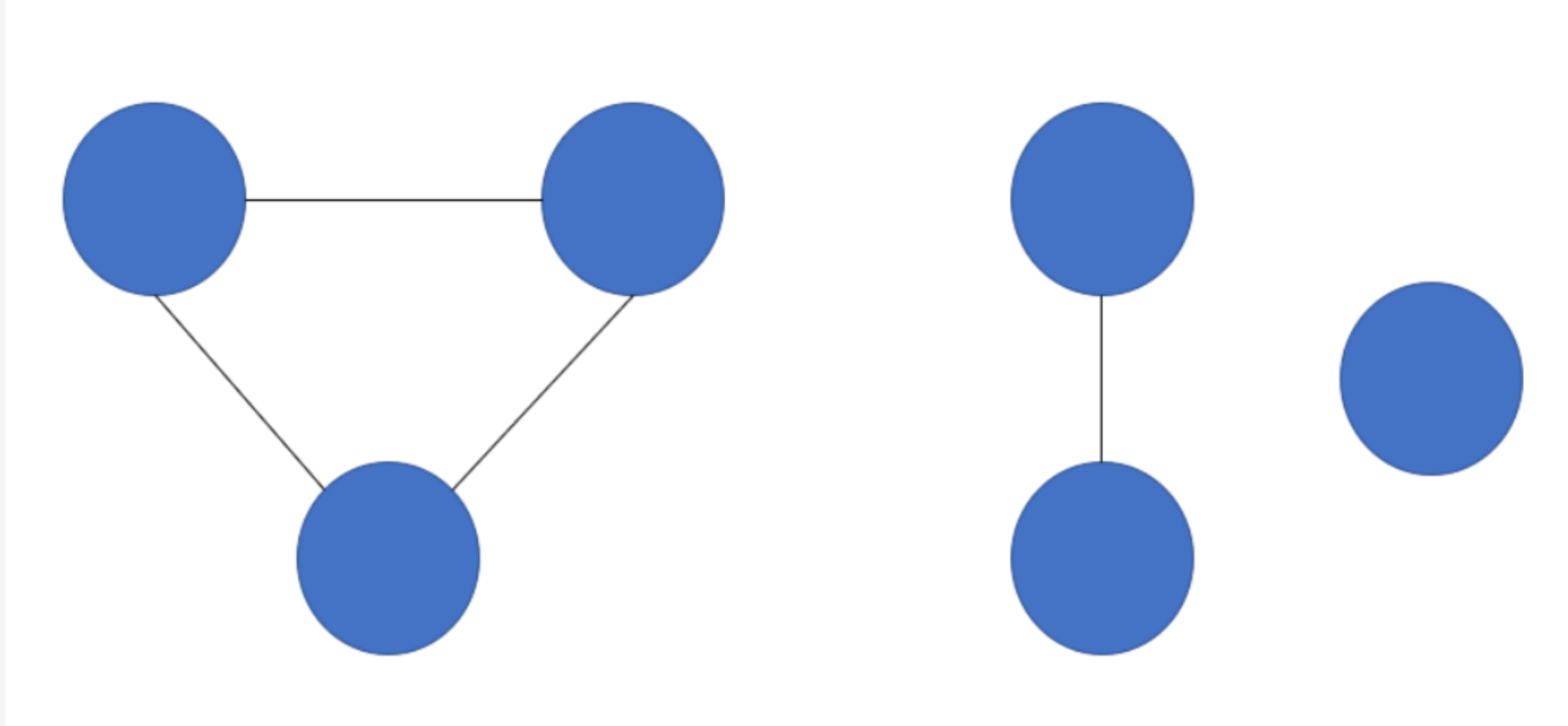


# 10. 그래프

## 그래프?

정점과 간선들로 이루어진 집합을 말한다.



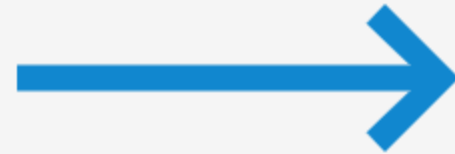
## 기초 용어 정리



정점(Node)



간선(Edge)



