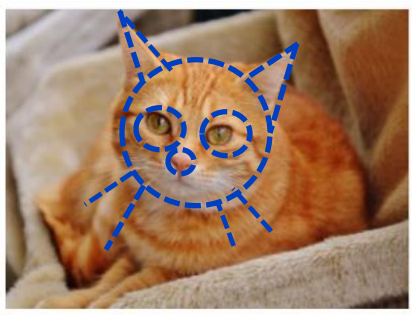
### 문 의

- dgist.imagechallenge.kr  
(T.053-785-6701~2)

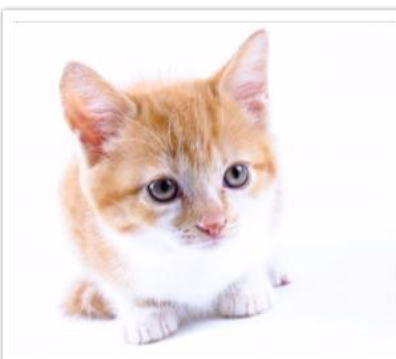
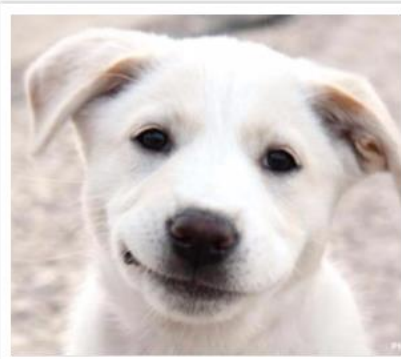
**DGIST** 기술창업교육센터

# 💡 데이터 학습 이해하기

## 02 기존의 프로그래밍 방식 (하드 코딩 규칙)

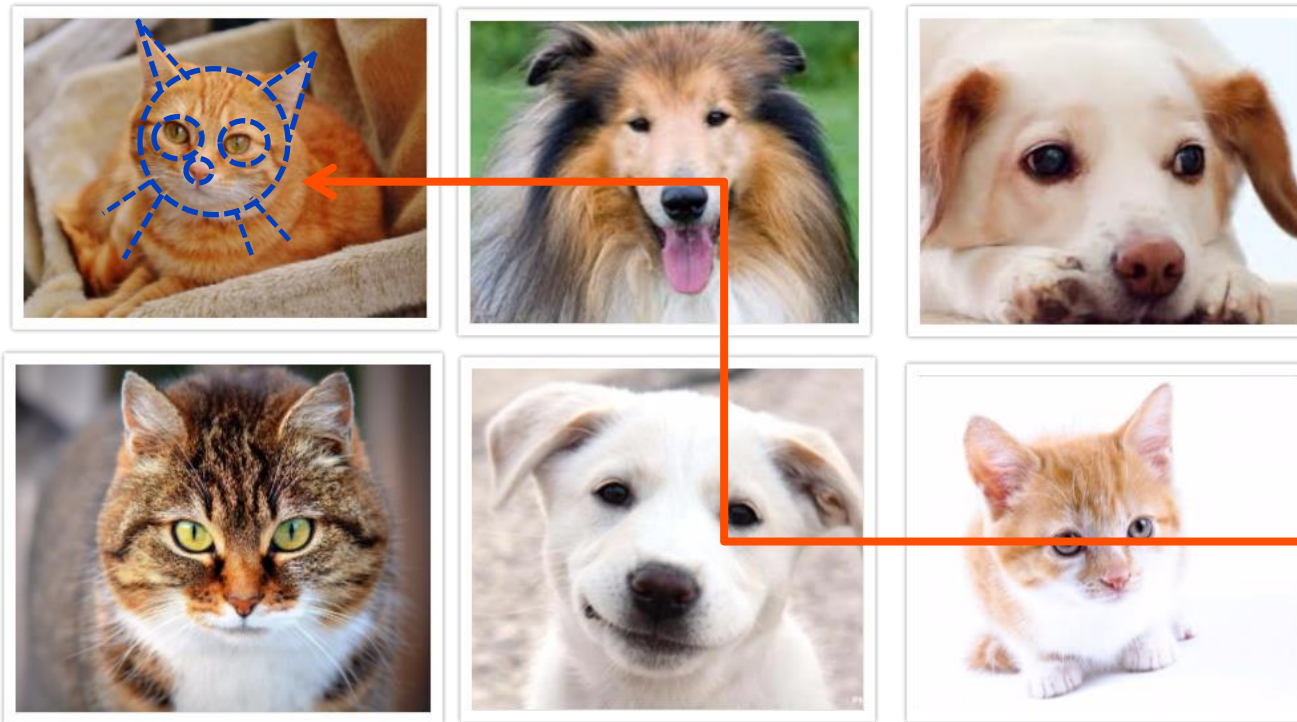


```
1 def prediction(이미지 as input):  
2     어떻게 코딩해야하지?  
3     return 결과
```

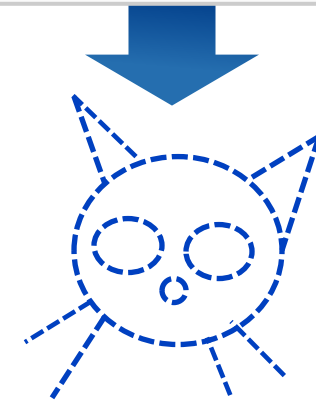


# 💡 데이터 학습 이해하기

## 02 기존의 프로그래밍 방식 (하드 코딩 규칙)



```
1 def prediction(이미지 as input):  
2     어떻게 코딩해야하지?  
3     return 결과
```

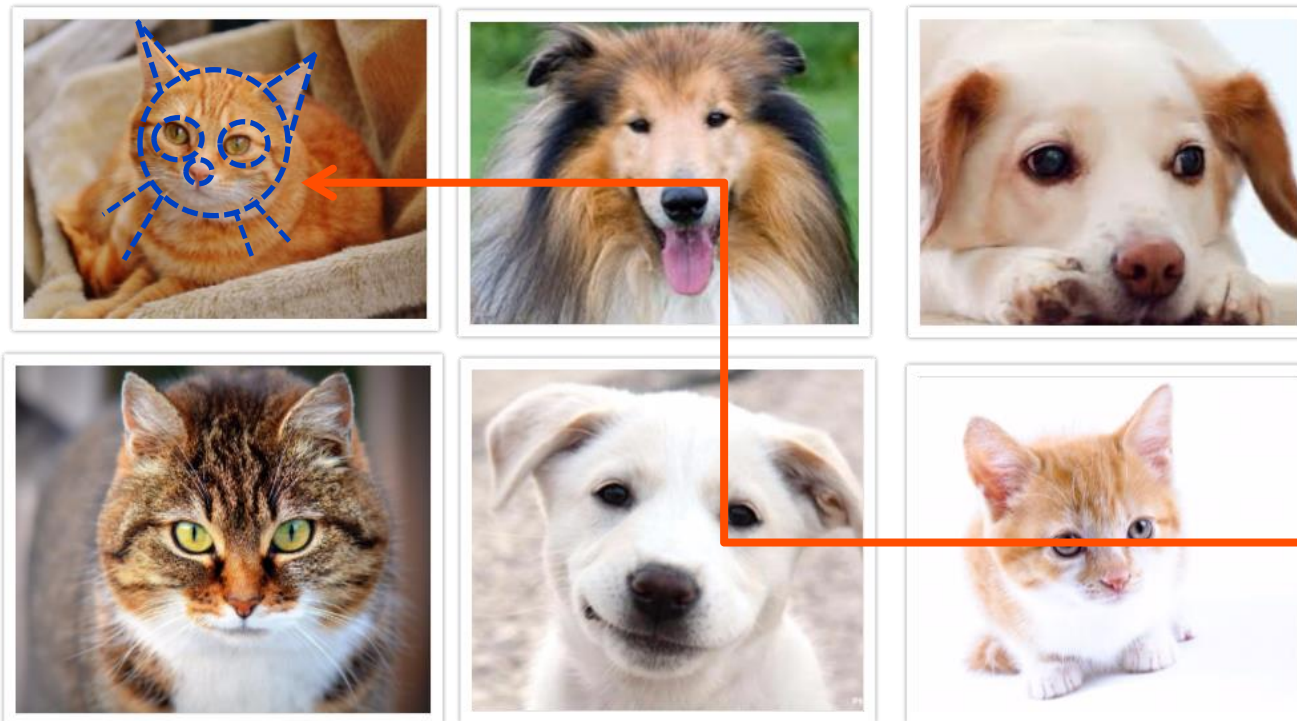


어떤 특징을 가지고 있는지 눈으로 잘 관찰하고  
열심히 픽셀 단위로 템플릿을 만들고 비교하는  
코드.  
(if-else 문의 집합체)

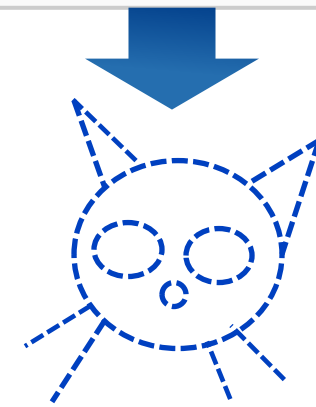


# 💡 데이터 학습 이해하기

## 02 기존의 프로그래밍 방식 (하드 코딩 규칙)



```
1 def prediction(이미지 as input):  
2     어떻게 코딩해야하지?  
3     return 결과
```



- ☑ If(보족한 두 귀) && if (빠져나온 수염)  
등으로  
맞출 순 있음.

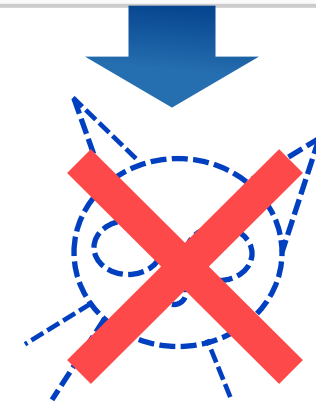
어떤 특징을 가지고 있는지 눈으로 잘 관찰하고  
열심히 픽셀 단위로 템플릿을 만들고 비교하는  
코드.  
(if-else 문의 집합체)

# 💡 데이터 학습 이해하기

## 02 기존의 프로그래밍 방식의 한계(1)



```
1 def prediction(이미지 as input):  
2     어떻게 코딩해야하지?  
3     return 결과
```



규칙을 일일이 수백 가지 정의한다고 하더라도  
뽀족한 귀, 삐죽 튀어나온 수염, 작은 발 등의  
규칙이 맞지 않기 시작함.



# 💡 데이터 학습 이해하기

## 02 기존의 프로그래밍 방식의 한계(2)



# 데이터 학습 이해하기

## 03 사람은 되는데 하드 코딩으로는 안 되는 이유

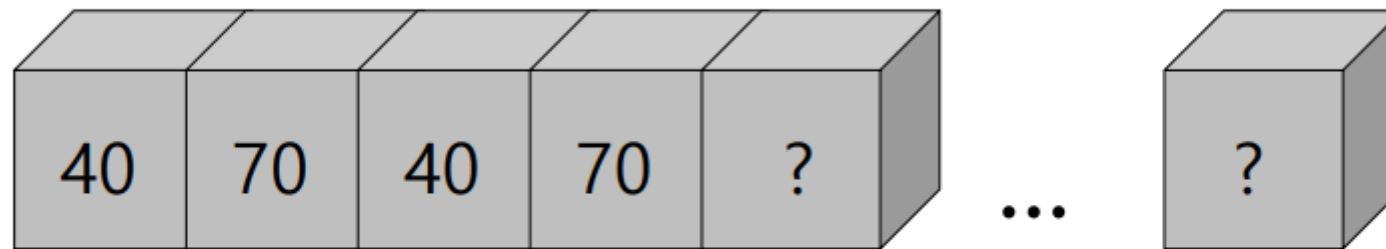
- 우리가 강아지와 고양이를 구분할 수 있는 것은 과거에 수많은 강아지와 고양이를 봤기 때문.
- 사람도 본 적 없는 두 동물을 구분하라고 하면 구분할 수 없음.
- 사람도 옛날에 한, 두 번 본 동물은 잘 구분 못해도 수백 번 본 동물들은 구분이 가능함.
- 기계한테도 이건 강아지고, 이건 고양이야라고 하면서 사진 여러장을 보여주면서 더 많은 사진을 배우면 배울 수록 점점 잘 맞추게 할 수는 없을까?



# 💡 데이터 학습 이해하기

## 03 정말 우리 사람도 학습이 가능할까요?

### Data 학습 #1

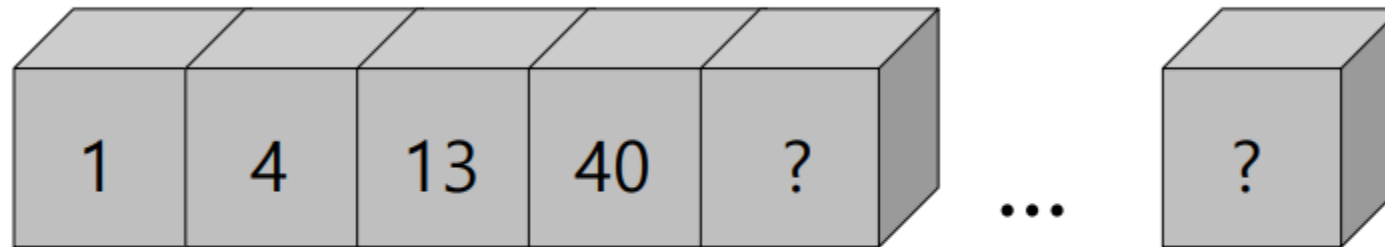


- 다음 숫자가 90일까요?
- 그럼 다음 숫자는 무엇일까요?
- 20번째 숫자는?
- 수학적으로 표현하면?

# 💡 데이터 학습 이해하기

## 03 정말 우리 사람도 학습이 가능할까요?

### Data 학습 #2



- 다음 숫자는 무엇일까요?
- 8번째 숫자는?
- 수학적으로 표현하면?

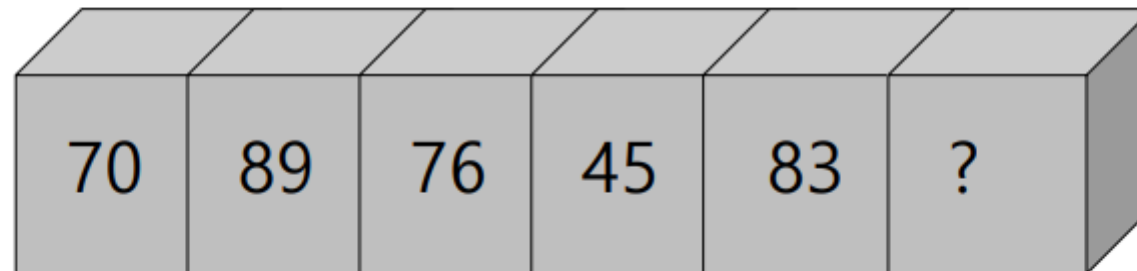


# 💡 데이터 학습 이해하기

## 03 정말 우리 사람도 학습이 가능할까요?

### Data 학습 #3

---



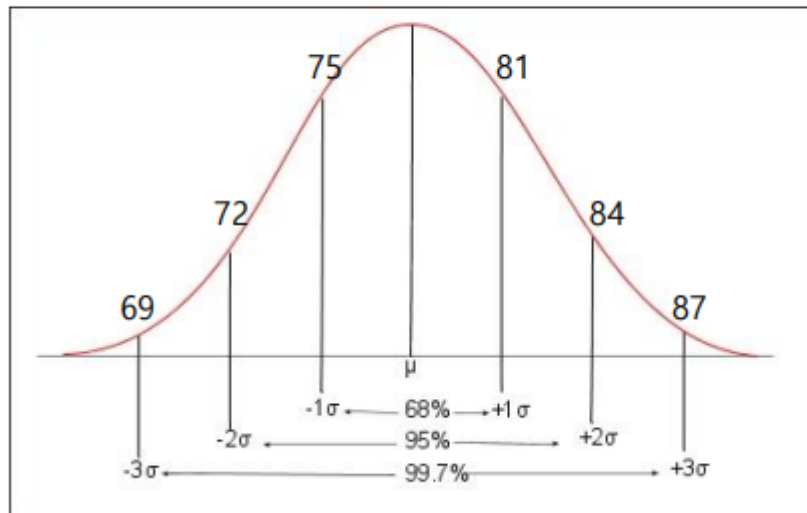
- 70 89 76 45 83
- 다음 숫자는?
- 다음 숫자는 90일까요?
- 수학적으로 표현하면?

# 💡 데이터 학습 이해하기

03 과거에는 통계 분석 모델로 데이터 분석을 수행했습니다.

## 평균, 표준편차를 이용한 데이터 분석

평균( $\mu$ )=78, 표준편차( $\sigma$ )=3

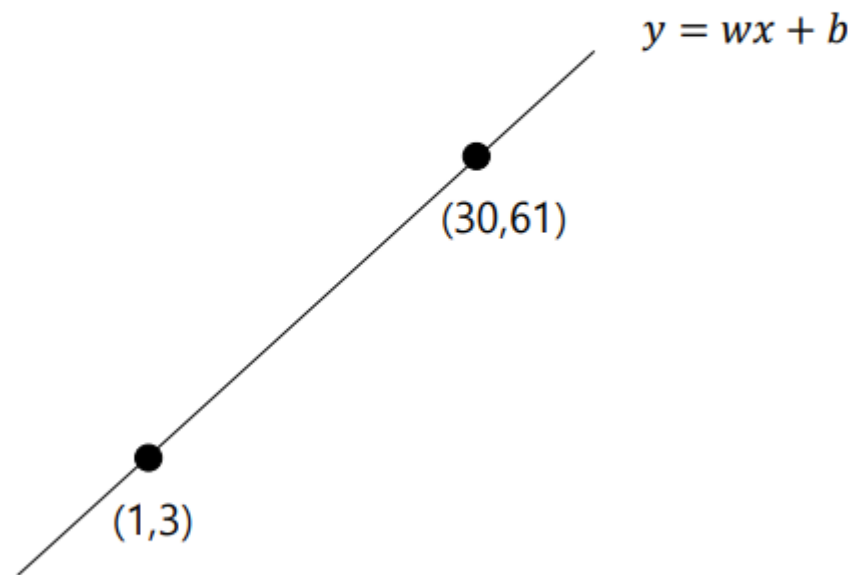


- 다음 시험에서 87점을 맞을 확률은?
- 시험 구간 95%에서 점수 범위는 어떻게 되는가?
- 확률적으로 0.3%이하면 비정상 점수로 판단한다면 몇 점이 비정상 점수인가?

# 💡 데이터 학습 이해하기

04 요즘의 데이터 분석은 선형모델에 집중을 합니다.

## 직선의 방정식을 이용한 데이터 학습



- 기울기  $w = \frac{3-61}{1-30} = \frac{-58}{-29} = 2$

- y절편  $b = y - wx = 3 - w = 3 - 2 = 1$



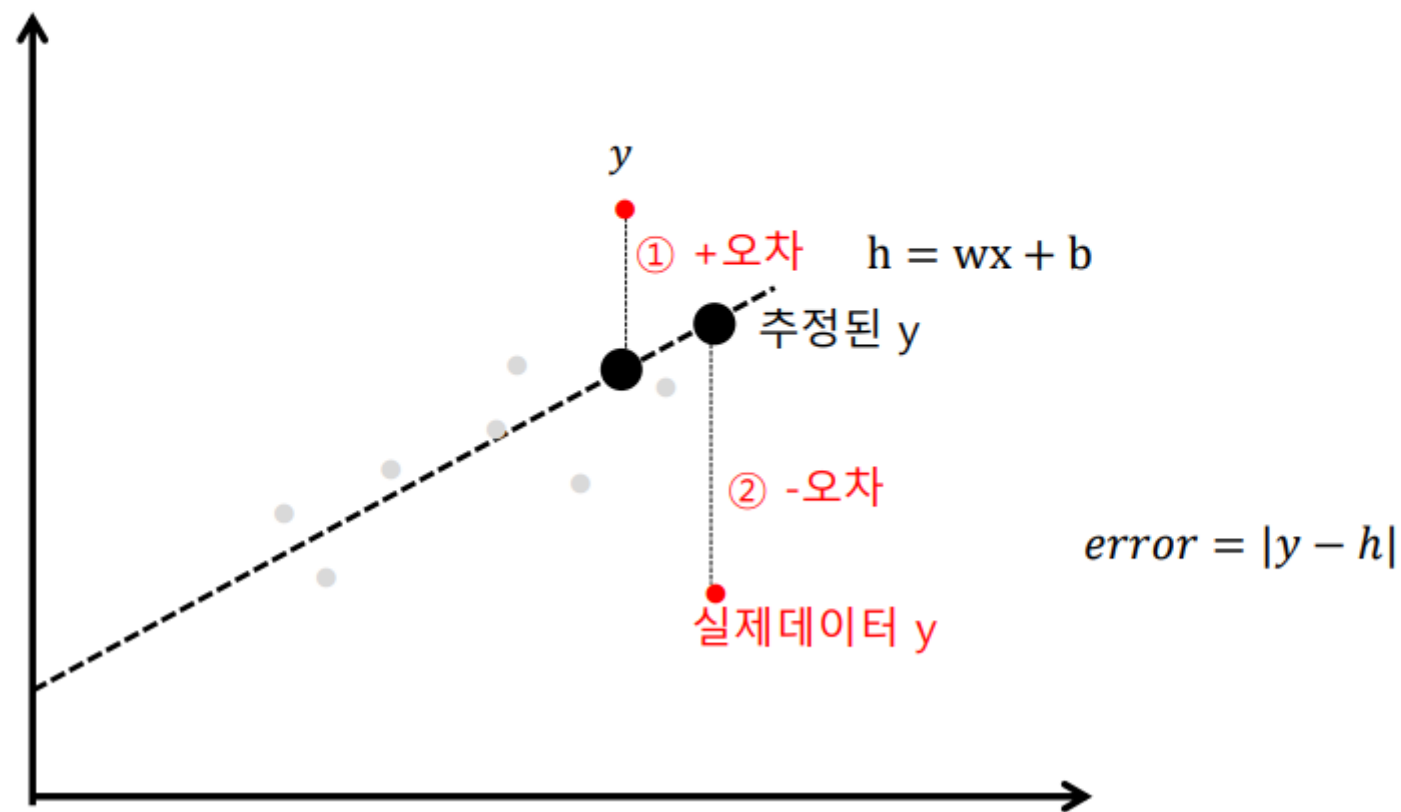
$$y = 2x + 1$$

x	y
1	3
2	5
3	7
4	9
5	11
10	21
20	41
30	61

# 💡 데이터 학습 이해하기

04 요즘의 데이터 분석은 선형모델에 집중을 합니다.

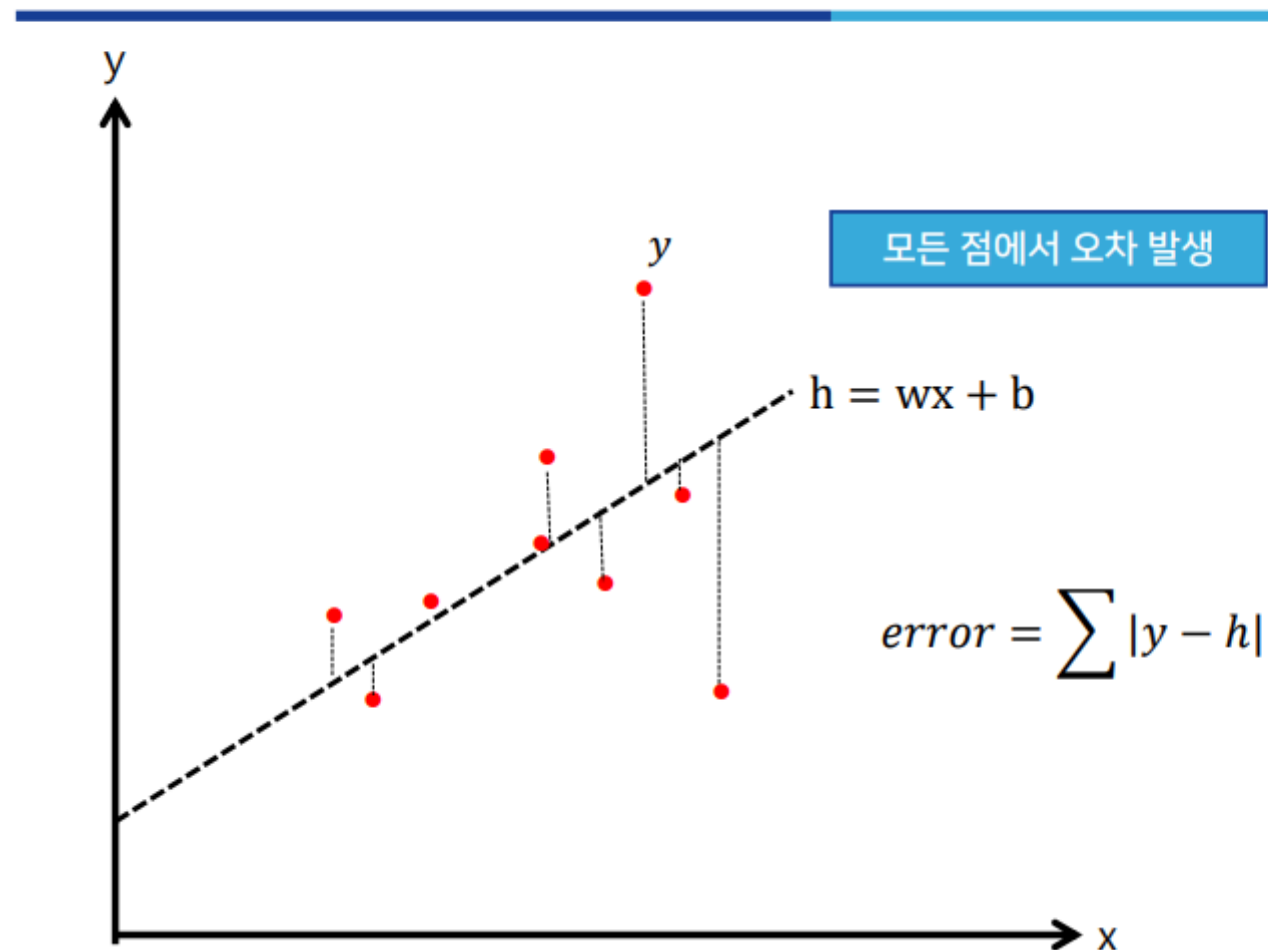
## 오차의 발생



# 💡 데이터 학습 이해하기

04 요즘의 데이터 분석은 선형모델에 집중을 합니다.

## 오차의 정의





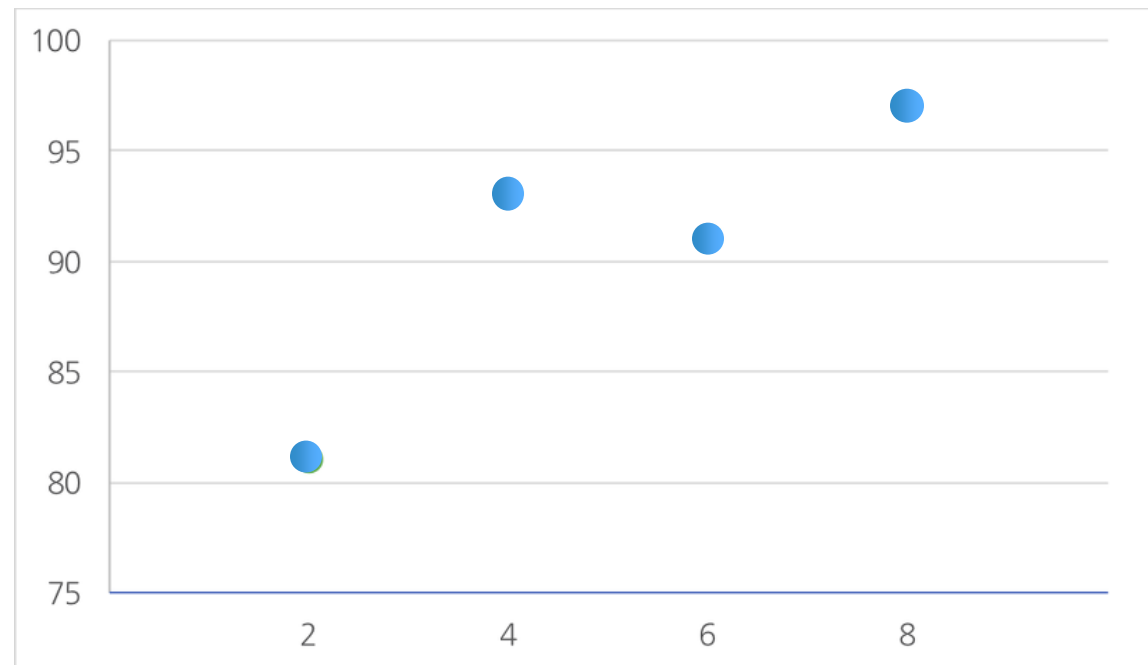
# 💡 데이터 학습 이해하기

## 04 가장 간단한 예제 - 규칙성 찾기

기계에게 알려주는 데이터



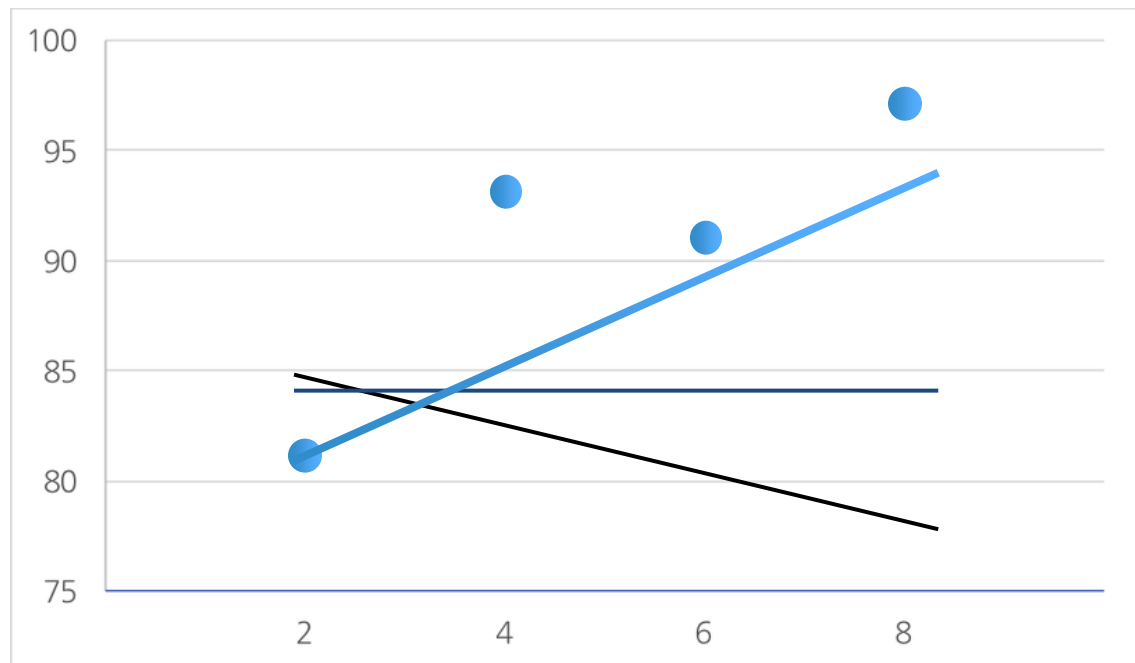
Hour	Score
2	81
4	93
6	91
8	97



Hour이 11일 때의 score는?

# 💡 데이터 학습 이해하기

## 04 가장 간단한 예제 - 규칙성 찾기



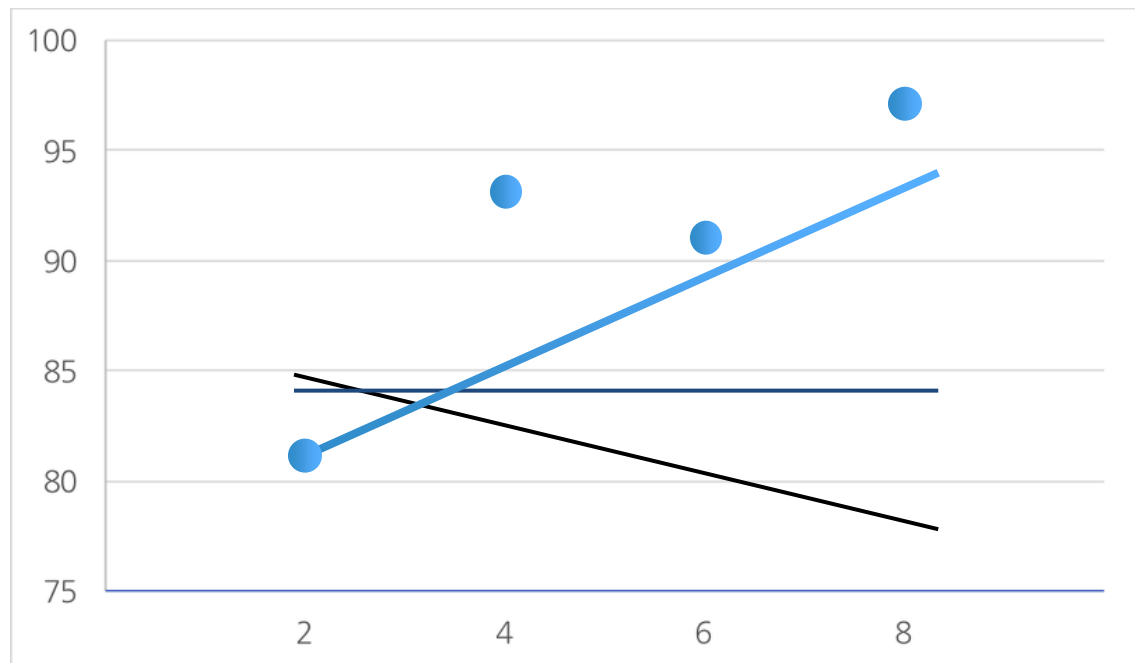
데이터를 가장 잘 반영한 직선을  
찾은 후에 11에서의 값을 보면 해결.

위에 3개의 랜덤으로 그린 직선이 있다.  
위 3개의 직선 중에서 가장 데이터를 잘 반영한 직선은?  
또한 데이터가 위의 추세라면 hour이 11일때의 y값은?

Hour이 11일 때의 score는?

# 💡 데이터 학습 이해하기

## 04 가장 간단한 예제 - 규칙성 찾기



위의 3개의 직선은 각각  $W$ 와  $b$ 가 다름.

데이터를 가장 잘 반영한 직선을  
찾은 후에 11에서의 값을 보면 해결.

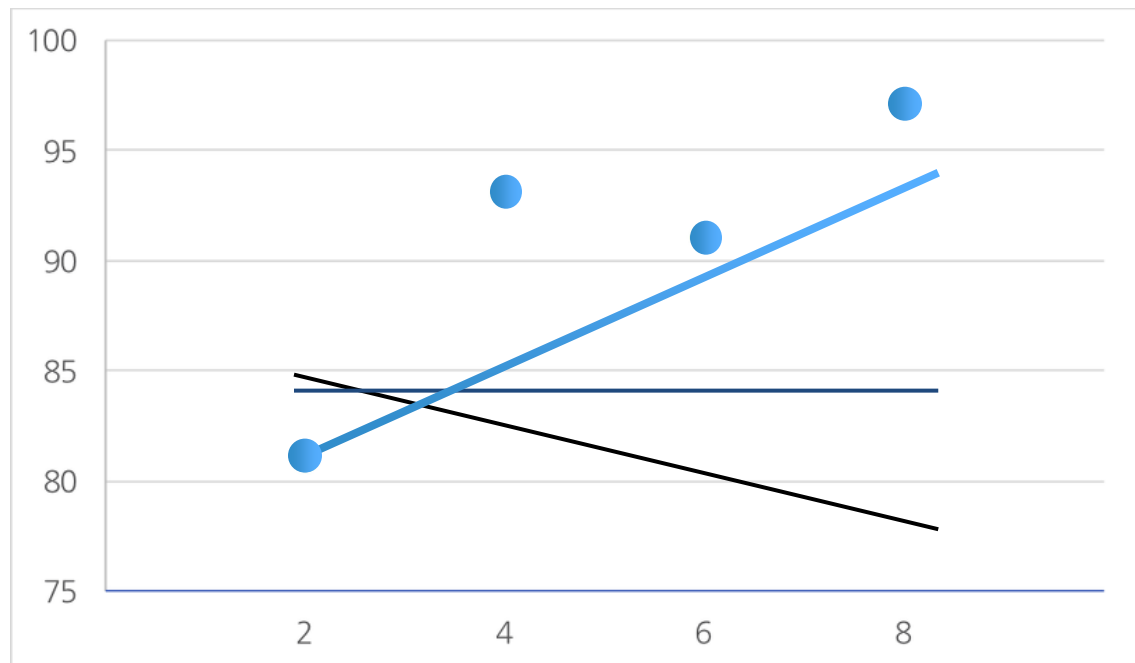
직선의 방정식  
 $Y = \mathbf{W}X + \mathbf{b}$

결국은 데이터를 잘 반영한  $\mathbf{W}$ 와  $\mathbf{b}$ 를 찾는  
문제.

Hour이 11일 때의 score는?

# 💡 데이터 학습 이해하기

## 04 가장 간단한 예제 - 규칙성 찾기



위의 3개의 직선은 각각  $W$ 와  $b$ 가 다름.

데이터를 가장 잘 반영한 직선을  
찾은 후에 11에서의 값을 보면 해결.

직선의 방정식  
 $Y = \mathbf{WX} + \mathbf{b}$

결국은 데이터를 잘 반영한  $\mathbf{W}$ 와  $\mathbf{b}$ 를 찾는  
문제.

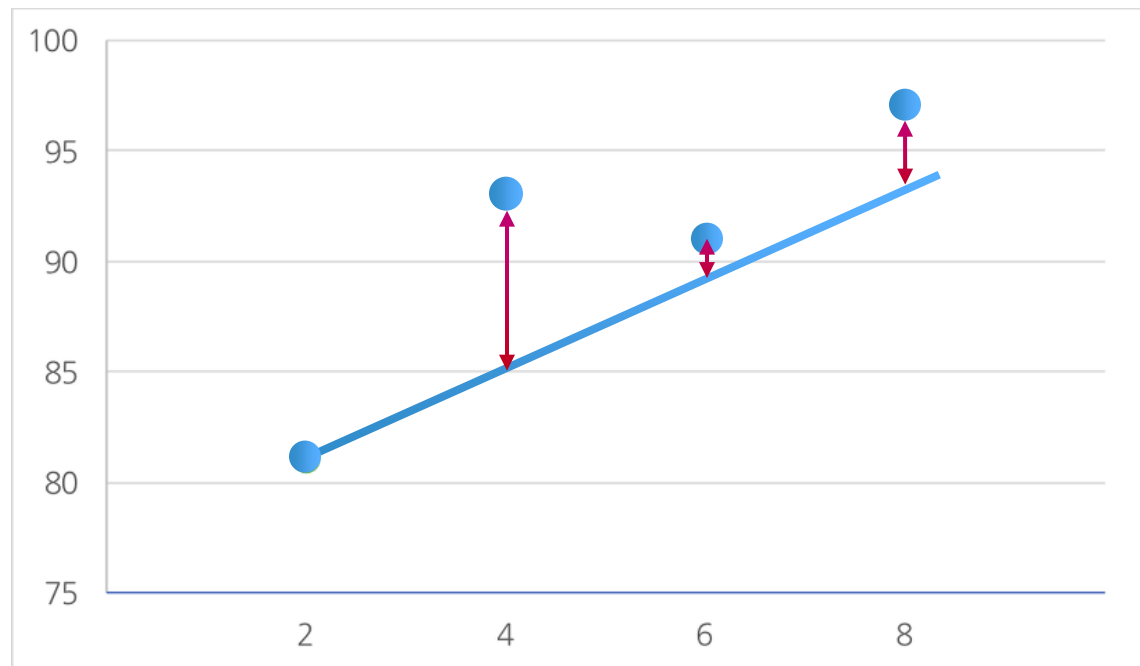
- = 데이터에 없던 질문에도 답이 가능.
- = 규칙성을 찾는 문제
- = 인공지능의 학습 단계

Hour이 11일 때의 score는?

