# Java Queue<E> Method Signatures and Examples

## offer(E e)

설명: 큐에 요소를 삽입. 공간 부족 시 false 반환.

예제 코드:

Queue<String> queue = new LinkedList<>();  
queue.offer("Apple");

## add(E e)

설명: 큐에 요소를 삽입. 공간 부족 시 예외 발생.

예제 코드:

Queue<String> queue = new LinkedList<>();  
queue.add("Banana");

## poll()

설명: 큐의 앞 요소를 제거하고 반환. 비어있으면 null 반환.

예제 코드:

Queue<String> queue = new LinkedList<>();  
queue.offer("Cat");  
String head = queue.poll(); // "Cat"

## remove()

설명: 큐의 앞 요소를 제거하고 반환. 비어있으면 예외 발생.

예제 코드:

Queue<String> queue = new LinkedList<>();  
queue.offer("Dog");  
String head = queue.remove(); // "Dog"

## peek()

설명: 큐의 앞 요소를 제거하지 않고 조회. 비어있으면 null 반환.

예제 코드:

Queue<String> queue = new LinkedList<>();  
queue.offer("Egg");  
String head = queue.peek(); // "Egg"

## element()

설명: 큐의 앞 요소를 제거하지 않고 조회. 비어있으면 예외 발생.

예제 코드:

Queue<String> queue = new LinkedList<>();  
queue.offer("Fish");  
String head = queue.element(); // "Fish"

## size()

설명: 큐에 들어 있는 요소의 개수를 반환.

예제 코드:

Queue<String> queue = new LinkedList<>();  
queue.offer("A");  
queue.offer("B");  
int size = queue.size(); // 2

## isEmpty()

설명: 큐가 비어있는지 확인.

예제 코드:

Queue<String> queue = new LinkedList<>();  
boolean empty = queue.isEmpty(); // true

## contains(Object o)

설명: 특정 요소가 큐에 포함되어 있는지 확인.

예제 코드:

Queue<String> queue = new LinkedList<>();  
queue.offer("Apple");  
boolean hasApple = queue.contains("Apple"); // true

## remove(Object o)

설명: 특정 요소를 큐에서 제거 (앞 요소가 아님).

예제 코드:

Queue<String> queue = new LinkedList<>();  
queue.offer("A");  
queue.remove("A");

## clear()

설명: 큐의 모든 요소 제거.

예제 코드:

Queue<String> queue = new LinkedList<>();  
queue.offer("A");  
queue.clear();

## iterator()

설명: 큐 요소들을 순회할 수 있는 이터레이터 반환.

예제 코드:

Queue<String> queue = new LinkedList<>();  
queue.offer("A");  
queue.offer("B");  
for (String item : queue) {  
 System.out.println(item);  
}

## toArray()

설명: 큐를 Object 배열로 변환.

예제 코드:

Queue<String> queue = new LinkedList<>();  
queue.offer("A");  
Object[] arr = queue.toArray();

## <T> toArray(T[] a)

설명: 큐를 지정된 타입의 배열로 변환.

예제 코드:

Queue<String> queue = new LinkedList<>();  
queue.offer("A");  
String[] arr = queue.toArray(new String[0]);

## containsAll(Collection<?> c)

설명: 지정된 컬렉션의 모든 요소가 큐에 포함되어 있는지 확인.

예제 코드:

Queue<String> queue = new LinkedList<>();  
queue.offer("A");  
queue.offer("B");  
List<String> list = List.of("A", "B");  
boolean result = queue.containsAll(list);

## addAll(Collection<? extends E> c)

설명: 컬렉션의 모든 요소를 큐에 추가.

예제 코드:

Queue<String> queue = new LinkedList<>();  
List<String> list = List.of("A", "B");  
queue.addAll(list);

## removeAll(Collection<?> c)

설명: 컬렉션의 모든 요소를 큐에서 제거.

예제 코드:

Queue<String> queue = new LinkedList<>();  
queue.offer("A");  
queue.offer("B");  
List<String> list = List.of("A");  
queue.removeAll(list);

## retainAll(Collection<?> c)

설명: 컬렉션에 포함된 요소만 유지.

예제 코드:

Queue<String> queue = new LinkedList<>();  
queue.offer("A");  
queue.offer("B");  
List<String> list = List.of("A");  
queue.retainAll(list);

## equals(Object o)

설명: 다른 큐와 동일한지 비교.

예제 코드:

Queue<String> queue1 = new LinkedList<>();  
Queue<String> queue2 = new LinkedList<>();  
queue1.offer("A");  
queue2.offer("A");  
boolean same = queue1.equals(queue2);

## hashCode()

설명: 큐의 해시코드를 반환.

예제 코드:

Queue<String> queue = new LinkedList<>();  
queue.offer("A");  
int hash = queue.hashCode();