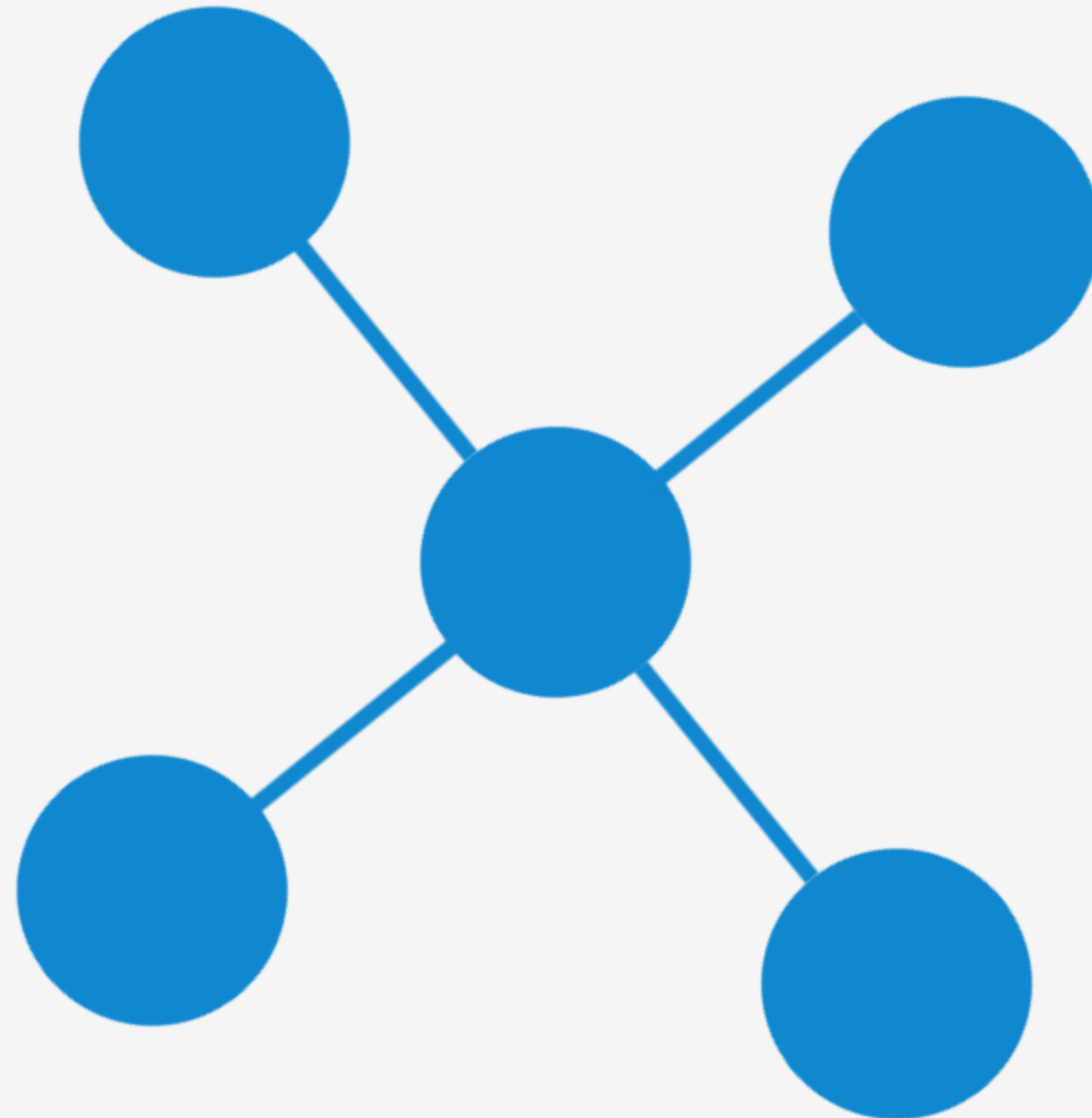
단방향 간선



양방향 간선

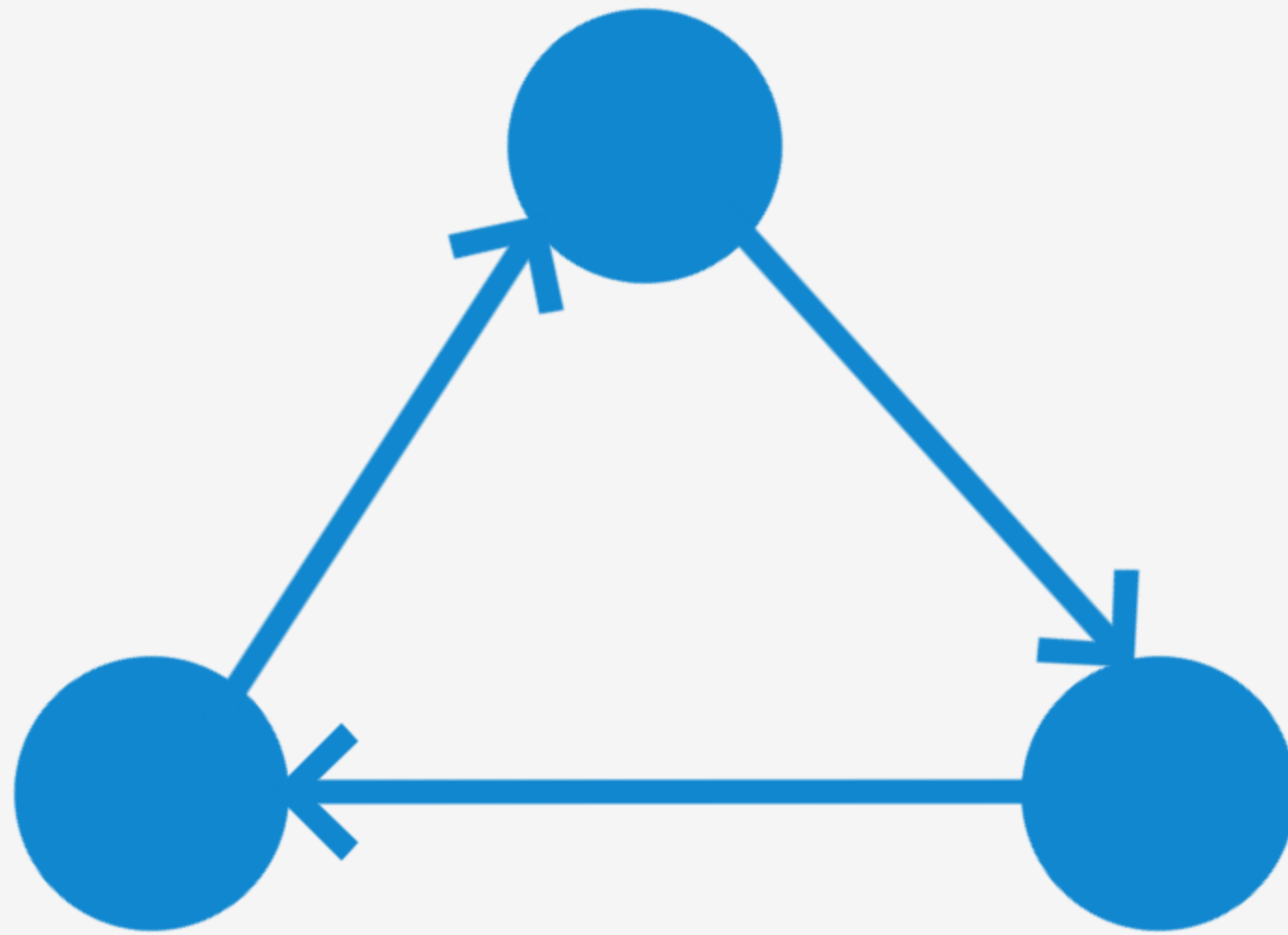
## 기초 용어 정리

차수(Degree)