# 💡 데이터 학습 이해하기

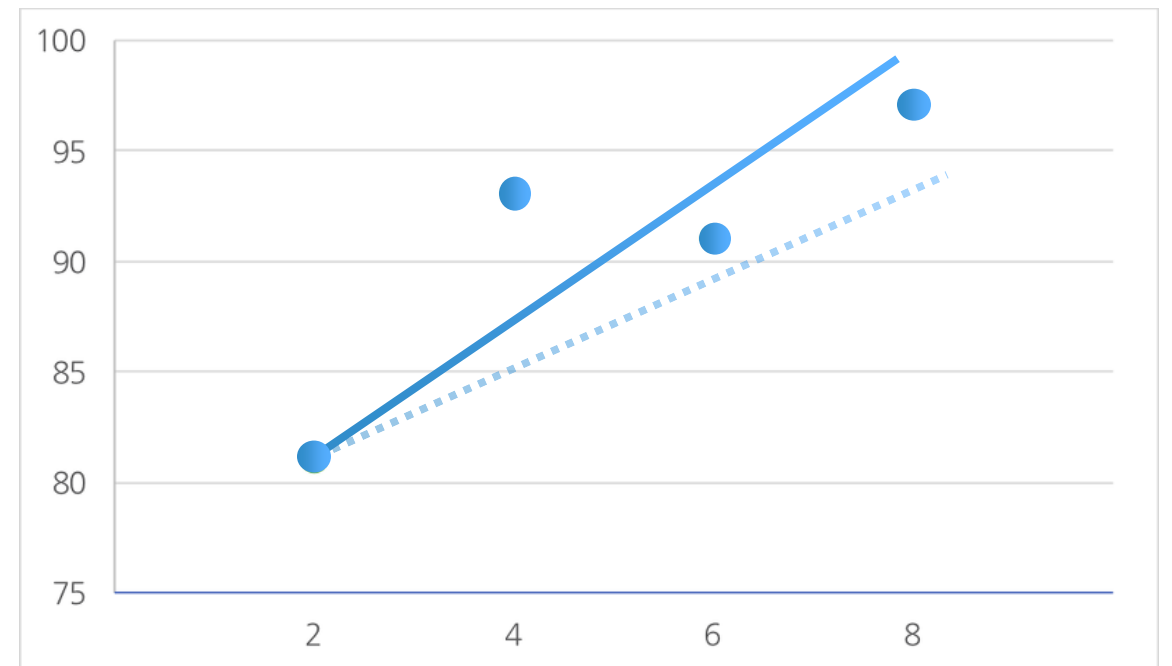
## 04 가장 간단한 예제 - 규칙성 찾기(오차의 최소화)

랜덤으로 그린 직선( $W$ 와  $B$ 가 random값)



↕ : 실제값과 예측값의 오차를 나타냄.

오차를 최소화하도록 하는 직선  
(적절한  $W$ 와  $b$ 의 값을 찾아냄)



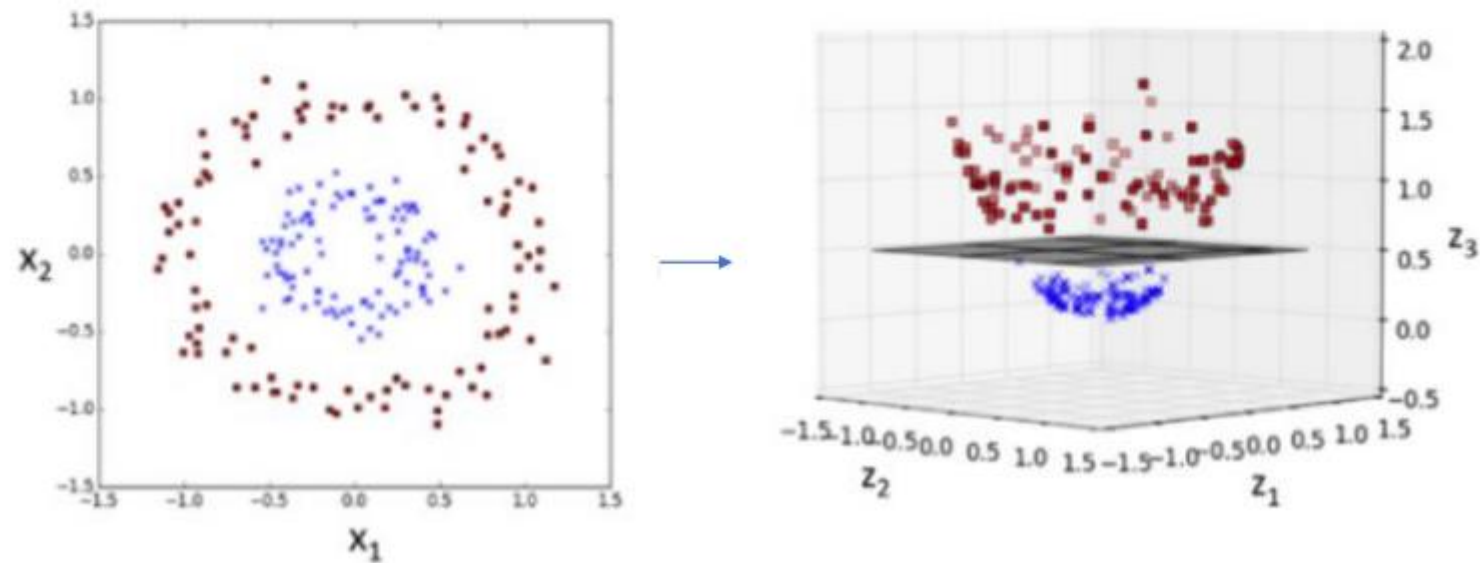
이제 이 직선으로는 score를 예측 가능.  
데이터로부터 규칙을 찾아낸 것.



## 데이터 학습 이해하기

04 선형 모델이 최선일까요? 저희가 다루는 데이터가 모두 선형성만 보이는 것은 아니겠죠?

### 고차원 변환을 통해 선형 모델링

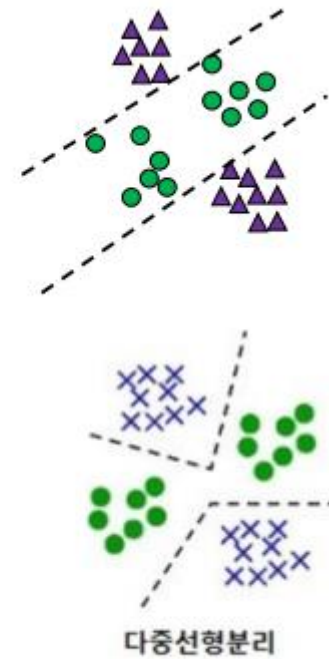
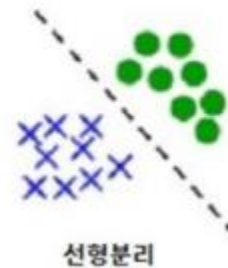
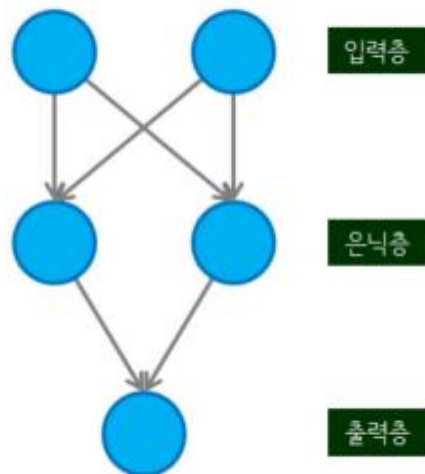


# 💡 데이터 학습 이해하기

04 선형모델이라고 직선을 하나만 쓰는 것은 아닙니다.

## 다층신경망(Multi-Layer Perception)

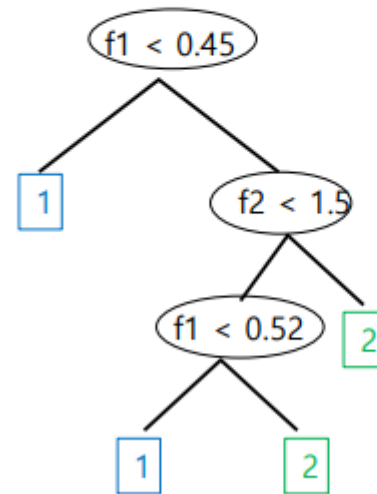
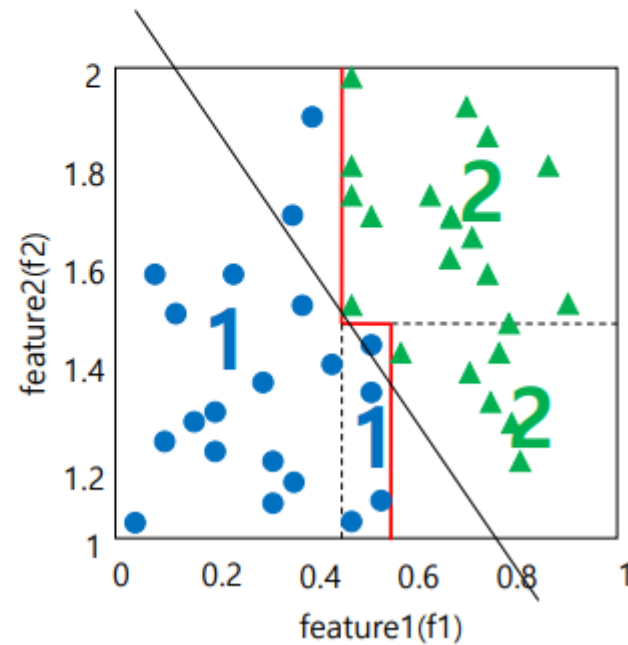
다층 신경망을 통한 비선형 문제 해결



# 💡 데이터 학습 이해하기

04 선형모델이라고 직선을 하나만 쓰는 것은 아닙니다.

## 다층 분류기를 이용한 문제 해결



# 💡 데이터 학습 이해하기

04 선형모델이라고 직선을 하나만 쓰는 것은 아닙니다.

## 딥러닝(Deep Learning)

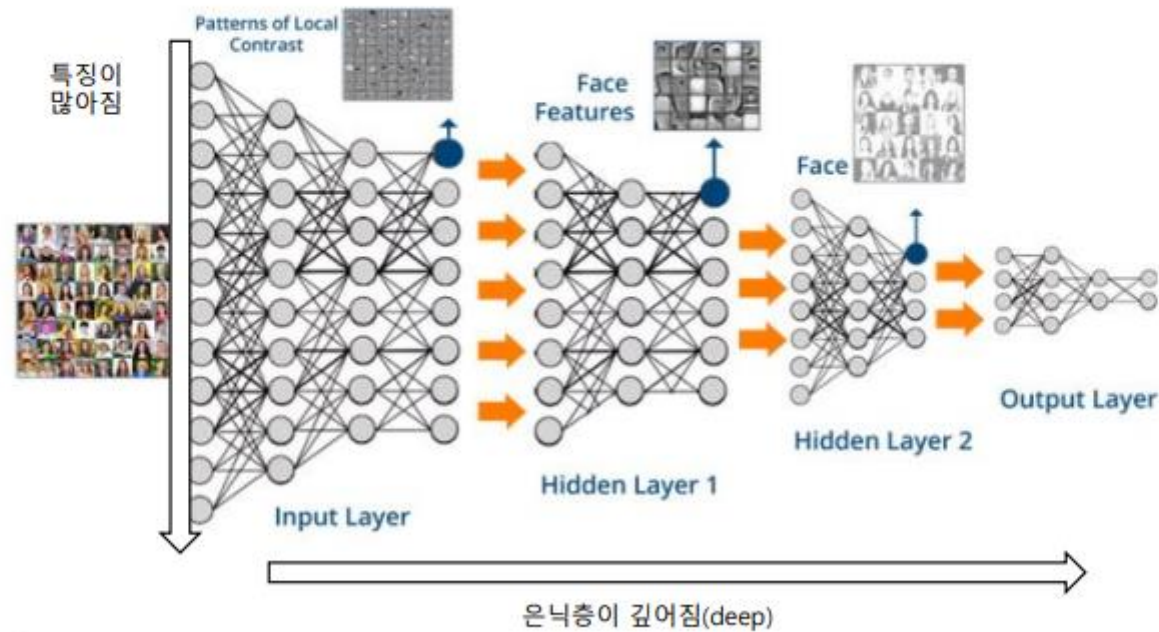


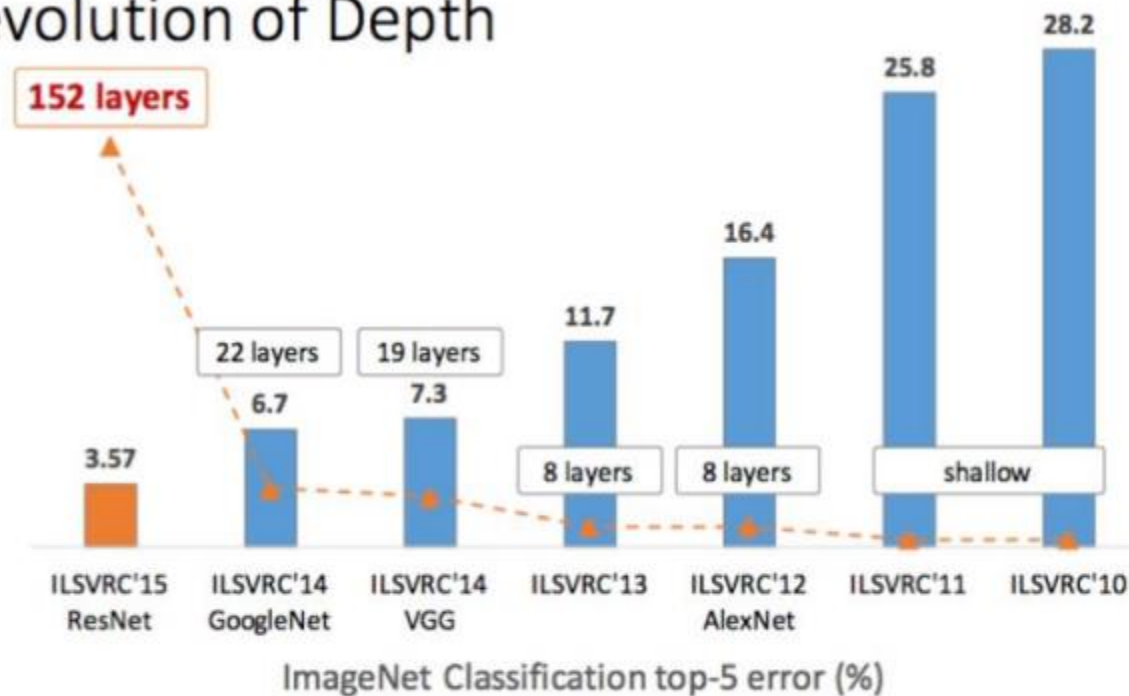
Image from: <https://cdn.edureka.co/blog/wp-content/uploads/2017/05/Deep-Neural-Network-What-is-Deep-Learning-Edureka.png>

# 💡 데이터 학습 이해하기

04 선형모델이라고 직선을 하나만 쓰는 것은 아닙니다.

## 딥러닝(Deep Learning)

Revolution of Depth



<https://medium.com/@Lidinwise/the-revolution-of-depth-fac174924f5>





# AI의 한계와 더 나은 방법

# AI의 한계와 더 나은 방법

## 01 AI 활용의 어려움

### 사람들이 AI를 쓰지 않는 '진짜' 솔직한 이유

슬라이드 1/5



많은 사람들이 AI의 잠재력을 인정하면서도  
실제 사용에는 여전히 큰 장벽이 존재합니다

**26%**

AI를 한 번도  
사용하지 않음

**72%**

코파일럿 도입 조직 중  
활용에 어려움을 겪음

# AI의 한계와 더 나은 방법

## 01 AI 활용의 어려움

### AI 사용 현황

슬라이드 2/5

AI 사용자 비율

49%

#### 주요 통계

- 2022년 11월 챗GPT 공개 이후 관심 증가
- 응답자 49%만 AI 도구 실제 사용
- 26%는 한 번도 사용한 적 없음

#### 도입 장벽

- MS365 코파일럿 도입 조직의 72%가 어려움 겪음
- 실제 적극적 사용자는 소수에 불과
- 교육 참가자들도 실제 사용은 저조

# AI의 한계와 더 나은 방법

## 01 AI 활용의 어려움

### AI 사용을 망설이는 이유

슬라이드 3/5

#### 사용하고 싶지만 못하는 경우

##### 사용법이 어렵다

기존 도구와 다른 방식, 프롬프트 작성 필요

##### 정답이 없음

사용 방법이 계속 바뀌어 익히기 어려움

##### 체계적 교육 부족

개인적으로 시도해보는 경우가 많음

#### 사용하고 싶지 않은 경우

##### AI를 신뢰하지 않음

응답 정확성 의심, 할루시네이션 우려

##### 검수 부담

AI 결과물 검증이 귀찮게 느껴짐

##### 기존 방식 고수

새로운 도구 사용이 귀찮고 익숙한 방식 선호

# AI의 한계와 더 나은 방법

## 01 AI 활용의 어려움

### AI 활용을 촉진하기 위한 방법

슬라이드 4/5



#### 체계적인 교육

AI 사용법과 사례를 배우고 신뢰 형성



#### 의도적인 사용 시도

다양한 시도를 통해 익숙해지는 과정 필요



#### 실험적인 접근

사용해보기 전에는 그 효과를 알 수 없음



#### 신뢰를 갖고 사용

AI 결과물을 믿기로 결심하는 태도 필요

#### AI 도입 과정

1

교육

2

실험

3

신뢰 형성

4

통합

5

일상화

# 💡 AI의 한계와 더 나은 방법

## 01 AI 활용의 어려움

### 결론

슬라이드 5/5



#### 주요 발견점

- ✓ AI 도구의 잠재력은 인정받지만 **사용법의 어려움**과 **신뢰 부족**이 주요 장벽
- ✓ 새로운 기술 수용에는 **체계적인 교육**과 **실험적 접근**이 필수적
- ✓ 조직 내 AI 도입은 기술적 문제보다 **문화와 습관**의 변화가 핵심

#### 📌 데이터 기반 AI 활용의 중요성

##### 데이터 수집



양질의 데이터 수집이 AI 성능의 기반

##### 데이터 가공



구조화된 형태로 정제하여 활용 가능성 증대

##### AI 연계



맥락에 맞는 데이터와 AI의 통합이 핵심

**효과적인 AI 활용을 위한 데이터 전략:** AI의 효율적 활용을 위해서는 단순히 AI 도구를 사용하는 것을 넘어, **양질의 데이터**를 **체계적으로 수집**하고, **목적에 맞게 가공**하여, **AI와 효과적으로 연계**하는 종합적인 접근이 필요합니다. 데이터의 품질과 관련성이 AI 성과의 핵심 결정 요소입니다.



# AI의 한계와 더 나은 방법

## 02 더 나은 방법

### AI의 한계와 더 나은 방법

AI가 정말로 어떻게 작동하는지, 그리고 어떻게 개선할 수 있는지



#### AI에 대한 오해와 진실

- AI가 어떻게 작동하는지에 대한 많은 오해가 있어요
- 컴퓨터가 정말 '생각'하는 것이 아니에요
- AI는 우리가 생각하는 것보다 더 단순한 원리로 작동해요
- AI의 한계를 이해하는 것이 중요해요

# AI의 한계와 더 나은 방법

## 02 더 나은 방법

### AI의 한계와 더 나은 방법

AI가 정말로 어떻게 작동하는지, 그리고 어떻게 개선할 수 있는지



#### AI의 실제 한계

- AI는 실제로 매우 좁은 범위의 일만 할 수 있어요
- ChatGPT는 실제로 이해하는 것이 아니라 패턴을 찾아내는 거예요
- AI는 엄청난 양의 전기를 소비하면서 작동해요
- AI는 단어들 사이의 관계는 알지만, 그 의미는 진짜로 모르죠

# AI의 한계와 더 나은 방법

## 02 더 나은 방법

### AI의 한계와 더 나은 방법

AI가 정말로 어떻게 작동하는지, 그리고 어떻게 개선할 수 있는지



#### 인간과 AI의 차이

- 운전할 때 우리는 단순히 보는 것보다 더 많은 것을 해요
- 신호등이 무엇인지, 그것이 왜 중요한지 이해하죠
- 다양한 상황에서 무엇이 더 중요한지 판단할 수 있어요
- 우리는 경험을 통해 배우고 직관적으로 결정을 내려요

# AI의 한계와 더 나은 방법

## 02 더 나은 방법

### AI의 한계와 더 나은 방법

AI가 정말로 어떻게 작동하는지, 그리고 어떻게 개선할 수 있는지



#### 온톨로지란 무엇일까?

- 온톨로지는 '사물이 무엇인지' 정의하는 방법이에요
- 예를 들어, '의자'가 무엇인지 컴퓨터에게 설명하는 것이죠
- 사물들 사이의 관계도 정의해요 (의자는 앉는 곳이다)
- 이렇게 명확한 정의는 AI가 더 정확하게 작동하게 도와줘요

# AI의 한계와 더 나은 방법

## 02 더 나은 방법

### 온톨로지 쉬운 예시



#### 의자란?

사람이 앉을 수 있는 가구

#### 의자의 특성:

- 앉을 수 있다
- 다리가 있다 (보통 4개)
- 등받이가 있을 수 있다

#### 의자의 관계:

- 의자는 가구의 일종이다
- 책상과 함께 사용할 수 있다
- 소파는 의자의 한 종류이다

이렇게 사물을 정의하고 관계를 설명하는 것이 온톨로지에요!

# AI의 한계와 더 나은 방법

## 02 더 나은 방법

### AI의 한계와 더 나은 방법

AI가 정말로 어떻게 작동하는지, 그리고 어떻게 개선할 수 있는지



#### 더 나은 AI를 위한 방법

- 더 많은 데이터만으로는 AI의 문제를 해결할 수 없어요
- AI에게 세상을 체계적으로 설명해주는 것이 필요해요
- 사물과 그들의 관계를 명확하게 정의해주는 것이 중요해요
- AI는 도구일 뿐, 모든 문제의 해결책은 아니에요



# AI의 한계와 더 나은 방법

## 02 더 나은 방법

### 인간 vs. AI


#### 인간의 지능


- 경험을 통해 배워요
- 직관이 있어요
- 맥락을 이해해요
- 상식을 가지고 있어요
- 창의적인 생각을 해요


#### 현재 AI의 한계


- 데이터 패턴만 인식해요
- 진짜 이해는 없어요
- 많은 전력을 소모해요
- 좁은 범위의 일만 잘해요
- 진짜 창의력이 없어요

### 결론: AI를 더 나아지게 하려면

 명확한 정의

 관계 설명

 체계적 접근

 인간의 지식 활용

# AI의 한계와 더 나은 방법

## 03 Data-centric AI

### AI시대에도 중요한 것은 Data

모든 비즈니스 및 AI 적용은 Data가 필요

- ➔ 주식 투자에도 과거 및 시장 Data가 필수
- ➔ AI는 '예측', '탐지', '생성'을 위한 학습 Data가 필요

**Data-Centric AI** (Andrew Ng 교수)

- ➔ Model-Centric AI는 모델 개선과 연산 자원 사용에 집중  
Data-Centric AI는 데이터 품질 개선을 통해 AI 성능을 높임
- ➔ 이 방법이 **AI 초보 기업** 및 **중소기업**에 특히 효과적  
데이터 품질 향상이 모델 개선보다 유리할 수 있음

기준	Model-centric AI	Data-centric AI
중점 부분	모델 아키텍처 개선	데이터 품질 다양성
핵심 전략	신경망 최적화	데이터 정제 및 레이블링
활용 자원	컴퓨팅 자원	다양한 데이터 효율적 사용
주요 목표	모델 능력 향상	데이터로 성능 개선

# 💡 AI의 한계와 더 나은 방법

## 03 Data-centric AI

### AI시대에도 중요한 것은 Data

모든 비즈니스 및 AI 적용은 Data가 필요

➔ 주식 투자에도 과거 및 시장 Data가 필수

➔ AI는 '예측', '탐지', '생성'을 위한 학습 Data가 필요

➔ Model-Centric AI는 모델 개선과 연산 자원 사용에 집중  
Data-Centric AI는 데이터 품질 개선을 통해 AI 성능을 높임

➔ 이 방법이 AI 초보 기업 및 중소기업에 특히 효과적 데이터 품질 향상이 모델 개선보다 유리할 수 있음

기준

Model-centric AI

Data-centric AI

중점

모델 아키텍처 개선

데이터 품질 다양성

활용  
자원

신경망 최적화

데이터 정제 및 레이블링

컴퓨팅 자원

다양한 데이터 효율적 사용

주요  
목표

모델 능력 향상

데이터로 성능 개선

# 💡 AI의 한계와 더 나은 방법

## 03 Data-centric AI



# 💡 AI의 한계와 더 나은 방법

## 03 Data-centric AI

### AI시대에도 중요한 것은 Data

모든 비즈니스 및 AI 적용은 Data가 필요

- ➔ 주식 투자에도 과거 및 시장 Data가 필수
- ➔ AI는 '예측', '탐지', '생성'을 위한 학습 Data가 필요

**Data-Centric AI** (Andrew Ng 교수)

- ➔ Model-Centric AI는 모델 개선과 연산 자원 사용에 집중

- ➔ Data-Centric AI는 **데이터** 품질 개선을 통해 AI 성능을 높임

기준	Model-centric AI	Data-centric AI
중점 부분	모델 아키텍처 개선	데이터 품질 다양성
핵심 전략	신경망 최적화	데이터 정제 및 레이블링
확용	컴퓨팅 자원 능력 향상	다양한 데이터 효율적 사용 데이터로 성능 개선





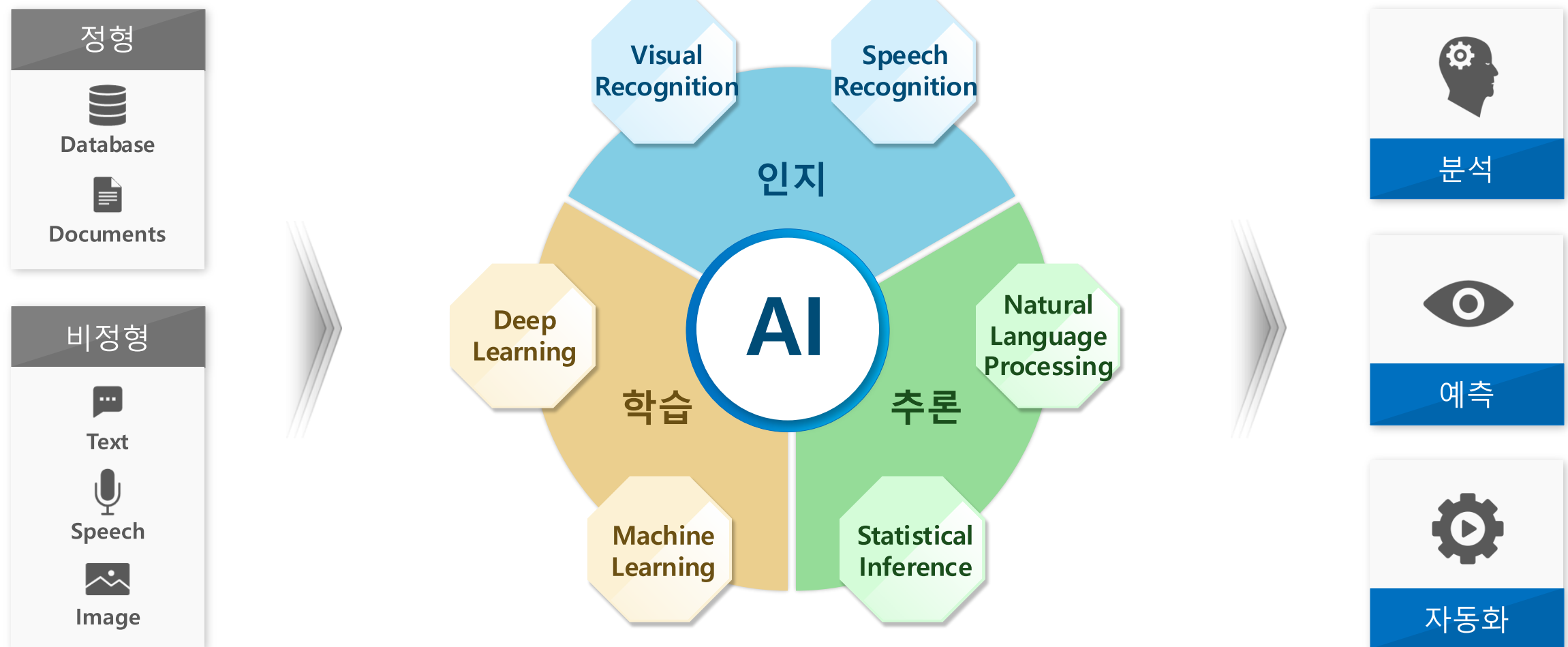
# AI, 머신러닝, 딥러닝



# 💡 AI와 머신러닝 그리고 딥러닝

## 01 인공지능과 머신러닝 : 인공지능의 개념

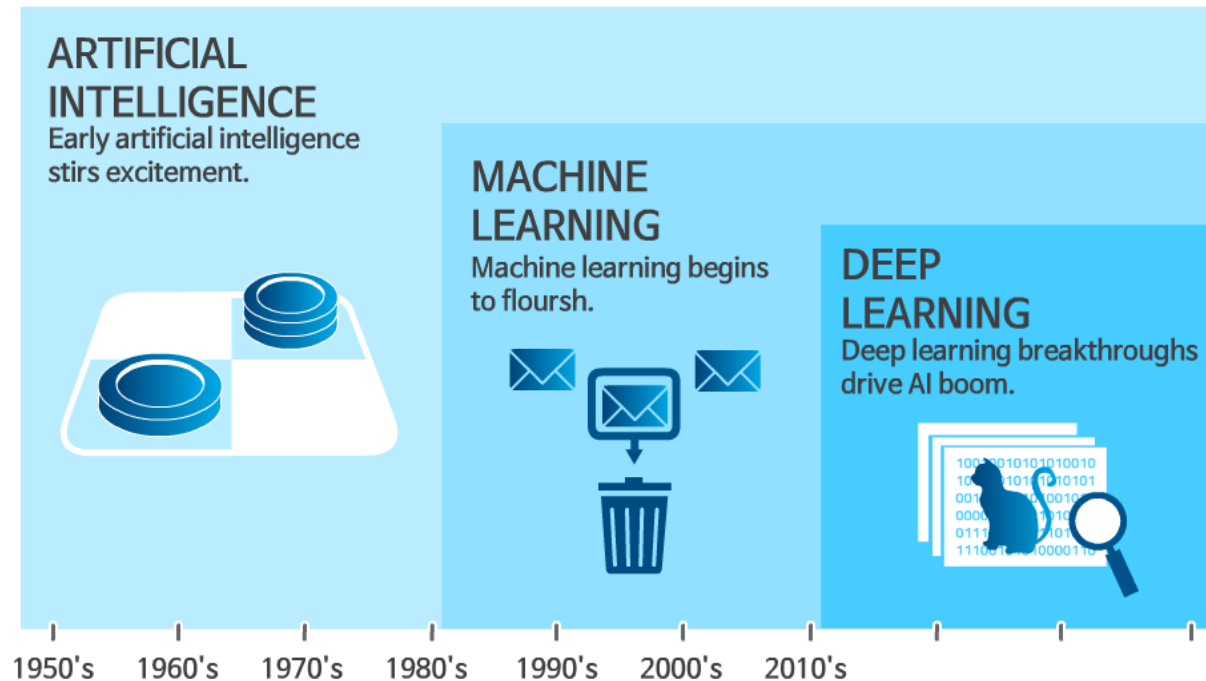
- 인간의 지능을 모방하여 만든 컴퓨터 프로그램



인공지능은 사람의 **인지, 이해, 추론, 학습능력**을 모방함으로써,  
더욱 효율적이고 정확도 높은 **분석 · 예측 · 자동화** 등 모든 범주에 적용 가능합니다.

# 💡 AI와 머신러닝 그리고 딥러닝

## 01 인공지능과 머신러닝 : 인공지능 기술의 분류



### Artificial Intelligence

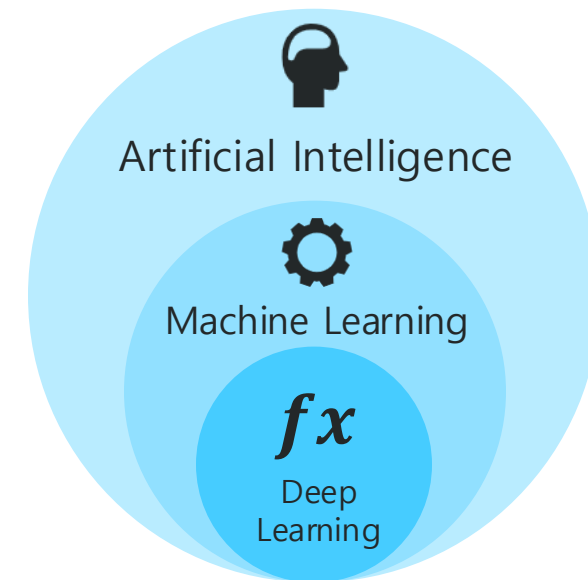
Any technique which enables computers to mimic human behavior.

### Machine Learning

Subset of AI techniques which use statistical methods to enable machines to improve with experiences.

### Deep Learning

Subset of ML which make the computation of multi-layer neural networks feasible.



# 💡 AI와 머신러닝 그리고 딥러닝

## 01 인공지능과 머신러닝 : 인공지능의 역사

- 90' 암흑기와 2010' 이후의 재조명



IBM Watson  
DeepQA  
won on  
jeopardy ['11]



Google DeepMind

Google DeepMind,  
AlphaGo ['12, '15]

규칙/논리적 프로그래밍

통계/확률적 기계학습

딥러닝

1956  
Birth

1970~1980  
Boom

1990~2000  
Winter

2000~2010  
Deep Learning

Cognitive  
Computing

J.McCarthy  
Dartmouth  
Conference

Expert systems  
Symbolic AI

Neural Networks  
Fuzzy Systems

Intelligent Agents  
Machine Learning



IBM "Deep Blue"  
beats human chess  
champion ['97]

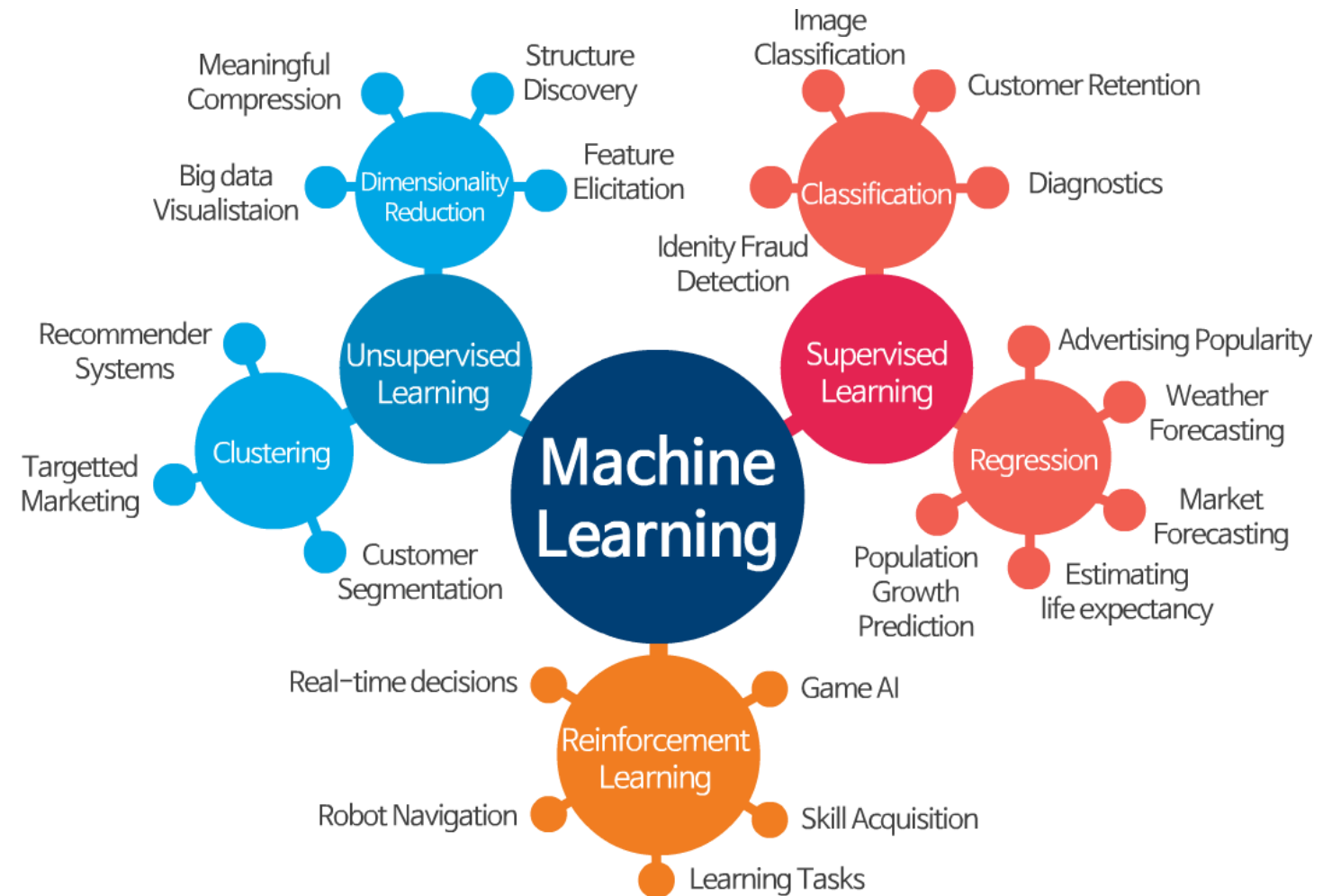
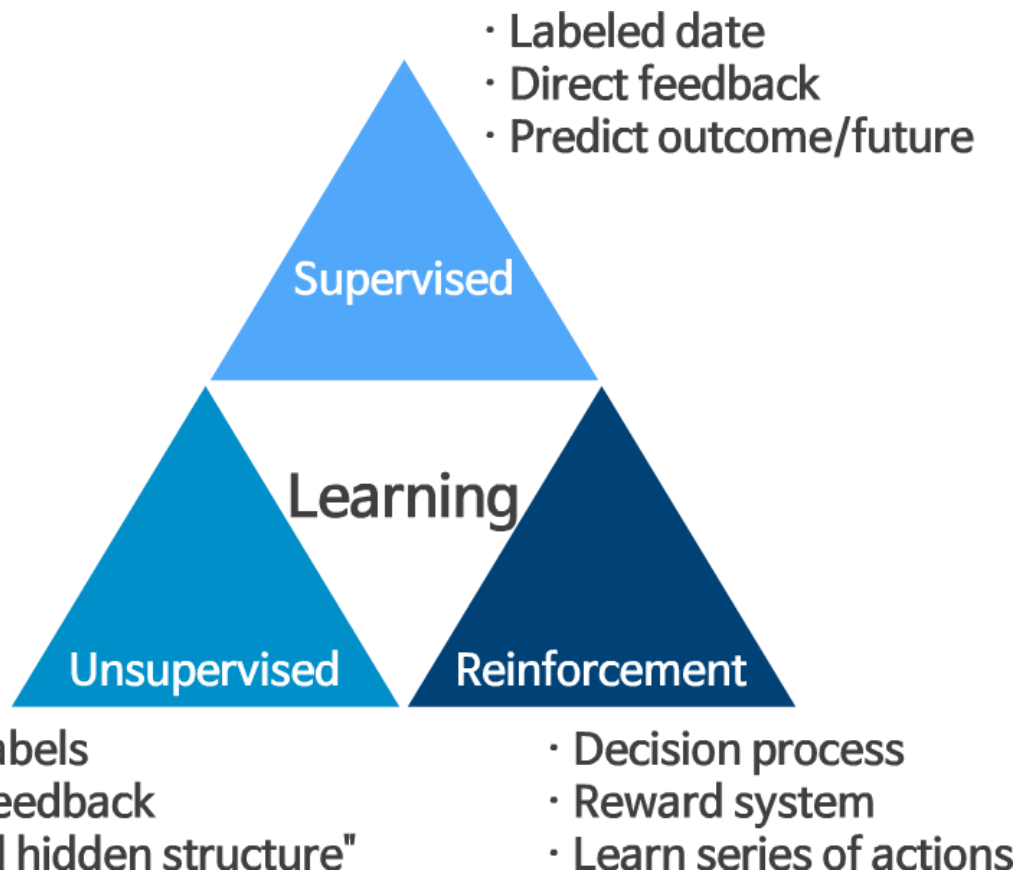
Deep Architecture  
breakthrough  
[Hinton, '06]



