

05

강

인공지능

# 지식과 인공지능

컴퓨터과학과 이병래교수

# 학습목차

1 지식표현의 개념

2 지식표현 방법의 종류

3 전문가 시스템

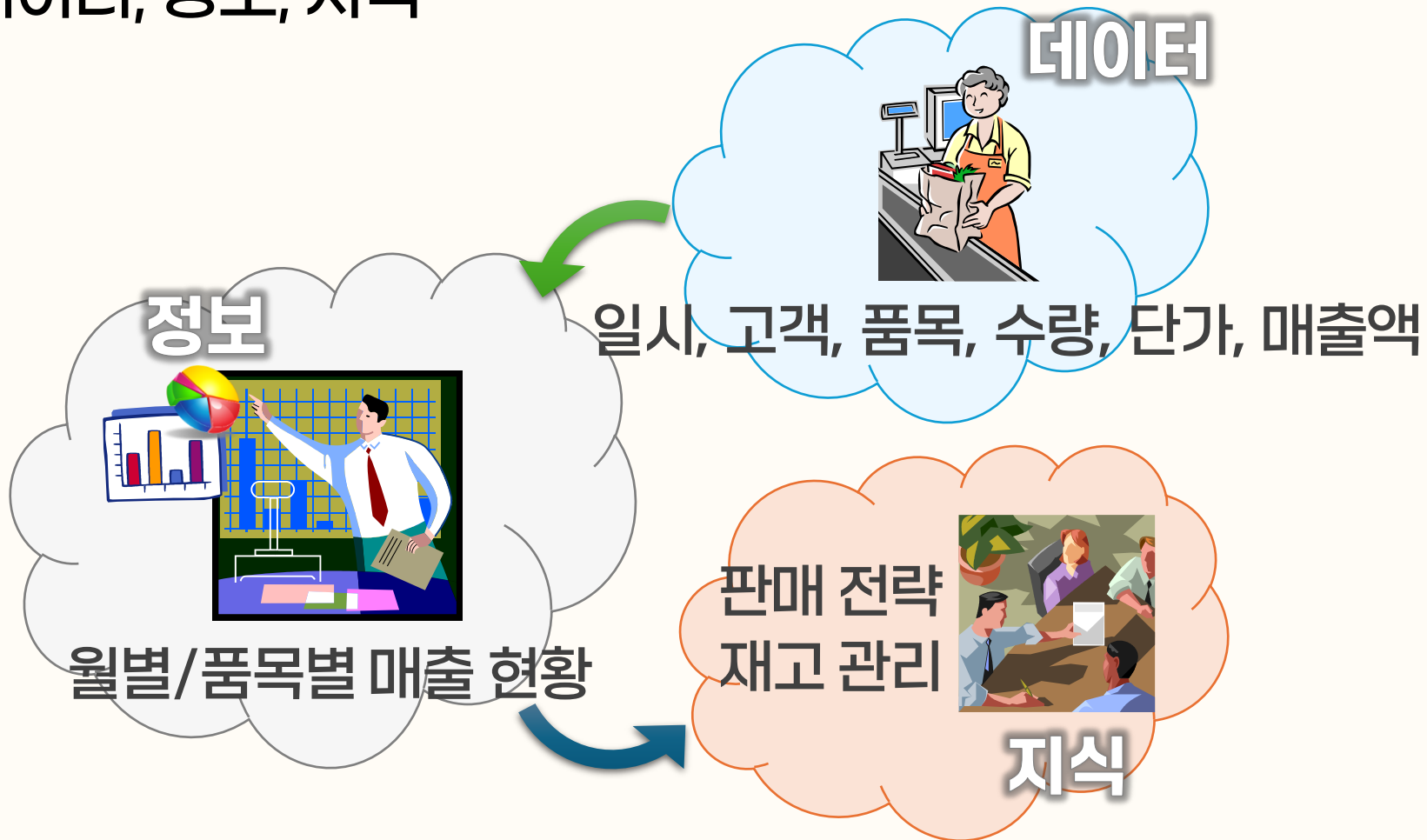




# 지식표현의 개념

# 1. 지식이란 무엇인가?

## ■ 데이터, 정보, 지식



# 1. 지식이란 무엇인가?

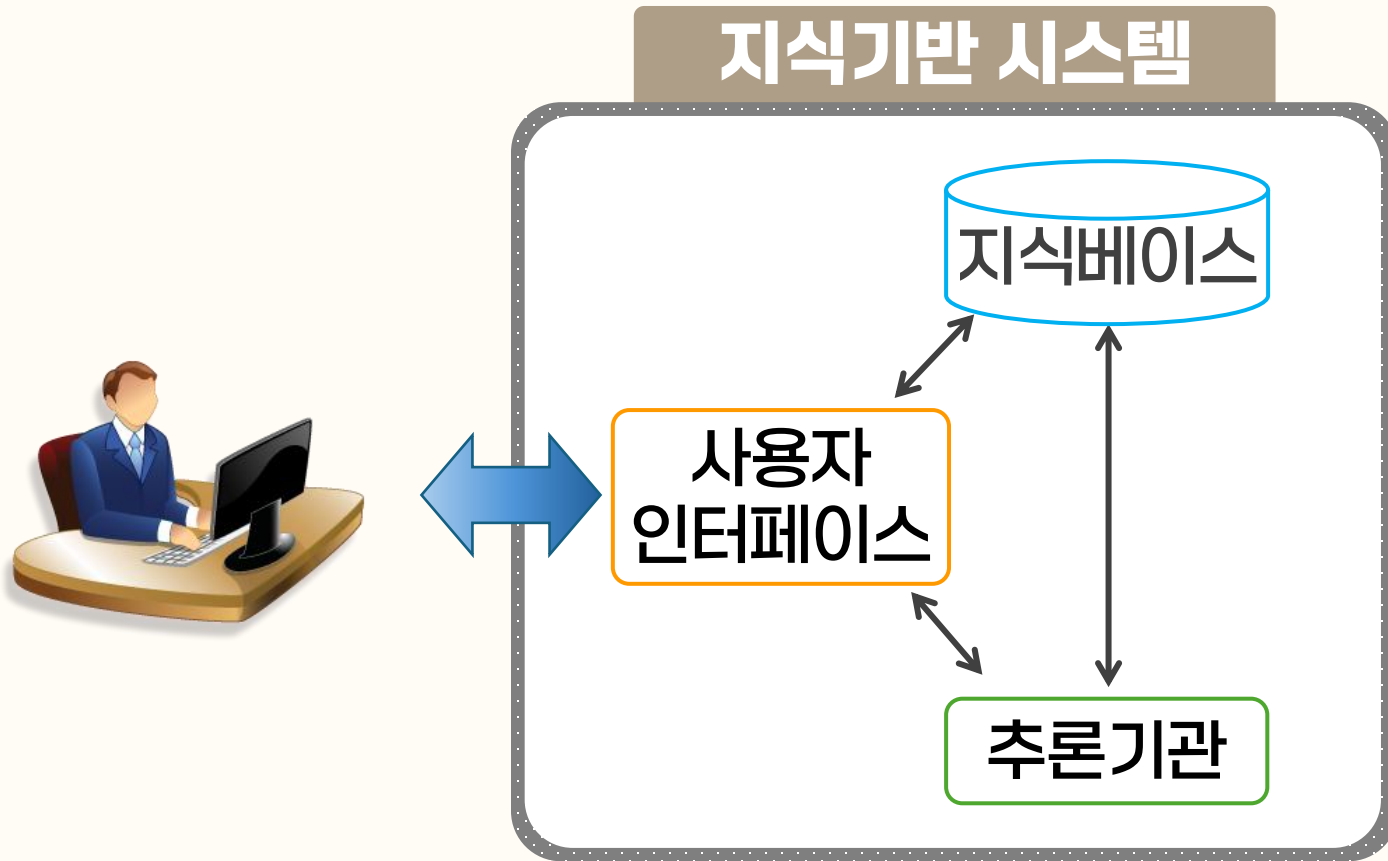
---

## ■ 데이터, 정보, 지식

- 지능적 문제풀이는 궁극적으로 그 문제와 관련된 지식들을 어떻게 저장하고 이용할 것인가에 달려 있음
  - ⇒ 지식공학(knowledge engineering)
- 특정한 문제 분야의 지식을 쉽게 접근할 수 있는 형태로 컴퓨터 내에 체계적으로 축적하여 사용
  - ⇒ 지식기반 시스템(knowledge-based systems)

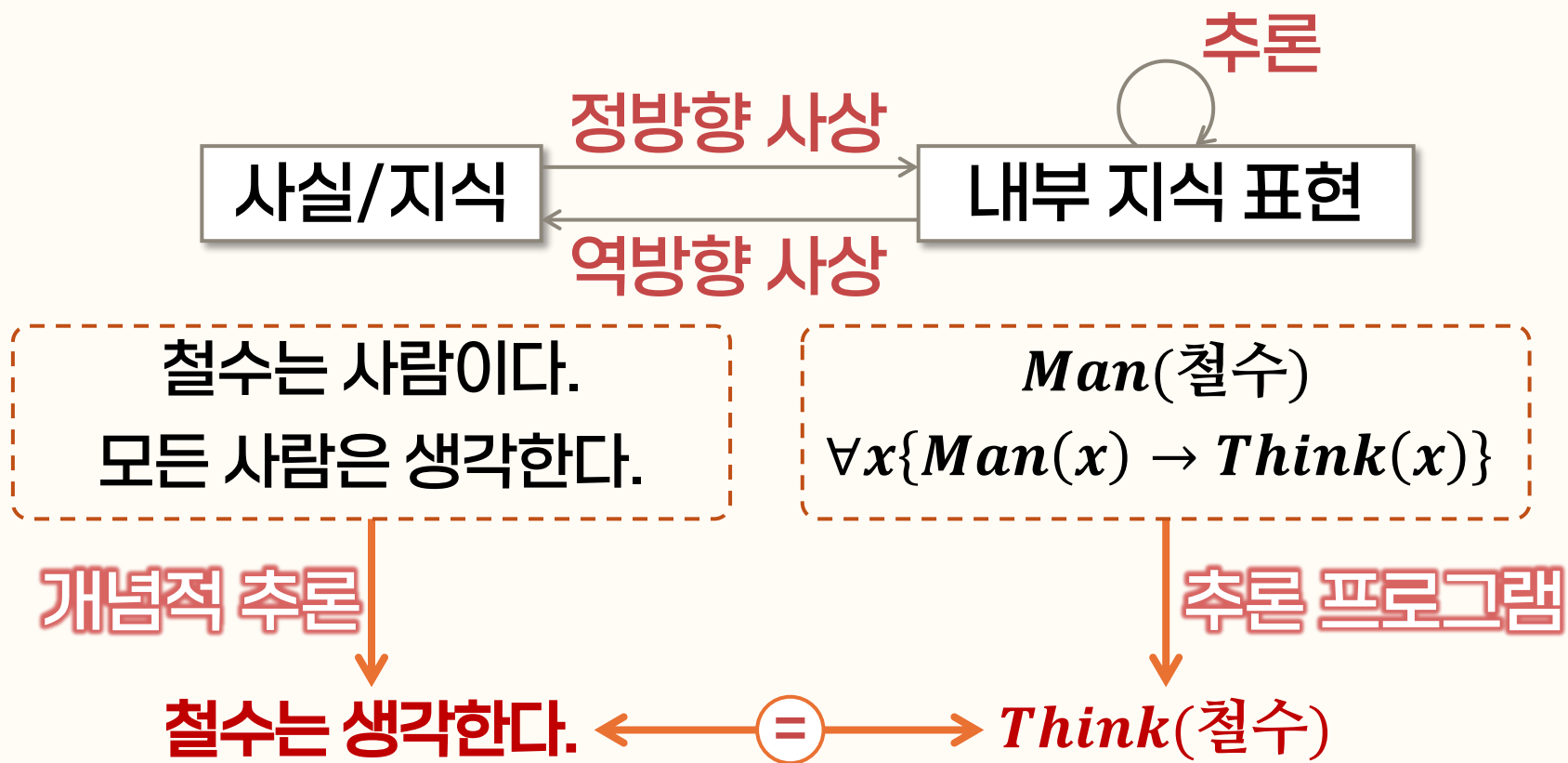
# 1. 지식이란 무엇인가?

## ■ 지식기반 시스템의 구성



## 2. 지식의 표현

### ■ 지식과 내부 지식표현 사이의 사상



## 2. 지식의 표현

### ❑ 지식표현 방법이 갖추어야 할 요건

- ① **표현방법의 적합성**: 문제영역에서 필요로 하는 모든 지식을 정확하게 표현할 수 있어야 함