## 기초 용어 정리

사이클(Cycle)



## 기초 용어 정리

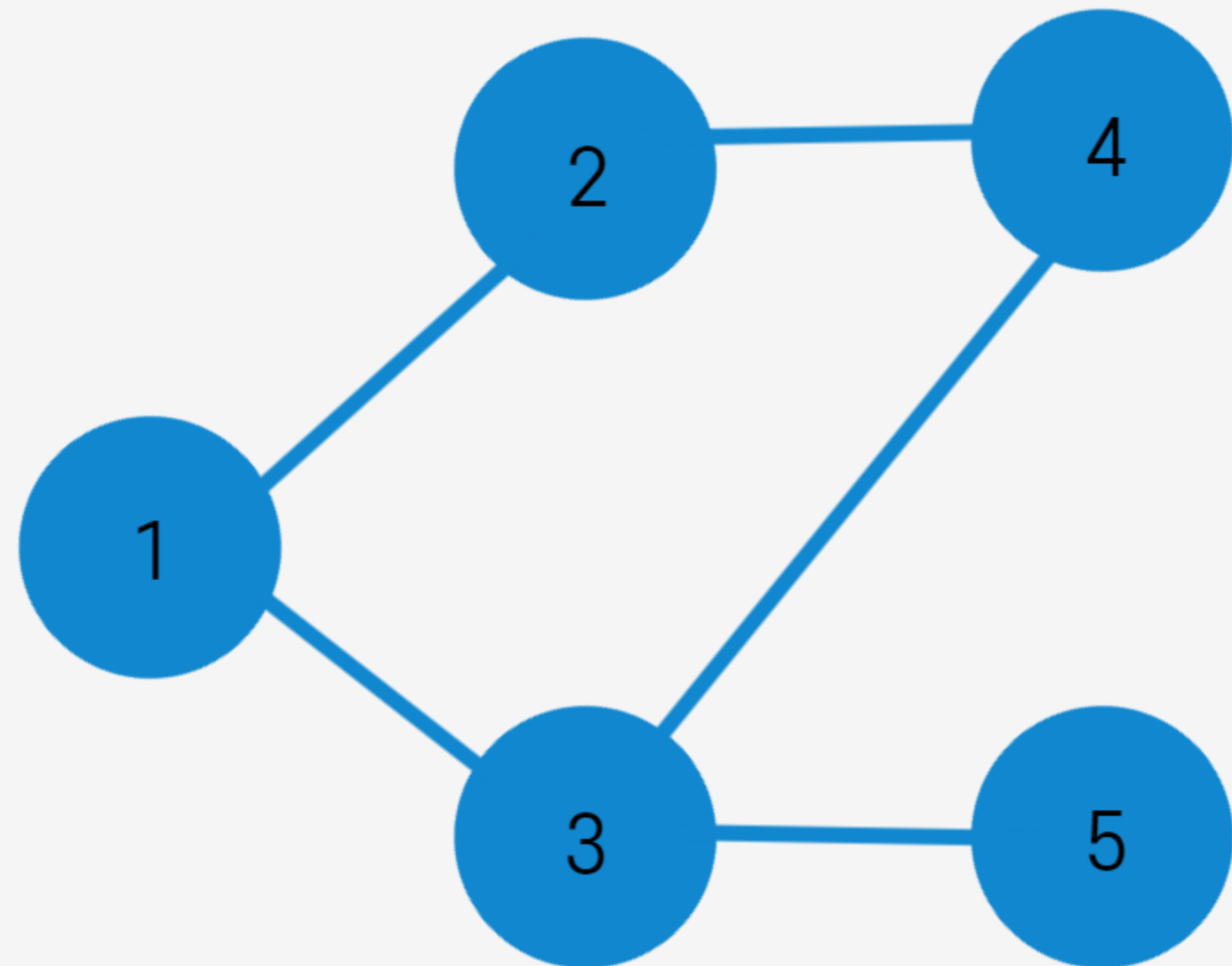
경로

1에서 4로 가는 경로

1->2->4

1->3->4

1->3->5->3->1->2->1->3->4 (?)