IBM TrueNorth ['14]

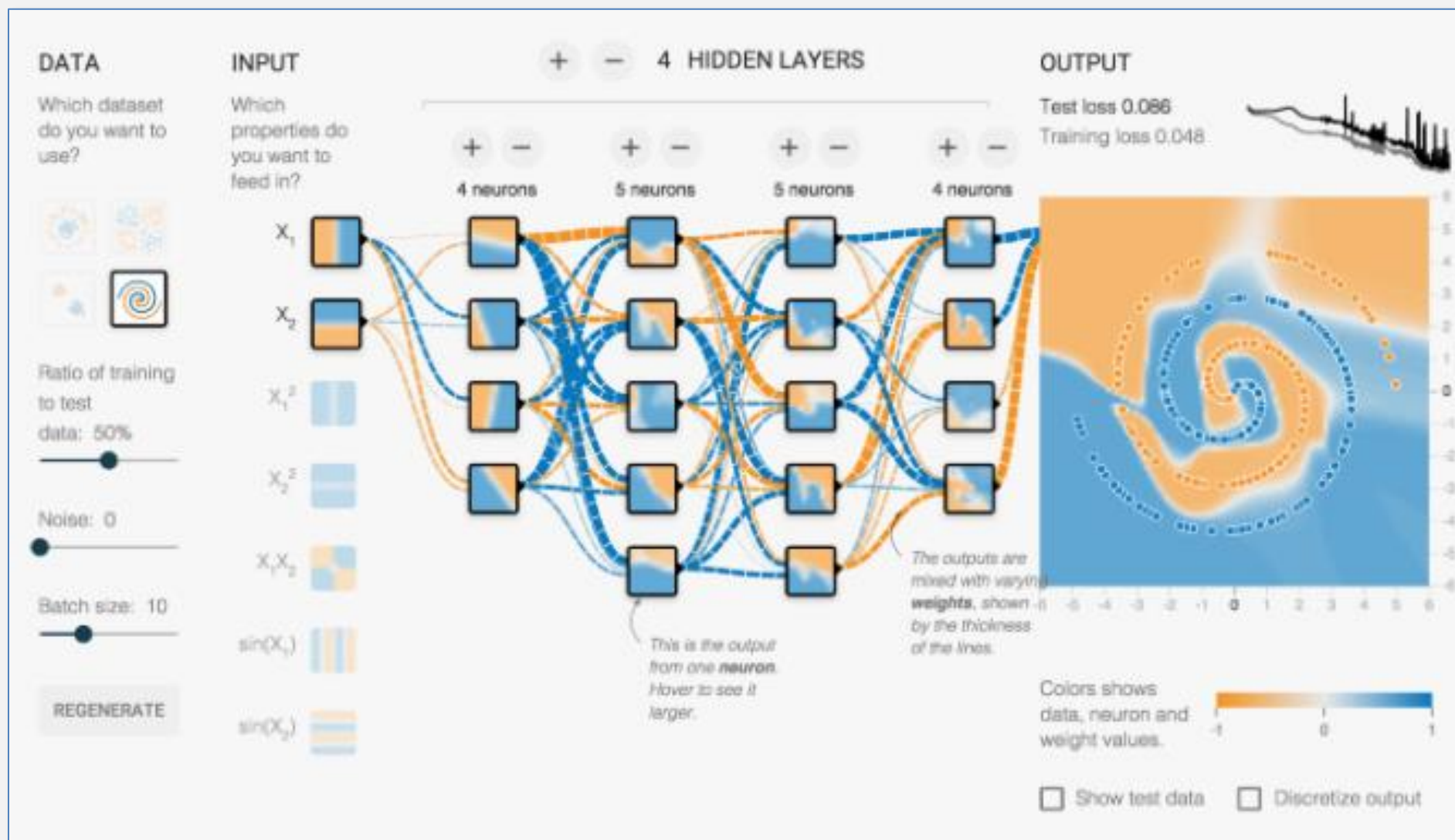
# 💡 AI와 머신러닝 그리고 딥러닝

## 01 인공지능과 머신러닝 : ML의 구분



# 💡 AI와 머신러닝 그리고 딥러닝

## 02 Deep Learning



# 💡 AI와 머신러닝 그리고 딥러닝

## 02 딥러닝 : 딥 러닝(Deep Learning)의 역사



퍼셉트론  
XOR 문제

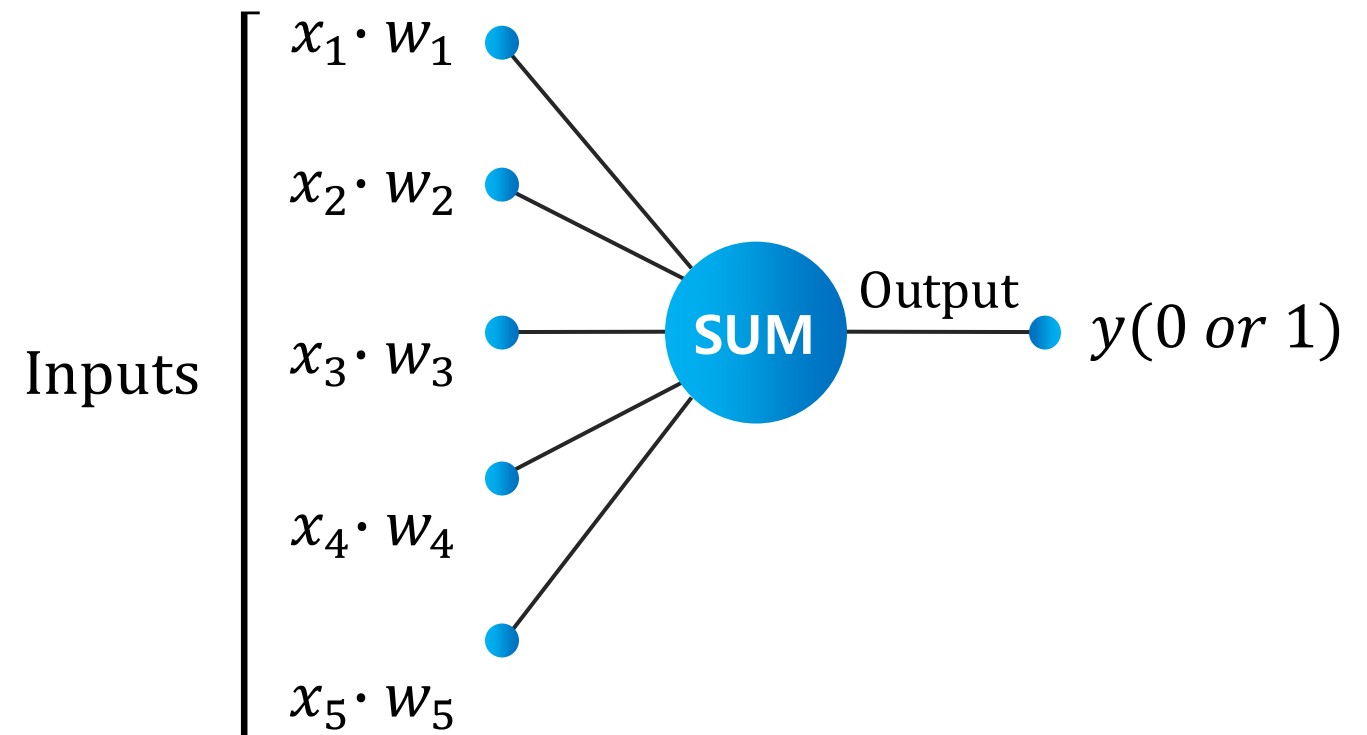
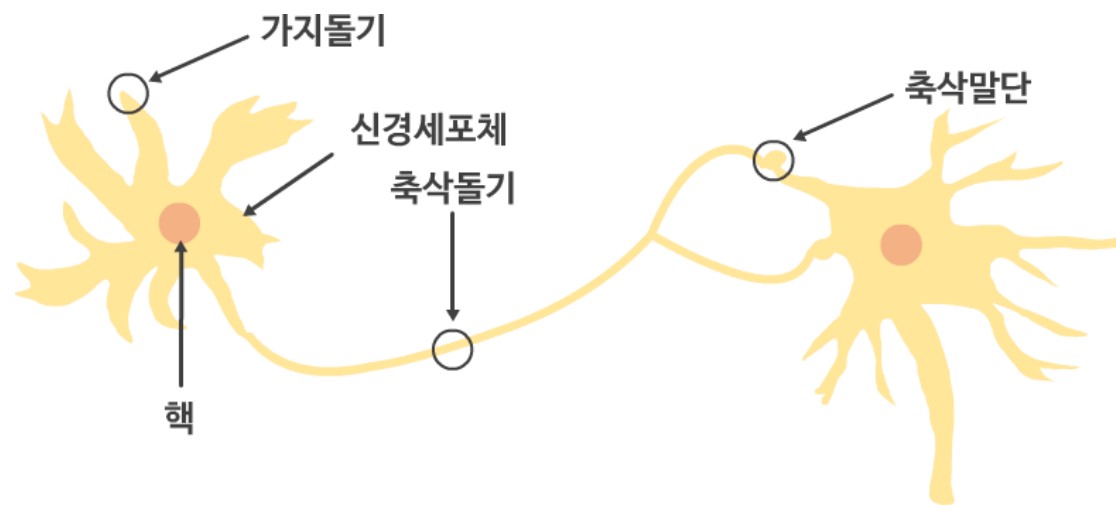
다층 퍼셉트  
가중치 계산 문  
제

깊은 신경망/  
딥 러닝의 등  
장



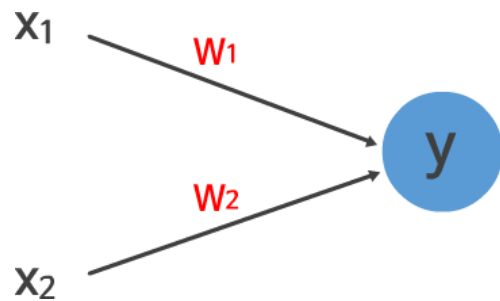
# 💡 AI와 머신러닝 그리고 딥러닝

## 02 딥러닝 : 1세대 - 퍼셉트론(1960년대)



# 💡 AI와 머신러닝 그리고 딥러닝

## 02 딥러닝 : 1세대 - 퍼셉트론(1960년대)



$w1=0.5$   $w2=0.5$   $\theta=0.7$  ➡

AND 게이트

X1	X2	y
0	0	0
0	1	0
1	0	0
1	1	1

$$output = \begin{cases} 1 & \text{if } \sum w_i x_i > \theta \\ 0 & \text{otherwise} \end{cases}$$

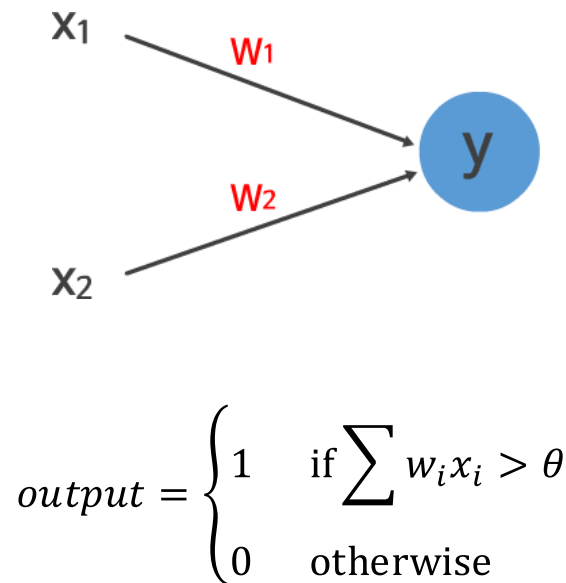
$w1=-0.5$   $w2=-0.5$   $\theta=-0.7$  ➡

NAND 게이트

X1	X2	y
0	0	1
0	1	1
1	0	1
1	1	0

# 💡 AI와 머신러닝 그리고 딥러닝

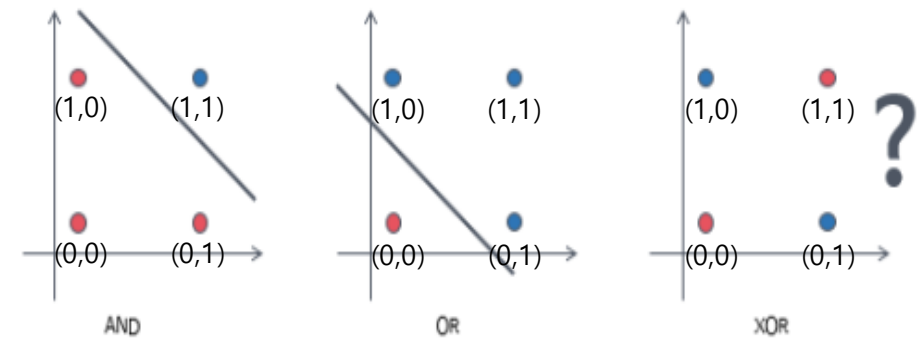
## 02 딥러닝 : 1세대 - 퍼셉트론(1960년대)



XOR 게이트

X1	X1	y
0	0	0
0	1	1
1	0	1
1	1	0

w1=? w2=? theta=?



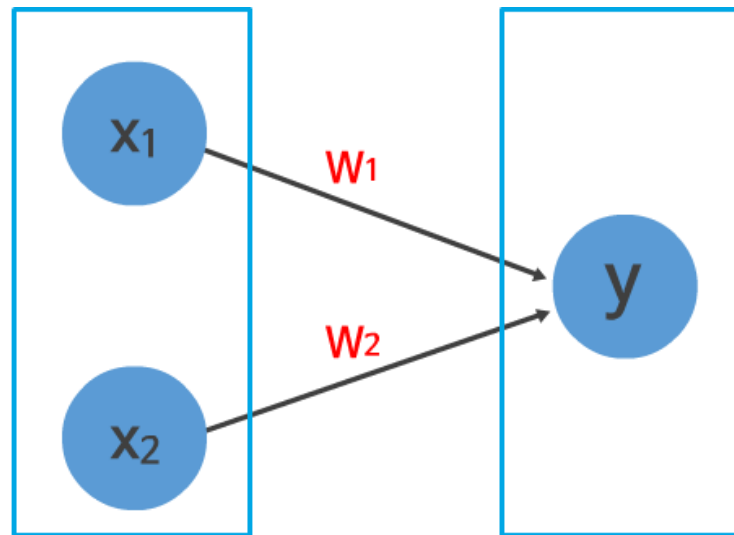
### 문제점

이 퍼셉트론에 그 어떤 가중치 **w**를 시도해도  
XOR 게이트 구현은 불가능.

직선 하나로 분류가 가능한 문제에 대해서만 문제를 풀 수 있음.

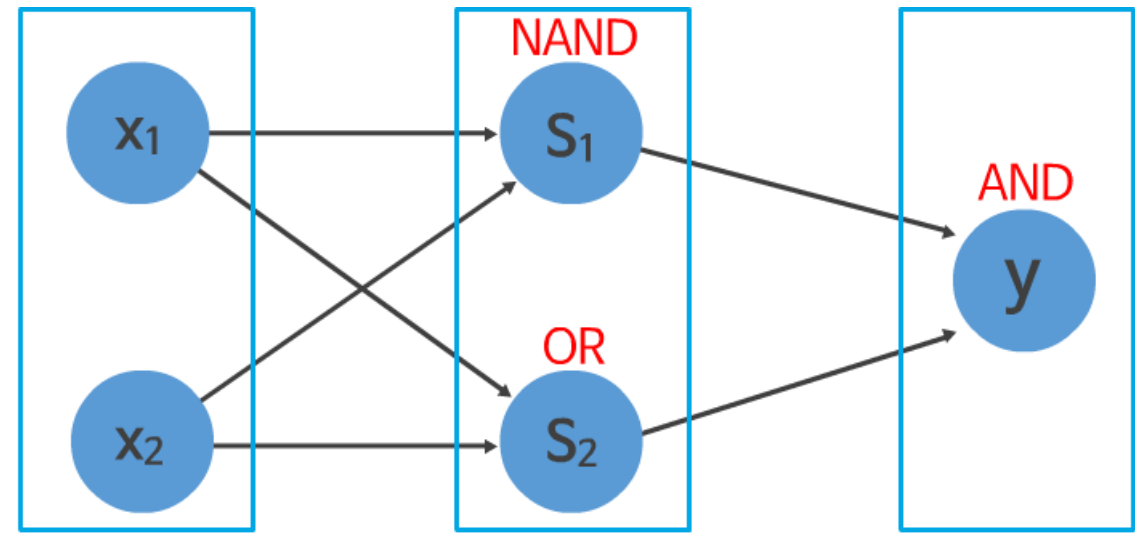
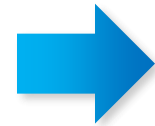
# 💡 AI와 머신러닝 그리고 딥러닝

## 02 딥러닝 : 2세대 - 다층 퍼셉트론(1980 ~ 1990)



입력층(input layer)    출력층(output layer)

XOR 게이트를 구현 할 수 없는 퍼셉트론



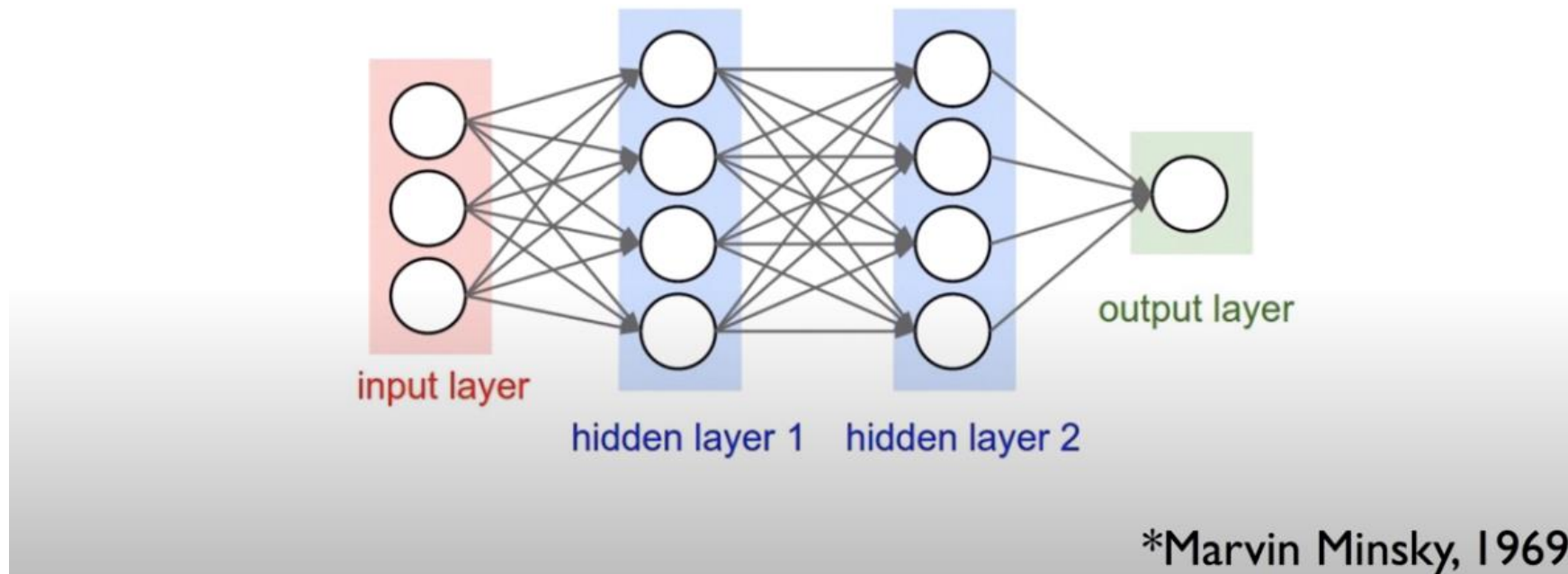
입력층(input layer)    은닉층(hidden layer)    출력층(output layer)

XOR 게이트를 구현할 수 있는 다층 퍼셉트론

# 💡 AI와 머신러닝 그리고 딥러닝

## 02 딥러닝 : 2세대 - 다층 퍼셉트론(1980 ~ 1990)

“No one on earth had found a viable way to train\*”



# 💡 AI와 머신러닝 그리고 딥러닝

## 02 딥러닝 : 3세대 - 깊은 신경망(2000 ~ 현재)

### CIFAR

- Canadian Institute for Advanced Research (CIFAR)
- CIFAR encourages basic research without direct application, was what motivated **Hinton** to move to Canada in 1987, and funded his work afterward.



CIFAR

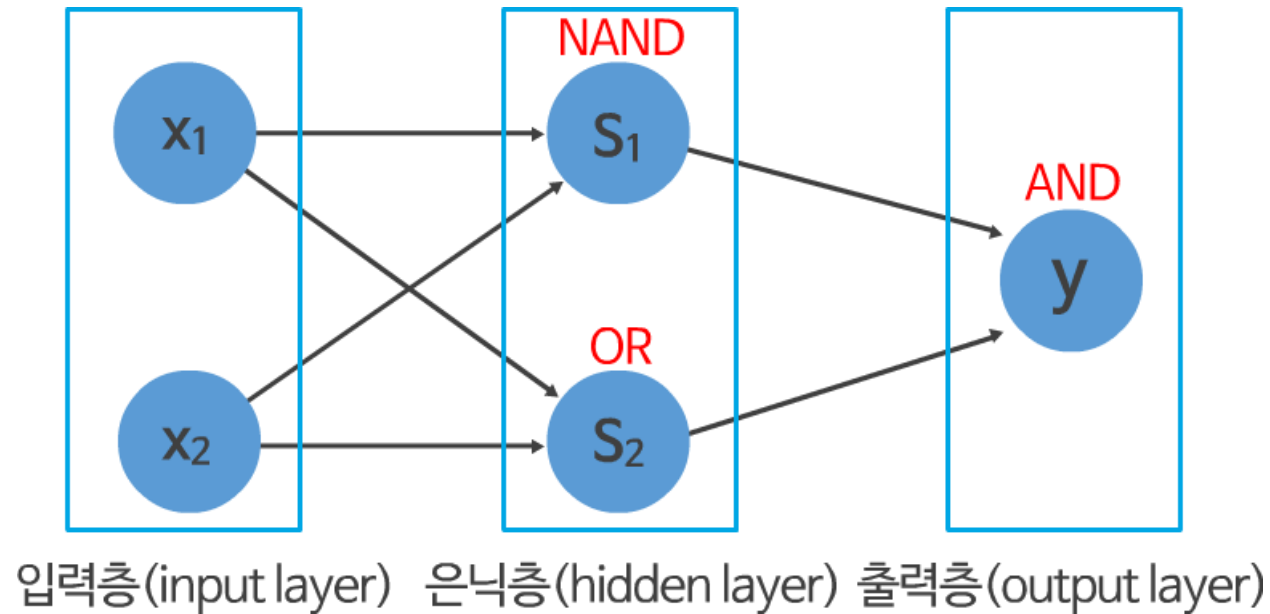
CANADIAN INSTITUTE  
for ADVANCED RESEARCH

<http://www.andreykurenkov.com/writing/a-brief-history-of-neural-nets-and-deep-learning-part-4/>

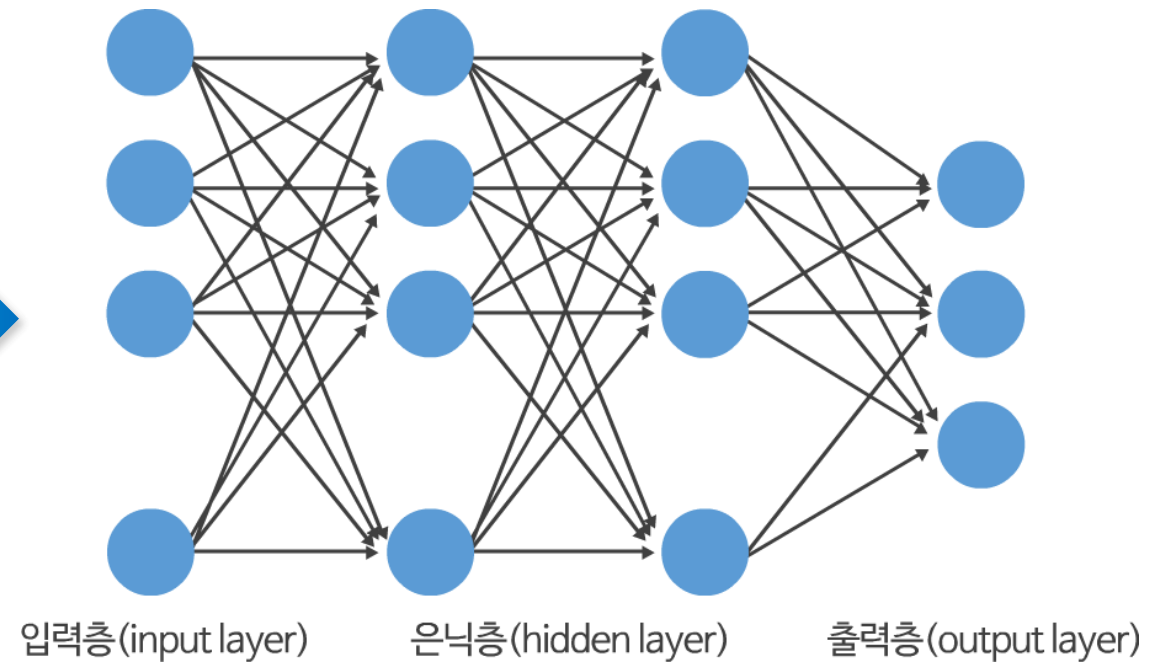
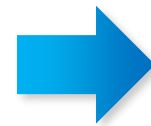


# 💡 AI와 머신러닝 그리고 딥러닝

## 02 딥러닝 : 3세대 - 깊은 신경망(2000 ~ 현재)



XOR 게이트를 구현할 수 있는 다층 퍼셉트론



이미지 인식, 텍스트 분류 등 다양한 문제를 풀 수 있는 깊은 신경망

# 💡 AI와 머신러닝 그리고 딥러닝

## 02 딥러닝 : 딥 러닝이 갑자기 떠오른 이유

### 알고리즘 개선

DBN, CNN 등 새로운  
딥러닝 알고리즘의  
등장으로 기존  
심층신경망의 문제점과  
한계를 해결

### 사물인터넷

센서를 통해 실시간으로  
데이터를 수집함으로써  
방대한 빅데이터를  
만들어내는 원동력으로  
작용

### 클라우드 플랫폼

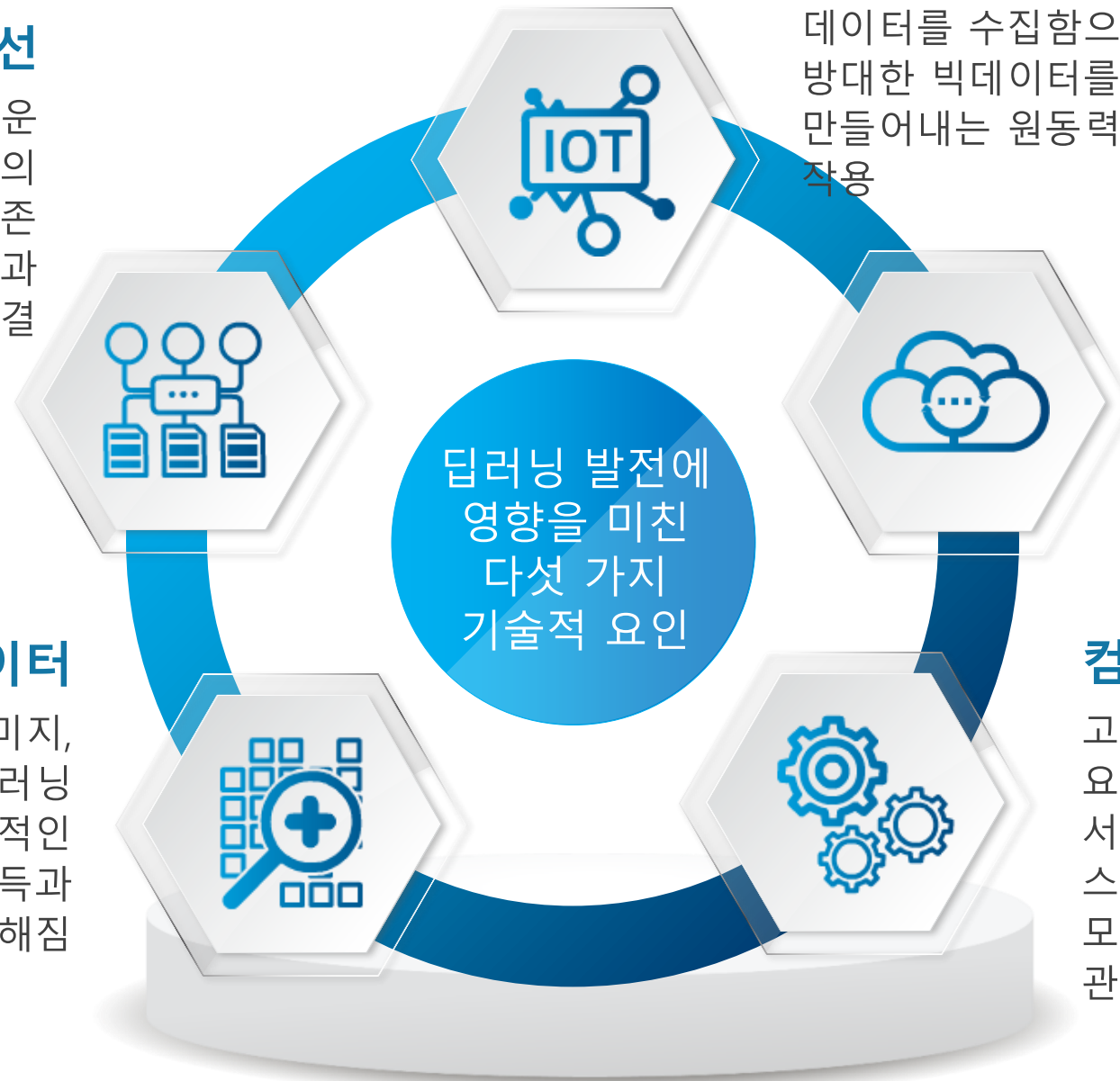
CPU와 GPU 프로세서의  
고성능화에 힘입어  
딥러닝 연산에 소요되는  
시간을 크게 단축

### 빅데이터

텍스트, 동영상, 이미지,  
음성파일 등 딥러닝  
정확도 향상에 필수적인  
방대한 데이터 획득과  
처리가 가능해짐

### 컴퓨터 성능 발전

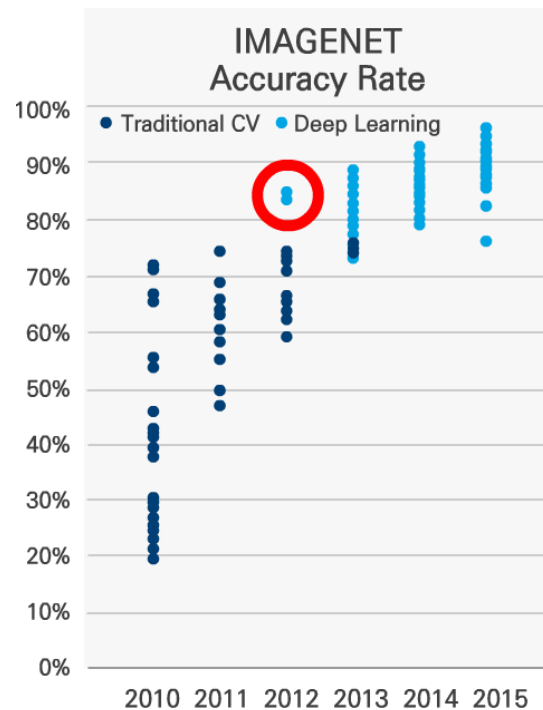
고성능 CPU와 GPU가  
요구되는 딥러닝 연산을  
서버에서 담당함으로써,  
스마트폰, 태블릿PC 등  
모바일 단말에서 딥러닝  
관련 서비스를 가능하게 함



# 💡 AI와 머신러닝 그리고 딥러닝

## 02 딥러닝 : 공부해야 하는 이유

- 은닉층을 계속해서 추가해도 가중치를 잘 찾아내는 방법들이 등장하고, 컴퓨터 성능이 좋아짐.
- 딥 러닝은 모델이 복잡해질수록 고성능의 인프라 자원을 필요로함. (2010년대에 급속도로 인프라의 발전.)
- 아이디어는 있었지만 현실적으로 시험해볼 수 없었던 모델들이 구현되면서 압도적인 성능을 보임.



**2012년**

이미지넷(이미지 인식 경진 대회)에 GPU 기반 딥 러닝 첫 등장과 함께

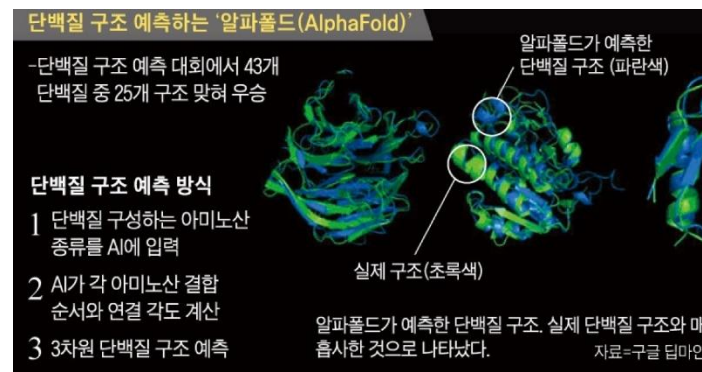


→ 딥 러닝 붐의 시작이 됨.



**2016년**

알파고 (은닉층 13개)



**2018년 12월**

단백질 구조 예측 대회에서 첫 참가한 딥 러닝 모델이 전통적인 방법을 사용하는 2위팀과 8배 이상의 성능 차이.





# 데이터를 이용한 AI 사례

# 부동산 데이터 어디까지 파봤니?

## 00 AI를 이용한 나만의 부동산 데이터 다루기

## 아파트 투자 매물 조회 서비스

👑 공유하지 마시고 사용 부탁드립니다. 👑

본 사이트의 정보는 참고용이지 어떠한 책임도 지지 않습니다.