  - ➡ 표현하고자 하는 지식의 실세계의 의미를 최대한 수용
- ② **추론의 적합성**: 표현된 지식을 이용하여 추론을 할 수 있는 메커니즘이 존재해야 함
- ③ **추론의 효율성**: 추론과정이 효율적으로 진행될 수 있어야 함
  - ➡ 추론에 유용한 부가정보를 지식의 표현구조에 적절히 포함시킬 수 있는 능력
- ④ **지식 획득 능력**: 새로운 지식을 쉽게 습득할 수 있어야 함



## 2. 지식의 표현

### ■ 지식의 형태

#### 절차적 지식

#### 선언적 지식

#### 신경망

- 어떠한 경우에 무엇을 어떻게 할 것인가에 대한 지식
- 프로그래밍 언어로 작성된 명령어의 집합
  - ➔ 지식 사용에 대한 제어 정보는 지식 자체에 내포되어 있음
- 추론의 적합성 면이나 지식 획득의 효율성 면에서 낮은 평가

## 2. 지식의 표현

### ■ 지식의 형태

절차적 지식

선언적 지식

신경망

- 상호 독립적, 단편적인 지식들을 나열한 형태의 지식  
⇒ 정적인 지식
- **추론기관**이라는 프로그램이 별도로 존재하며, 이 프로그램에 의해 지식이 추론에 사용됨
- 개별적으로 지식을 편집, 획득, 검색하는 것이 절차적 지식에 비해 용이함