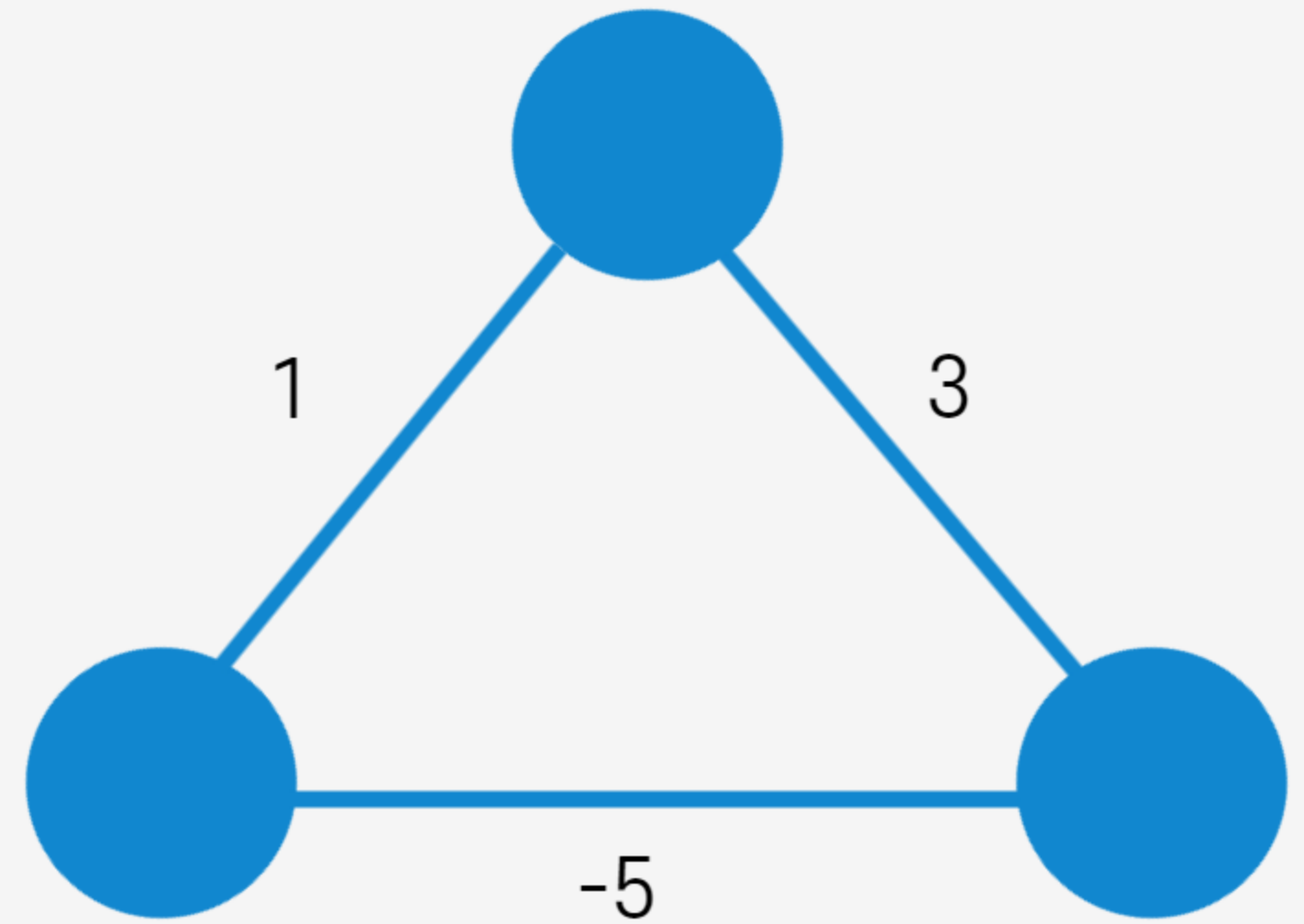


## 기초 용어 정리

가중치

간선(Edge)이 가지는 고유값

문제에 따라 거리, 비용 등 다양한 값을 갖는다.



## 복습

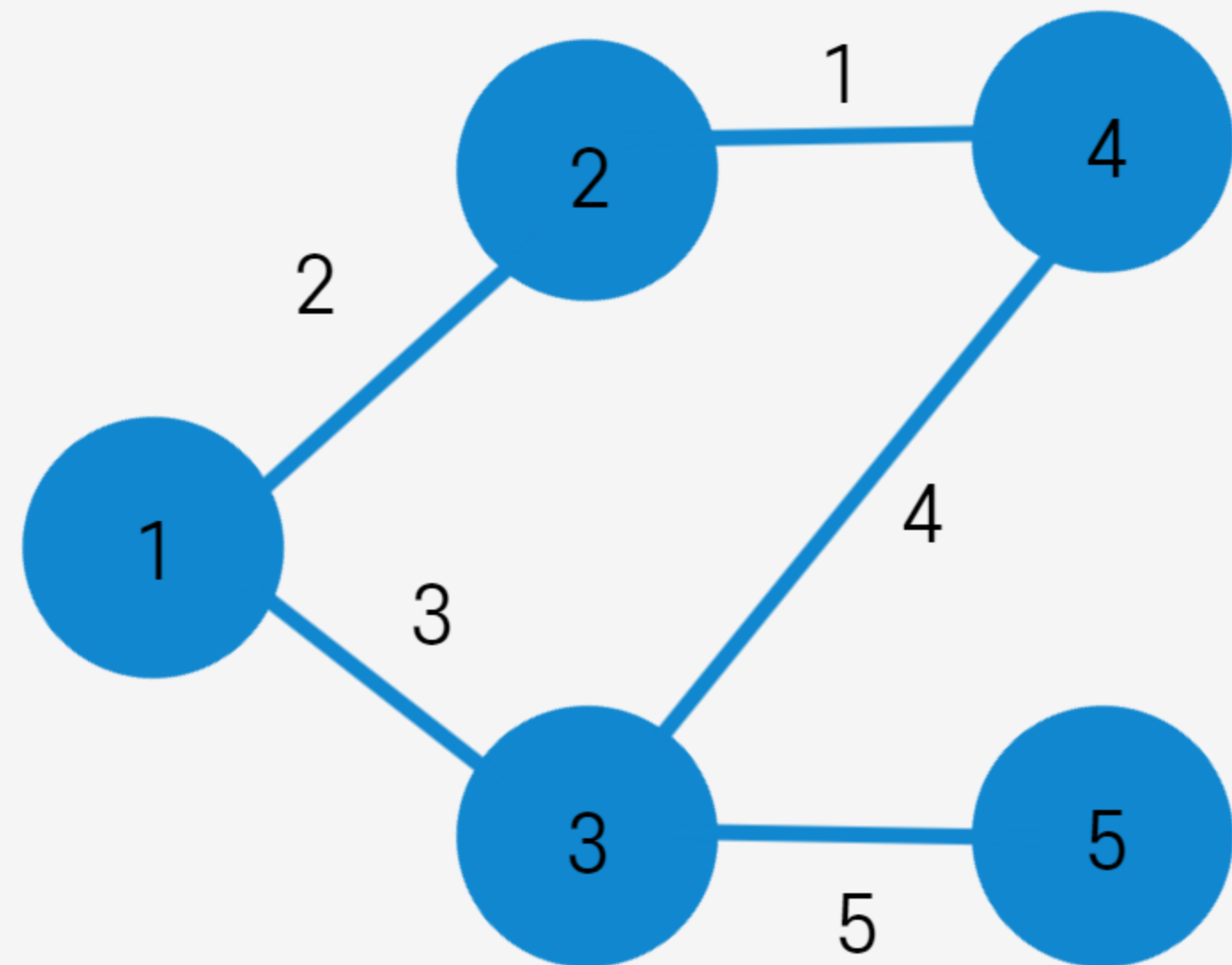
정점 ?