[Home](#)

 시각화

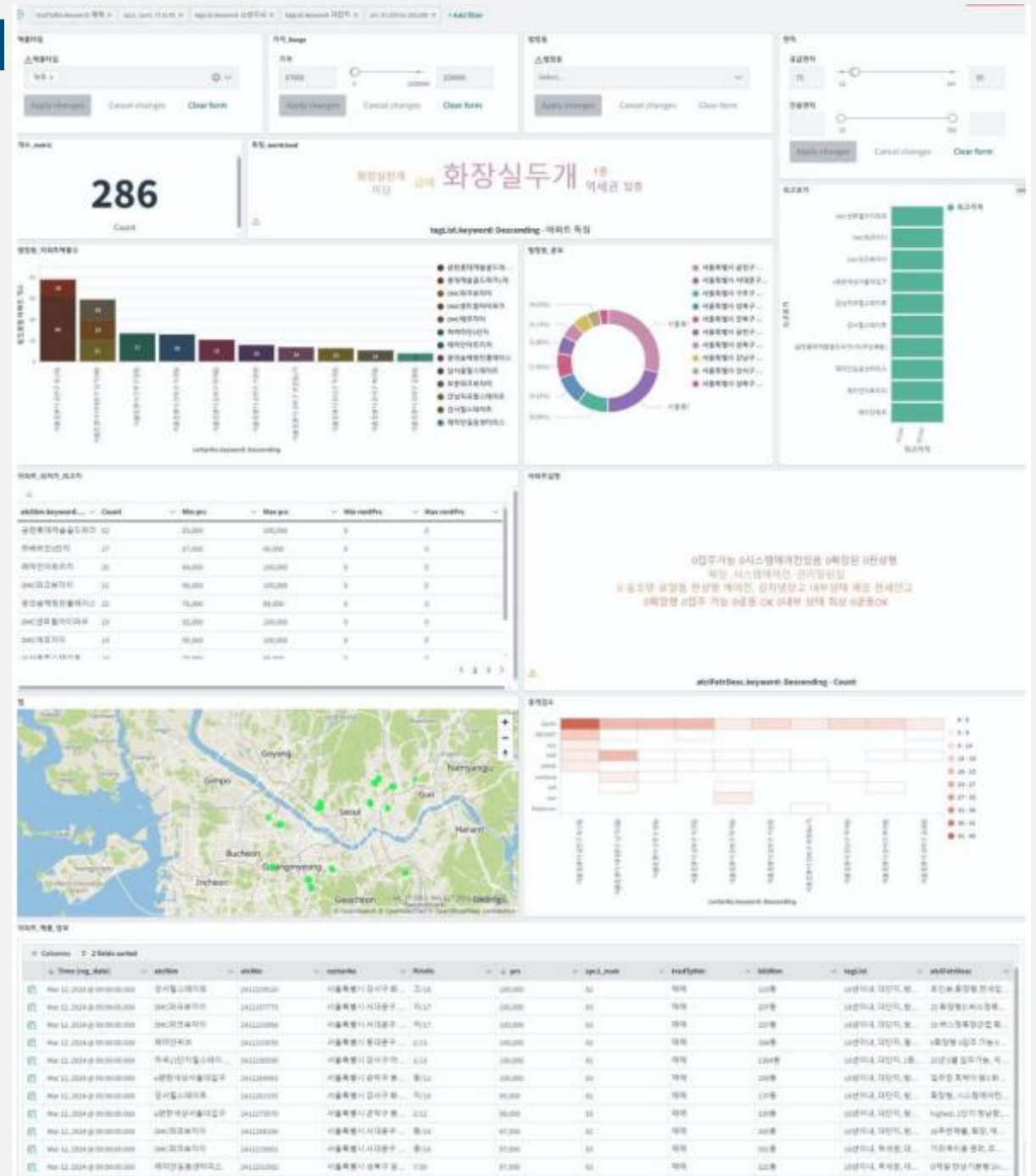
 AI챗봇

 급매

 갭투자

- 네이버 호가와 국토부 실거래가 기준 정보 제공(업데이트 주기: 매주 월요일)
- 아직 베타버전 (👉)입니다.
- 네이버 URL이 올바르게 않은 경우는 네이버부동산에서 아파트명을 수정하여 검색해보세요.
- 문의나 요구사항이 있으면 언제든지 연락주세요. ([humanist96@gmail.com](mailto:humanist96@gmail.com)) 🙋

<https://kevin-urgent-sale.streamlit.app/>





# 💡 부동산 데이터 어디까지 파봤니?

## 00 AI를 이용한 나만의 부동산 데이터 다루기

### 머신러닝 예측 개요

머신러닝 예측 페이지입니다. 🍌🍌🍌

🏠 Home

📊 주거형태별

📖 자치구역별

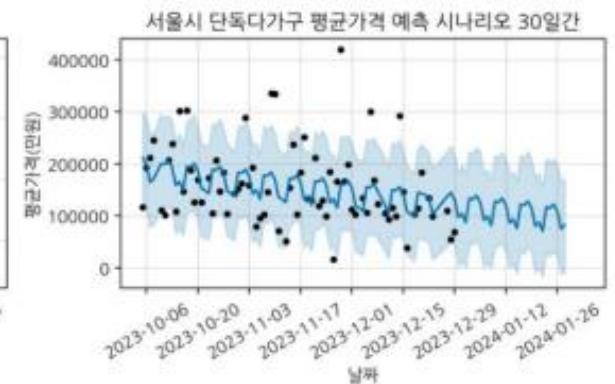
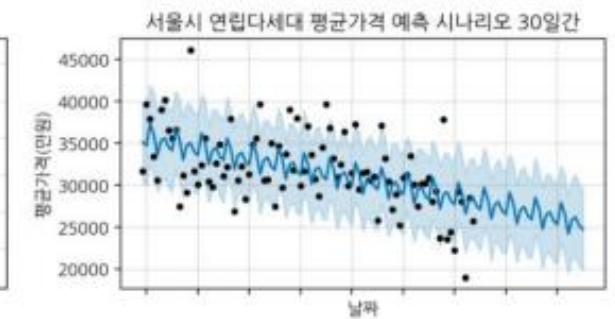
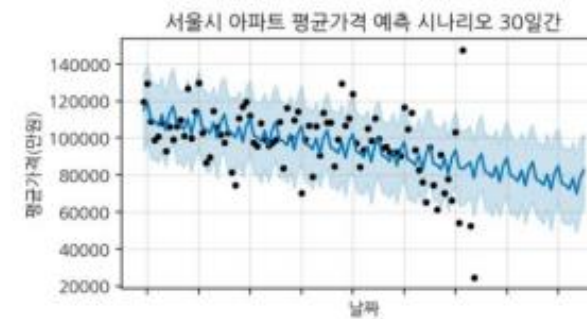
▶ 보고서

향후 예측 기간을 지정하세요(1일 ~ 30일)

30

폰트 선택

NanumGothic

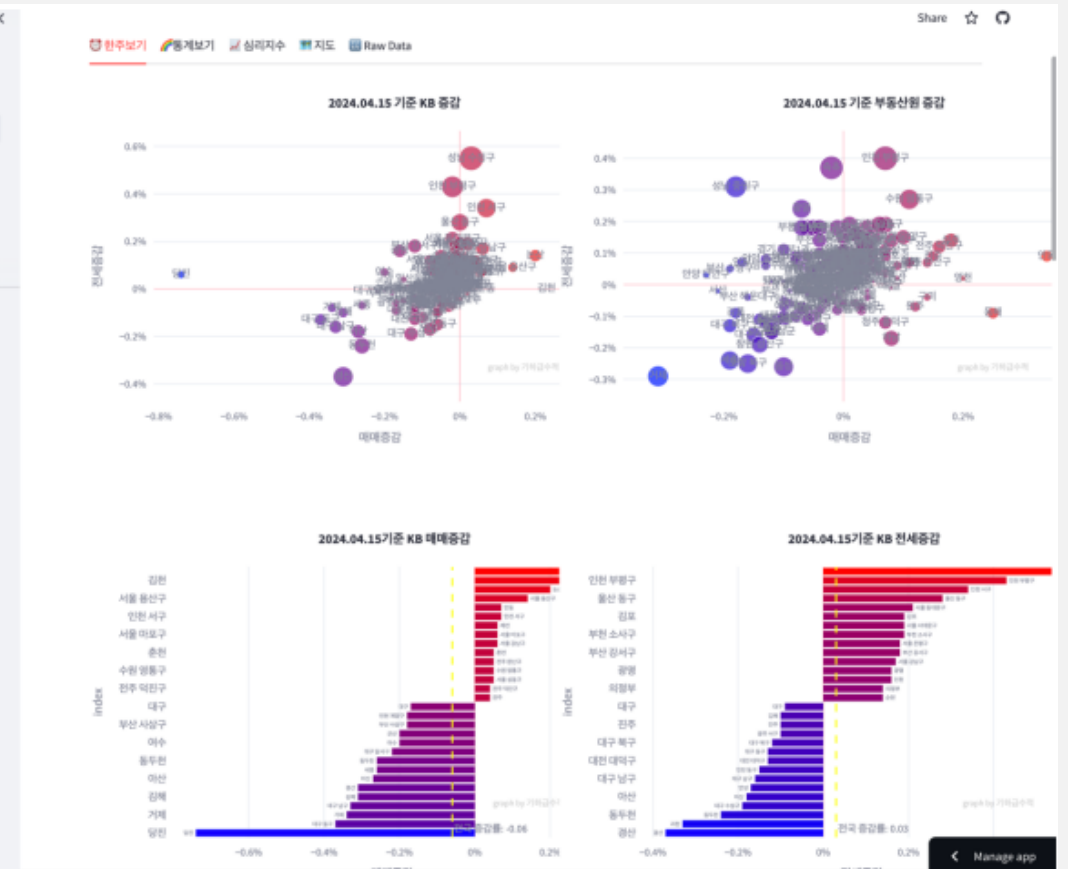
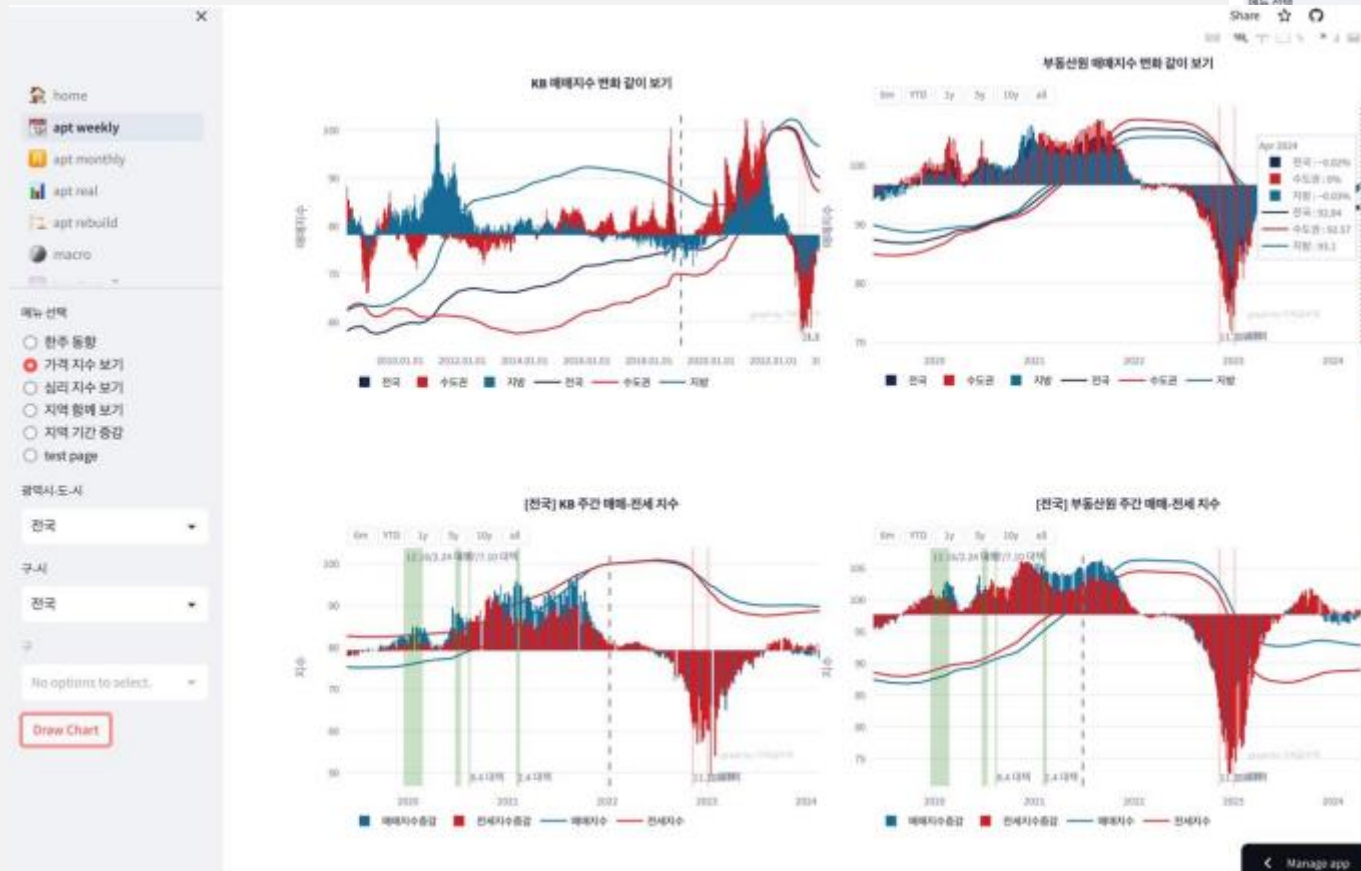


<https://kevin-seoul-realestate.streamlit.app/>



# 💡 부동산 데이터 어디까지 파봤니?

## 00 AI를 이용한 나만의 부동산 데이터 다루기



# 💡 부동산 데이터 어디까지 파봤니?

## 00 AI를 이용한 나만의 부동산 데이터 다루기



### 실시간 네이버 부동산 최저가 검색

작성자: jeong kyungseok ✉

서울에 한정하여 아파트명으로 검색하면 답변합니다.

청구3차

마포래미안푸르지오

역삼래미안

래미안옥수리버젚

 메시지 실시간 네이버 부동산 최저가 검색 

[ChatGPT - 실시간 네이버 부동산 최저가 검색](#)



gpts.webpilot.ai과(와) 대화 완료

다음은 "마포래미안푸르지오" 아파트 매매 정보입니다. 상위 3개는 🏆, 🥈, 🥉로 표시하였습니다.

아파트명	지역	가격(억)	층수	향	전용면적(m²)	태그	설명	부동산업체	링크
🏆 마포래미안푸르지오	서울특별시 마포구 아현동	13.5	1/19	남서향	80.0	10년 이내, 역세권, 대단지, 1층	소유자직접의뢰 V 초역세권 V 대중교통이용편리 V 세안고매매	(주)진명부동산중개법인	<a href="#">링크</a>
🥈 마포래미안푸르지오	서울특별시 마포구 아현동	13.5	저/19	남서향	80.0	10년 이내, 마당, 역세권, 대단지	판상형구조.정원조망권.아현역역세권.전세안고	거성공인중개사사무소	<a href="#">링크</a>
🥉 마포래미안푸르지오	서울특별시 마포구 아현동	13.8	고/23	서향	80.0	10년 이내, 대단지, 방세개	전세매매 거실주방마루교체 안방베란다2개 판상형	미소(362-0111)공인중개사사무소	<a href="#">링크</a>
마포래미안푸르지오	서울특별시 마포구 아현동	14.5	4/12	남동향	80.0	10년 이내, 역세권, 대단지	연동형 상가 윗동으로 헬스장 상가 이용 좋고 애오개 접근성 좋은등	래미안단지내공인중개사사무소	<a href="#">링크</a>
마포래미안푸르지오	서울특별시 마포구 아현동	14.5	6/22	남동향	80.0	10년 이내, 역세권	아현역 역세권 이대분사대주	래미안단지내공인중개사사무소	<a href="#">링크</a>



메시지 실시간 네이버 부동산 최저가 검색



# 💡 부동산 데이터 어디까지 파봤니?

## 00 AI를 이용한 나만의 부동산 데이터 다루기

🤖 케빈의 AI 부동산 뉴스레터 (25.01.16) ▶ 받은편지함 x 📧



humanist96@gmail.com

오전 8:42 (5시간 전)



나, mj7976, bom8190, arapesh0, mindul, sjjs96, mydori\_, gg96083, holyspark, icandoyu21, kihuns79 ▾

### 오늘의 핫이슈 🌟

새로운 뉴스로 가득한 하루, 궁금하시죠? 여기 가장 중요한 10가지 뉴스를 모아봤어요!

#### 1 새로운 출발, 롯데마트 천호점 오픈 🎉

롯데마트가 강동구 천호역 근처에 새로운 점포를 개장했습니다. 6년 만의 확장이어서 많은 관심이 쏠리고 있습니다.

[기사 보기](#)

#### 2 경제와 교통의 변화를 쫓다 🚗

서울역에서 수서까지 연결되는 교통 시스템이 올해 안에 공개될 예정입니다. 기대감이 고조되고 있어요!

[기사 보기](#)

#### 3 강남 지역의 부동산 가격 변화 📊

강남 부동산 시장이 변화의 시기를 맞고 있습니다. 초고가 아파트마저 영향받는 현실을 살펴봅시다.

[기사 보기](#)

#### 4 신축 아파트, 오르는 분양가 📈

서울의 아파트 분양가가 고공행진 중입니다. 앞으로의 분양 시장에 어떤 영향을 미칠까요?

[기사 보기](#)

### ★ 강동구 임장 블로그 탐색 ★

안녕하세요! 이번 주 뉴스레터에서는 서울의 매력적인 지역, 강동구에서의 임장 후기를 소개해드릴게요. 많은 글 중에서 유사성을 가진 글들을 선별하여 가장 흥미롭고 대표적인 이야기들을 전해드릴게요. 재밌게 읽으시고, 관심 있는 곳은 직접 방문해보세요. 😊

### 🏠 강동구 임장, 올림픽파크포레온 포함 리뷰

투자하는 백조의 기록장

[블로그 바로가기](#)

언젠가 한 번 가봐야지 하고 마음먹었던 강동구에 드디어 다녀왔습니다. 마음속 껌이지 않는 투자 열정! 고덕그라시움을 포함한 다양한 아파트를 구경하며 지역의 매력을 직접 체험해보았습니다!

### 📖 아는만큼 보인다 - 강동구 임장 이야기

꿈을 이루는 율리아

[블로그 바로가기](#)

강동구 임장을 열심히 다녔지만 기록이 없으면 기억하기 어렵죠. 그래서 이번엔 꼼꼼히 기록해보았습니다. 강동구 내의 인기있는 루트를 따라가며 임장 후 느낀 점들을 공유합니다.

### 📍 서울 강동구 임장 - 상일동 및 명일동 후기

유어의 신혼부부 머니로그

[블로그 바로가기](#)

강동구 성내동과 명일동을 중심으로 임장을 진행하고, 그 결과를 기록했습니다. 다양한 상

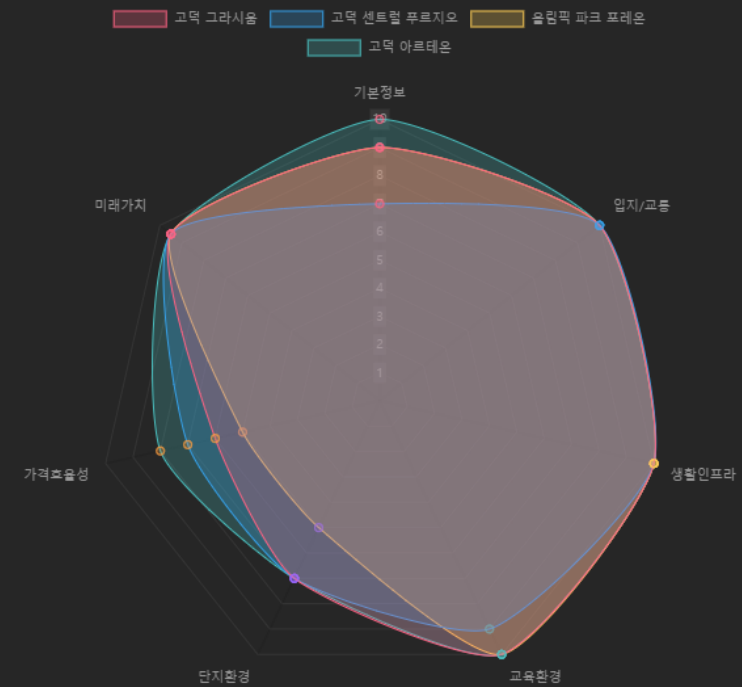
# 부동산 데이터 어디까지 파봤니?

## 00 AI를 이용한 나만의 부동산 데이터 다루기

썸네일	제목	채널명	링크	조회수	구독자 수	날짜	설
	10년 후 올라갈 부동산 '이 지역', 만약 현금 있으면 반드시 사세요 (빠충 김학렬 소장)	김작가 TV	<a href="https://www.youtube.com/watch?v=doZXY8yGNDY">https://www.youtube.com/watch?v=doZXY8yGNDY</a>	282471	2340000	2024. 7. 27. 오후 1:30:14	10학
	2025년 강동구 전세 대란 사태 벌어진다?	빌딩의신	<a href="https://www.youtube.com/watch?v=pAKayt9x46U">https://www.youtube.com/watch?v=pAKayt9x46U</a>	211276	59100	2023. 5. 17. 오후 7:00:27	20및
	강동구 고덕동 대장 아파트 아르데온의 거래 현황	빅톡TV	<a href="https://www.youtube.com/watch?v=IXB3iz1lvqE">https://www.youtube.com/watch?v=IXB3iz1lvqE</a>	161765	27600	2023. 2. 23. 오전 6:00:31	부서
	공급폭탄 강동구 아파트 분위기가 충격적인데...	표영호 tv / 경제적 자유와 행복을 위하여	<a href="https://www.youtube.com/watch?v=R3ZOdGF_MDA">https://www.youtube.com/watch?v=R3ZOdGF_MDA</a>	129276	595000	2024. 8. 17. 오전 9:30:12	1단문
	2년 뒤에 여기 정말 많이 달라질 거예요 #서울유랑지역   빌딩의신	빌딩의신	<a href="https://www.youtube.com/watch?v=qkveiMkQZww">https://www.youtube.com/watch?v=qkveiMkQZww</a>	102637	59100	2023. 5. 23. 오후 6:00:36	2단동
	강동구에 이 아파트 아는 사람!?	우리 동네 클라쓰	<a href="https://www.youtube.com/watch?v=eUUWXRb07QA">https://www.youtube.com/watch?v=eUUWXRb07QA</a>	94052	107000	2022. 11. 30. 오전 8:00:15	부안
	강동구 아파트 다 줘드립니다.	빌딩의신	<a href="https://www.youtube.com/watch?v=kTalrevmNCg">https://www.youtube.com/watch?v=kTalrevmNCg</a>	77635	59100	2023. 2. 8. 오전 6:00:30	강의

평가 항목	고덕 그라스יום	고덕 센트럴 푸르지오	올림픽 파크 포레온	고덕 아르테온
기본 정보	9	7	9	10
입지 및 교통	10	10	10	10
생활 인프라	10	10	10	10
교육 환경	10	9	10	10
단지 환경	7	7	5	7
가격 효율성	6	7	5	8
미래 가치	9.5	9.5	9.5	9.5
총점	61.5	59.5	58.5	64.5
백분율	87.9%	85.0%	83.6%	92.1%

### 3. 단지별 항목 비교 레이더 차트

[illegible]



# 💡 부동산 데이터 어디까지 파봤니?

## 00 AI를 이용한 나만의 부동산 데이터 다루기

<https://www.youtube.com/watch?v=Bf253eZwQZI>

YouTube

거주와 투자를 동시에 할 수 있는 내 집 마련의 실마리 | 신혼부부부터 세가족이... 5억이상 10억미만 세가족이 거주가능한 실거주겸 투자 서울 아파트...



오후 4:15 ✓

[요약 정리]

이 영상은 신혼부부나 3인 가족이 살기 좋은 10억원 미만의 아파트를 고르는 방법을 설명합니다. 특히 실제 살면서도 나중에 투자 가치도 있는 집을 고르는 방법을 알려줍니다.

주요 내용:

1. 예산에 맞는 집 고르기
2. 가족 구성원 수에 맞는 평수 선택
3. 실제 거주하면서도 투자 가치가 있는 부동산 고르기

[인사이트 및 착안사항]

1. 생활 측면

- 집을 고를 때는 교통, 학교, 마트 등 실제 생활하기 편한 곳인지 꼭 확인하기
- 가족 수가 늘어날 것을 고려해 약간 여유 있는 평수 선택하기
- 주차시설, 방음, 채광 등 실생활에 중요한 요소 체크하기

2. 투자 측면

- 새 아파트만 고집하지 말고 잘 관리된 중고 아파트도 고려하기
- 주변 개발 계획이나 교통 개선 계획이 있는지 살펴보기
- 비슷한 가격대의 다른 지역 아파트와 비교해보기

강동구 임장보고서 COT를 통해서 투자에 유용하도록 작성해줘

오전 9:52 ✓

임장보고서를 Chain of Thought (COT) 방식으로 분석하여 작성하겠습니다:

1. 지역 분석

- 위치: 서울 동부권 강동구
- 교통
  - \* 지하철: 5호선, 8호선, 9호선 환승 가능
  - \* 버스: 광역/간선/지선 노선 다양
- 생각 과정: 대중교통 접근성이 우수하여 출퇴근 수요 있음

2. 개발 호재

- 강동구청역 일대 복합개발
- 둔촌동 재건축 단지 완공 예정
- 고덕비즈밸리 조성
- 생각 과정: 향후 배후수요 증가 예상됨

3. 인프라

- 교육: 명문고 다수, 학원가 발달
- 생활: 롯데몰, 이마트, 롯데백화점 등
- 의료: 강동경희대병원, 상일동 의료클러스터
- 생각 과정: 생활 인프라 충분, 주거 만족도 높을 것

4. 시세 동향

- 최근 3년간 매매가 추이
- 전월세 시장 현황
- 인근 신축 분양가 비교
- 생각 과정: 시세 대비 투자 메리트 분석

5. 투자 포인트

- 장점:
  - 서울 동부권 新주거타운으로 성장
  - 교통/교육/생활 인프라 우수
  - 개발 호재로 미래가치 상승 기대

단점:

- 강남권 대비 상대적 고평가 우려
- 일부 지역 주차난/교통혼잡

6. 투자 전략  
결론:

### 서울 동부권 강동구: 투자 가치와 미래 전망

이 노트는 서울 동부권 강동구에 대한 임장보고서를 Chain of Thought(COT) 방식으로 분석하여 투자 가능성 및 시장 전망을 제시하는 것을 목적으로 합니다. 강동구의 위치, 개발 호재, 인프라, 시세 동향 및 투자 포인트에 대해 자세히 살펴볼 것입니다.

#### 지역 분석

- 위치: 서울 동부권 강동구는 대중교통 접근성이 뛰어나며 출퇴근 수요가 높은 지역입니다.
- 교통
  - 지하철: 5호선, 8호선, 9호선 환승 가능
  - 버스: 광역/간선/지선 노선 다양
- 생각 과정: 이 지역의 훌륭한 대중교통은 버스과 지하철을 모두 포함하여 교통 혼잡을 감소시키고 출퇴근 시간을 줄여줄 수 있습니다.

#### 개발 호재

- 강동구청역 일대: 복합개발 진행 중
- 둔촌동 재건축: 완공 예정 단지
- 고덕비즈밸리: 신규 조성 중인 업무 단지
- 생각 과정: 이러한 개발들은 곧 배후 수요의 증가로 이어질 것으로 예상됩니다. 미래 가치 상승의 기반이 될 수 있습니다.

#### 인프라

- 교육: 명문 고등학교와 발달한 학원가
- 생활: 롯데몰, 이마트, 롯데백화점 등의 편의시설
- 의료: 강동경희대병원, 상일동 의료 클러스터
- 생각 과정: 생활 인프라가 풍부해 주거 만족도가 높을 것으로 예상됩니다. 이는 장기적인 수요를 뒷받침합니다.

#### 시세 동향

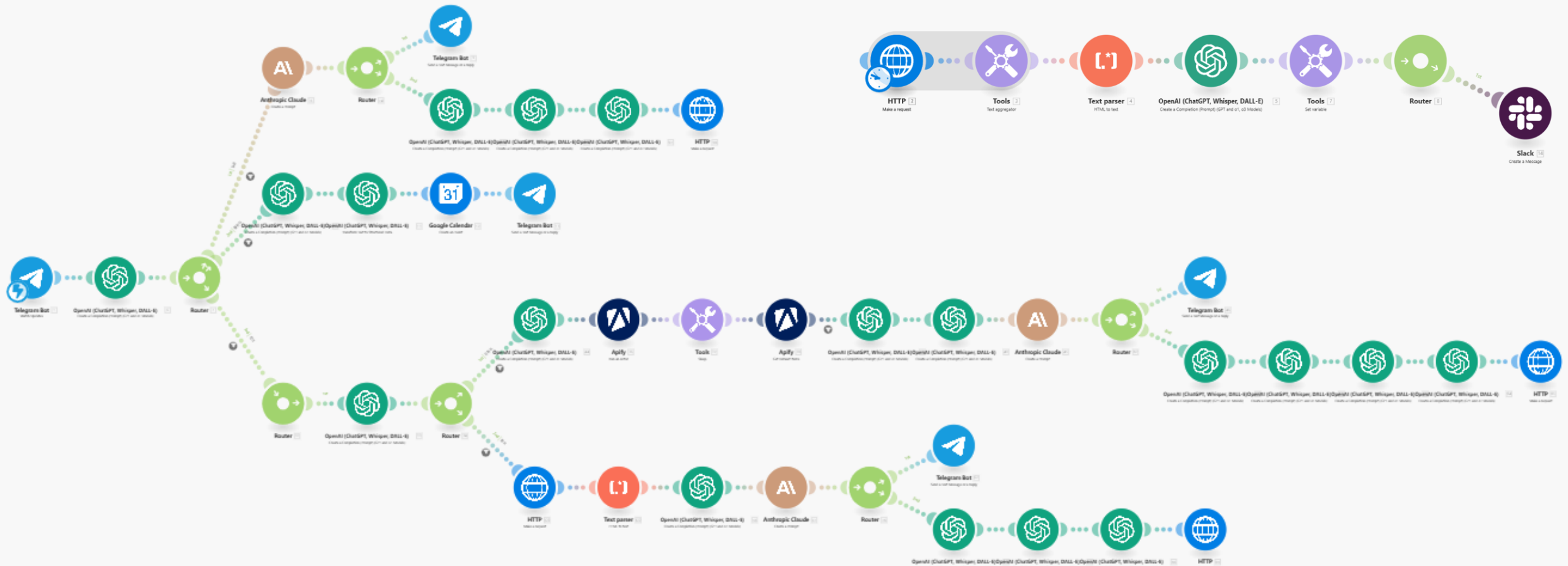
- 최근 3년간 매매가 추이: 지속적인 상승세
- 전월세 시장 현황: 수요에 맞추어 안정적 가격대

# 💡 부동산 데이터 어디까지 파봤니?

## 00 AI를 이용한 나만의 부동산 데이터 다루기

### ← 야후 파이낸싱 뉴스 자동 요약

DIAGRAM HISTORY INCOMPLETE EXECUTIONS





# 💡 부동산 데이터 어디까지 파봤니?

## 00 AI를 이용한 나만의 부동산 데이터 다루기

The image displays a Notion workspace on the left and a Claude chat interface on the right, both in dark mode.

**Notion Workspace:**

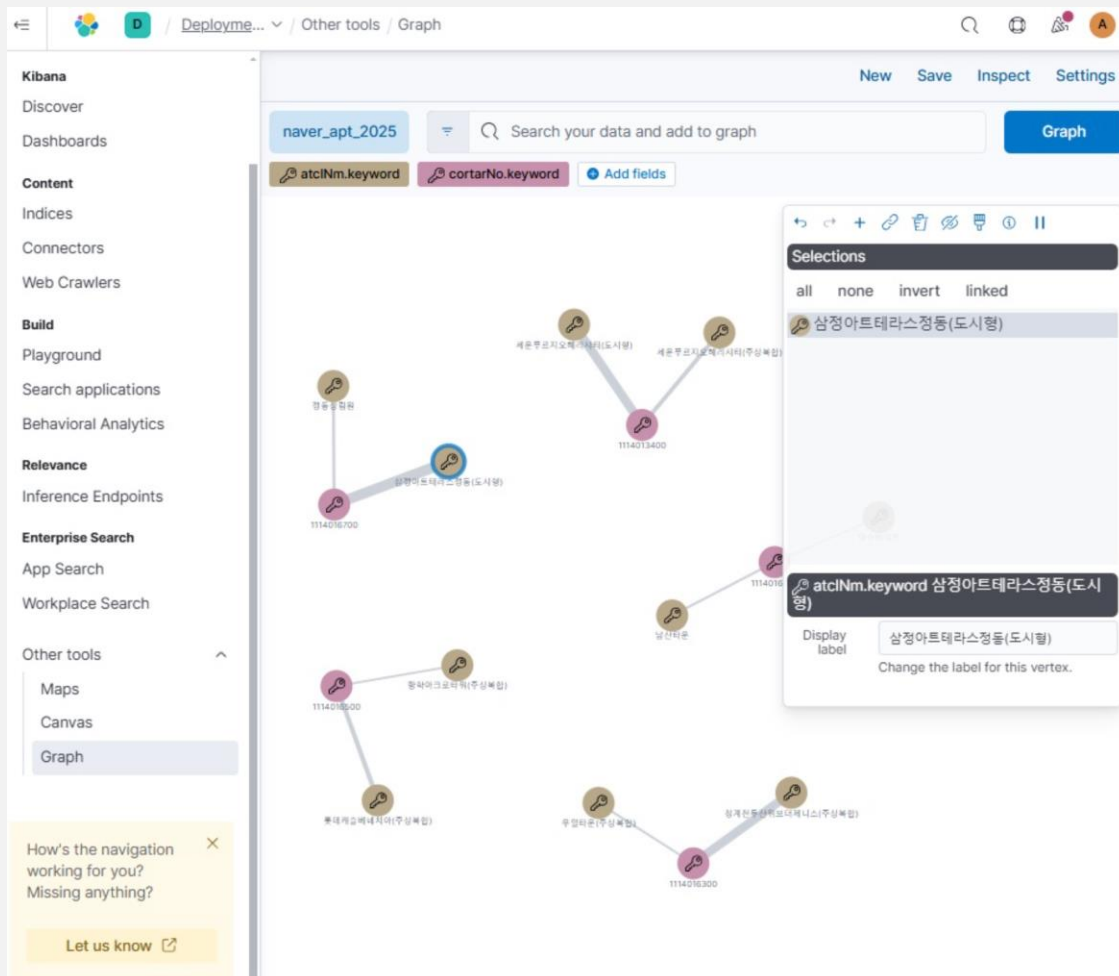
- Left Sidebar:** Contains a file explorer with items like "AI를 이용한 데이터 기반 부동산 투자\_v0.8 PDF", "KakaoTalk\_20241213\_143722502 JPG", and "Kevin 부동산 공부 파일 샘플 자료". It also has sections for "Blog", "News", "Youtube", "Leo", "Attached file", "Guideline", "MOC Note", "smart-chats", "Tasks", and "출석부".
- Main Content Area:** Titled "links:Boo-Ob Process Guideline". It features a "Boo-Ob MOC Home" section with the subtitle "- Real Estate, Obsidian Specialist". Below this is a numbered list of topics: 1. 어느 유형의 부동산 매물과 지역에 (100 부동산 테마, 200 부동산 지역), 2. 어떤 거래 방식을 취하며 (300 부동산 거래), 3. 어떤 문헌과 데이터를 사용하고 (400 부동산 문헌, 500 부동산 데이터), 4. 어떤 연구방법론과 도구를 활용하여 (600 부동산 방법론, 700 도구), 5. 어떤 결과를 도출 할 것인가? (800 성과물, 900 창작물).
- Bottom Section:** Includes "Boo-Ob Membber list" and "Boo-Ob MOC list" with a list of items like "100 부동산 테마", "110 주거용 부동산", "111 아파트", etc.

**Claude Chat Interface:**

- Chat Window:** Shows a conversation with "claude-3.5-sonnet (Recommended)". The user input is a URL: "@https://new.land.naver.com/complexes/623?ms=37.5061383,127.0952994,16&a=APT:ABYGJGC:PRE&e=RETAIL".
- Response:** The AI responds with "디에이치 아너힐즈(구 개포주공1단지)의 네이버 부동산 정보를 정리하겠습니다." and provides a detailed summary of the property, including location, size, price, and features.
- Summary Content:**
  - # 디에이치 아너힐즈 분석**
  - ## 1. 단지 개요**
    - 위치: 서울 강남구 개포동 660
    - 규모: 1,320세대, 32개동
    - 준공: 2023년 3월
    - 용적률: 199%
    - 건폐율: 17%
  - ## 2. 가격 동향**
    - 매매가: 35억 ~ 65억
    - 평당가: 약 8,000만원 수준
    - 전세가율: 45~50% 수준
    - 8월 이후 실거래가 상승세
  - ## 3. 투자 포인트**
    - 강남 재건축 완료 단지
    - 탄탄한 학군 (개포초/중)
    - 지하철 3호선 도보권
    - 단지 내 상가 보유
    - 대규모 공원 인접
  - ## 4. 위험 요소**
    - 고가 주택 규제
    - 대출 제한
    - 보유세 부담

# 기타 응용 사례

## 04 Elasticsearch를 이용한 RAG 챗봇



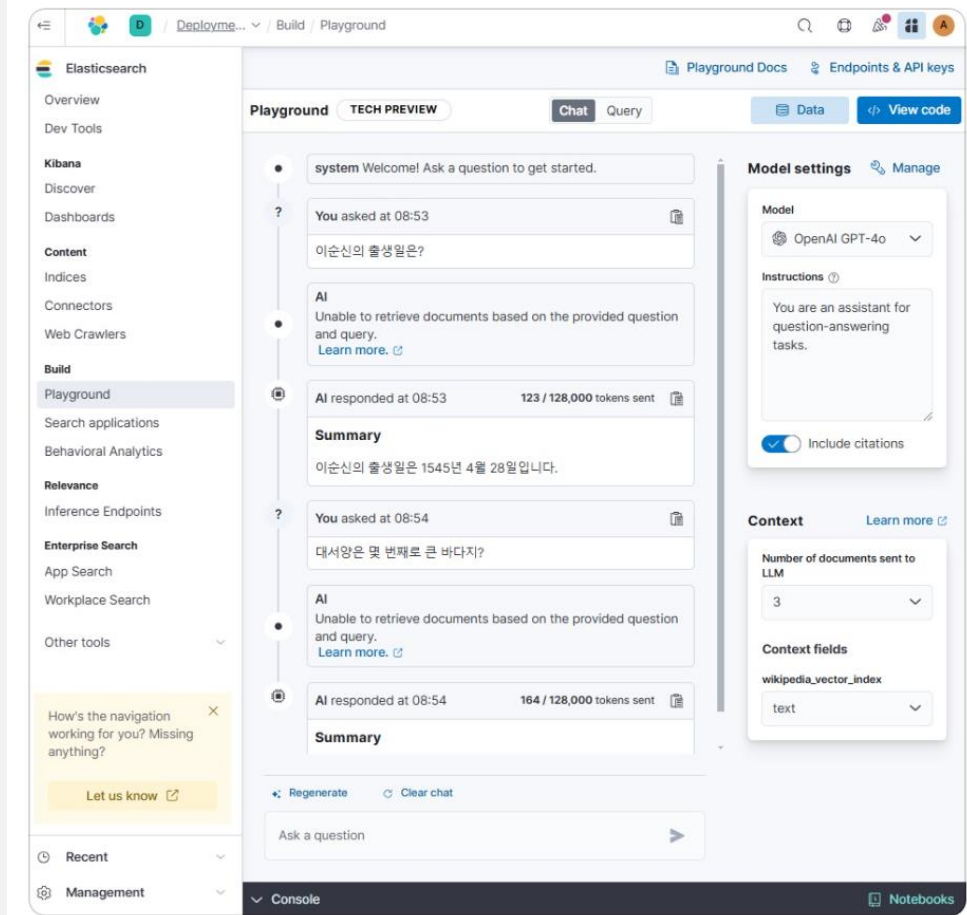
elastic cloud에 영문 위키피디아로 답변 가능한 Q&A 챗봇(RAG) 구축  
10분만에 완성~

영문위키피디아 : 2만 5천건

[weaviate.io/devel...](https://weaviate.io/devel...)

나만의 방대한 데이터를 쉽게 관리하며  
LLM을 이용하여 내가 원하는 정보를 쉽게 찾을 수 있다면..

관심있는 분은 하트 ^^



# 기타 응용 사례

## 04 Elasticsearch를 이용한 RAG 챗봇

### 영문 위키피디아 이용한

### 한글로 답변하는 AI

부제 : Semantic search and Retrieval augmented generation using Elasticsearch and OpenAI

명확한 정답이 있는 답변을 잘합니다. 좋은 질문 예 :

- How big is the Atlantic ocean?
- 대한민국의 수도는?
- 이순신의 출생년도는?
- 도요타에서 가장 많이 팔리는 차는?

#### 장점

- 전문 영어 문서(예: 해외 기업투자분석리포트 등)를 대상으로 한글로 검색하고 답변 받기 용이합니다.
- 전통적인 검색방식(키워드 매칭)을 탈피하여 의미가 유사한 문서를 찾고 이를 통한 답변 가능
- 일반 검색에 비하여 2~3 단어의 이상의 문장형 검색 질의에 답변 잘함
- 다국어(multi-lingual AI) 검색 가능(단, 학습문서가 영어라서 영어로 질의해야 답변 잘해요.)
- LLM(Large Language Model) AI의 답변이 아닌 내가 원하는 문서를 통해서 답변 가능하여 할루시네이션(Hallucination)을 최소화
- Private한 사내 문서나 특정 도메인에 특화된 답변이 가능합니다.(사내 문서가 외부로 노출되지 않음)

#### 단점

- 데이터가 충분하지 않아 다양한 질문에 정확한 답을 못할 수 있음.
- 유사도 검색의 특성상 적절하지 않은 검색결과가 있다면 잘못된 답변을 할 수 있음

#### 데이터 출처

- [https://cdn.openai.com/API/examples/data/vector\\_database\\_wikipedia\\_articles\\_embedded.zip](https://cdn.openai.com/API/examples/data/vector_database_wikipedia_articles_embedded.zip)
- 데이터 설명 : <https://weaviate.io/developers/weaviate/tutorials/wikipedia>
- 데이터 건수 : 25,000건 (데이터의 양을 늘리면, 다양한 질문에 대한 답변 가능)

#### Prompt

이순신의 출생년도는?

Submit

이순신의 출생년도는 1536년 입니다.

### 검색해본 위키 문서 List

ID: 22421 Title: Yi I Url: <https://simple.wikipedia.org/wiki/Yi%20I> Score: 0.9028312

ID: 40687 Title: 1415 Url: <https://simple.wikipedia.org/wiki/1415> Score: 0.8957019

ID: 39133 Title: 1551 Url: <https://simple.wikipedia.org/wiki/1551> Score: 0.8900757

ID: 15081 Title: 1232 Url: <https://simple.wikipedia.org/wiki/1232> Score: 0.8894432

ID: 20088 Title: 509 Url: <https://simple.wikipedia.org/wiki/509> Score: 0.88554895

ID: 4678 Title: Kim Jong-il Url: <https://simple.wikipedia.org/wiki/Kim%20Jong-il> Score: 0.8849724

ID: 46365 Title: Yun Poson Url: <https://simple.wikipedia.org/wiki/Yun%20Poson> Score: 0.88478947

ID: 46358 Title: Kim Young-sam Url: <https://simple.wikipedia.org/wiki/Kim%20Young-sam> Score: 0.8843397

<https://wiki-ai-search.streamlit.app/>

<https://colab.research.google.com/drive/12bxT-pnxKkfPEut4SDLJvCvfh0MEomW7?usp=sharing>

# 기타 응용 사례

## 04 솔루션 Q&A 챗봇

### Private Data QA Chat

안녕하세요! 주어진 문서에 대해 궁금하신 것이 있으면 언제든지 물어봐주세요!

기타 특수채, 광주지방채, ABS AAA  
위 세개 신용그룹에 대한 커브를 구하고 싶습니다. 3741에서는 없는 것 같네요.

안녕하세요. CHECKExpert+ 입니다.  
3741화면에서는 시가평가사에서 산출하는 그룹에 대해서만 커브를 제공하고 있습니다.  
말씀하신 분류체계는 시가평가사가 아닌 CHECK자체적으로 신용등급 등으로 산출한 분류체계  
여서 시가평가사에서 산출하는 그룹과 달라서 커브가 존재하지 않습니다.  
감사합니다.

안녕하세요 혹시 CD, CP 거래내역은 어디서 확인할 수 있을까요?

CD, CP 거래내역은 CHECK의 메뉴에서, 채권 > 자금/단기시장 > CD/CP 경로로 들어가신 후,  
3072화면에서 한꺼번에 조회가 가능합니다.

3-10년 장단기금리스프레드 추이를 보고싶은데 어디서 보면 될까요?