## 2. 지식의 표현

### ■ 지식의 형태

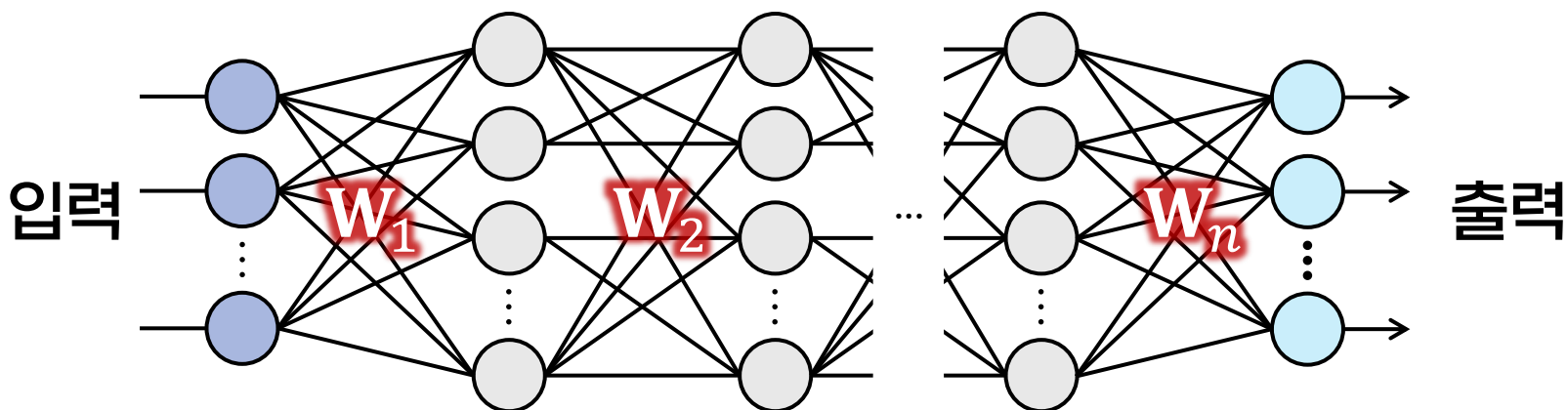
절차적 지식

선언적 지식

신경망

- 뉴런의 출력이 다른 뉴런으로 전달되는 연결의 가중치의 형태로 분산되어 저장

➡ 개별 지식이 잘 드러나지 않음





## 지식표현 방법의 종류

# 1. 논리

## ■ 형식논리학(formal logic)

- 기호와 논리연산자를 이용하여 논리적 연역 체계를 표현
  - 기호: 명제, 객체, 관계 등을 표현
  - 논리연산자

명제		$X \wedge Y$	$X \vee Y$	$\sim X$	$X \rightarrow Y$	$X \leftrightarrow Y$
$X$	$Y$					
$T$	$T$	$T$	$T$	$F$	$T$	$T$
$T$	$F$	$F$	$T$	$F$	$F$	$F$
$F$	$T$	$F$	$T$	$T$	$T$	$F$
$F$	$F$	$F$	$F$	$T$	$T$	$T$

# 1. 논리

## ■ 명제논리를 이용한 지식 표현

- 명제를 기호로 표현

- 명제: 참 또는 거짓을 명백히 판단할 수 있는 문장

'이것은 유리이다.' → *GLASS*

- 논리연산자를 이용하여 복합명제 표현

- $\wedge$ (AND),  $\vee$ (OR),  $\sim$ (NOT),  $\rightarrow$ (조건명제),  $\leftrightarrow$ (동치) 등

'유리는 잘 깨진다.' → *GLASS*  $\rightarrow$  *FRAGILE*

# 1. 논리

## ■ 술어논리를 이용한 지식 표현

- 명제를 술어와 객체로 분리하여 표현

‘고양이는 포유류이다.’  $\Rightarrow$  *Mammal*(CAT)

‘철수는 사람이다.’  $\Rightarrow$  *Man*(철수)

- 변수를 사용할 수 있음

‘모든 사람은 생각한다.’  $\Rightarrow \forall x \{Man(x) \rightarrow Think(x)\}$

# 1. 논리

## ■ 추론

### • 연역법칙

$$\left. \begin{array}{l} X \\ X \rightarrow Y \end{array} \right\} \Rightarrow \text{결론: } Y$$

$$\left. \begin{array}{l} \textit{Man}(\text{철수}) \\ \forall x \{ \textit{Man}(x) \rightarrow \textit{Think}(x) \} \end{array} \right\} \Rightarrow \text{결론: } \textit{Think}(\text{철수})$$



## 2. 규칙

### ■ 규칙(rule)이란?

- 주어진 상황을 위한 권고·지시·전략을 나타내는 정형화된 표현방법
- 'IF ㉠ THEN ㉡' 형태의 표현
  - ㉠ : 가정, 전제조건, LHS
  - ㉡ : 결론, 실행할 동작, RHS
- 지식 베이스: 현재의 상황을 나타내는 사실들과 이에 대해 적용할 규칙들로 구성됨
- 추론기관: 현재 상태에 의해 만족되는 규칙을 선택하여 실행함

## 2. 규칙

### ■ 추론의 종류

연역법(deduction)

유도법(abduction)

귀납법(induction)

- 'IF A THEN B'라는 규칙과 'A'라는 사실로부터 'B'를 결론으로 제시하는 추론 방법

*IF Mother( $x$ ) THEN Woman( $x$ )*  
*Mother(Marie)* } *Woman(Marie)*

- 연역에 의한 추론은 항상 옳음

## 2. 규칙

### ■ 추론의 종류

연역법(deduction)

유도법(abduction)

귀납법(induction)

- 'IF A THEN B'라는 규칙과 'B'라는 사실로부터 'A'를 결론으로 제시하는 추론 방법

**IF** *Mother*(*x*) **THEN** *Woman*(*x*)  
*Woman*(Marie) } *Mother*(Marie)

- 추론 결과가 항상 옳지는 않음

➡ 유사추론

## 2. 규칙

### ■ 추론의 종류

연역법(deduction)

유도법(abduction)

귀납법(induction)

- 관측된 사실로부터 새로운 법칙을 만들어 내는 것

IF *Sparrow*( $x$ ) THEN *CanFly*( $x$ )

IF *Dove*( $x$ ) THEN *CanFly*( $x$ )

IF *Eagle*( $x$ ) THEN *CanFly*( $x$ )

⇒ IF *Bird*( $x$ ) THEN *CanFly*( $x$ )