간선 ?

경로 ?

사이클 ?

가중치 ?





## 그래프 표현 방법

어떻게 그래프를 코드로 나타낼 수 있을까?

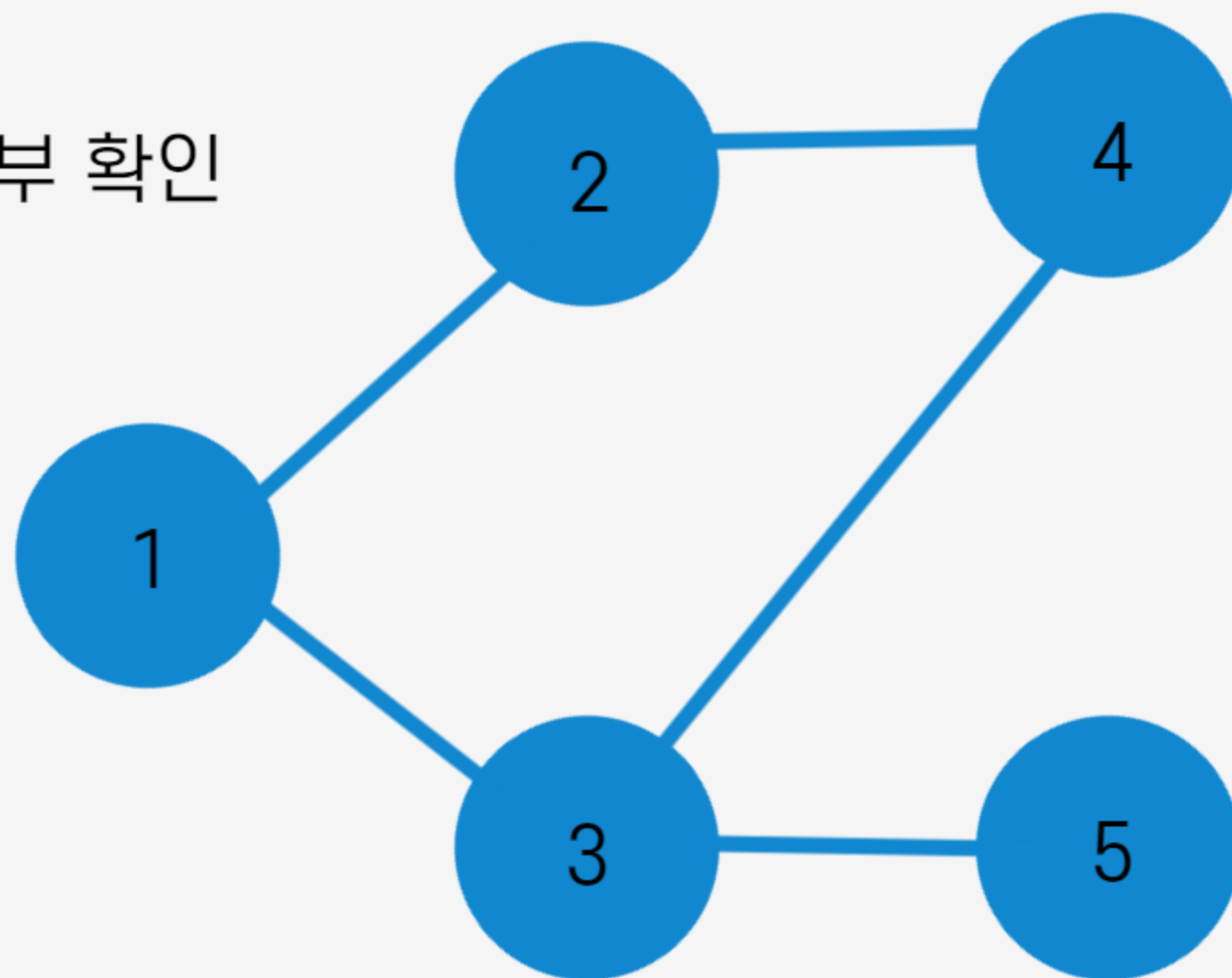
인접 행렬과 인접 리스트 방법이 있다.