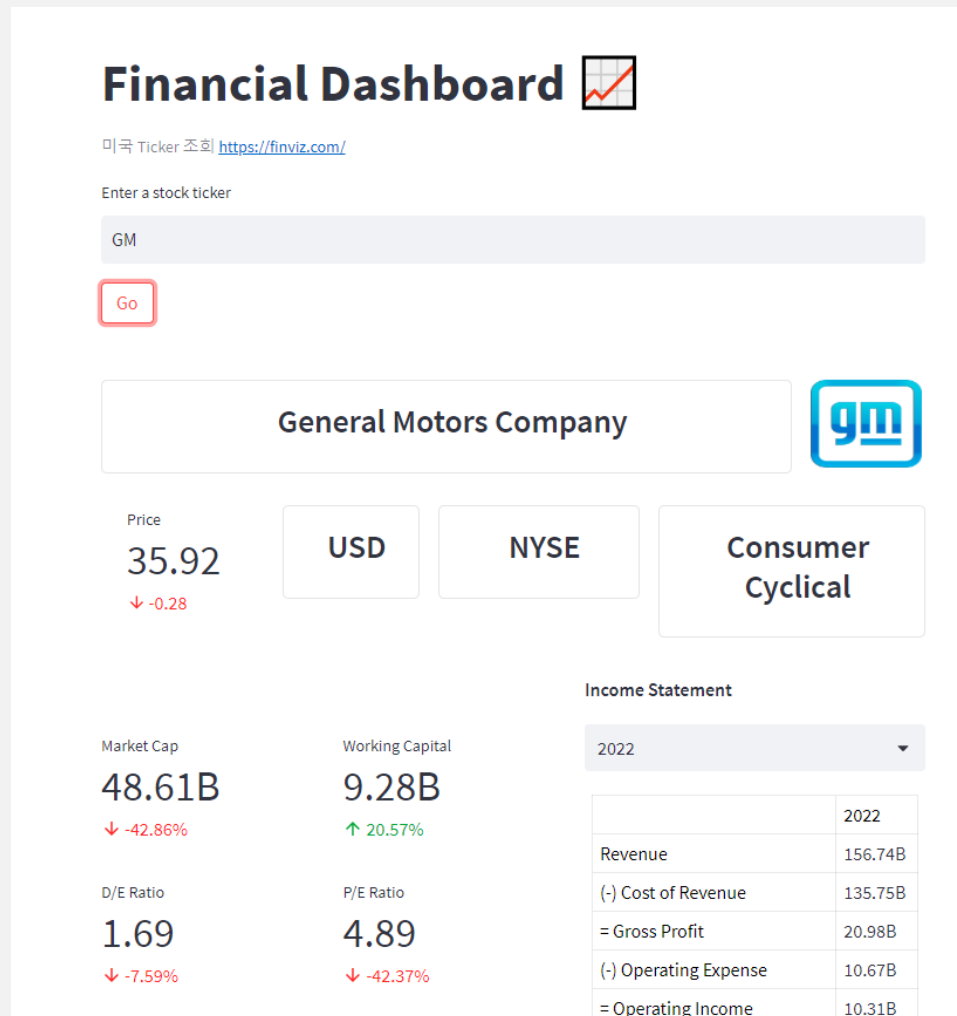
3-10년 장단기금리스프레드 추이를 조회하고 싶으시다면, 3530화면에서 가능합니다. 국고채  
3-10년 스프레드를 보고 싶으시다면, 차례로 시가그룹-민평(3사)또는 민평(4사)-국고채-3년과  
10년을 선택하시면 하단에 차트와 데이터값을 조회할 수 있습니다.

오늘 국고 20-3 조회하는데 채권종목명이 빨간 글씨로 되어있어 어떤 내용일때 이렇게 표시  
되는건지 궁금합니다.  
아마 이런 내용 관련 도움말이나 어딘가에 나와있을 듯 한데, 찾아보려 하는데 잘 안 찾아지  
네요. 화면 이용하다보면 팝업 같은걸로도 설명 안 나오는 이런 것들은 확인하는 방법이 어  
떻게 있는지 알려주시면 감사하겠습니다!

<https://kevin-rag-langchain.streamlit.app/>

# 기타 응용 사례

## 04 미국 주식 정보 데이터 시각화



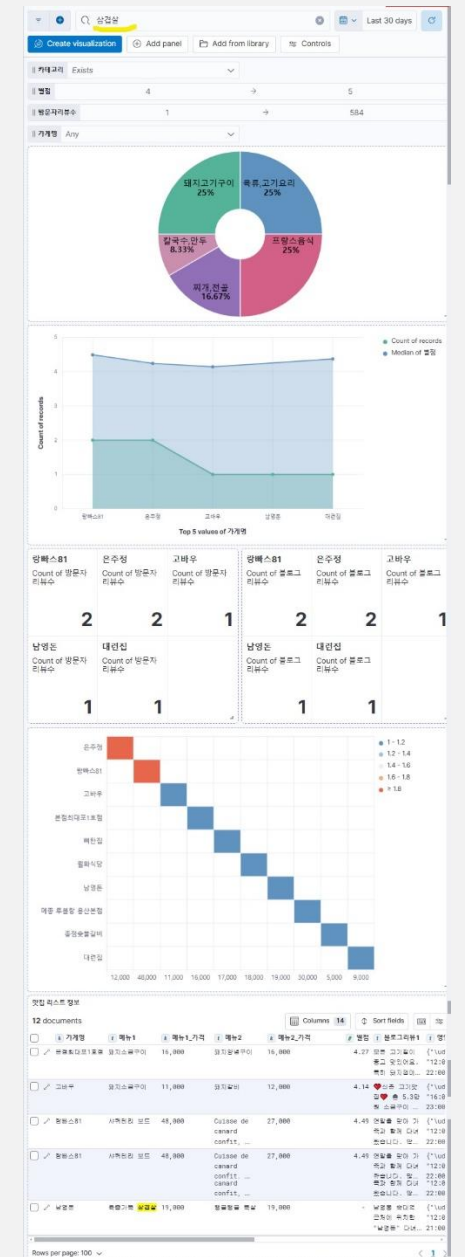
<https://kevin-fin-dash.streamlit.app/>





## 04 맛집 정보 챗봇과 데이터 시각화

K	<div> <div>You asked at 10:49</div> <div>아마키치</div> </div>						
	<div> <div>AI Searching for ""아마키치"에 대한 정보를 제공해 주시겠습니까?"</div> <div>AI Grounding answer based on 3 document sources</div> </div>						
K	<div> <div>AI responded at 10:49</div> <div>1,301 / 128,000 tokens sent</div> </div>						
	<div> <div>Summary</div> <p>아마키치는 점심에는 덮밥집으로, 저녁에는 이자카야로 운영되는 일식당입니다. 식당 정책이 까다로운 편이고 웨이팅이 자주 있지만, 직원들이 친절하고 음식도 맛있다고 합니다. 메뉴로는 카니미소소바(22,000₩)와 부타가쿠니(18,500₩) 등이 있습니다 [1].</p> <div>Citations</div> <div>ID</div> <div>8zTpG5UBKKkq140RLWog</div> <div>Snippet</div> <p>Kitsch한 아마키치 ♡ 카니미소소바 (22,000₩) ♡ 부타가쿠니 (18,500₩) 점심에는 덮밥집, 저녁에는 이자카야로 운영되는 일식당. 식당 정책이 까다로운 편이고 웨이팅이 자주 있으나, 직원 분들은 친절하시고 음식도 맛있다. 음식이 맛있어요+1 개의 리뷰가 더 있습니다 펼쳐보기 이미지 수 11</p> </div> <div> <div>You asked at 10:50</div> <div>삼겹살</div> </div> <tr> <td rowspan="2">K</td><td> <div> <div>AI Searching for "삼겹살에 대한 정보를 알려줄 수 있나요?"</div> <div>AI Grounding answer based on 3 document sources</div> </div> </td><td></td></tr> <tr> <td> <div> <div>AI responded at 10:50</div> <div>1,885 / 128,000 tokens sent</div> </div> </td><td></td></tr> <tr> <td>K</td><td> <div> <div>Summary</div> <p>울지로 방산시장 내에 있는 김치찌개 맛집에서는 삼겹살과 함께 고기가 듬뿍 들어간 김치찌개를 제공합니다. 이곳은 삼겹살 1인분에 김치찌개까지 포함되어 있어 가격이 상당히 저렴한 편입니다. 또한, 신선한 고기와 다양한 반찬이 제공되며, 찜에 싸먹는 재미도 느낄 수 있습니다. 이 식당은 찜싸먹는 김치찌개로 유명하며, 가성비가 뛰어난 곳으로 평가받고 있습니다 [2][3].</p> <div>Citations</div> <div>ID</div> <div>PzTpG5UBKKkq140RLWwg</div> <div>Snippet</div> <p>울지로 방산시장 내 김치찌개 맛집. 약 3개월만에 방문한듯하다. 1인분에 8,000원인가 9,000원일때 부터 다녔는데 어느새 18,000원이 되었다. 물론 삼겹살 1인에 고기 듬뿍 김치찌개까지 나오니 상당히 저렴한가격이다. 장사 잘 되는집의 고기는 신선해서 맛이 없을수가 없다. 반찬도 여러종류가 넉넉히 나오니 찜에 넣어 먹으면 그맛이 더 살아난다. 고기를 다 먹고나면 찜개가 나오는데 안에 들어있는 고기를 남은 찜에 싸먹으면 별미이다. 원래부터 찜싸먹는 김치찌개로 유명한 곳이라 다들 잘 아실것이다. 직접 키우는 찜채소는 다른 고기 집에 비해 종류가 다양해 싸먹는 재미를 느낄 수 있다. 가게가 넓어 조금은 시끌시끌하지만 그게 시장에있는 가게의 맛이 아닐까한다. 삼겹살에 맛있는 김치찌개까지 가성비 으뜸인 곳이다. 음식이 맛있어요+4 개의 리뷰가 더 있습니다 펼쳐보기 이미지 수 20</p> <div>aDTpG5UBKKkq140RMWos</div> <div>Snippet</div> <p>울지로 방산시장 내 김치찌개 맛집. 약 3개월만에 방문한듯하다. 1인분에 8,000원인가 9,000원일때 부터 다녔는데 어느새 18,000원이 되었다. 물론 삼겹살 1인에 고기 듬뿍 김치찌개까지 나오니 상당히 저렴한가격이다. 장사 잘 되는집의 고기는 신선해서 맛이 없을수가 없다. 반찬도 여러종류가 넉넉히 나오니 찜에 넣어 먹으면 그맛이 더 살아난다. 고기를 다 먹고나면 찜개가 나오는데 안에 들어있는 고기를 남은 찜에 싸먹으면 별미이다. 원래부터 찜싸먹는 김치찌개로 유명한 곳이라 다들 잘 아실것이다. 직접 키우는 찜채소는 다른 고기 집에 비해 종류가 다양해 싸먹는 재미를 느낄 수 있다. 가게가 넓어 조금은 시끌시끌하지만 그게 시장에있는 가게의 맛이 아닐까한다. 삼겹살에 맛있는 김치찌개까지 가성비 으뜸인 곳이다. 음식이 맛있어요+4 개의 리뷰가 더 있습니다 펼쳐보기 이미지 수 20</p> </div> </td></tr>	K	<div> <div>AI Searching for "삼겹살에 대한 정보를 알려줄 수 있나요?"</div> <div>AI Grounding answer based on 3 document sources</div> </div>		<div> <div>AI responded at 10:50</div> <div>1,885 / 128,000 tokens sent</div> </div>		K
K	<div> <div>AI Searching for "삼겹살에 대한 정보를 알려줄 수 있나요?"</div> <div>AI Grounding answer based on 3 document sources</div> </div>						
	<div> <div>AI responded at 10:50</div> <div>1,885 / 128,000 tokens sent</div> </div>						
K	<div> <div>Summary</div> <p>울지로 방산시장 내에 있는 김치찌개 맛집에서는 삼겹살과 함께 고기가 듬뿍 들어간 김치찌개를 제공합니다. 이곳은 삼겹살 1인분에 김치찌개까지 포함되어 있어 가격이 상당히 저렴한 편입니다. 또한, 신선한 고기와 다양한 반찬이 제공되며, 찜에 싸먹는 재미도 느낄 수 있습니다. 이 식당은 찜싸먹는 김치찌개로 유명하며, 가성비가 뛰어난 곳으로 평가받고 있습니다 [2][3].</p> <div>Citations</div> <div>ID</div> <div>PzTpG5UBKKkq140RLWwg</div> <div>Snippet</div> <p>울지로 방산시장 내 김치찌개 맛집. 약 3개월만에 방문한듯하다. 1인분에 8,000원인가 9,000원일때 부터 다녔는데 어느새 18,000원이 되었다. 물론 삼겹살 1인에 고기 듬뿍 김치찌개까지 나오니 상당히 저렴한가격이다. 장사 잘 되는집의 고기는 신선해서 맛이 없을수가 없다. 반찬도 여러종류가 넉넉히 나오니 찜에 넣어 먹으면 그맛이 더 살아난다. 고기를 다 먹고나면 찜개가 나오는데 안에 들어있는 고기를 남은 찜에 싸먹으면 별미이다. 원래부터 찜싸먹는 김치찌개로 유명한 곳이라 다들 잘 아실것이다. 직접 키우는 찜채소는 다른 고기 집에 비해 종류가 다양해 싸먹는 재미를 느낄 수 있다. 가게가 넓어 조금은 시끌시끌하지만 그게 시장에있는 가게의 맛이 아닐까한다. 삼겹살에 맛있는 김치찌개까지 가성비 으뜸인 곳이다. 음식이 맛있어요+4 개의 리뷰가 더 있습니다 펼쳐보기 이미지 수 20</p> <div>aDTpG5UBKKkq140RMWos</div> <div>Snippet</div> <p>울지로 방산시장 내 김치찌개 맛집. 약 3개월만에 방문한듯하다. 1인분에 8,000원인가 9,000원일때 부터 다녔는데 어느새 18,000원이 되었다. 물론 삼겹살 1인에 고기 듬뿍 김치찌개까지 나오니 상당히 저렴한가격이다. 장사 잘 되는집의 고기는 신선해서 맛이 없을수가 없다. 반찬도 여러종류가 넉넉히 나오니 찜에 넣어 먹으면 그맛이 더 살아난다. 고기를 다 먹고나면 찜개가 나오는데 안에 들어있는 고기를 남은 찜에 싸먹으면 별미이다. 원래부터 찜싸먹는 김치찌개로 유명한 곳이라 다들 잘 아실것이다. 직접 키우는 찜채소는 다른 고기 집에 비해 종류가 다양해 싸먹는 재미를 느낄 수 있다. 가게가 넓어 조금은 시끌시끌하지만 그게 시장에있는 가게의 맛이 아닐까한다. 삼겹살에 맛있는 김치찌개까지 가성비 으뜸인 곳이다. 음식이 맛있어요+4 개의 리뷰가 더 있습니다 펼쳐보기 이미지 수 20</p> </div>						







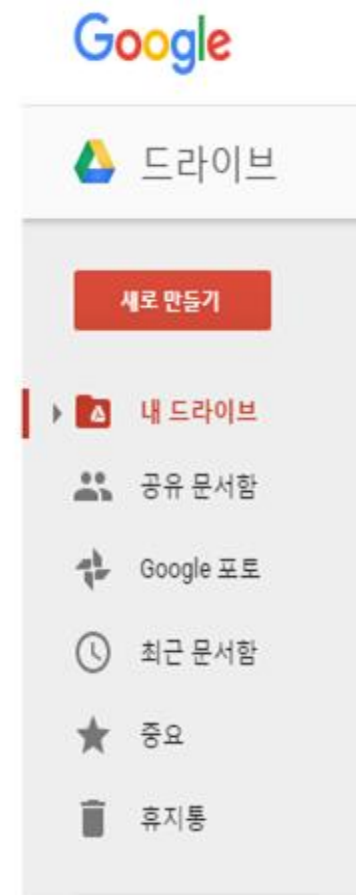
# 수집코드 실행 환경

# 실습환경 구축 및 준비

## 1. Google Drive

Google 드라이브란,  
Google사이트에서 계정을 만들면 이용할 수 있는 온라인 대용량  
저장소입니다.

<https://drive.google.com>



# 실습환경 구축 및 준비

## 1. Google Drive

### 장점

- 1) 문서 작성에서 엄청난 힘을 발휘해요!
- 2) 모바일 환경에서의 작업 진행의 반영이 유리해요!
- 3) G mail 등의 기타 구글 서비스와 연동이 편해요!
- 4) 연결할 수 있는 앱이 가장 많아요!



드라이브



# 실습환경 구축 및 준비

## 2. Google Colaboratory

**Google Colaboratory**란,

클라우드 기반의 무료 Jupyter 노트북 개발 환경이다.

코랩 + 구글드라이브 + 도커 + 리눅스 + 구글클라우드의 기술스택

<https://colab.research.google.com/>



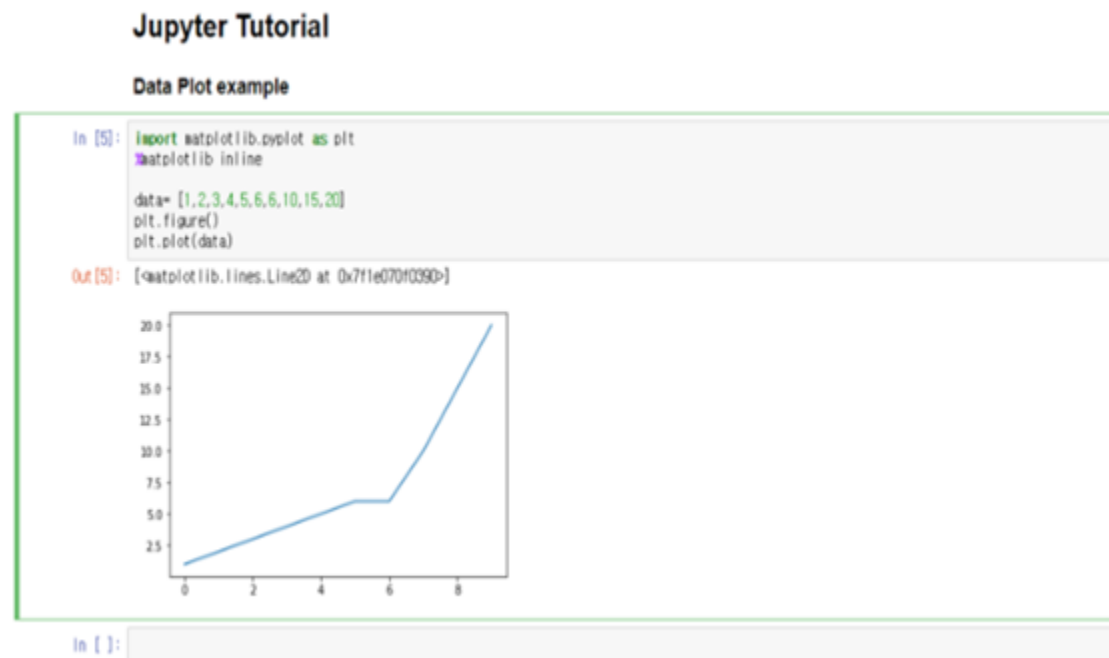
# 💡 실습환경 구축 및 준비

## 2. Google Colaboratory

### Jupyter Notebook 이란,

라이브 코드, 등식, 시각화와 설명을 위한 텍스트 등을 포함한 문서를 만들고 공유가 가능한 오픈소스 웹 어플리케이션

<https://jupyter.org/>

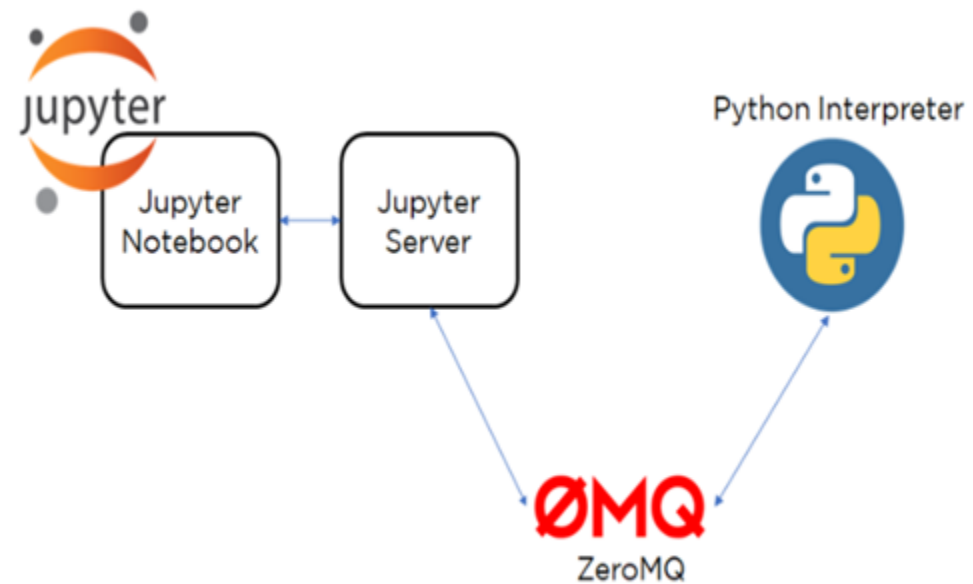


# 💡 실습환경 구축 및 준비

## 2. Google Colaboratory

### Jupyter Notebook 구조

Jupyter는 본래 iPython이었지만 하나의 서버에서 다수의 커널을 지원하게 되면서 이름이 변경



Jupyter 구조



# 실습환경 구축 및 준비

## 2. Google Colaboratory

### 장점

1. 공짜
2. 쉽다. 환경설정 및 구동 준비가 5분.
3. 클라우드 기반
4. 학습 및 공유에 최고!



# 실습환경 구축 및 준비

## 2. Google Colaboratory

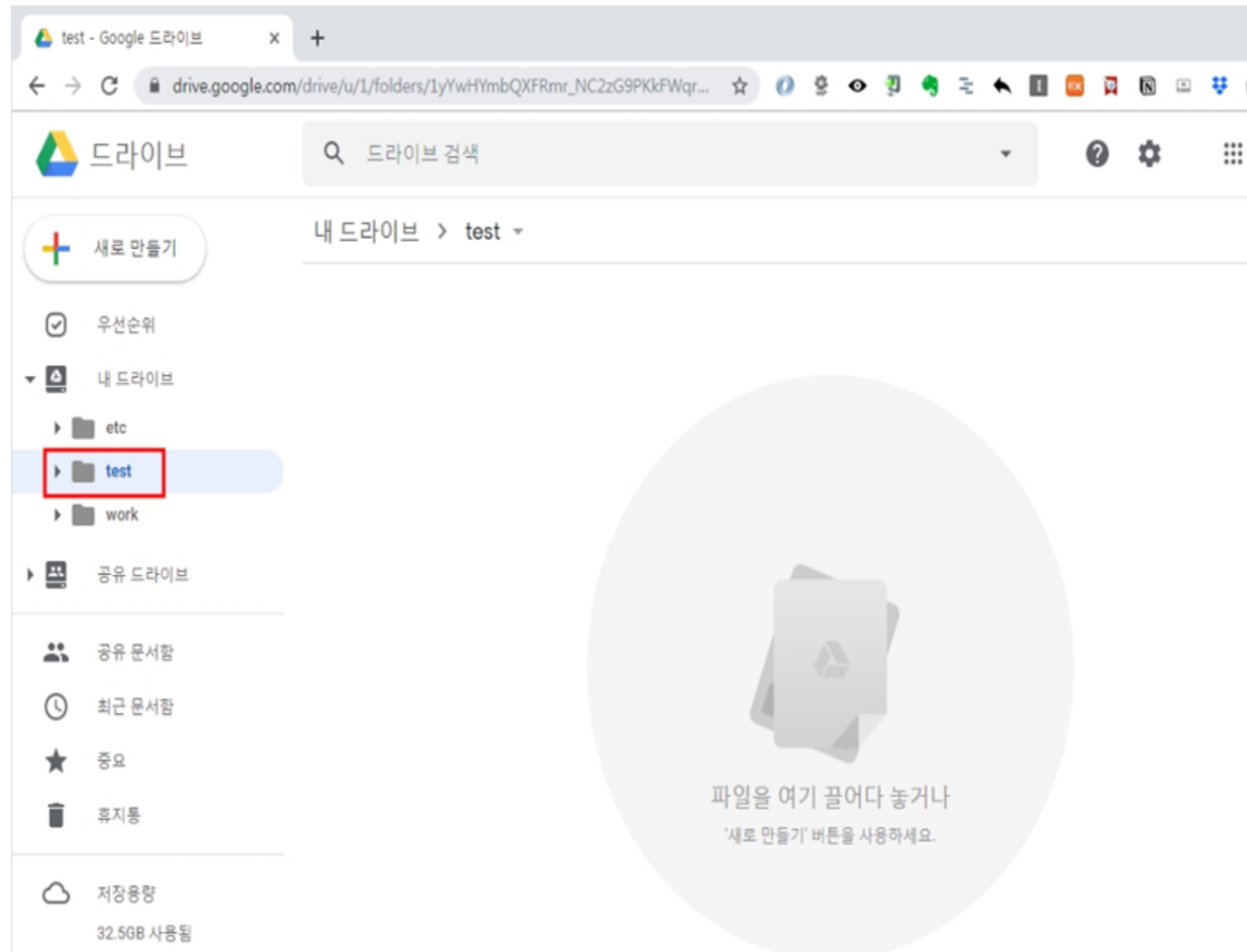
### 주의할 점

1. 최대 세션 유지시간은 12시간
2. 세션이 끊기면? 작업중이던 데이터 날아감
3. 망분리, 보안 이슈가 있는 경우 데이터 업로드 불가



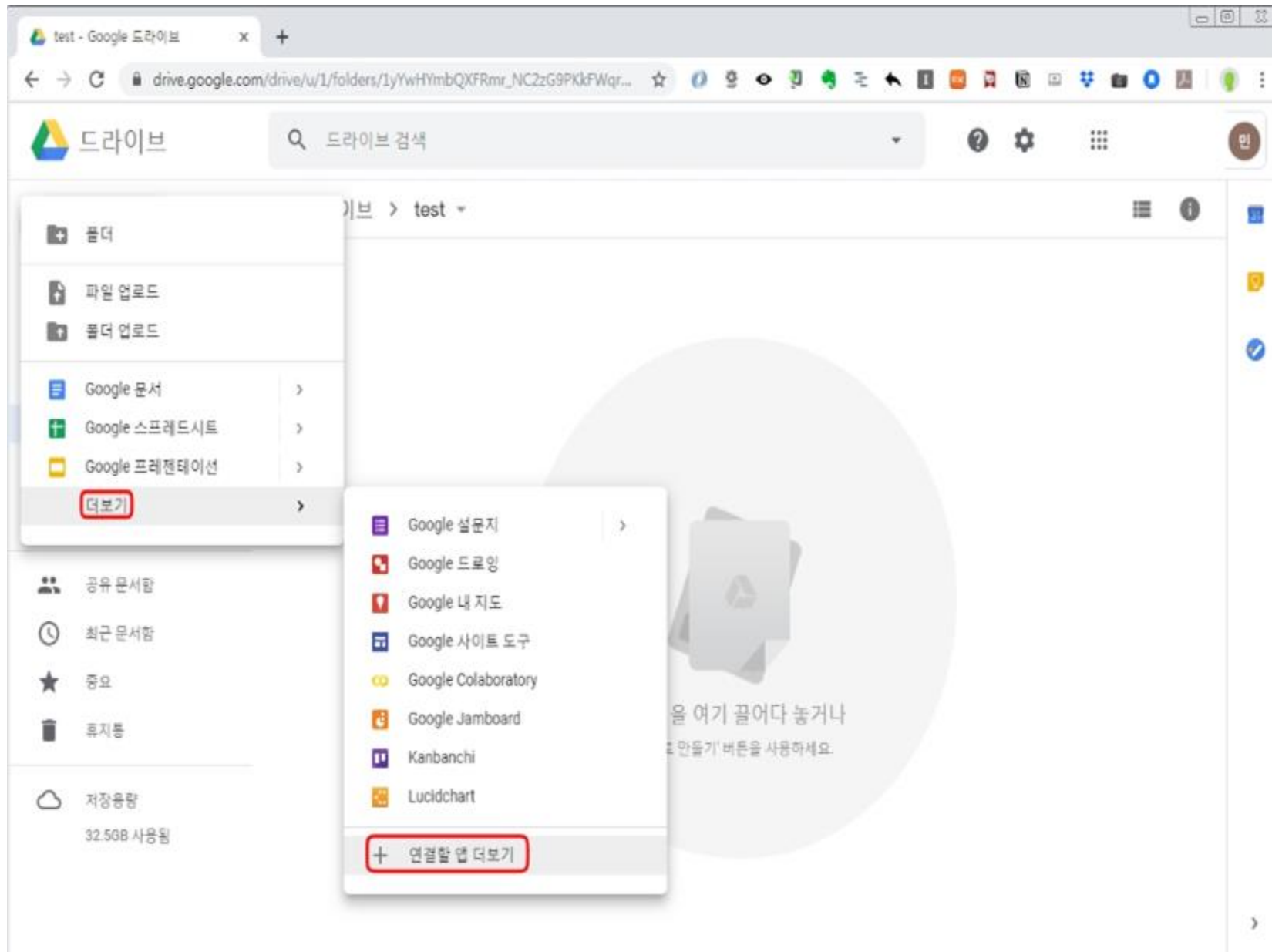
## 실습환경 구축 및 준비

1. <https://drive.google.com>에 접속 후, 우클릭하여 다음과 같이 test 폴더를 만든다



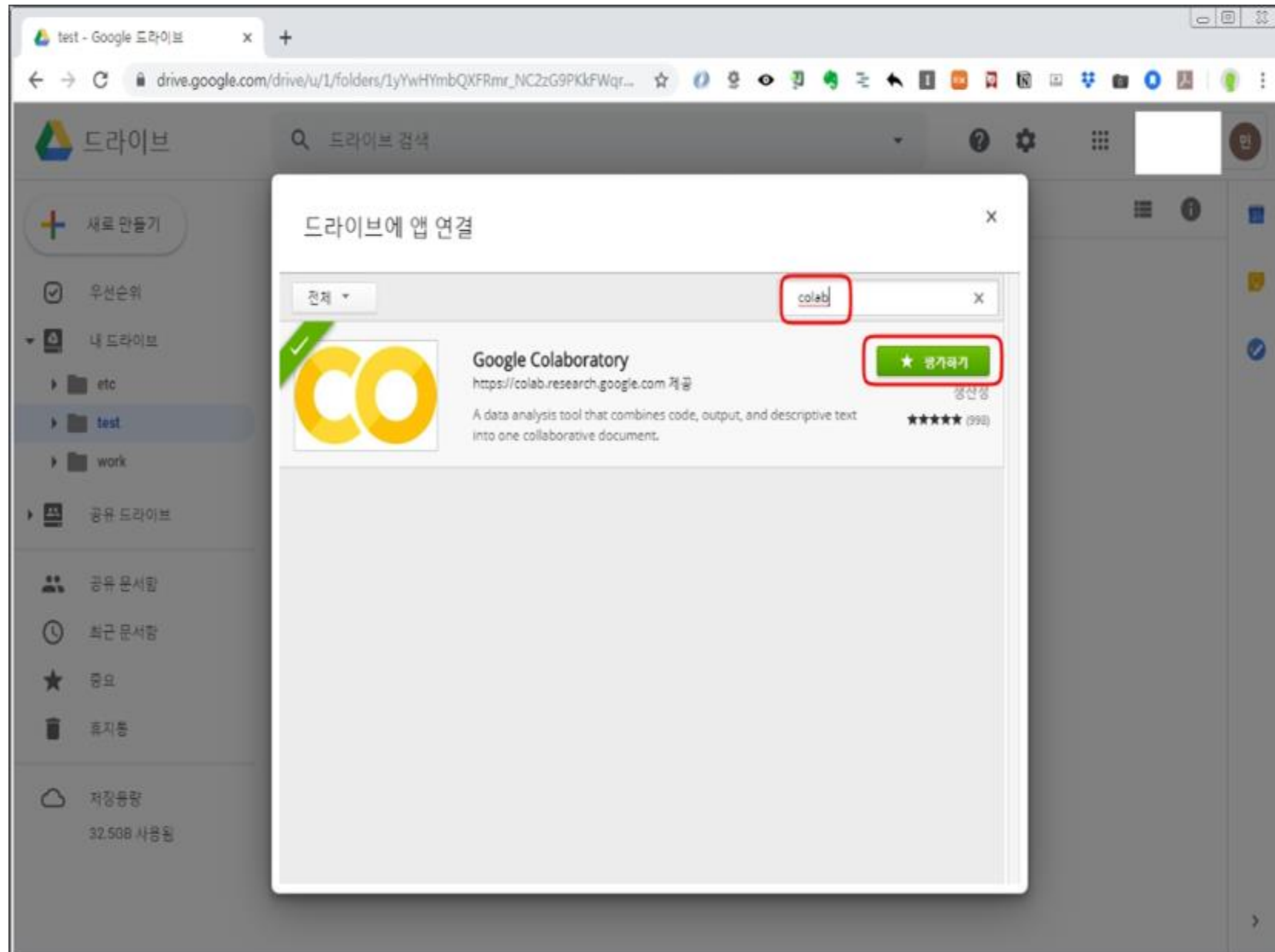
## 💡 실습환경 구축 및 준비

2. 좌측 상단의 [+새로만들기] 버튼 > 더보기 > 연결할 앱 더보기를 선택한다.



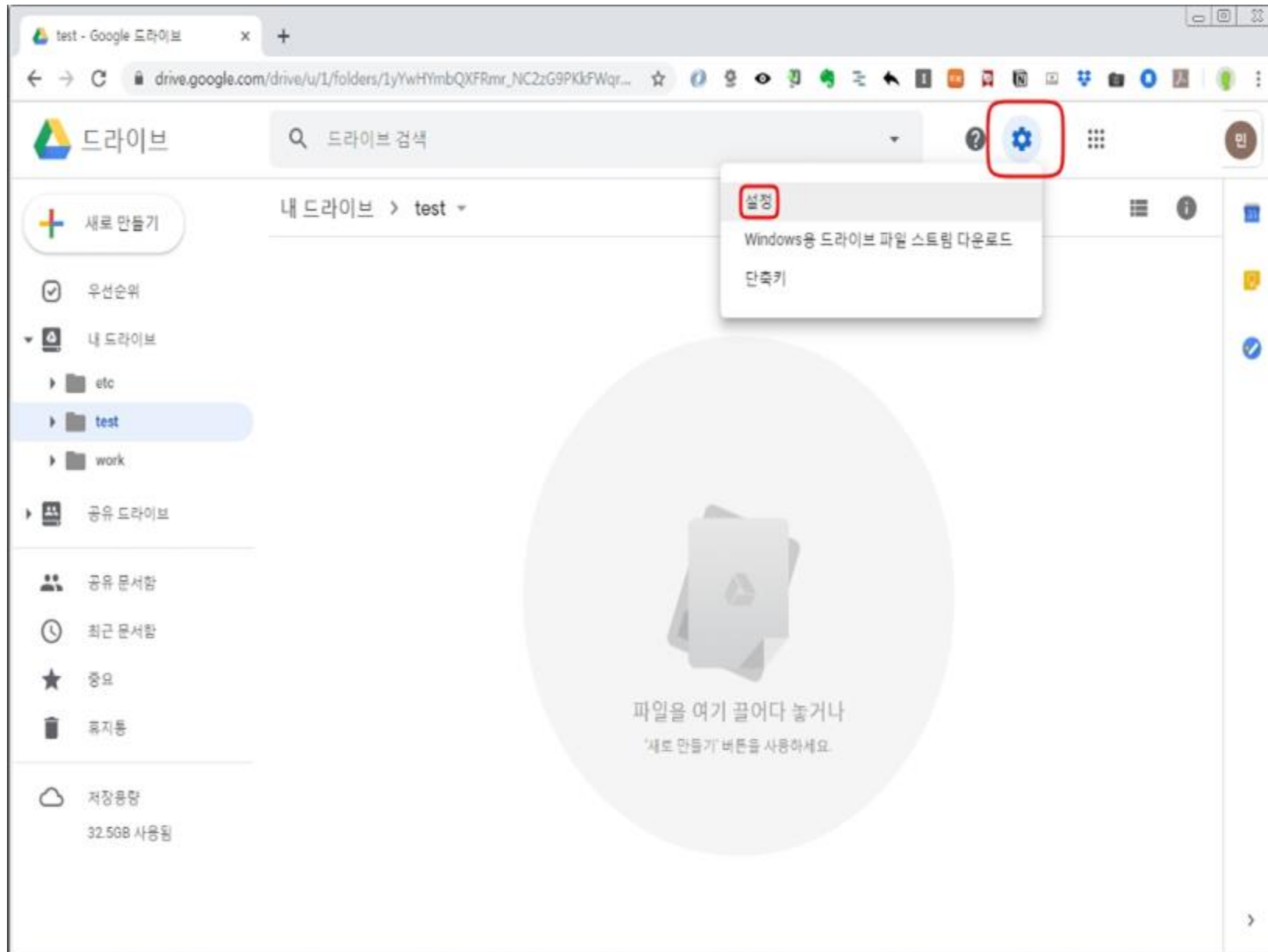
## 💡 실습환경 구축 및 준비

3. 새로 뜬 팝업의 우측 상단에 colab이라는 검색어를 입력하여 Google Colaboratory 앱 설치



## 💡 실습환경 구축 및 준비

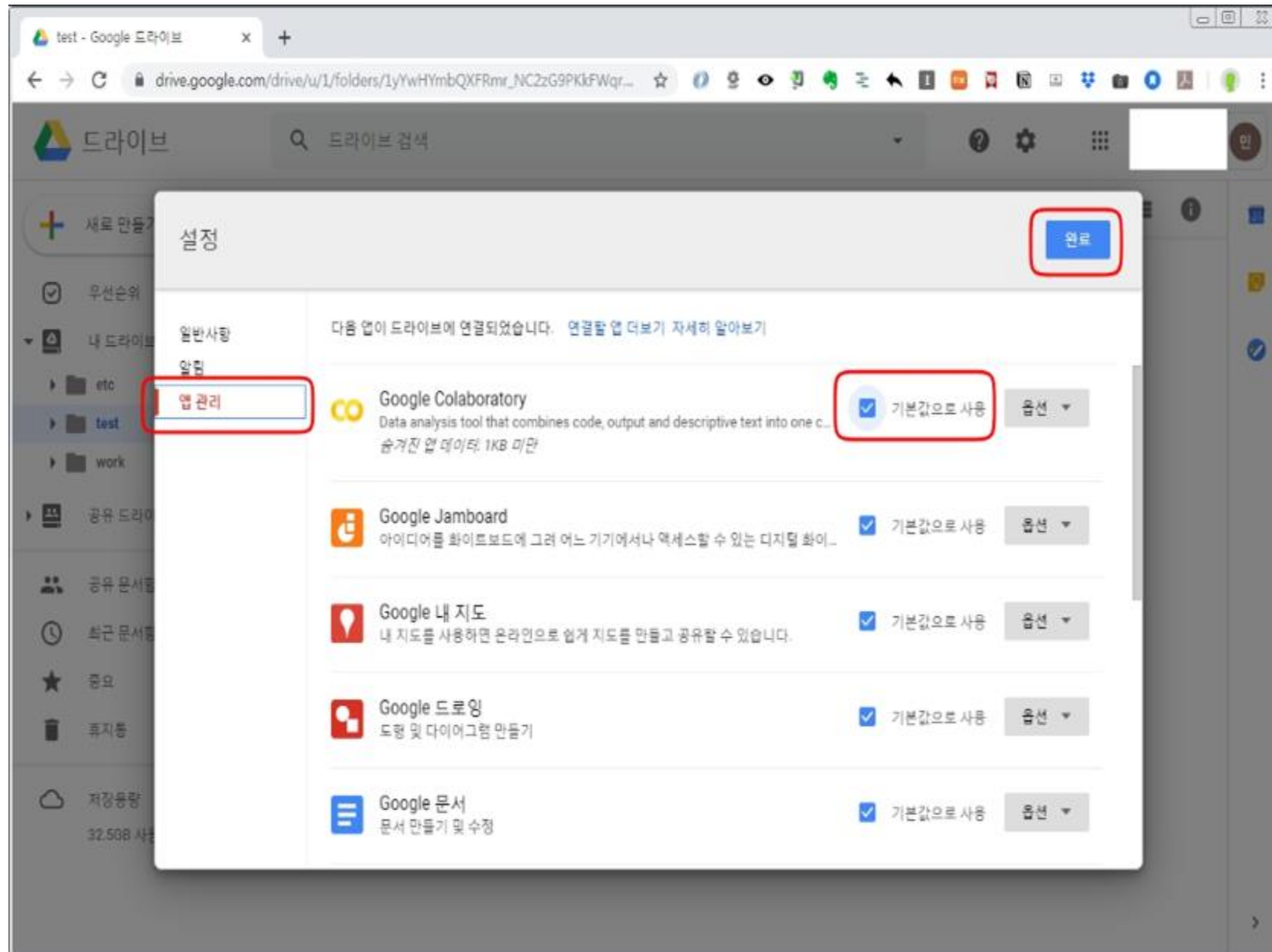
4. 드라이브 메인화면으로 이동 > 톱니바퀴 모양 버튼 클릭 > 설정을 클릭한다.