- 학습과 관련된 추론 방법

## 2. 규칙

### 추론 방향

#### 전방향 추론

#### 후방향 추론

- 주어진 사실들로부터 만족되는 규칙을 규칙의 조건부와 정합에 의해 선택
- 선택된 규칙의 결론부의 내용을 실행하거나 사실에 추가

**IF** *Mother*(*x*) **THEN** *Woman*(*x*)  
*Mother*(Marie) } *Woman*(Marie)

## 2. 규칙

### 추론 방향

#### 전방향 추론

#### 후방향 추론

- 목표로 하는 결론이 현재 상태 또는 알려진 사실들로부터 유도해 낼 수 있는가를 알아내고자 함
- 결론부로부터 가정부 방향으로 진행하는 추론

? *Woman(Marie)*

*IF Mother(x) THEN Woman(x)*

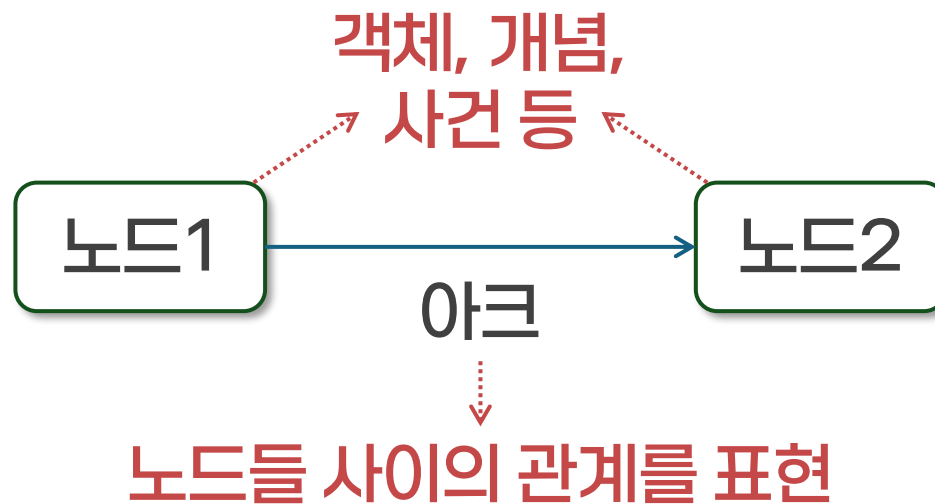
*Mother(Marie)*

} *Woman(Marie)*는 참

### 3. 시맨틱 네트와 프레임

#### ■ 시맨틱 네트(Semantic Net)

- 개념 사이의 관계를 표현하는 지식 표현 방법
  - 노드와 아크를 이용하여 방향성 그래프로 표현



### 3. 시맨틱 네트워크와 프레임

#### ■ 시맨틱 네트워크(Semantic Net)

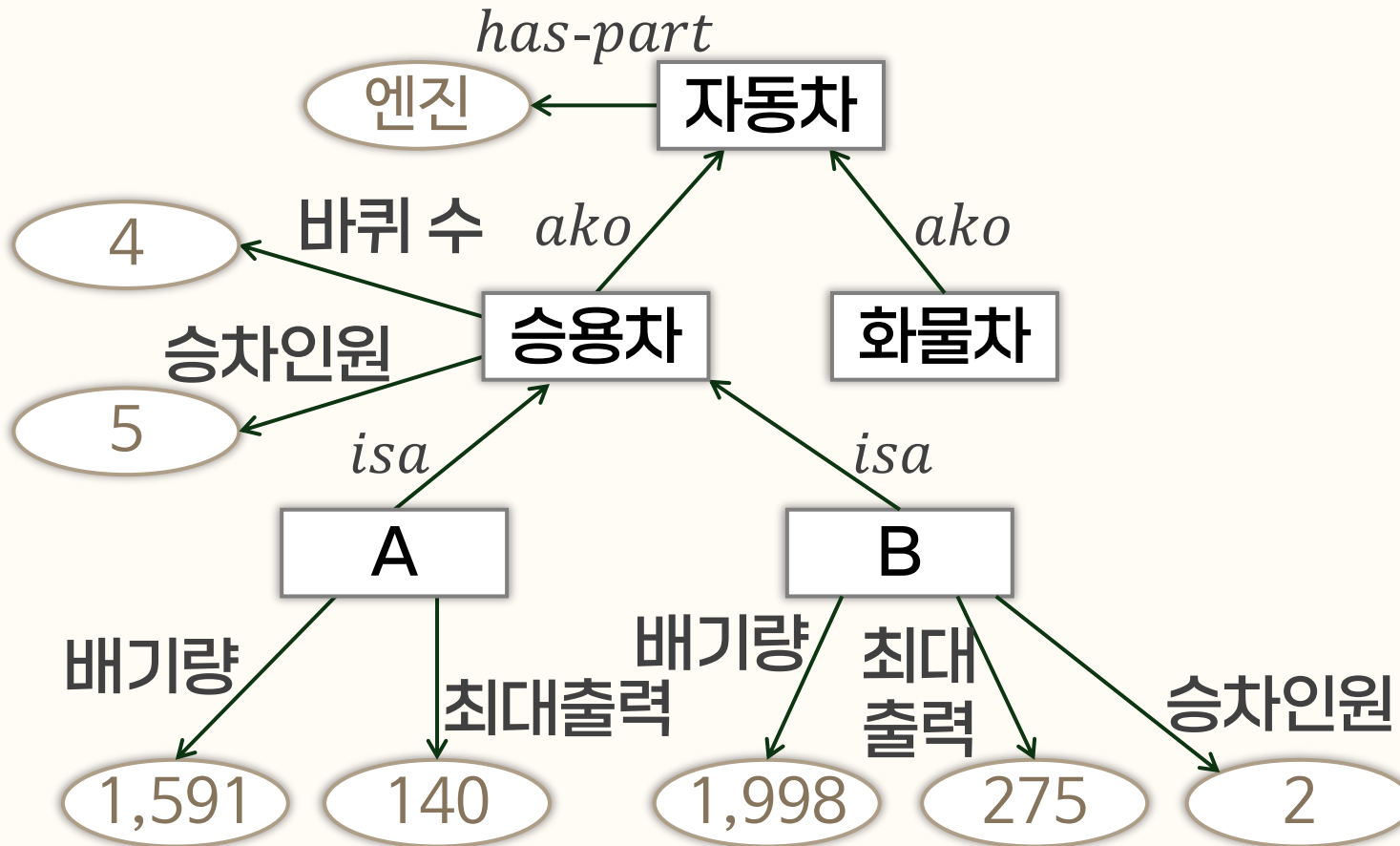
- 대표적인 아크 속성의 예

속성	의미	예
<i>ako</i>	상위 개념의 하위 클래스 ( <i>a kind of</i> )	
<i>isa</i>	어떤 클래스의 하나의 사례 ( <i>is a</i> )	
<i>has-part</i>	어떤 객체의 부속품	