## 인접 행렬

정점의 개수  $N$ 개에 대해  $N \times N$  행렬(배열)을 이용하여 표현한다.

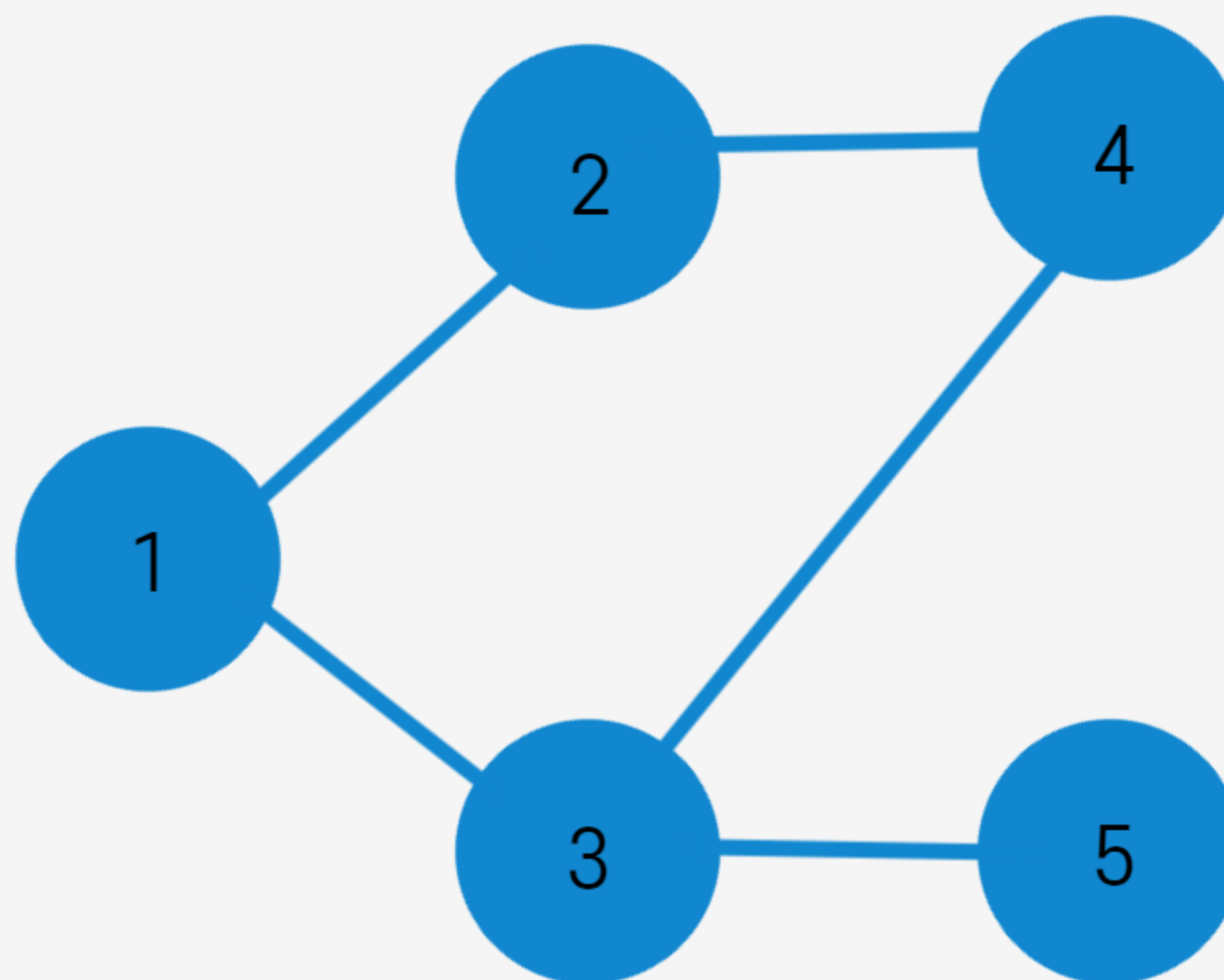
$V[i][j]$  :  $i$ 번 노드 ~  $j$ 번 노드를 잇는 간선의 여부 확인  
(1 : 있음, 0 : 없음)