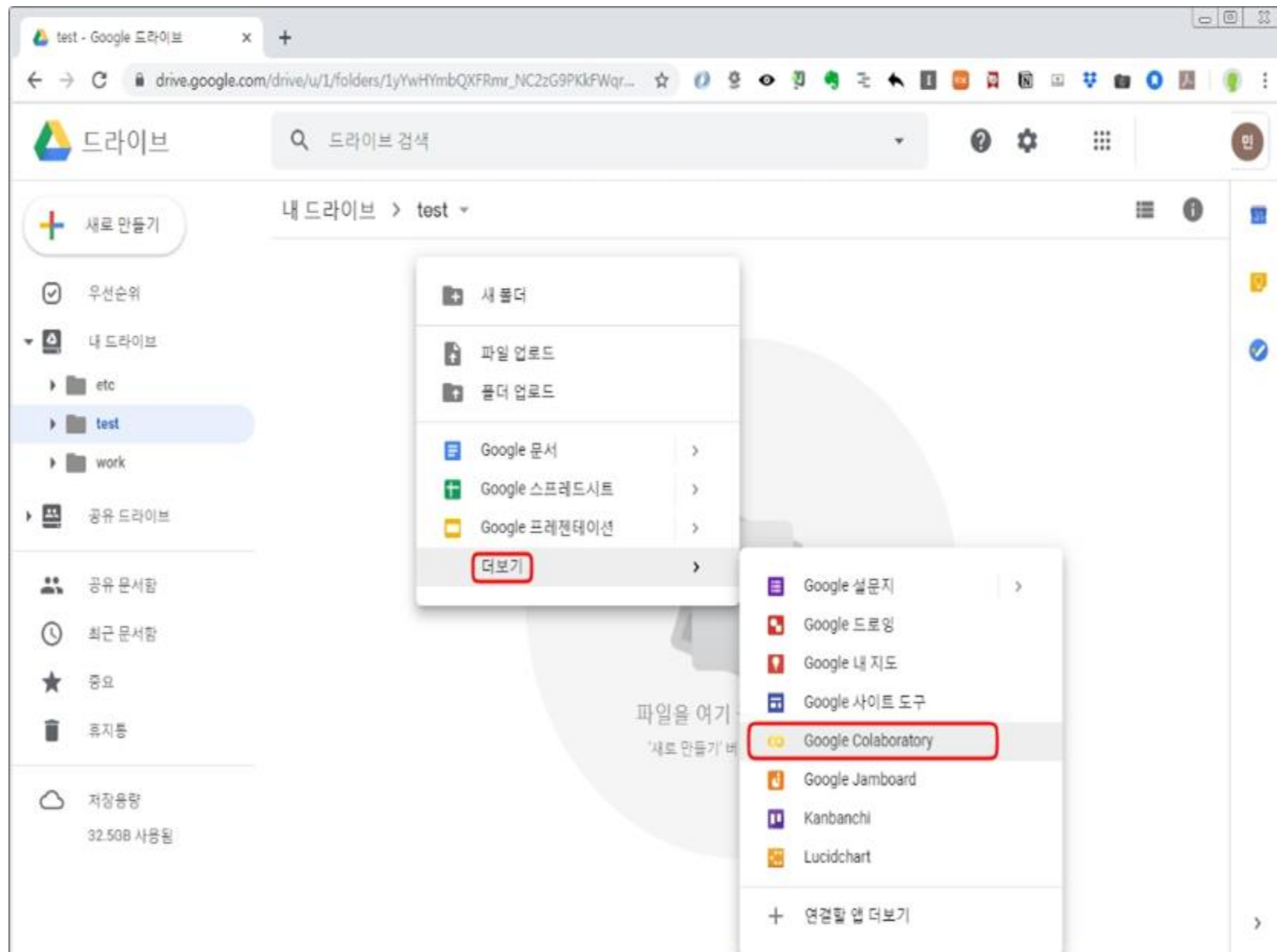
## 실습환경 구축 및 준비

5. 팝업의 좌측 메뉴 앱관리 클릭 > Google Colaboratory 우측의 기본값으로 사용 체크박스 클릭 > 완료



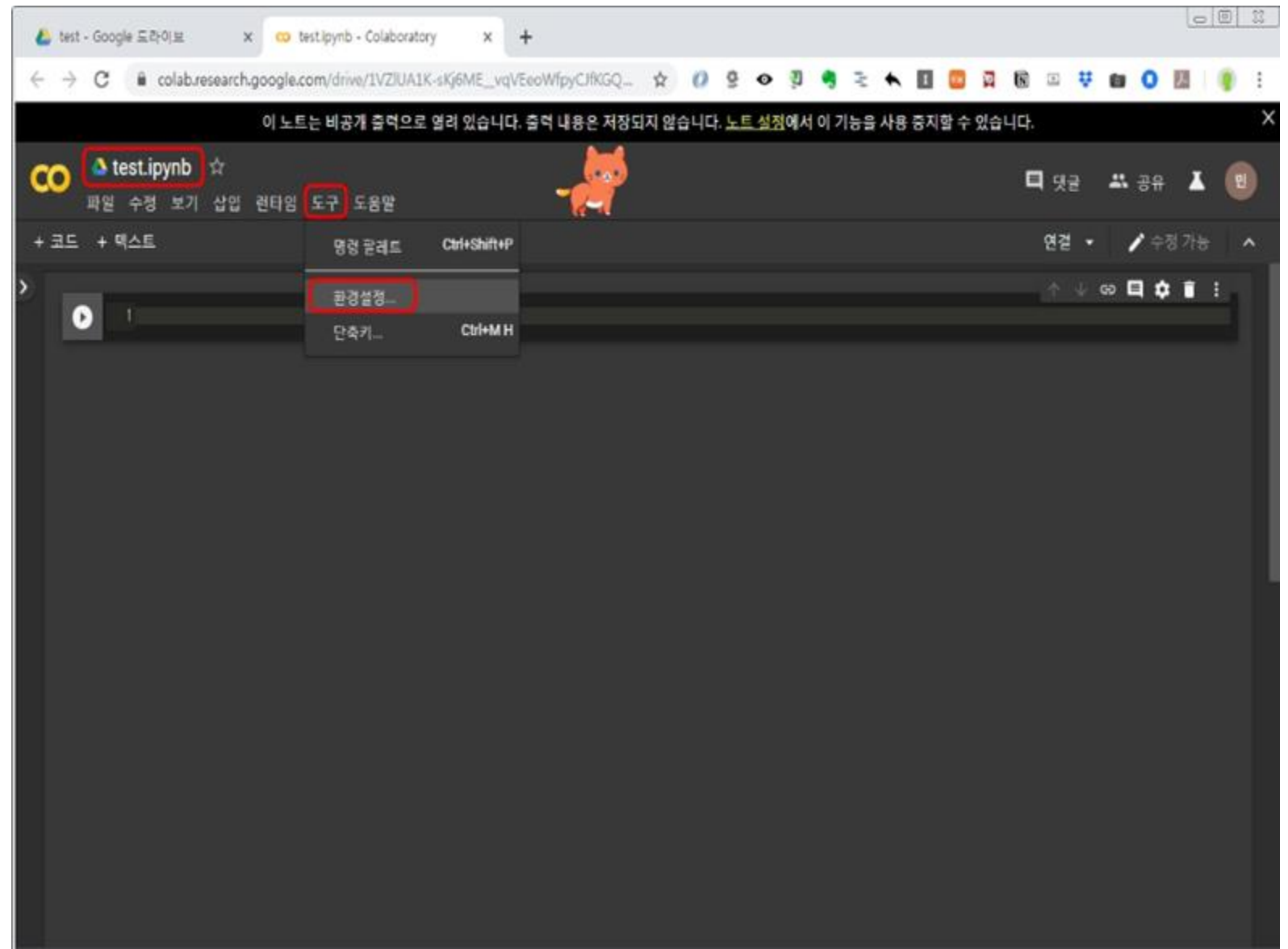
## 💡 실습환경 구축 및 준비

6. 드라이브 메인화면으로 이동 > 우클릭 > 더보기 > Google Colaboratory 클릭



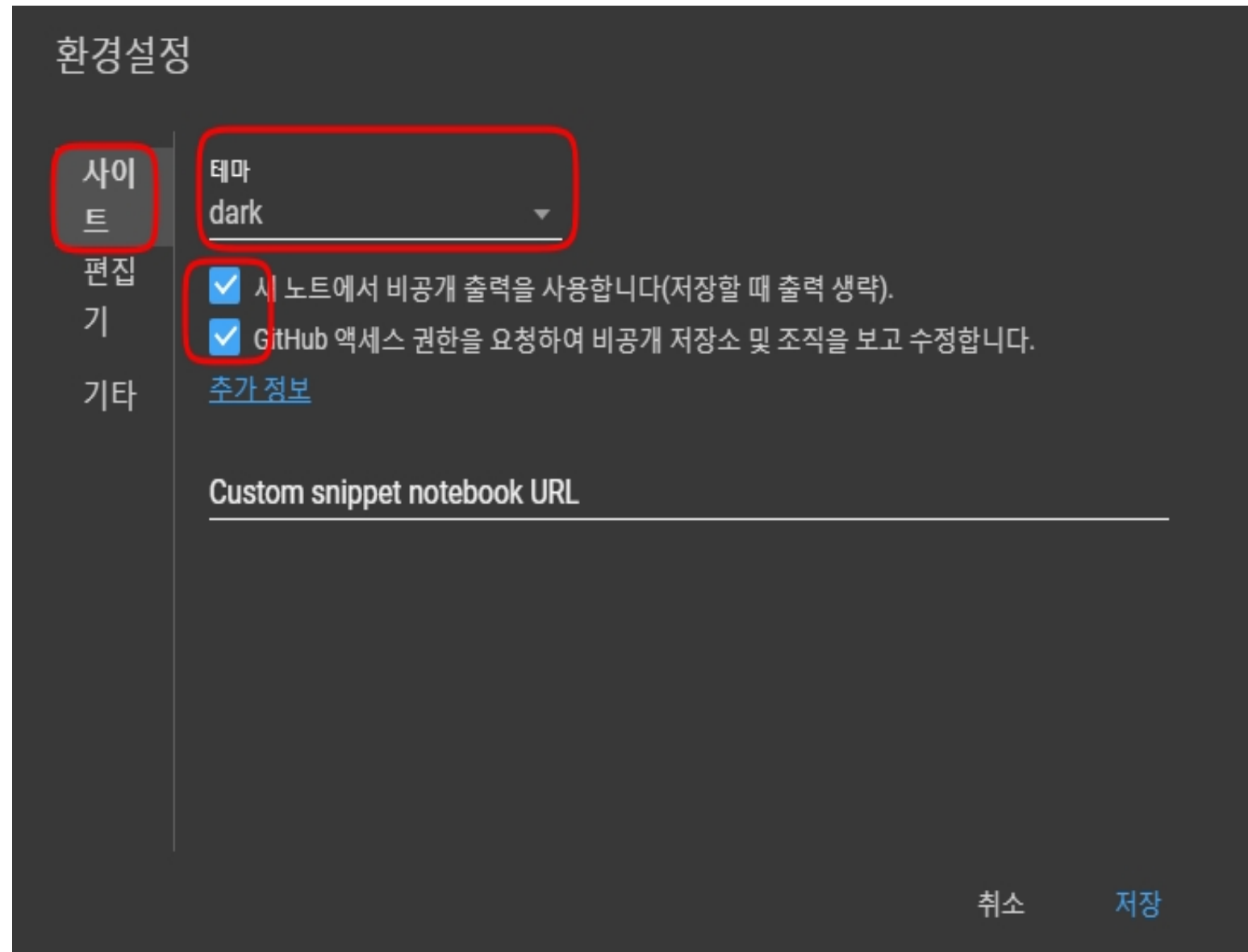
## 💡 실습환경 구축 및 준비

7. .ipynb 확장자 파일의 주피터 노트북이 등장. 파일이름을 클릭하여 test.ipynb로 이름을 변경  
상단 메뉴의 도구 > 환경설정을 클릭한다



## 실습환경 구축 및 준비

8. (좌측) 편집기 탭 클릭 > 들여쓰기 4선택 > 체크박스 2개 체크



환경설정

사이드바  
편집기  
기타

테마  
dark

☒ 시 노트에서 비공개 출력을 사용합니다(저장할 때 출력 생략).  
☒ GitHub 액세스 권한을 요청하여 비공개 저장소 및 조직을 보고 수정합니다.

[추가 정보](#)

Custom snippet notebook URL

취소 저장

## 실습환경 구축 및 준비

9. (좌측) 기타 탭 클릭 > 원하는 설정을 적용 > 저장을 클릭한다.

### 환경설정

사이드바

편집기

기타

들여쓰기 너비(공백 개수)  
4

☒ 행 번호 표시  
☒ 코드 셀에서 자동으로 괄호 및 인용부호 닫기

세로 눈금자 열  
80

취소

저장

## 실습환경 구축 및 준비

### 10. 기타 효과 설정

환경설정

사이드  
트  
편집  
기

기타

파워 레벨  
Many power (use with cal ▾)

☐ 크기 모드

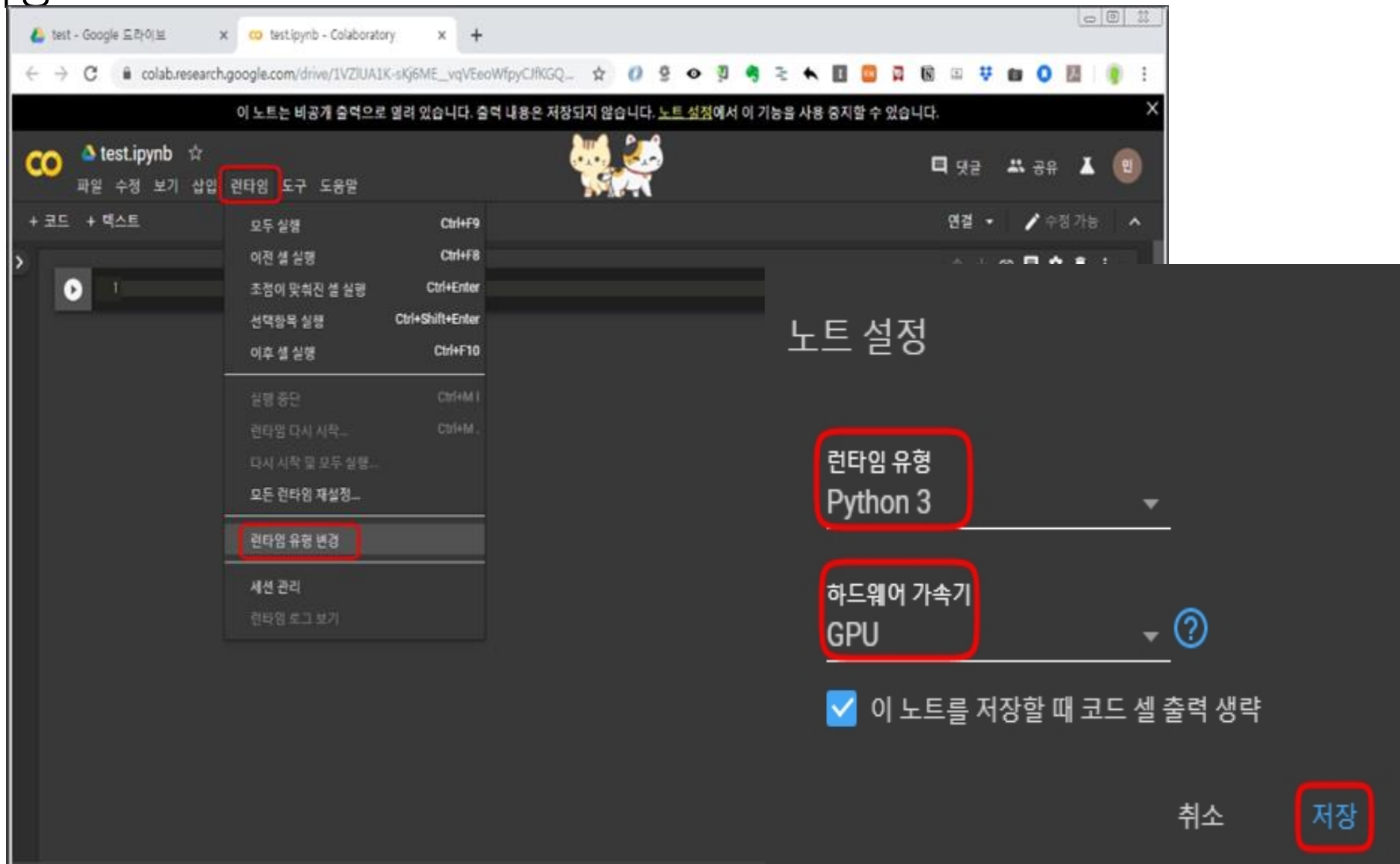
☒ 아기고양이 모드

취소 저장



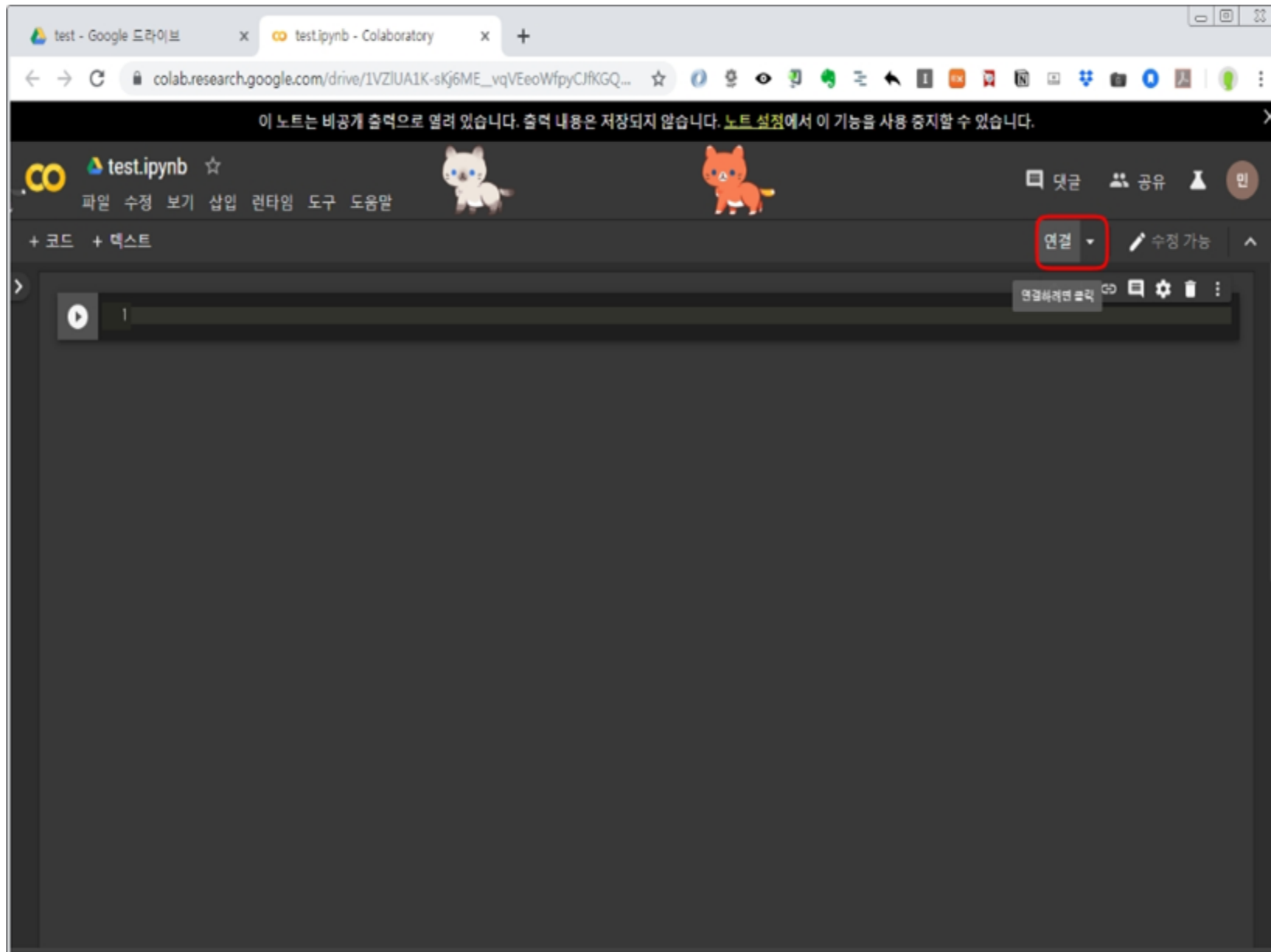
## 💡 실습환경 구축 및 준비

11. 상단 메뉴 런타임 > 런타임 유형 변경을 클릭, 런타임 유형은 Python3를, 가속기는 GPU를 선택 > 저장



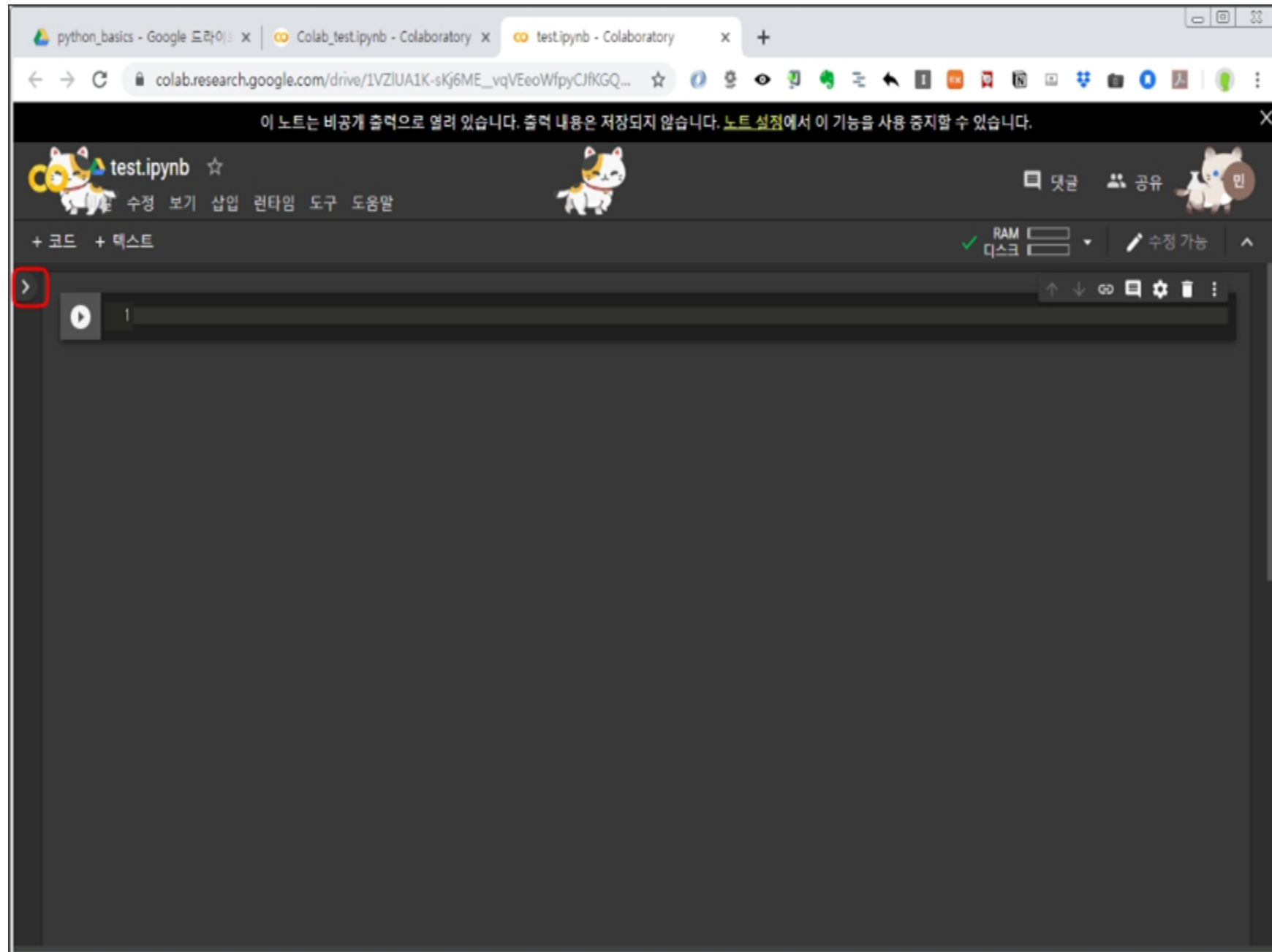
## 실습환경 구축 및 준비

12. 아래 그림과 같이 연결 버튼을 클릭,  
할당중.. > 연결중... > 초기화중.. 으로 텍스트가 변경되며, 최종 RAM, 디스크 사용량 막대  
그래프



## 실습환경 구축 및 준비

13. 아래와 같이 [>] 모양의 버튼을 클릭하면, 목차, 코드스니펫, 파일등의 기능을 활용할 수 있다.





# 검색 vs 데이터분석 vs AI



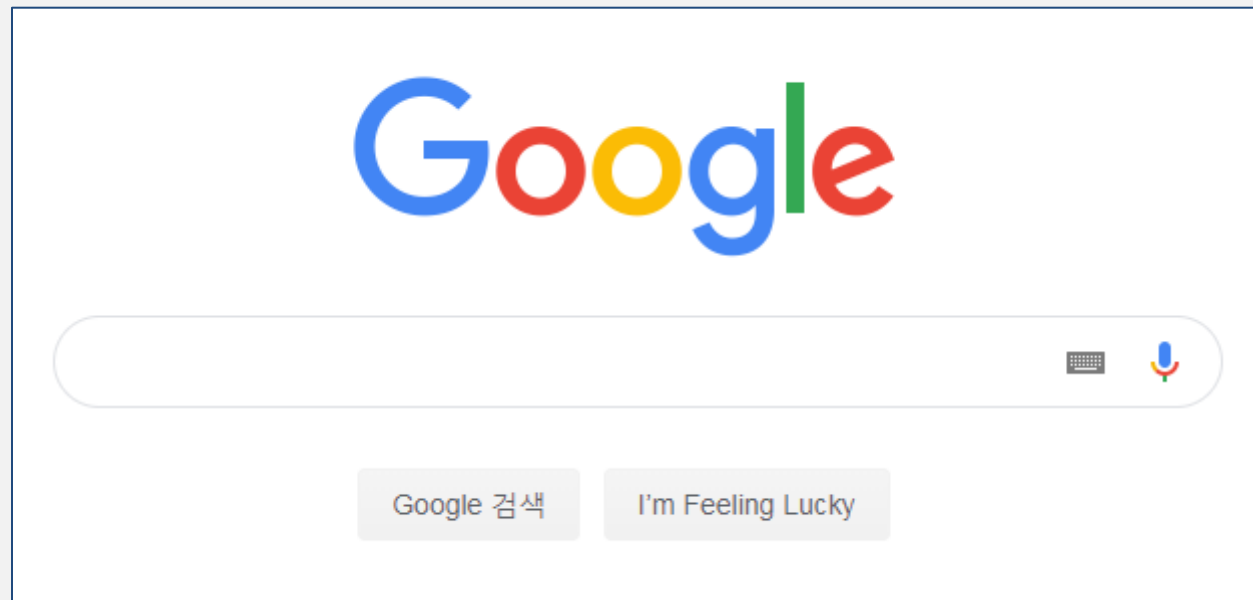
# 검색과 탐색

## 01 Search , Data Analysis, AI



# 검색과 탐색

## 01 Search , Data Analysis, AI





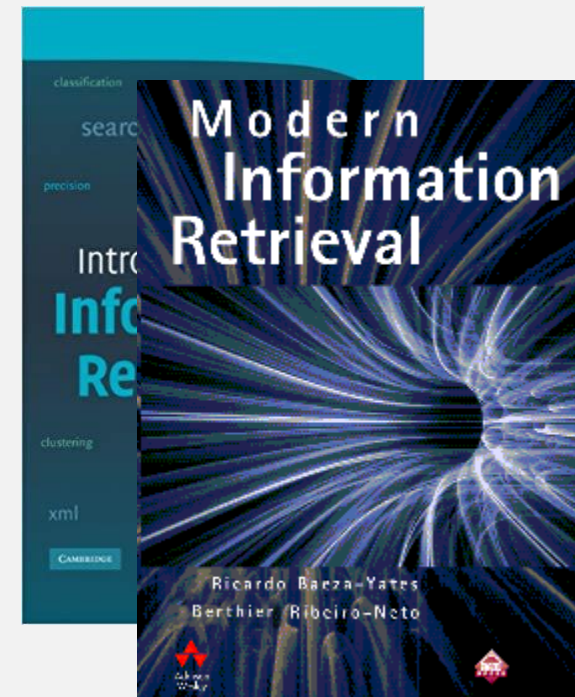
# 💡 검색 주요 개념

## 02 검색 주요 개념

### *Brief Contents*

1	Boolean retrieval	1
2	The term vocabulary and postings lists	19
3	Dictionaries and tolerant retrieval	49
4	Index construction	67
5	Index compression	85
6	Scoring, term weighting and the vector space model	109
7	Computing scores in a complete search system	135
8	Evaluation in information retrieval	151
9	Relevance feedback and query expansion	177
10	XML retrieval	195
11	Probabilistic information retrieval	219
12	Language models for information retrieval	237
13	Text classification and Naïve Bayes	253
14	Vector space classification	289
15	Support vector machines and machine learning on documents	319
16	Flat clustering	349
17	Hierarchical clustering	377
18	Matrix decompositions and latent semantic indexing	403
19	Web search basics	421
20	Web crawling and indexes	443
21	Link analysis	461

## *IR ( Information Retrieval )*



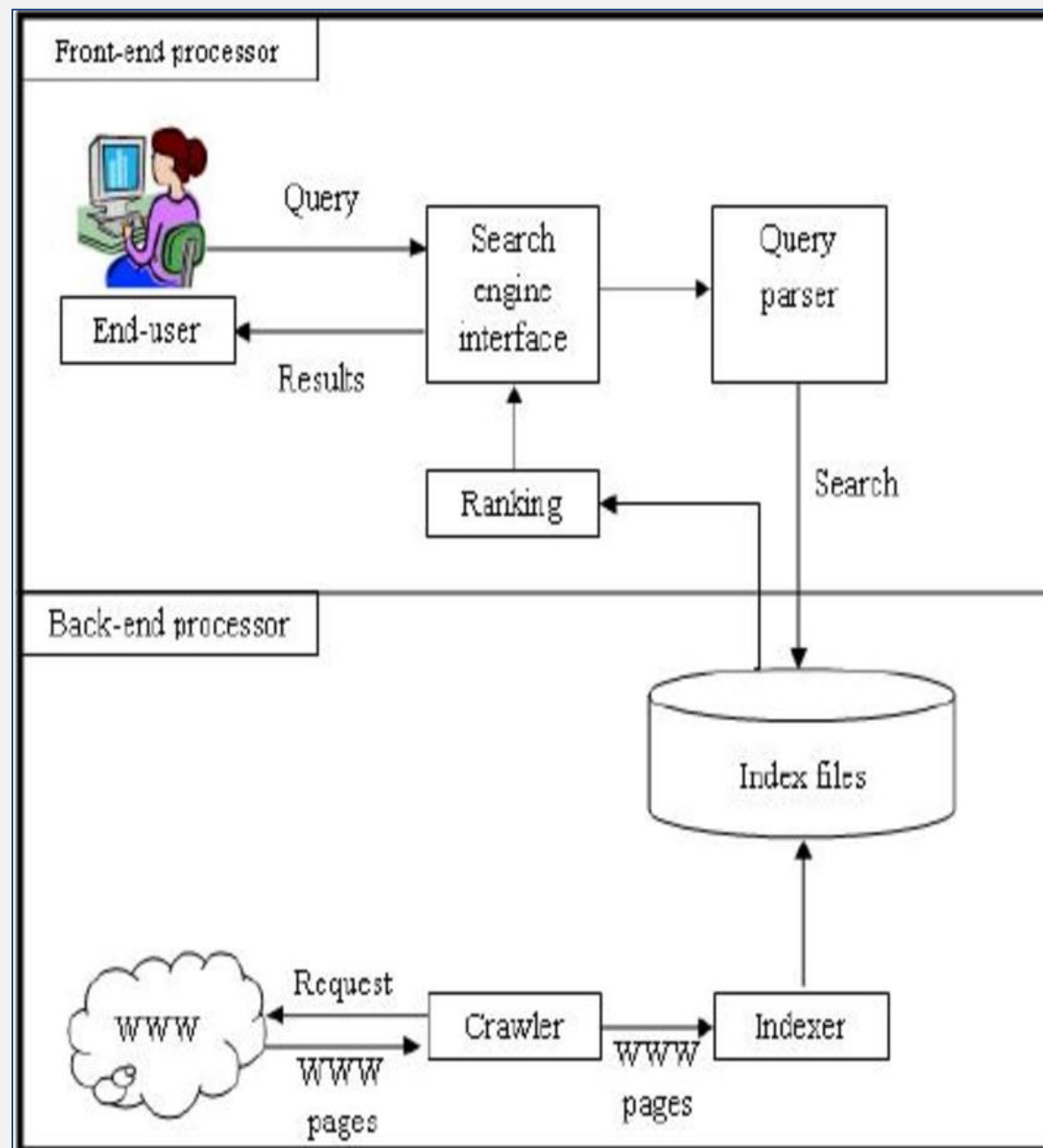
### 정보 검색의 4요소

- 정보(문서) 수집                      - Crawling
- 단위 정보(단어) 추출                - Stemming
- 정보 색인                                - Indexing
- 검색과 랭킹                            - Searching

## 💡 검색 주요 개념

### 02 검색 주요 개념

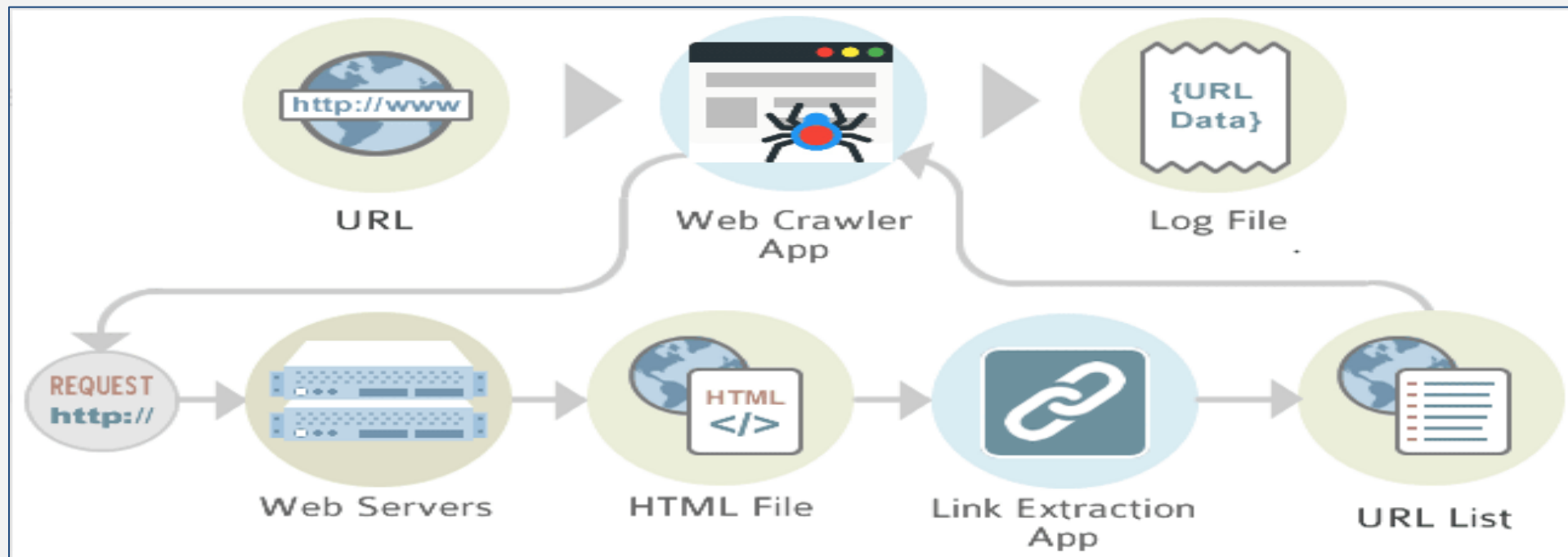
# *Search Workflow*



# 💡 검색 주요 개념

## 02 검색 주요 개념

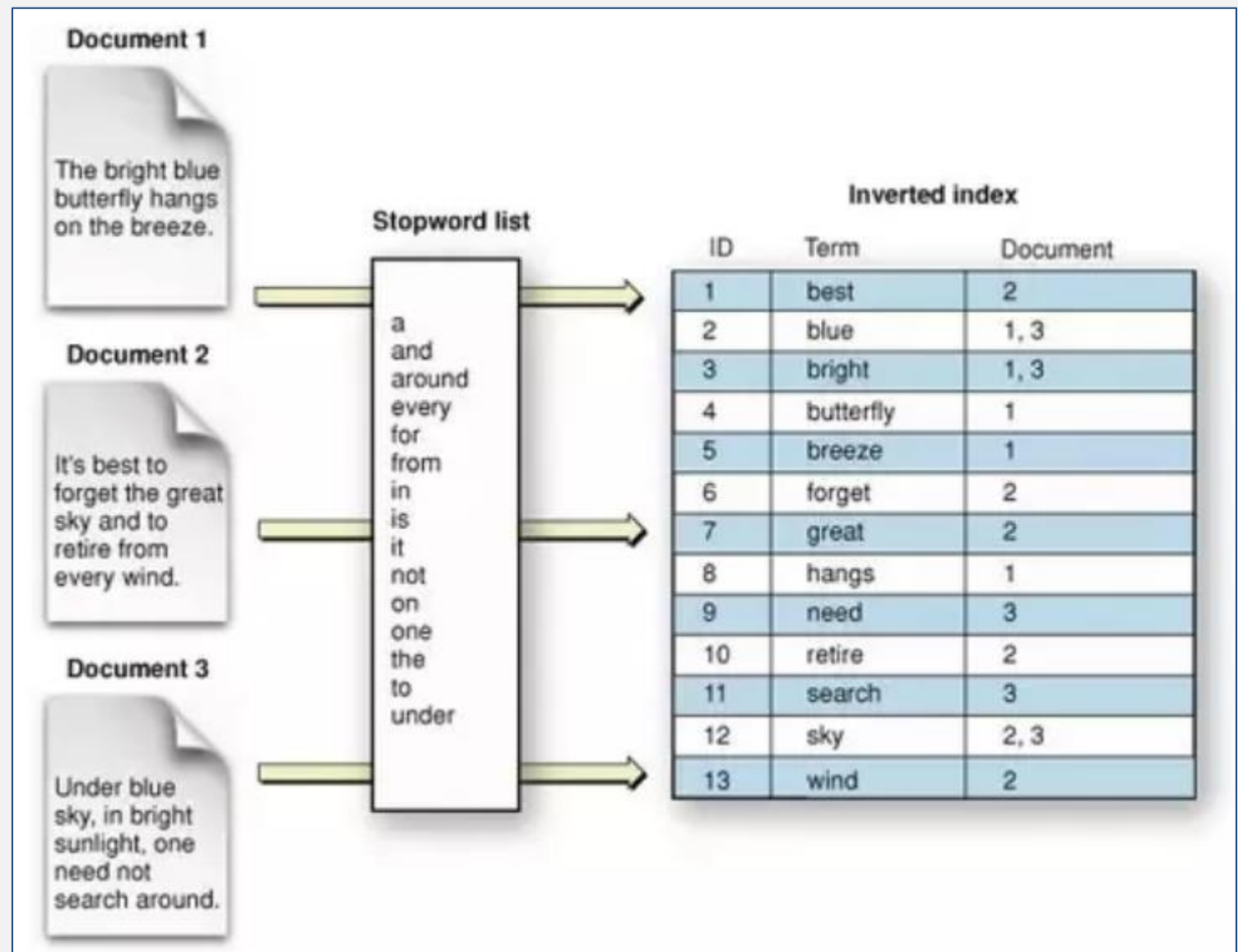
### *Crawler*



# 💡 검색 주요 개념

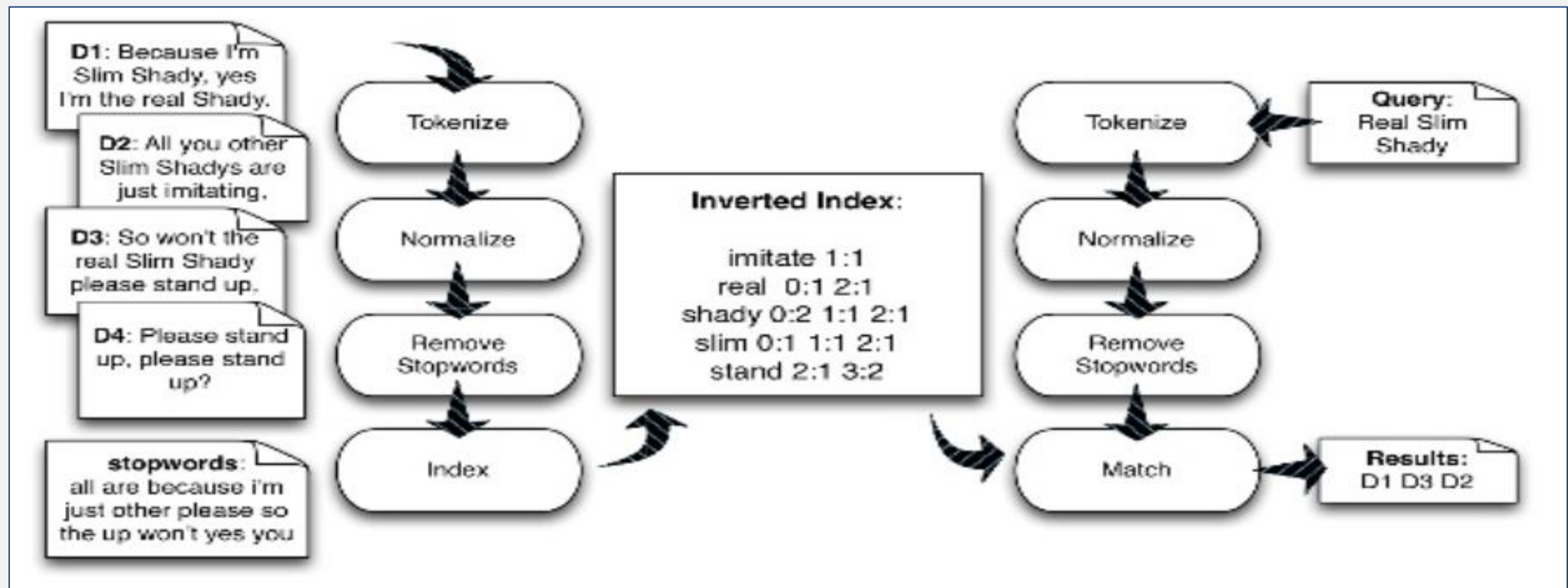
## 02 검색 주요 개념

### *Indexer*



# 💡 검색 주요 개념

## 02 검색 주요 개념



*Query  
Processor*



# 💡 검색 주요 개념

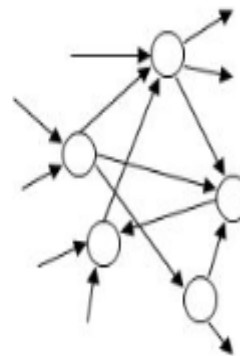
## 02 검색 주요 개념

### □ Content-based (query dependent)

- TF, IDF
- BM25
- Classical IR
- ...