### 3. 시맨틱 네트워크와 프레임

#### ■ 시맨틱 네트워크를 이용한 지식표현 예



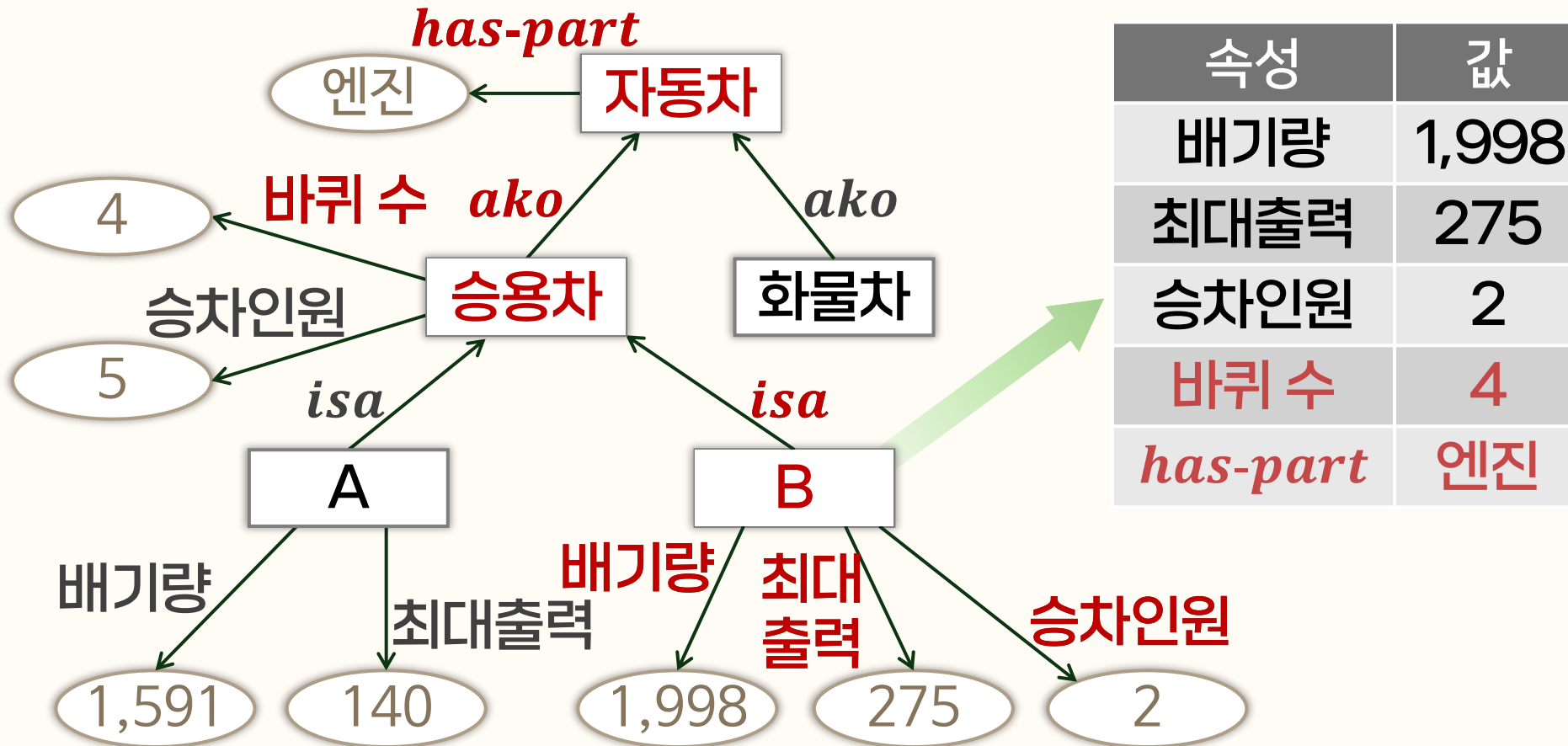
### 3. 시맨틱 네트워크와 프레임

#### ■ 시맨틱 네트워크와 특성상속(property inheritance)

- 상위 클래스의 속성과 값을 하위 클래스 또는 사례가 이어받도록 하는 추론 형태
- 상위 개념의 지식을 하위 개념이 공유하는 중앙집중적 방법의 지식 표현
- 특성 상속이 이루어지는 아크: *isa* 아크 및 *ako* 아크
- 상위 개념은 일반적인 속성을, 하위 개념은 일반적 속성값과 다르거나 고유한 속성을 연결

### 3. 시맨틱 네트워크와 프레임

#### ■ 시맨틱 네트워크와 특성상속(property inheritance)

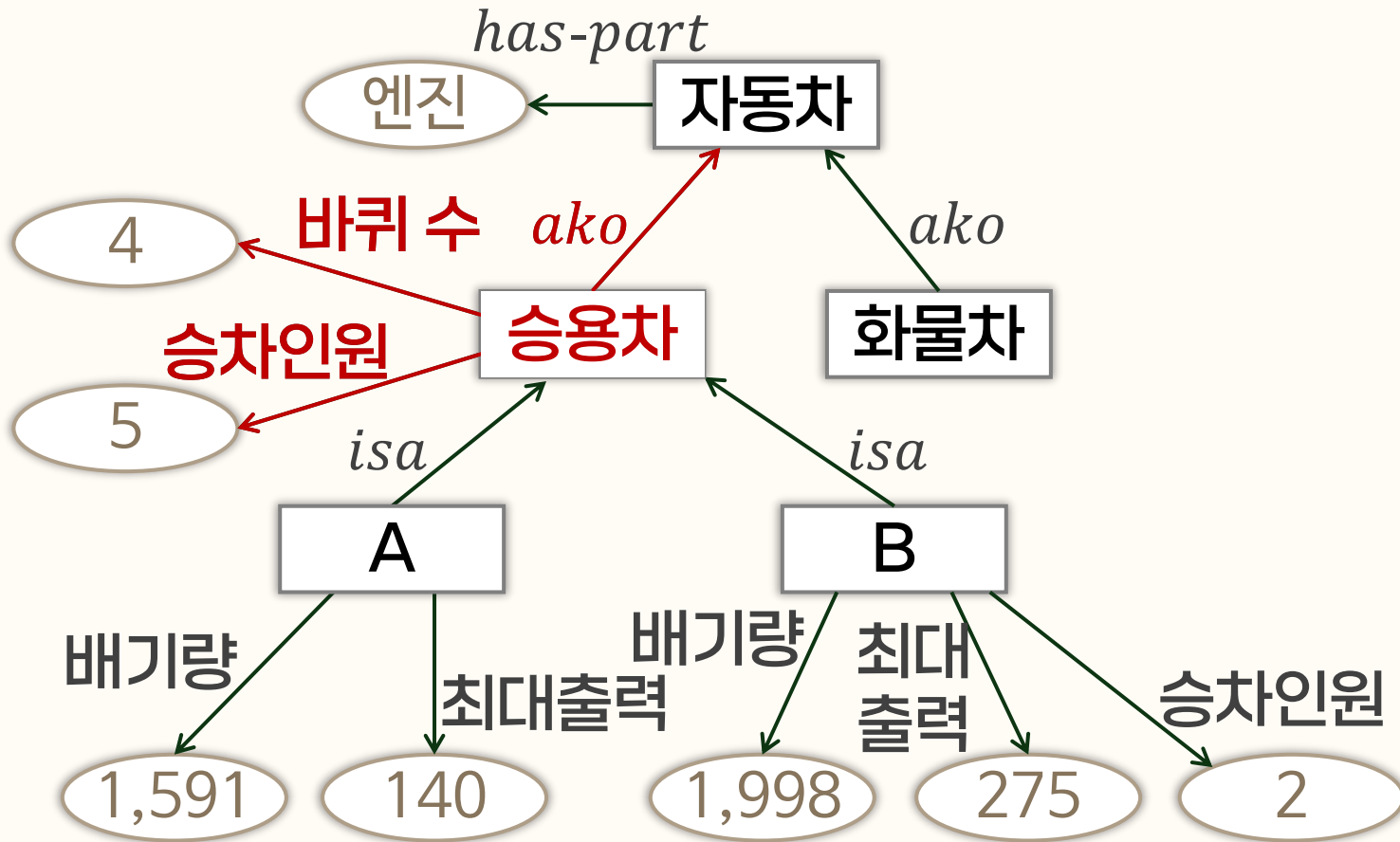


### 3. 시맨틱 네트와 프레임

- 시맨틱 네트와 특성상속(property inheritance)
  - 특성상속을 이용한 중앙집중식 지식 공유의 장점
    - ① 지식을 구성하기 쉬움
    - ② 표현된 지식에 잘못이 있을 경우 이를 쉽게 수정할 수 있음
    - ③ 시간이 흐름에 따라 최신의 지식을 유지하기가 쉬움
    - ④ 지식의 분배가 자동적으로 이루어짐