## 인접 행렬

아래 그래프에 대한 행렬의 표현

	1	2	3	4	5
1	0	1	1	0	0
2	1	0	0	1	0
3	1	0	0	1	1
4	0	1	1	0	0
5	0	0	1	0	0

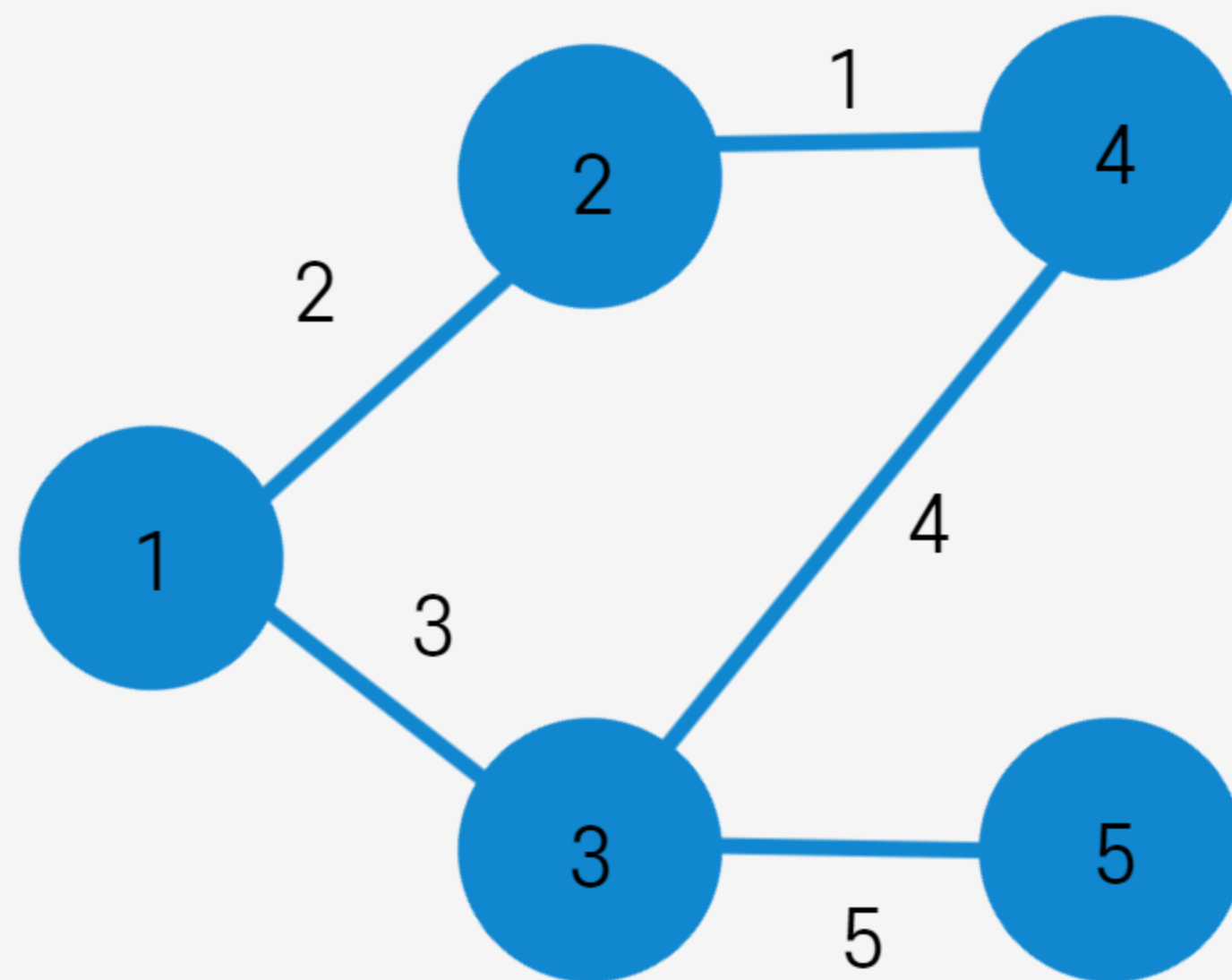


## 인접 행렬

가중치가 있는 경우

	1	2	3	4	5
1	?	2	3	?	?
2	2	?	?	1	?
3	1	?	?	1	1
4	?	1	1	?	?
5	?	?	1	?	?

? : 문제의 입력에서 절대 나올 수 없는 수



### 인접 리스트

가변길이의 벡터를 통해 구현한다.