### □ Connectivity based (web)

- PageRank
- HITS
- ...

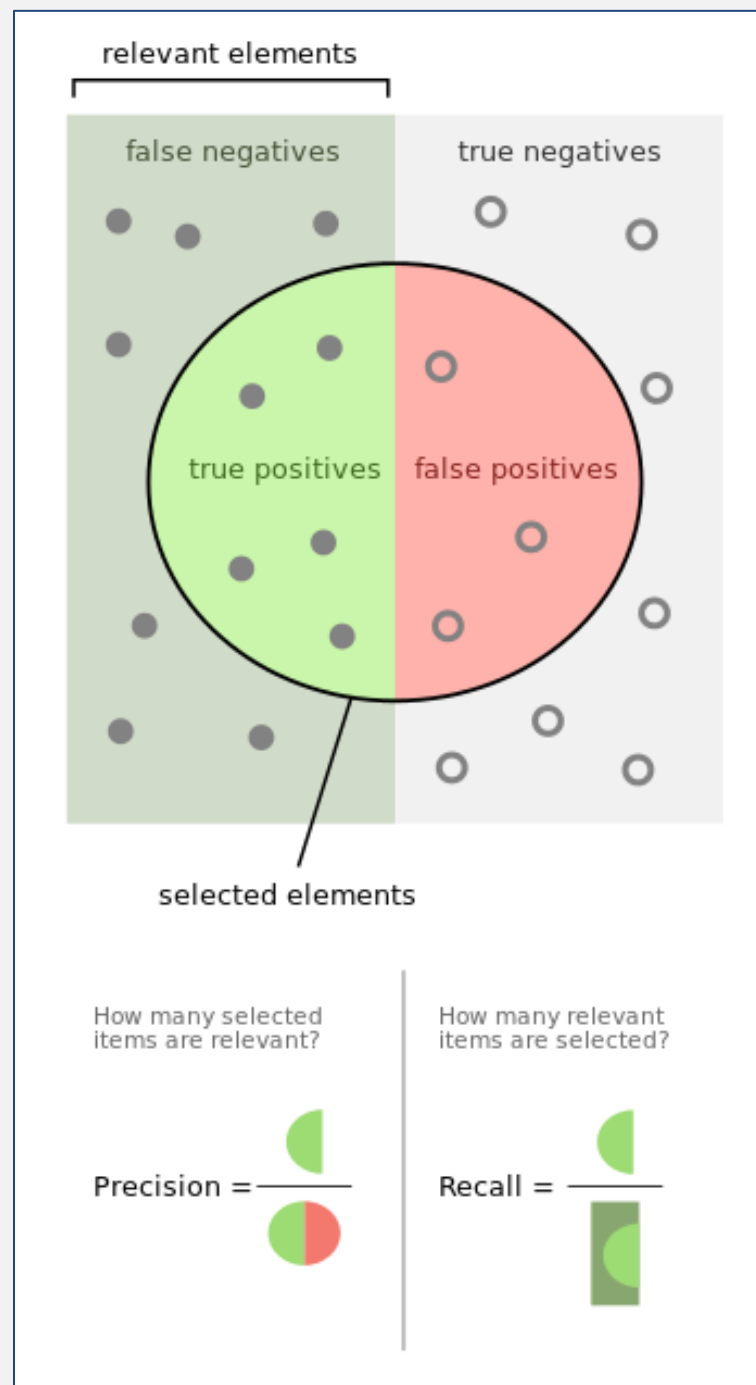


***Search  
Models  
( Raking )***

# 💡 검색 주요 개념

## 02 검색 주요 개념

### *Search Evaluation*



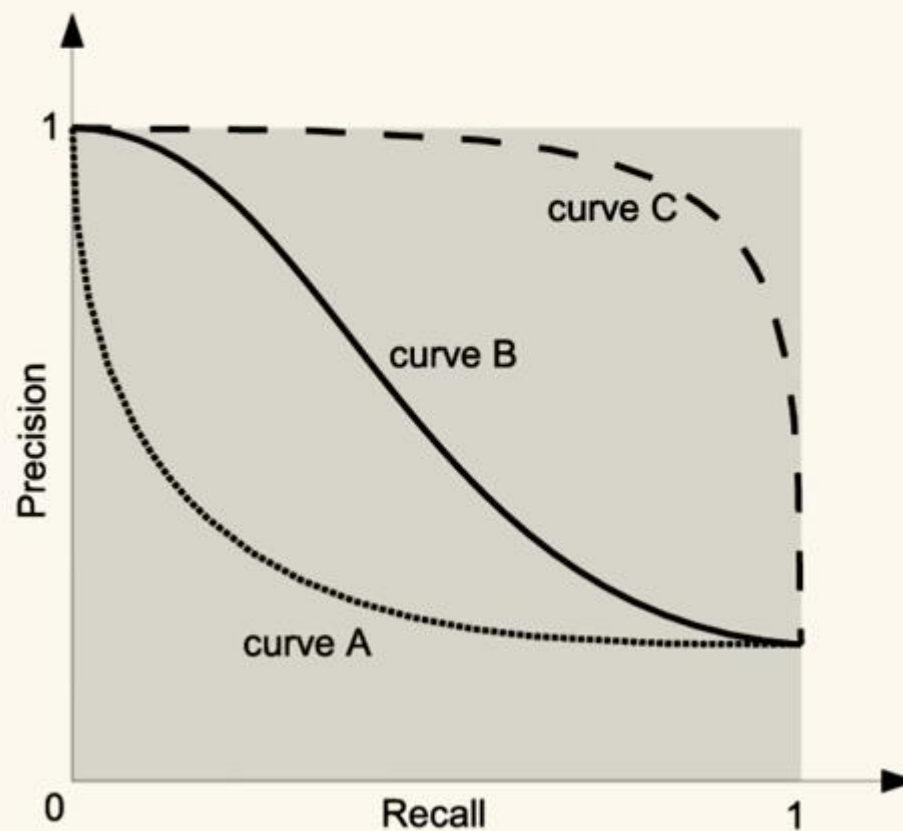
# 💡 검색 주요 개념

## 02 검색 주요 개념

좋은 검색 기술은 뭘까?

원하는 정보를

- 빠르고,
- 정확하고,
- 많이...



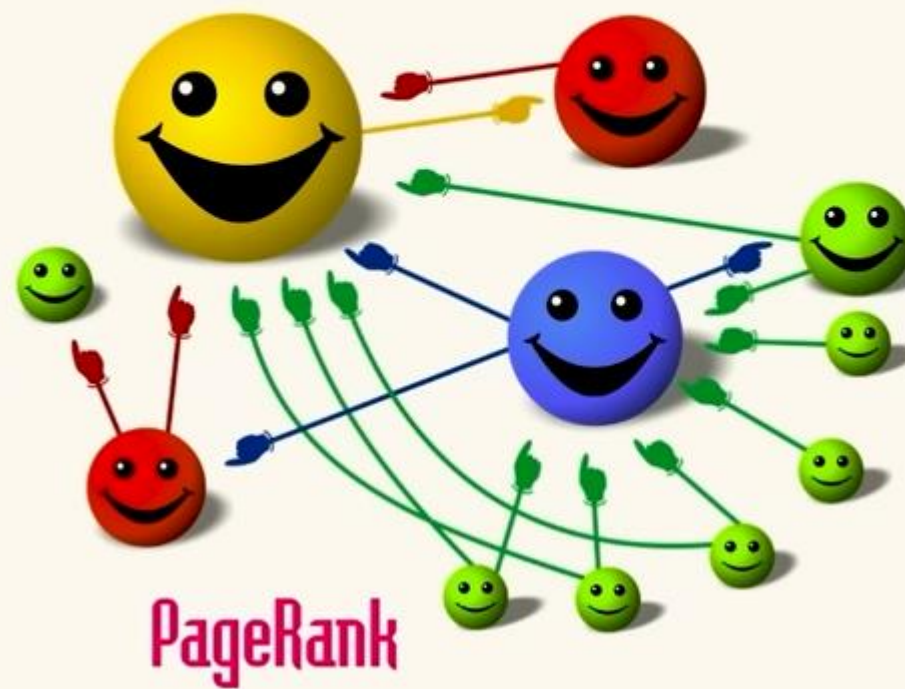
# 💡 검색 주요 개념

## 02 검색 주요 개념

그래서

신님은...

( TF x IDF ) &



# 💡 검색 주요 개념

## 02 검색 주요 개념

### 그렇게 좋은 '구글 검색'의 한계?

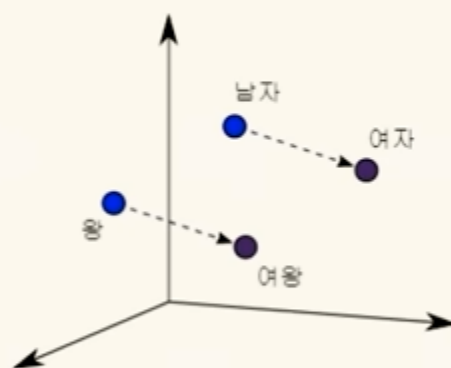
- 많아도 너무 많은 정보
- 동형이의 / 이형동의
- 일일이 클릭하고, 읽고  
내가 답을 찾아야...
- 파편화된 정보의 이해



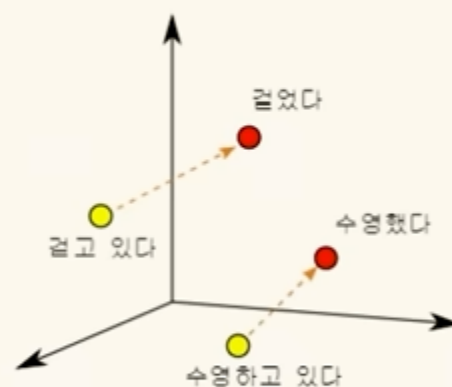
# 💡 검색 주요 개념

## 02 검색 주요 개념

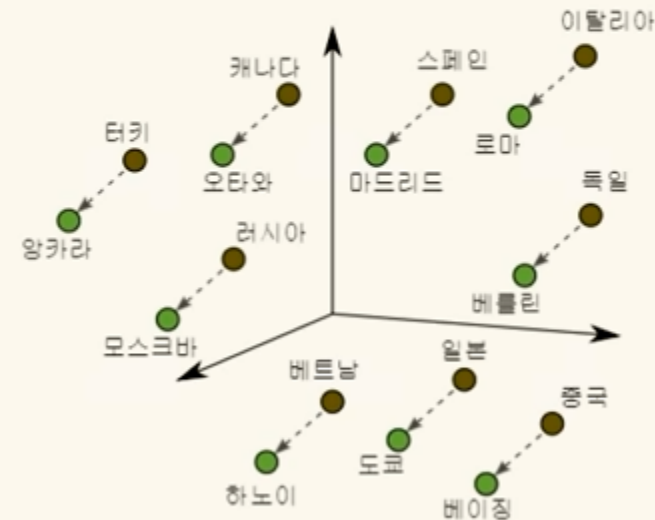
### 벡터 : 인공지능의 언어



남자-여자



동사 시제



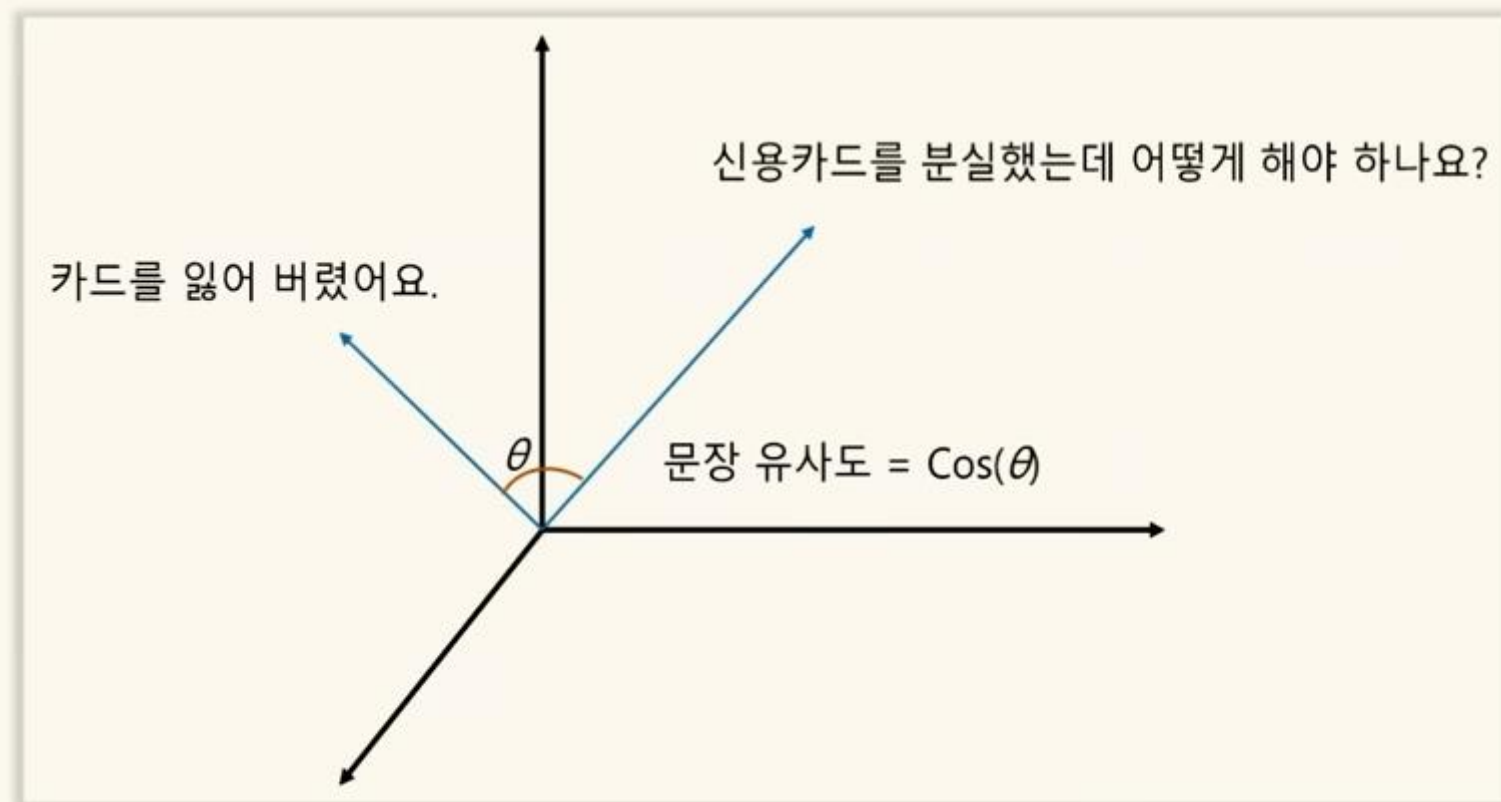
국가-수도



## 💡 검색 주요 개념

### 02 검색 주요 개념

#### 검색? : 벡터 공간 유사도



# 검색 주요 개념

## 02 검색 주요 개념

생성AI (ChatGPT)에서 갑자기 왜 AI검색??

YOU<sup>.com</sup>

goover



perplexity

SearchGPT 



# Elastic

---

1. Elastic Overview
2. ELASTIC Stack
3. CRUD Operation
4. Analyzer

## 01 Elastic Overview

### *Lucene*

- As any good story begins, “Once upon a time...”
  - More precisely: in 1999, Doug Cutting created an open-source project called **Lucene**
- Lucene is:
  - a **search engine library** entirely written in Java
  - a top-level Apache project, as of 2005
  - great for full-text search
- But, Lucene is also:
  - a library (you have to incorporate it into your application)
  - challenging to use
  - not originally designed for scaling



## 01 Elastic Overview

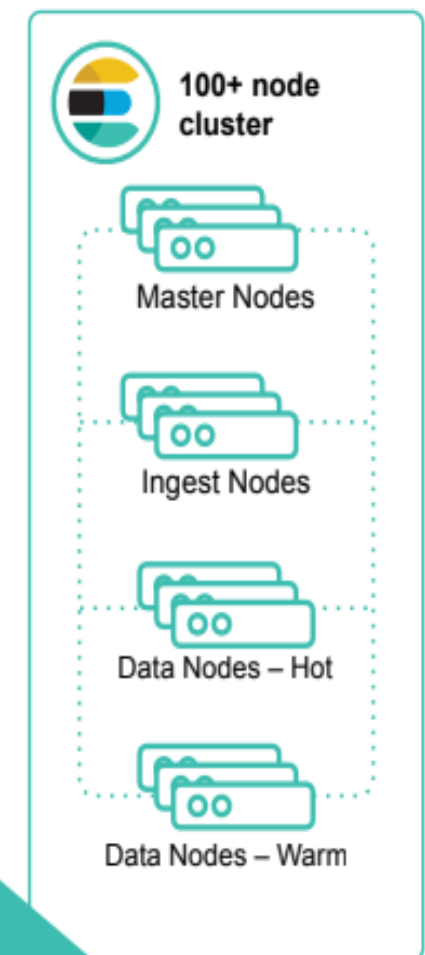
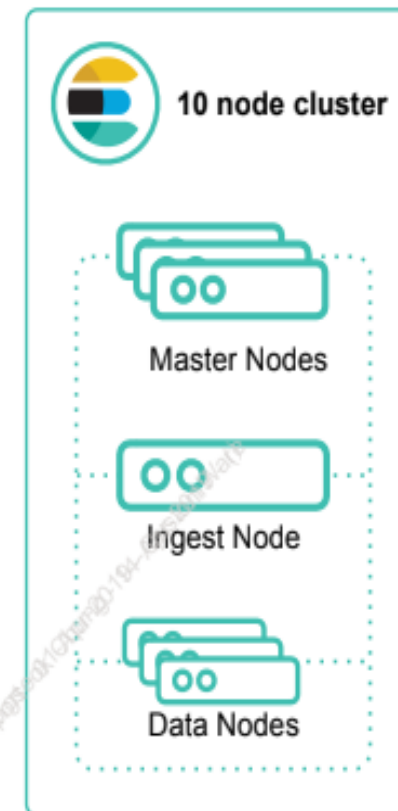
- In 2004, Shay Banon developed a product called **Compass**
  - Built on top of Lucene, Shay's goal was to have search integrated into Java applications as simply as possible
- The need for **scalability** became a top priority
- In 2010, Shay completely rewrote Compass with two main objectives:
  1. *distributed from the ground up in its design*
  2. *easily used by any other programming language*
- He called it **Elasticsearch**
  - ...and we all lived happily ever after!
- Today Elasticsearch is the most popular enterprise search engine

*The Birth of  
ElasticSearch*

## 01 Elastic Overview

### *Distributed Processing*

- Elasticsearch is distributed and scales horizontally:



A **node** is an instance of Elasticsearch

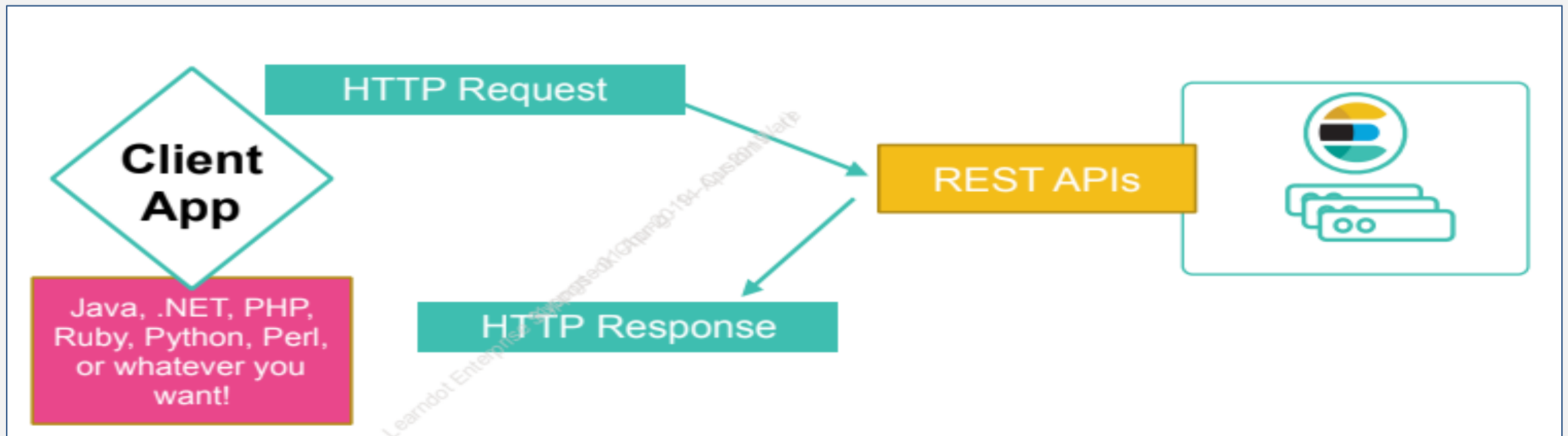
A **cluster** is a collection of Elasticsearch nodes

Your cluster can grow as your needs grow

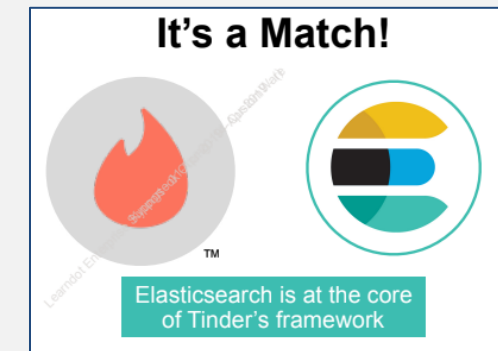


## *Restful APIs*

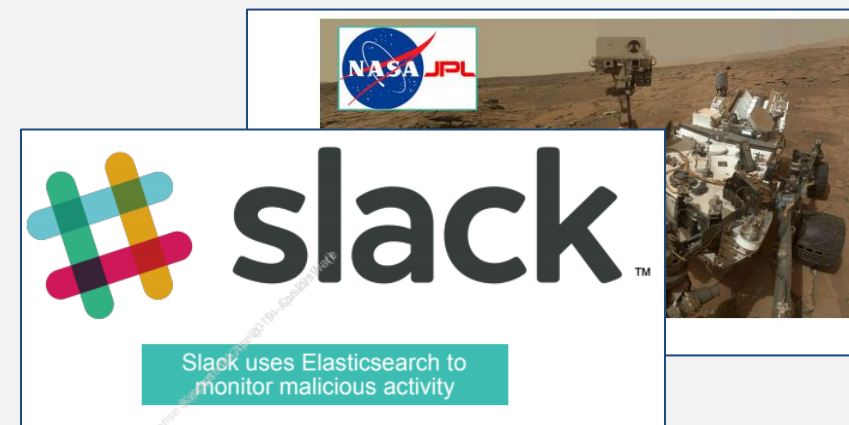
*( Easily used by other languages )*



## 01 Elastic Overview



## *More than Search*



## 01 Elastic Overview

- Use **GET** to retrieve an indexed document
  - notice we specify the *index*, the *type* and an *ID*
  - response code is **200** if the document is found, **404** if not

"I am looking for  
blog 1."

```
GET my_blogs/_doc/1
```

```
{
  "_index" : "my_blogs",
  "_type" : "_doc",
  "_id" : "1",
  "_version" : 3,
  "found" : true,
  "_source" : {
    "title": "Elasticsearch 5.0.0-beta1 released",
    "category": "Releases",
    "date": "September 26, 2016",
    "author": {
      "first_name": "Clinton",
      "last_name": "Gormley",
      "company": "Elastic"
    }
  }
}
```

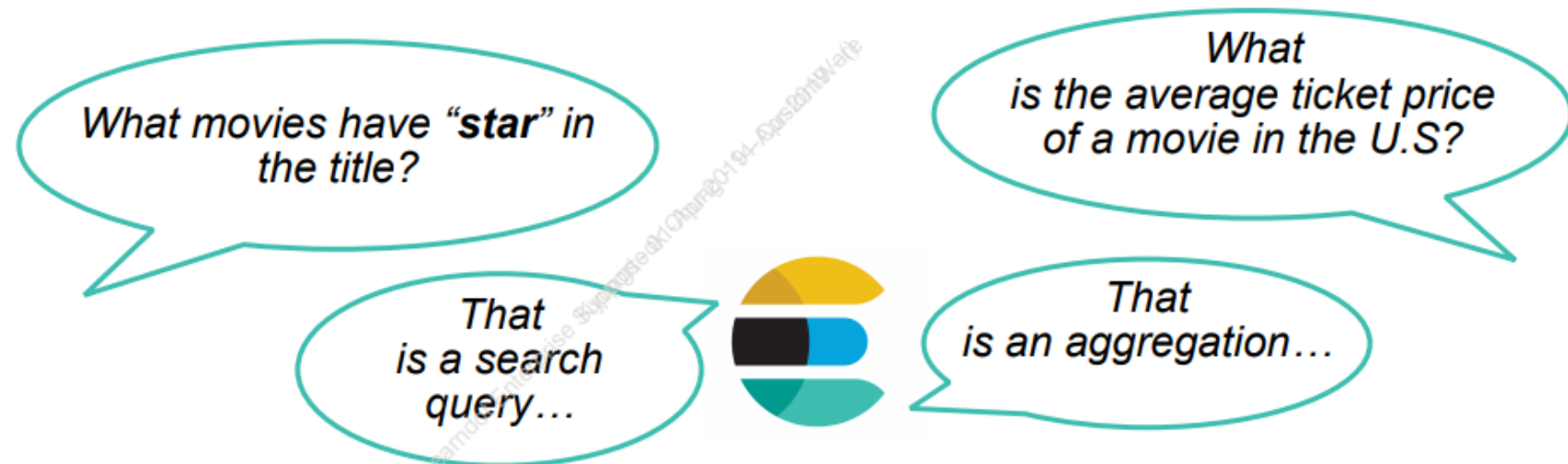
The original document is  
returned in the **\_source** field

# Retrieval Documents

## 01 Elastic Overview

- We have been focusing on search, but Elasticsearch has another powerful capability known as aggregations
- **Aggregations** are a way to perform analytics on your indexed data

### *Aggregations*





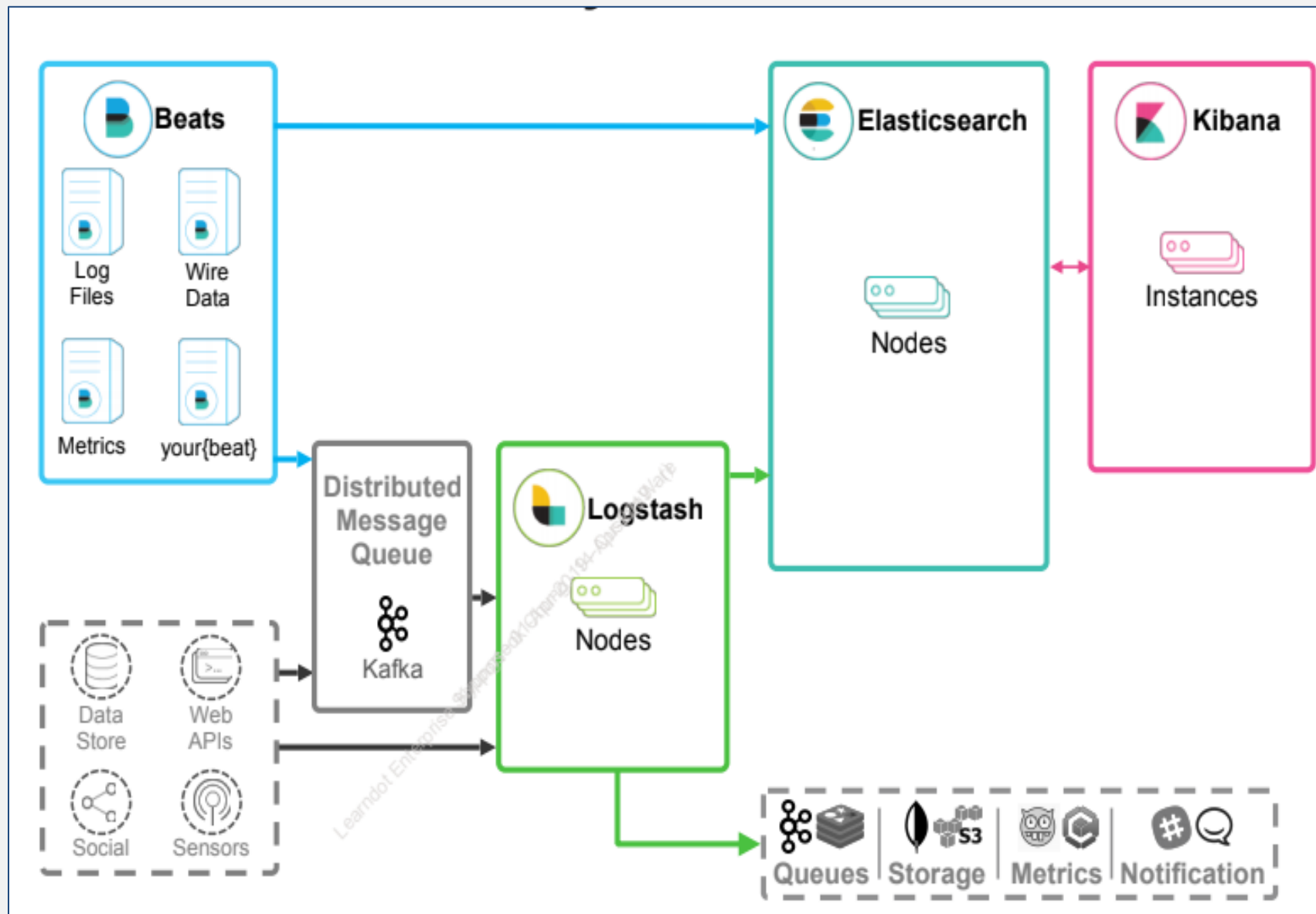


# Elastic

---

1. Elastic Overview
2. ELASTIC Stack
3. CRUD Operation
4. Analyzer

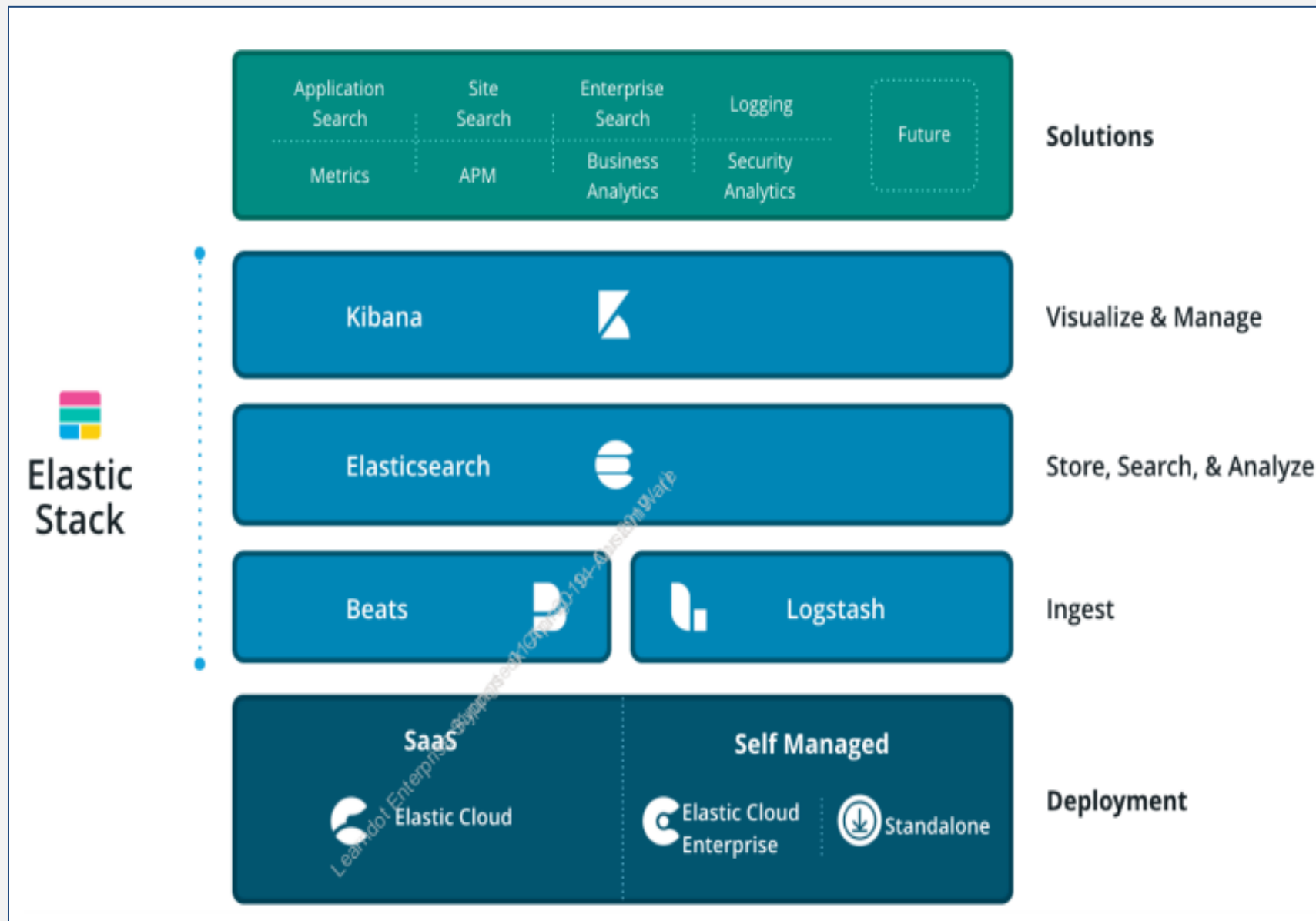
## 02 Elastic Stack



*Elastic  
Stack  
Components*



## 02 Elastic Stack



*Elastic  
Stack  
Layers*

## 02 Elastic Stack

*Logstash*



SQL database  
with a table  
named **"blogs"**



logstash

my\_cluster



elasticsearch

LearnDocker Enterprise & Kubernetes  
Supporting 100+ Docker & Kubernetes  
2019-2020

### *Beats*



web log files

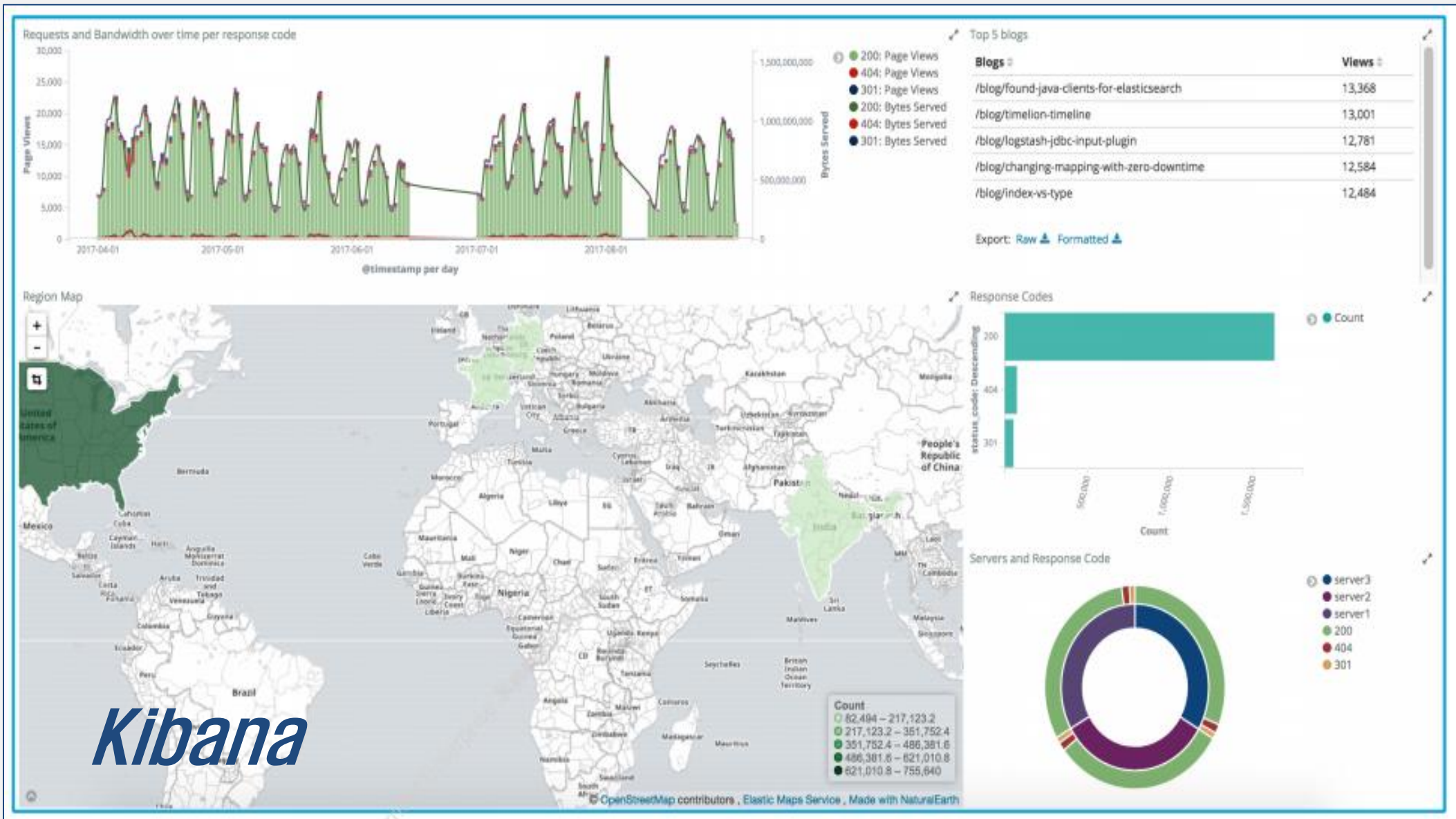


my\_cluster



elasticsearch

# 01 Elastic Stack







# Elastic

---

1. Elastic Overview
2. ELASTIC Stack
3. CRUD Operation
4. Analyzer

## 01 CRUD Operation

### *Documents must be JSON Object*

A JSON object...

```
{  
  "title": "Solving the Small but...",  
  "category": "Culture",  
  "date": "December 22, 2017",  
  "author": {  
    "first_name": "Daniel",  
    "last_name": "Cecil",  
    "company": "Elastic"  
  }  
}
```