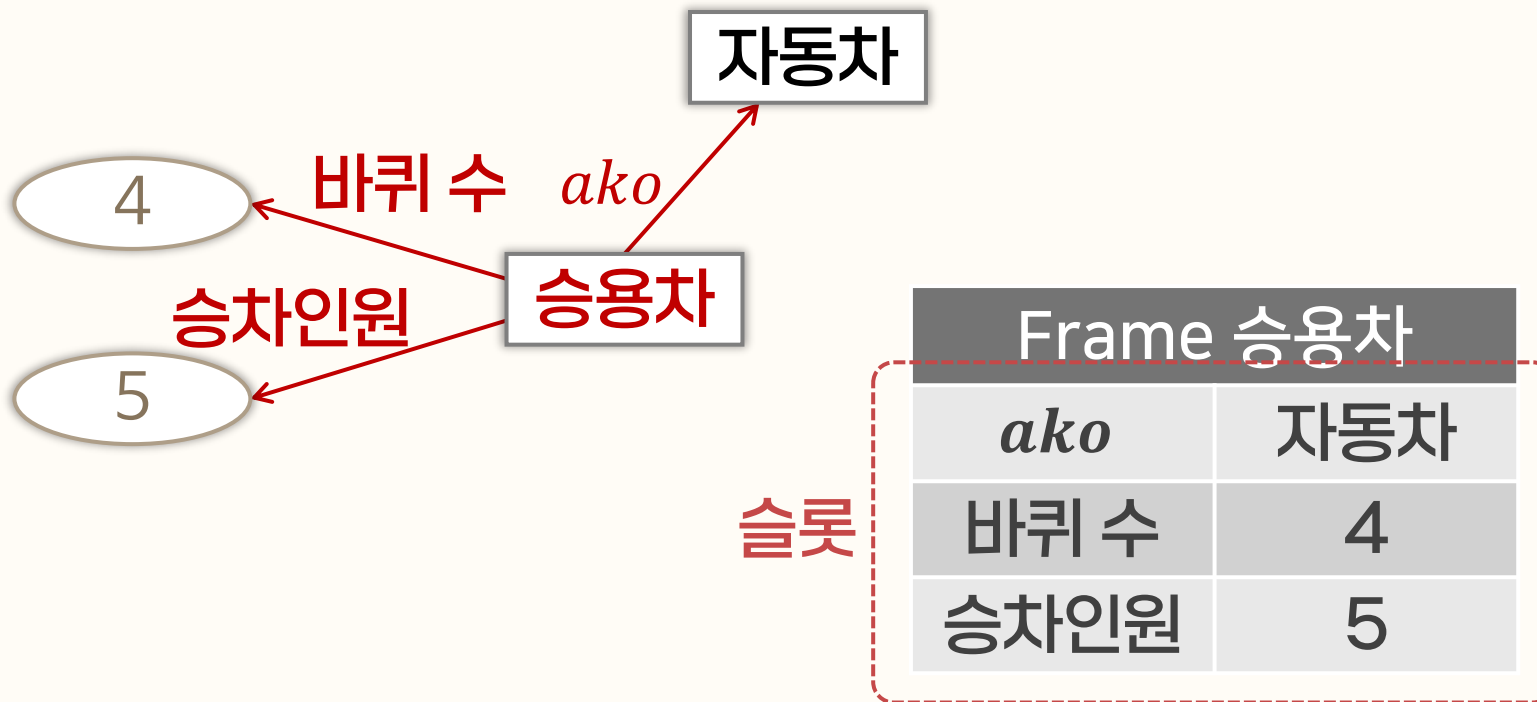
### 3. 시맨틱 네트와 프레임

#### ■ 프레임(Frame)



### 3. 시맨틱 네트와 프레임

#### ■ 프레임(Frame)



### 3. 시맨틱 네트와 프레임

#### ■ 프레임(Frame)

- 표현 대상 개념의 속성을 나타내는 슬롯들의 집합
- 관련된 프레임들이 상위 개념, 하위 개념, 사례(instance)로 분류되어 연결
  - ➡ 클래스, 부 클래스, 사례 프레임
- 슬롯의 값에 기본값을 지정할 수 있음
- 특성상속을 이용하여 중앙집중 지식 공유
- 부가 프로시저를 통해 절차적 지식을 함께 표현

### 3. 시맨틱 네트와 프레임

#### 프레임의 예

Frame 인간	
<i>ako</i>	포유류
*이동	직립보행
*지능	(기본값 = 100)

Frame 성인남자	
<i>ako</i>	인간
*연령	
*키	(기본값 = 170)
*체중	(기본값 = 65)
*결혼여부	
*배우자	

Frame 홍길동	
<i>instance</i>	성인남자
연령	35
키	175
체중	70
결혼여부	기혼
배우자	이영숙



### 3. 시맨틱 네트와 프레임

---

#### ■ 프레임과 부가 프로시저

- 슬롯의 사용과 관련하여 수행해야 할 동작을 지시하는 프로시저

➡ 절차적 지식을 함께 표현할 수 있음

- 부가 프로시저의 종류

- 필요(if-needed) 프로시저
- 판독(if-read) 프로시저
- 기록(if-written) 프로시저
- 제거(if-removed) 프로시저

### 3. 시맨틱 네트워크와 프레임

#### ■ 프레임과 부가 프로시저

- 예: if-needed 프로시저의 활용

Frame 성인남자	
<i>ako</i>	인간
*연령	
*키	(기본값 = 170)
*체중	(기본값 = 65)
*결혼여부	
*배우자	

if-needed:

```
if 연령 > 35 then
  체중 ← 키 - 100;
else
  체중 ← 키 - 110;
end-if
```

### 3. 시맨틱 네트와 프레임

#### ■ 프레임과 부가 프로시저

- 예: if-written 프로시저의 활용

Frame 성인남자	
<i>ako</i>	인간
*연령	
*키	(기본값 = 170)
*체중	(기본값 = 65)
*결혼여부	
*배우자	

if-written:

if 결혼여부 = 기혼 then

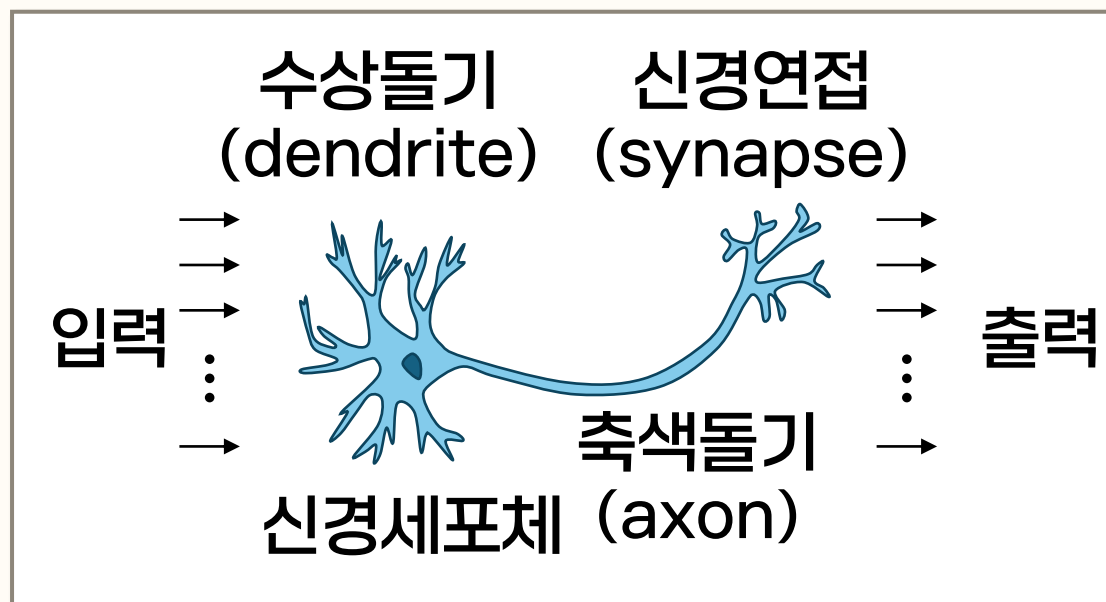
배우자 이름을 질문하여 배우자 슬롯에 넣음;  
해당되는 배우자 프레임에게 배우자 슬롯에  
이 프레임의 이름을 넣으라는 메시지를 보냄;