정점 N개인 그래프의 인접 리스트 초기 선언 - `vector<int>[N]`

$v[i] = \{i\text{와 연결되어 있는 모든 정점}\}$

## 인접 리스트

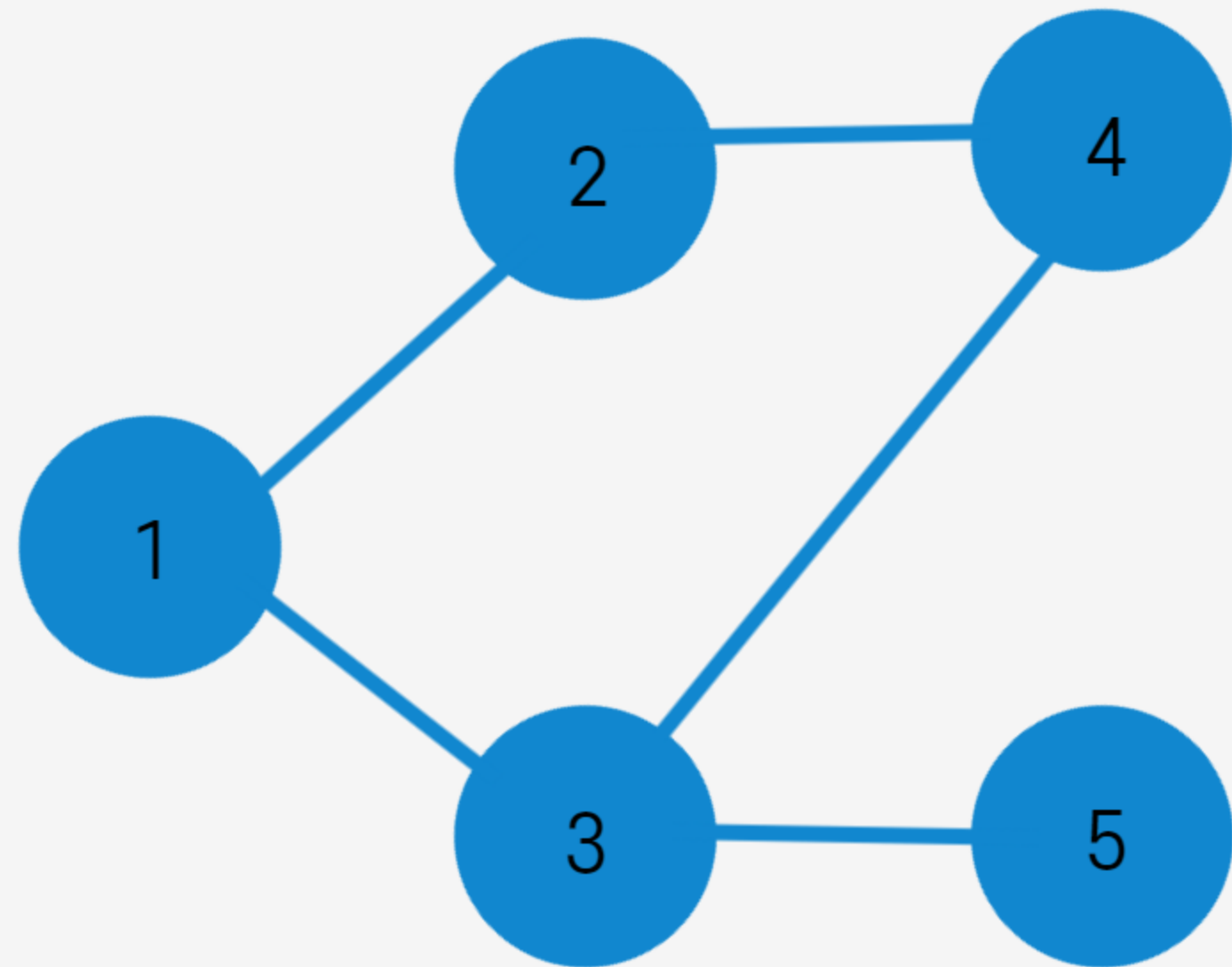
$v[1] = \{2, 3\}$

$v[2] = \{1, 4\}$

$v[3] = \{1, 4, 5\}$

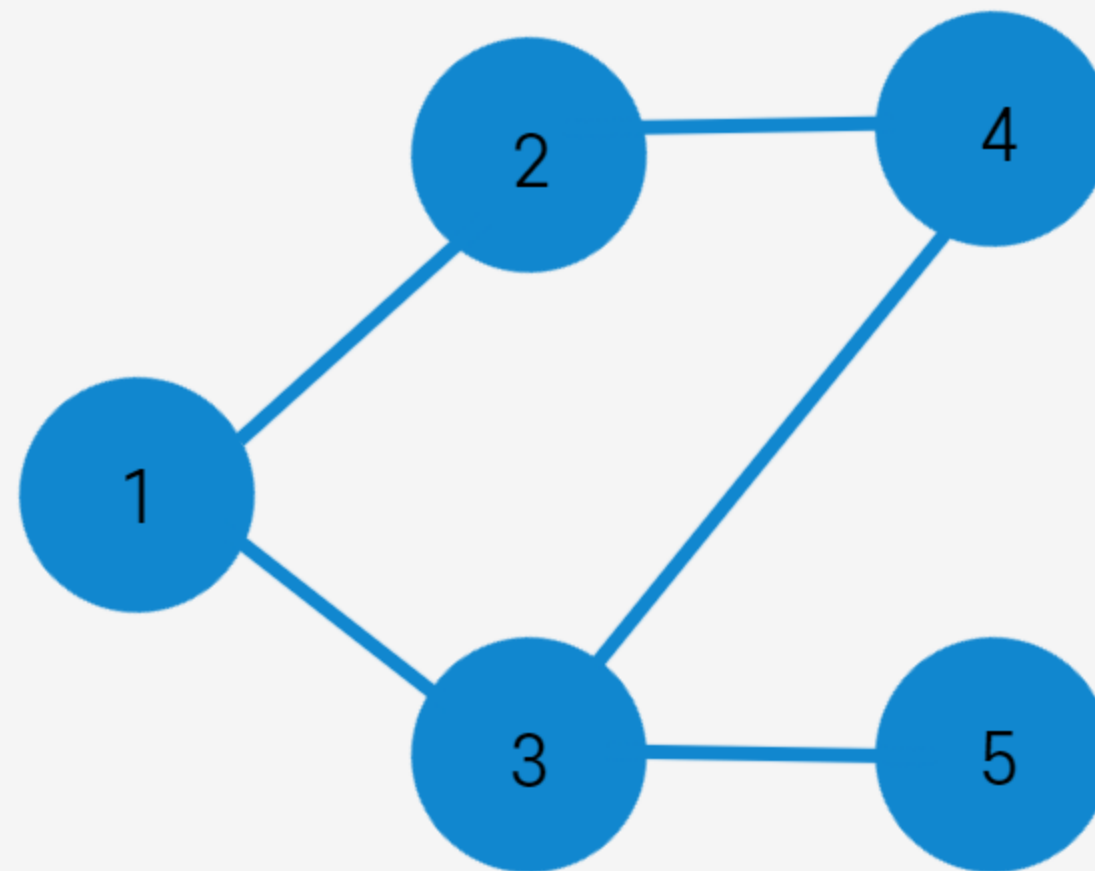
$v[4] = \{2, 3\}$

$v[5] = \{3\}$



## 인접 리스트 구현

입력	코드
5	<code>cin&gt;&gt;n;</code>
1 2	<code>cin&gt;&gt;a&gt;&gt;b;</code>
1 3	<code>v[a].push_back(b);</code>
2 4	<code>v[b].push_back(a);</code>
3 4	
3 5	



## 비교

공간 복잡도

인접 행렬 :  $N \times N$  행렬 사용  $\rightarrow O(N^2)$

인접 리스트 : 간선 개수  $\times 2$  만큼의 공간 필요  $\rightarrow O(E)$

$\rightarrow$  인접 리스트가 훨씬 효율적!