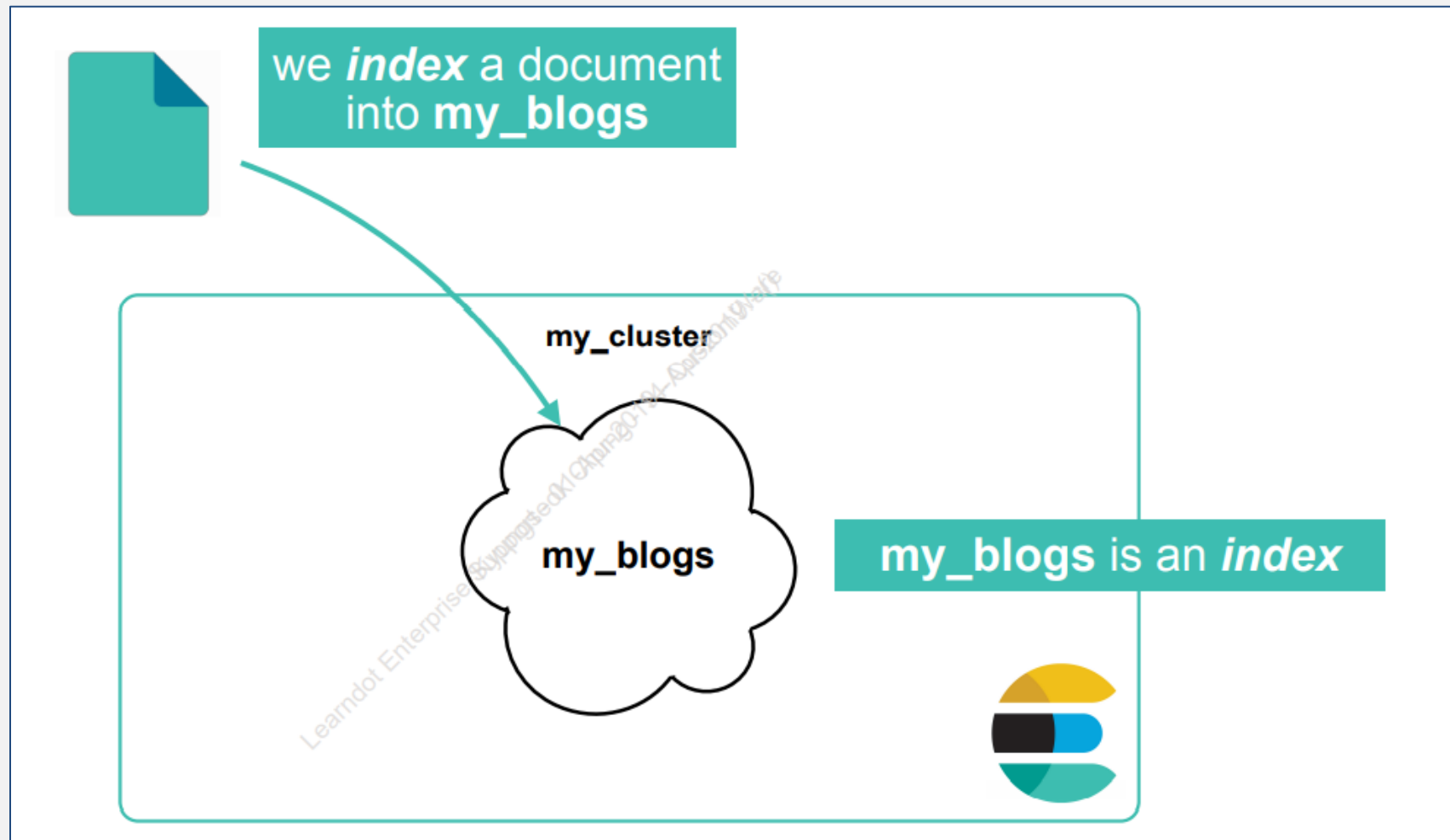
...is stored in Elasticsearch as  
a document





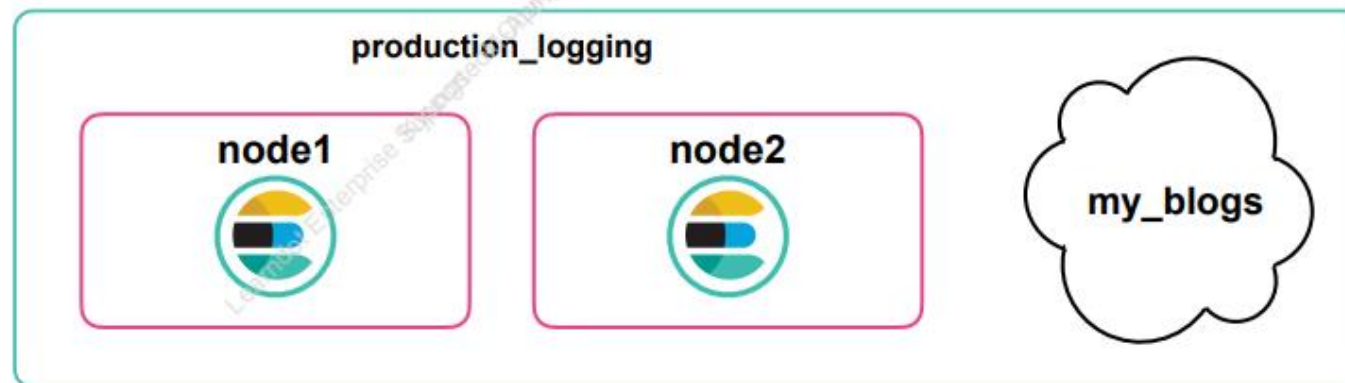
## 01 CRUD Operation

### *Documents are Indexed into an Index*



## 01 CRUD Operation

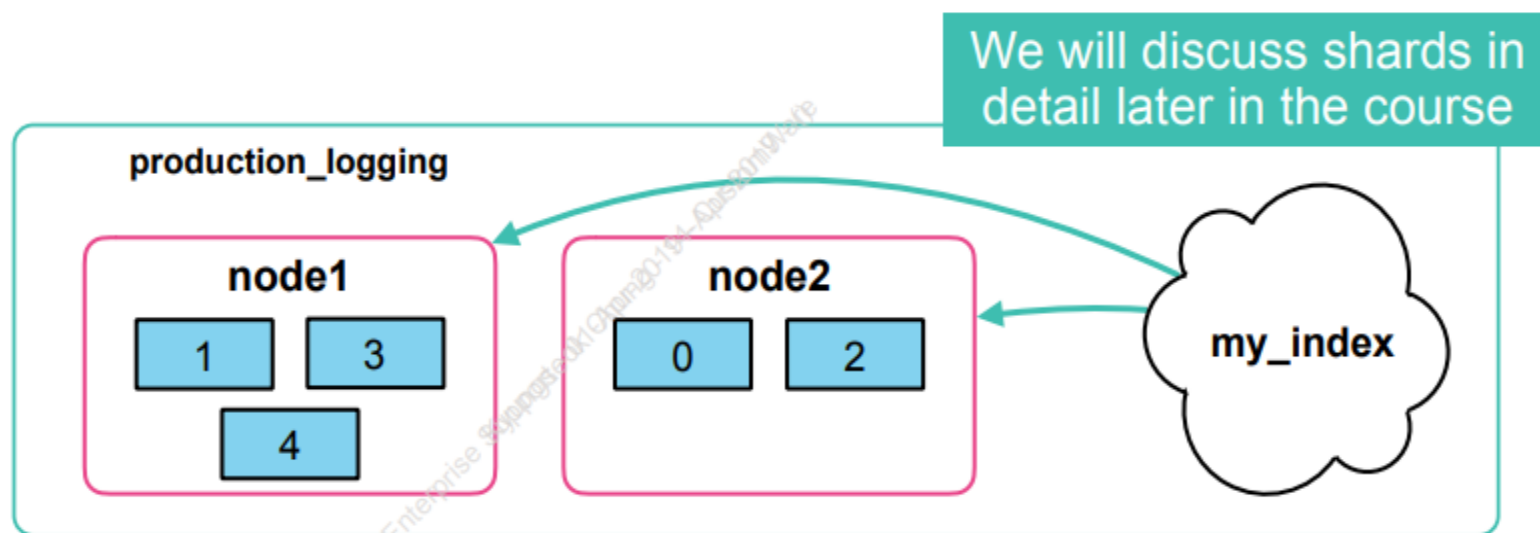
- An **index** in Elasticsearch is a **logical** way of grouping data
  - an index has a **mapping** that defines the fields in the index
  - an index is a **logical namespace** that maps to where its contents are stored in the cluster
- There are two different concepts in this definition:
  - an index has some type of **data schema** mechanism
  - an index has some type of mechanism to **distribute data** across a cluster



*Index*

## 01 CRUD Operation

- Elasticsearch is a ***distributed*** document store
- A ***shard*** is a single piece of an Elasticsearch index
  - Indexes are partitioned into shards so they can be distributed across multiple nodes



*Shard*

## 01 CRUD Operation

Index	<pre>PUT my_blogs/_doc/4 {   "title" : "Elasticsearch released",   "category": "Releases" }</pre>	<pre>POST my_blogs/_doc {   "title" : "Elasticsearch released",   "category": "Releases" }</pre>
Create	<pre>PUT my_blogs/_doc/4/_create {   "title" : "Elasticsearch released",   "category": "Releases" }</pre>	
Read	<pre>GET my_blogs/_doc/4</pre>	
Update	<pre>POST my_blogs/_doc/4/_update {   "doc" : {     "title" : "Elasticsearch 6.2 released"   } }</pre>	
Delete	<pre>DELETE my_blogs/_doc/4</pre>	
Search	<pre>GET my_blogs/_search</pre>	

## 01 CRUD Operation

### *Query string VS Query DSL*

- **Query string** is a simplified approach to search documents
  - can be used in the URL
  - simple and easy-to-use
  - but hard to write complex queries
  - unforgiving for small typos (quotes, parentheses, ...)
- **Query DSL** (Domain Specific Language) allows you to write queries in a JSON format
  - must send a body
  - exposes the entire collection of Elasticsearch APIs
  - very powerful

## 01 CRUD Operation

### *Query string VS Query DSL*

```
GET blogs/_search
{
  "query": {
    "match": {
      "content": "ingest nodes"
    }
  }
}
```

Query DSL

```
GET blogs/_search?q=content:(ingest nodes)
```

query string



## 01 CRUD Operation

### *Response*

```
"hits": {  
  "total": 241,  
  "max_score": 9.3005905,  
  "hits": [  
    {  
      "_index": "blogs",  
      "_type": "_doc",  
      "_id": "7ItcB2cBZSWiUi4z_dX",  
      "_score": 9.3005905,  
      "_source": {  
        "title": "A New Way To Ingest - Part 2"  
      }  
    },  
    {  
      "_index": "blogs",  
      "_type": "_doc",  
      "_id": "2otcB2cBZSWiUi41Pkb",  
      "_score": 8.616863,  
      "_source": {  
        "title": "Ingest Node: A Client's Perspective"  
      }  
    },  
    ...  
  ]  
}
```

241 documents were a match

but only the top 10 documents will be returned by default

Each hit has a `_score`

Results are sorted by `_score` (by default)



# Elastic

---

1. Elastic Overview
2. ELASTIC Stack
3. CRUD Operation
4. Analyzer

# Mapping

GET my\_index/\_mapping

This particular mapping has a **type** named “\_doc”

The “**properties**” section contains the fields and data types in your documents

```
{
  "my_index": {
    "mappings": {
      "_doc": {
        "properties": {
          "comment": {
            "type": "text",
            "fields": {
              "keyword": {
                "type": "keyword",
                "ignore_above": 256
              }
            }
          },
          "details": {
            "properties": {
              "created_at": {
                "type": "date"
              },
              "employee": {
                "type": "boolean"
              }
            }
          }
        }
      }
    }
  }
}
```



## 신영 뉴스 *Mapping*

```
"mappings" : {
  "properties" : {
    "uid" : {
      "type" : "keyword"
    },
    "title" : {
      "type" : "text",
      "term_vector" : "with_positions_offsets_payloads",
      "store" : true,
      "analyzer" : "nori_analyzer"
    },
    "topic" : {
      "type" : "keyword"
    },
    "content" : {
      "type" : "text",
      "term_vector" : "with_positions_offsets_payloads",
      "store" : true,
      "analyzer" : "nori_analyzer"
    },
    "symbol_list" : {
      "type" : "nested"
    },
    "industry_list" : {
      "type" : "nested"
    },
    "keyword_list" : {
      "type" : "keyword"
    },
    "class_list" : {
      "type" : "nested"
    },
    "publisher" : {
      "type" : "keyword"
    },
    "date" : {
      "type" : "date"
    }
  }
}
```

## 01 Analyzer

### *Text Analyze*

Use the `_analyze` API

Test the `"simple"` analyzer on the given `"text"`

```
GET _analyze
{
  "analyzer": "simple",
  "text": "How to configure ingest nodes?"
}
```

"How to configure ingest nodes?"

"simple"  
analyzer

how

to

configure

ingest

nodes

## Analyzer Setting : nori\_analyzer

인간이 문명의 이기를 사용할때에 그것이 인간자신을 위해  
지혜있게 사용되어야 한다는 것이다.



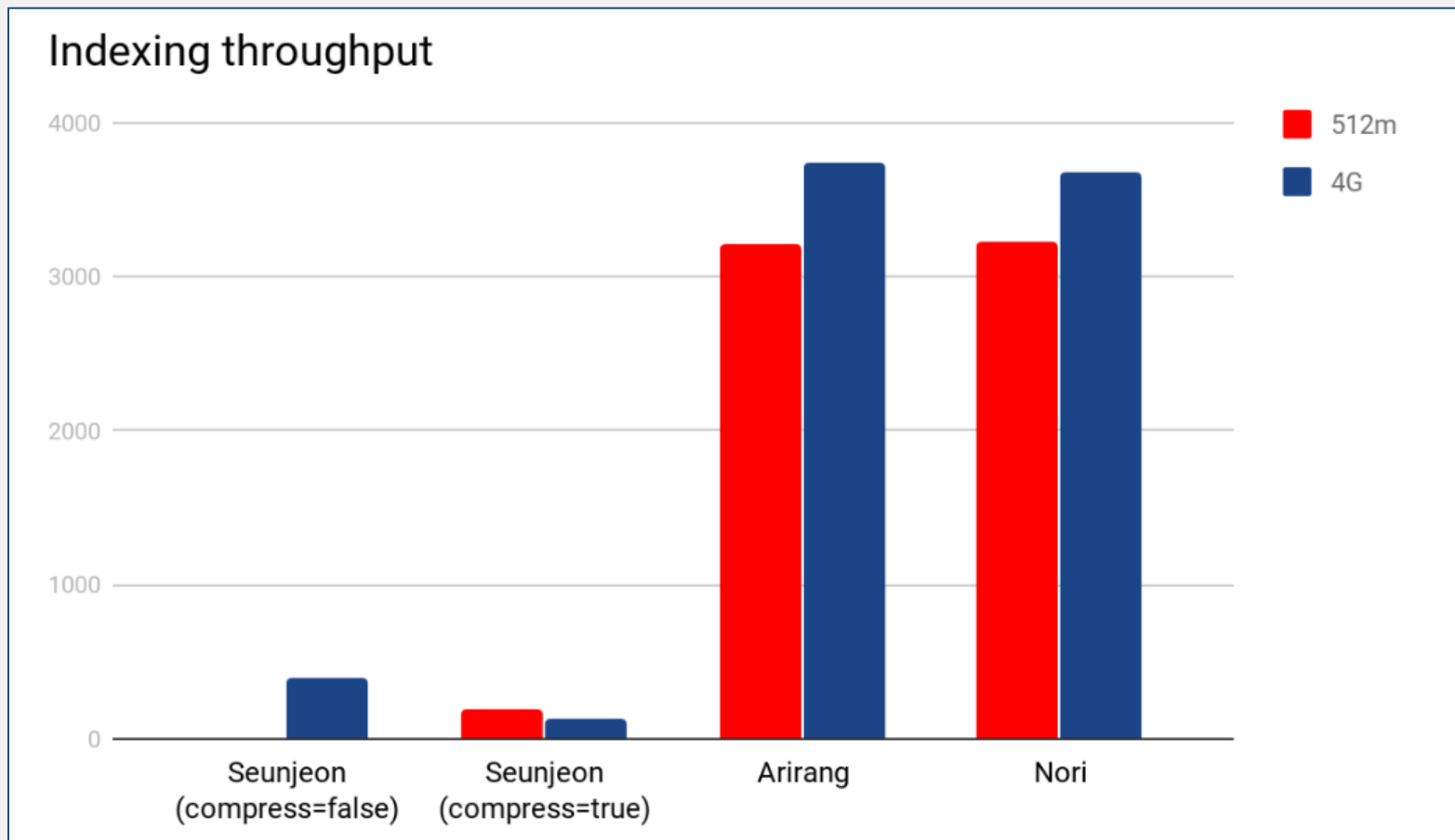
인간(명사)이(주격조사) 문명(명사)의(관형조사) 이기(명사)를(목적격조사)  
사용(명사)하(동사형파생접미사) ㄹ(관형사형전성어미)때(명사)에(부사격조사)  
그것(명사)이(주격조사) 인간(명사) 자신(명사)을(목적격조사) 위해(동사)이(연결어미)  
지혜(명사)있(동사)게(연결어미) 사용(명사)되(동사형파생접미사)어야(연결어미) 하(동사) ㄴ(관형사형전성어미)  
것(의존명사)이(공정지정사)다(어말어미).(마침표)

```
"analysis" : {
  "filter" : {
    "nori_posfilter" : {
      "type" : "nori_part_of_speech",
      "stoptags" : [
        "E",
        "IC",
        "J",
        "MAG",
        "MM",
        "NA",
        "NR",
        "SC",
        "SE",
        "SF",
        "SP",
        "SSC",
        "SSO",
        "SY",
        "VA",
        "VCN",
        "VCP",
        "VSV",
        "V",
        "VX",
        "XPN",
        "XR",
        "XSA",
        "XSN",
        "XSV",
        "NP"
      ]
    }
  },
  "tokenizer" : {
    "nori_analyzer" : {
      "filter" : [
        "nori_posfilter"
      ],
      "type" : "custom",
      "tokenizer" : "korean_nori_tokenizer"
    },
    "korean_nori_tokenizer" : {
      "type" : "nori_tokenizer",
      "user_dictionary" : "user_dictionary.txt",
      "decompound_mode" : "none"
    }
  }
},
```



## 01 Analyzer

### *Morpheme Analyzer : Mecab VS Nori*





# 아파트 데이터 분석

## By Elastic

---

# 실습환경과 Demo

## 01 참고 문서

1. [Elastic Cloud 환경 구축](#)
2. [N사 시세 정보 import](#)
3. [데이터 시각화](#)





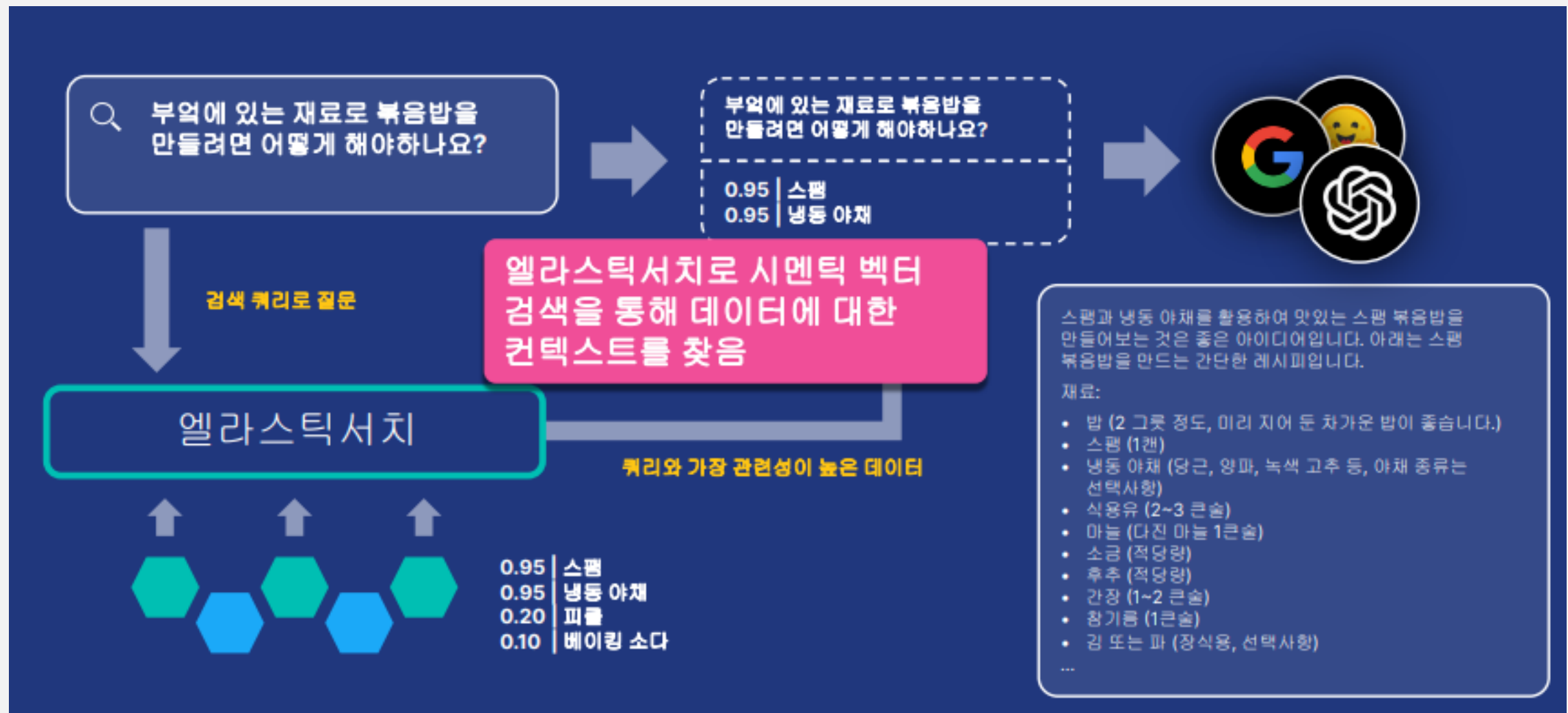
# 의미 기반 챗봇 만들기

## By Elastic

---

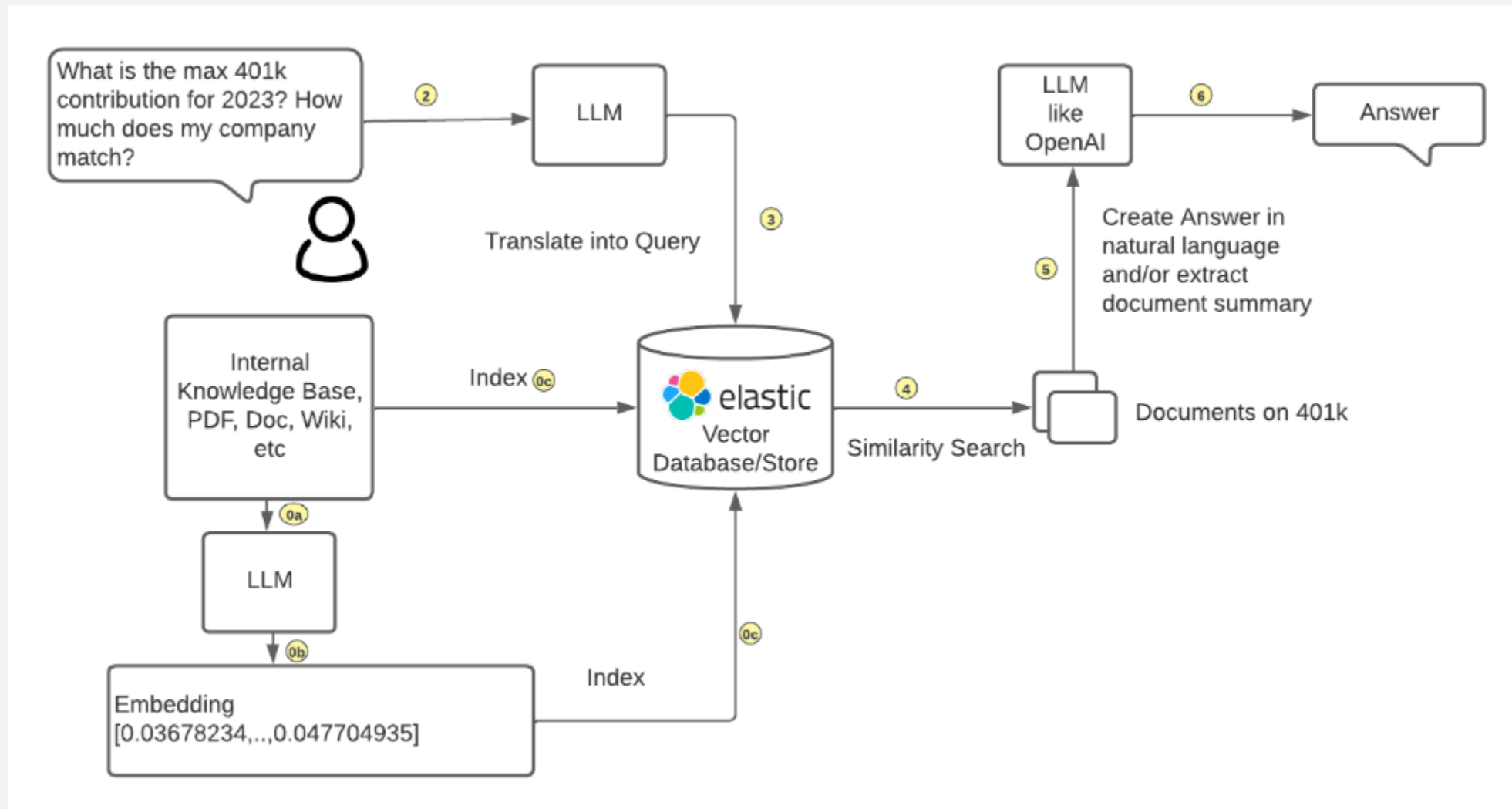
# 💡 실습환경과 Demo

## 01 Vector DB = Elasticsearch



# 💡 실습환경과 Demo

## 01 Vector DB = Elasticsearch





## 실습환경과 Demo

### 02 Embedding Model = Wikipedia by OpenAI

#### Download the dataset

We will use this [Simple English Wikipedia dataset hosted by OpenAI](#) (~700MB zipped, 1.7GB CSV file) that includes vector embeddings. These are the columns of interest, where `content_vector` is a [vector embedding](#) with [1536 elements \(dimensions\)](#), generated using OpenAI's `text-embedding-ada-002` model:

id	url	title	text	content_vector
1	<a href="https://simple.wikipedia.org/wiki/April">https://simple.wikipedia.org/wiki/April</a>	April	"April is the fourth month of the year..."	[-0.011034, -0.013401, ..., -0.009095]

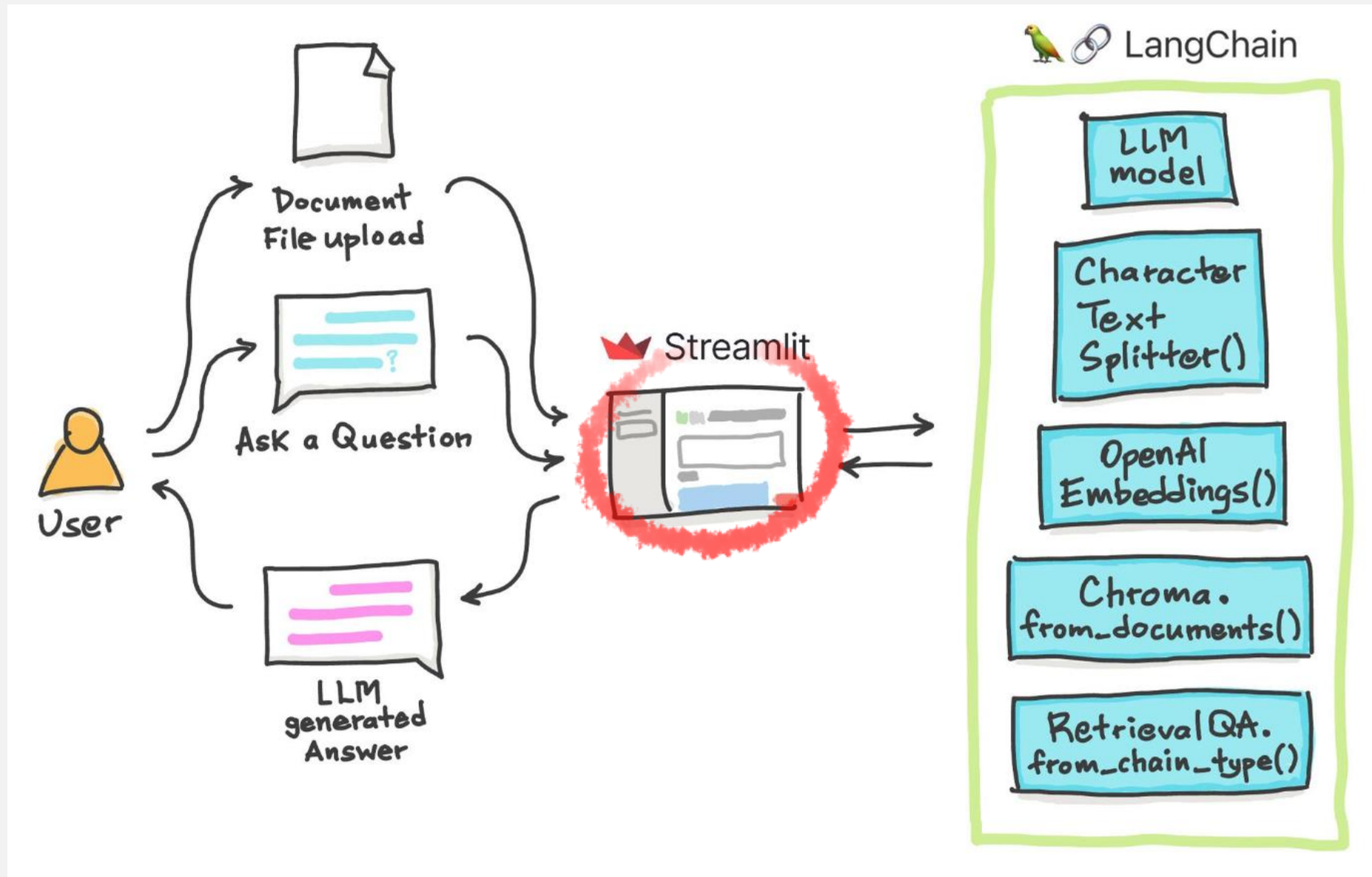
If you haven't already, make sure to download the dataset and unzip the file. You should end up with `vector_database_wikipedia_articles_embedded.csv` in your working directory. The records are mostly (but not strictly) sorted by title.

[Download Wikipedia dataset ZIP](#)

<https://weaviate.io/developers/weaviate/tutorials/wikipedia>

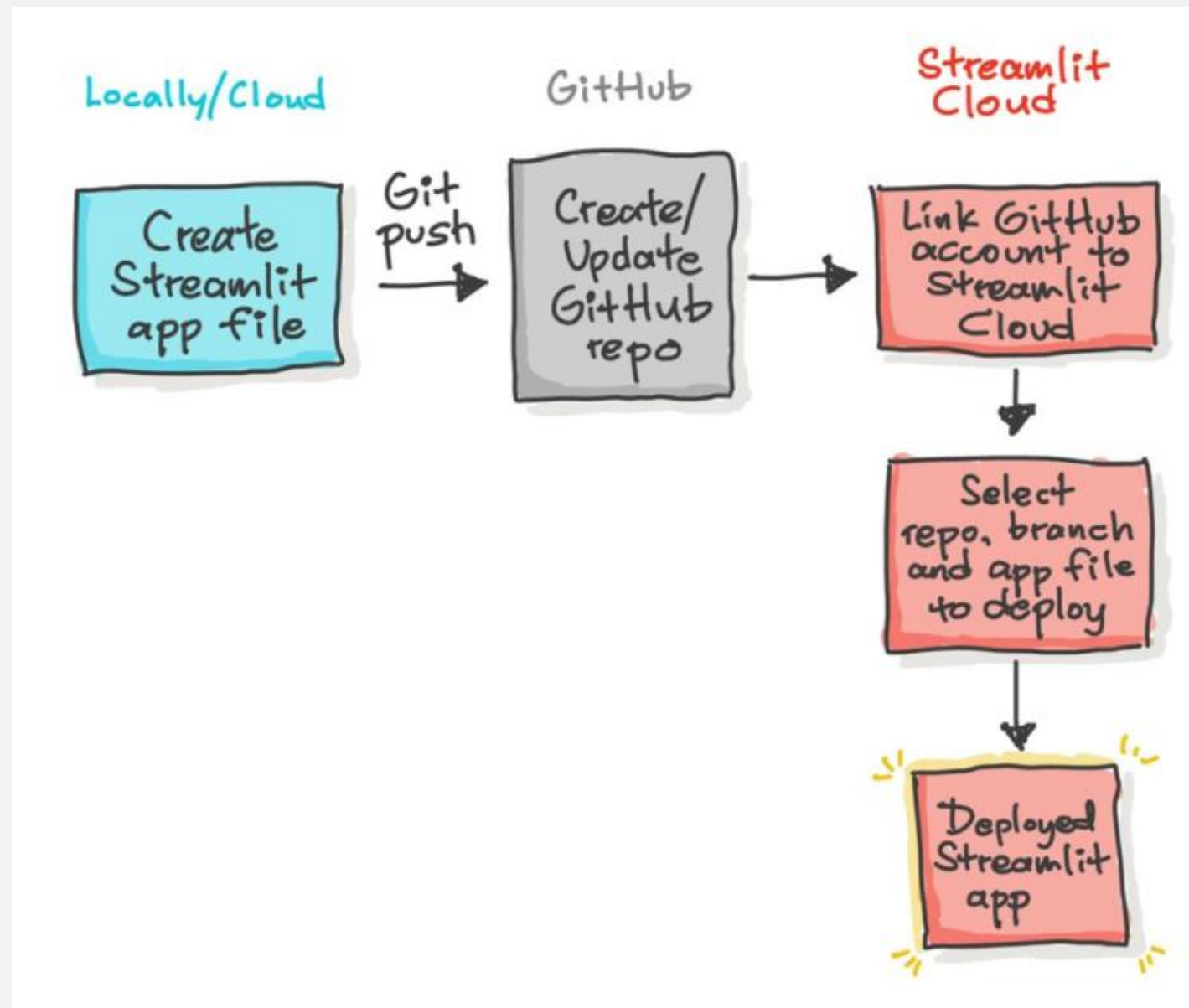
# 💡 실습환경과 Demo

## 03 Streamlit과 LangChain (with Python)



## 💡 실습환경과 Demo

### 03 Streamlit과 LangChain (with Python)



# 💡 실습환경과 Demo

## 04 Demo

### 영문 위키피디아 이용한

### 한글로 답변하는 AI

**부제 : Semantic search and Retrieval augmented generation using Elasticsearch and OpenAI**

명확한 정답이 있는 답변을 잘합니다. 좋은 질문 예 :

- How big is the Atlantic ocean?
- 대한민국의 수도는?
- 이순신의 출생년도는?
- 도요타에서 가장 많이 팔리는 차는?

#### 장점

- 전문 영어 문서(예: 해외 기업투자분석리포트 등)를 대상으로 한글로 검색하고 답변 받기 용이합니다.
- 전통적인 검색방식(키워드 매칭)을 탈피하여 의미가 유사한 문서를 찾고 이를 통한 답변 가능
- 일반 검색에 비하여 2~3 단어의 이상의 문장형 검색 질의에 답변 잘함
- 다국어(multi-lingual AI) 검색 가능(단, 학습문서가 영어라서 영어로 질의해야 답변 잘해요.)
- LLM(Large Language Model) AI의 답변이 아닌 내가 원하는 문서를 통해서 답변 가능하여 할루시네이션(Hallucination)을 최소화
- Private한 사내 문서나 특정 도메인에 특화된 답변이 가능합니다.(사내 문서가 외부로 노출되지 않음)

#### 단점

- 데이터가 충분하지 않아 다양한 질문에 정확한 답을 못할 수 있음.
- 유사도 검색의 특성상 적절하지 않은 검색결과가 있다면 잘못된 답변을 할 수 있음

#### 데이터 출처

- [https://cdn.openai.com/API/examples/data/vector\\_database\\_wikipedia\\_articles\\_embedded.zip](https://cdn.openai.com/API/examples/data/vector_database_wikipedia_articles_embedded.zip)
- 데이터 설명 : <https://weaviate.io/developers/weaviate/tutorials/wikipedia>
- 데이터 건수 : 25,000건 (데이터의 양을 늘리면, 다양한 질문에 대한 답변 가능)

Prompt

이순신의 출생년도는?

Submit

이순신의 출생년도는 1536년 입니다.

### 검색해본 위키 문서 List

ID: 22421 Title: Yi I Url: <https://simple.wikipedia.org/wiki/Yi%20I> Score: 0.9028312

ID: 40687 Title: 1415 Url: <https://simple.wikipedia.org/wiki/1415> Score: 0.8957019

ID: 39133 Title: 1551 Url: <https://simple.wikipedia.org/wiki/1551> Score: 0.8900757

ID: 15081 Title: 1232 Url: <https://simple.wikipedia.org/wiki/1232> Score: 0.8894432

ID: 20088 Title: 509 Url: <https://simple.wikipedia.org/wiki/509> Score: 0.88554895

ID: 4678 Title: Kim Jong-il Url: <https://simple.wikipedia.org/wiki/Kim%20Jong-il> Score: 0.8849724

ID: 46365 Title: Yun Poson Url: <https://simple.wikipedia.org/wiki/Yun%20Poson> Score: 0.88478947

ID: 46358 Title: Kim Young-sam Url: <https://simple.wikipedia.org/wiki/Kim%20Young-sam> Score: 0.8843397

<https://wiki-ai-search.streamlit.app/>

<https://colab.research.google.com/drive/12bxT-pnxKkfPEut4SDLJvCvfh0MEomW7?usp=sharing>





Q&A

감사합니다