end-if

## 4. 신경회로망에서의 지식 표현

### ■ 인공 신경회로망(artificial neural network)

- 신경 구조를 모델링하여 지능적 처리에 응용하고자 하는 시도

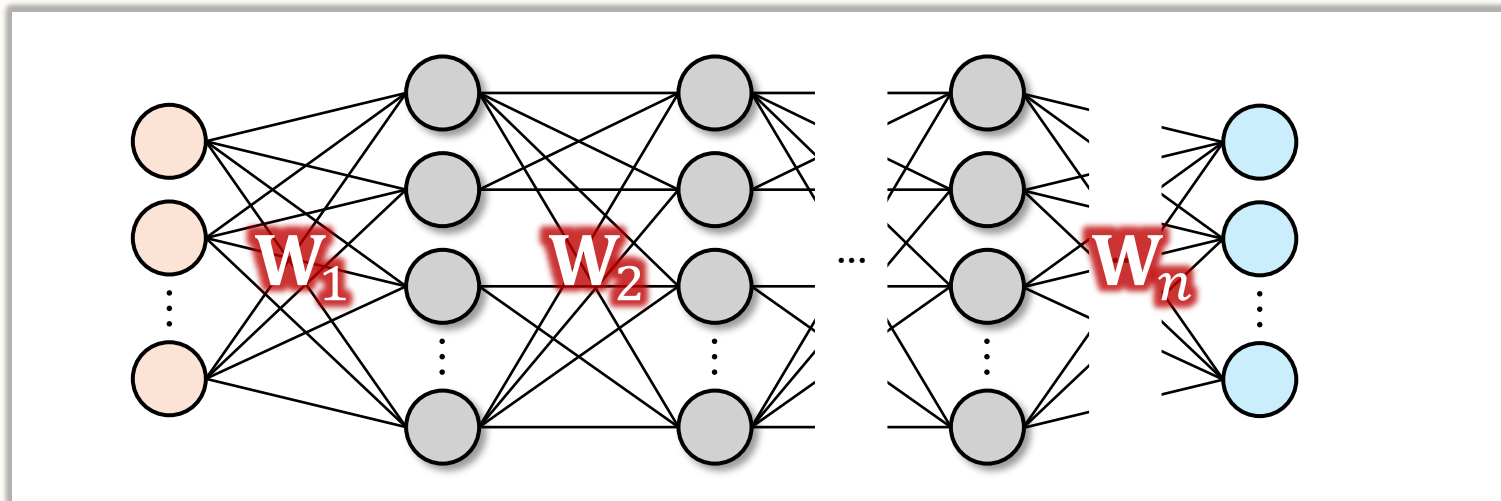


신경세포는 다수의 다른 신경세포와 신경연접을 통해 연결

## 4. 신경회로망에서의 지식 표현

### ■ 인공 신경회로망(artificial neural network)

- 신경 구조를 모델링하여 지능적 처리에 응용하고자 하는 시도



➡ 신경연접의 연결가중치 벡터 형태로 지식을 분산 저장함



## 전문가 시스템

# 1. 전문가 시스템 개요

## ■ 전문가 시스템(expert system)이란?

- 주어진 문제 분야에서 인간 전문가의 문제해결 지식, 전략 등을 시뮬레이션함으로써 문제풀이, 의사결정을 지원하는 지식기반 시스템



### 현장 전문가

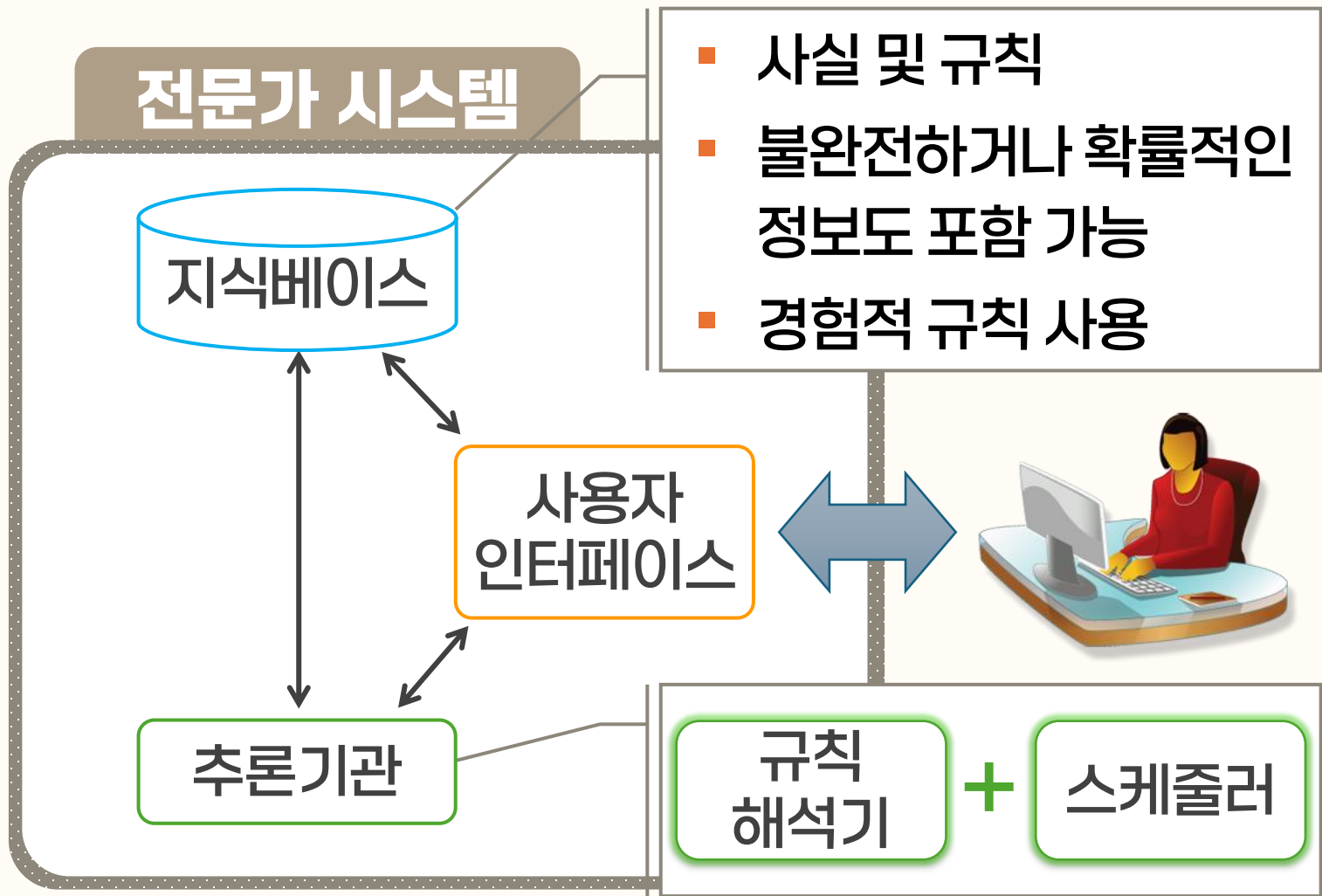
문제 분야에서 사용하는  
규칙, 전략 등을 제공

### 지식공학자



전문가의 지식을  
지식베이스화하여  
전문가 시스템을 만듦

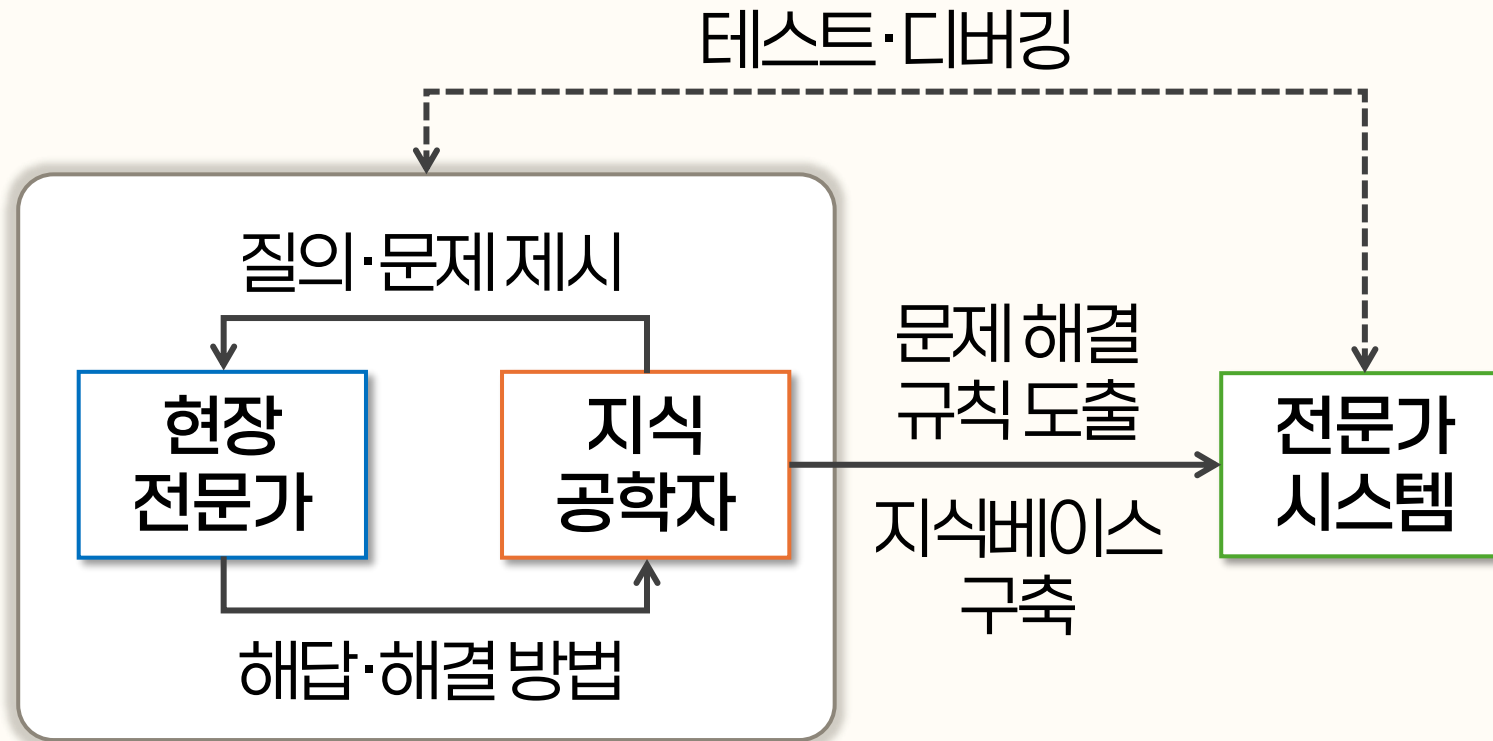
## 2. 전문가 시스템의 구조





### 3. 전문가 시스템의 개발

#### ■ 현장전문가와 지식공학자의 역할



# 정리하기

- ✓ 지식기반 시스템의 핵심 구성요소는 지식베이스와 추론기관이다.
- ✓ 지식표현방법은 지식의 실세계의 의미를 최대한 수용할 수 있어야 하고, 적절한 추론 메커니즘이 존재해야 하며, 추론과정이 효율적으로 진행될 수 있어야 한다. 또한 새로운 지식을 쉽게 습득할 수 있는 표현방법이 좋다.
- ✓ 절차적 지식은 어떠한 경우에 무엇을 어떻게 할 것인가에 대한 지식으로, 지식 사용에 대한 제어 정보는 지식 자체에 내포된다.
- ✓ 선언적 지식은 상호 독립적, 단편적인 지식들을 나열해 놓은 형태로서, 추론은 별도의 추론기관에 의해 이루어진다.

# 정리하기

- ✓ 논리를 이용한 지식표현방법은 명제기호와 논리연산자를 이용한 명제논리나 객체와 술어를 이용하는 술어논리를 이용하며, 기본적인 추론 메커니즘은 연역법칙이다.
- ✓ 규칙을 이용한 지식표현에서는 주어진 상황을 위한 권고·지시·전략을 'IF 조건 THEN 결론' 형태의 규칙으로 표현한다.
- ✓ 시맨틱 네트는 객체, 개념, 사건 등을 표현하는 노드와 노드들 사이의 관계를 표현하는 아크를 이용하여 지식을 표현한다.
- ✓ 프레임은 객체나 클래스를 표현하는 속성들을 나타내는 슬롯들의 집합으로 지식을 표현한다. 부가 프로시저를 이용하여 절차적 지식을 함께 표현할 수 있다.

# 정리하기

- ✔ 시맨틱 네트나 프레임은 특성상속을 이용하여 상위 개념의 지식을 하위 개념이 공유하는 중앙집중적 방법의 지식 표현을 할 수 있다.
- ✔ 인공 신경회로망에서는 신경연접의 연결가중치 벡터 형태로 지식을 분산 저장한다.
- ✔ 전문가 시스템은 초기 인공지능에서 지식기반 시스템의 가능성을 보여준 사례로서, 주어진 문제 분야에서 인간 전문가의 문제해결 지식, 전략 등을 시뮬레이션하도록 설계한다.

06강

다음시간안내 ▶▶▶

# 논리에 의한 지식표현