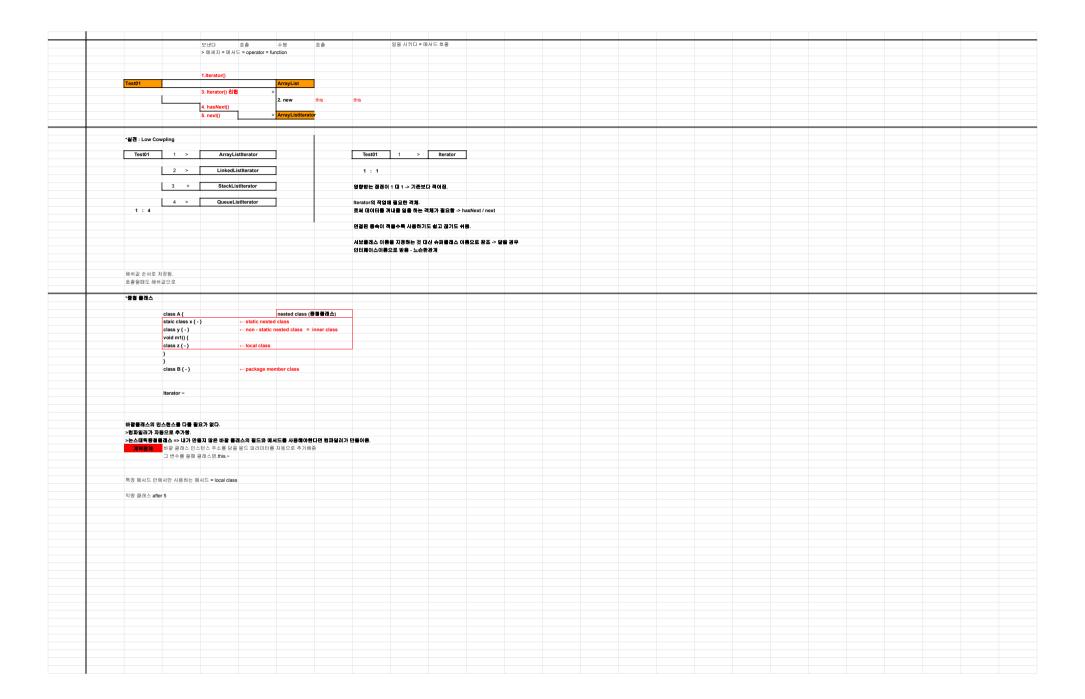
i												
		*031 데이터 조회 :	조건에 삼관없이	일관된 방법으로	조회하기.							
		- 게시글을 조회할	때 역순으로 조	회한다.: Iterator II	턴 적용							
		배열을 거꾸로 만든	등이 기담하다									
		0	1	2	3 4	5 length -1 - i						
					3 4	o lengar-1-1	-					
		_				_						
		0	1	/	3 4	5						
		-		/								
		> 문제 : 데이터 조	회 방식에 따라	코딩이 달라진다.								
			방식 : 데이터 3	조회 방식에 상관없	이 동일한 방식으로 조회하고 싶	G.						
		> 해결왝										
		>>> 데이터 조희를										
		>>> 데이터 조희 병	방식을 일관되게	하기 위해 인터페	이스를 이용하여 규격화 한다.							
	*Interator 패턴 2	적용 전										
		ArrayList										
			데이터 조회	>> get(인맥스)	·결론							
		LinkedList			>데이터 저장	방식(Data Strucutre) = 자료구조에	따라 데이터를 조회하는 방식이다르다.					
						회를 위한 코딩이 일관성이 없다.						
		Stack	데이터 조회	>> pop()								
		Queue	데이터 조회	>> poll()								
		HashSet	데이터 不하	>> toArray								
		Audiloct		>>>HashSat®	레이터를 저장할 때 데이터를 가기	17						
				>>> [등지 개시시	= ^- e= 배 네이너를 기/ = Hashrodo= 웹 시가 - best	u alue = 정수로 된 ID값]의 결과값을 인	작다 마셨으시 5시민					
						조회활 수 없다. 그래서 get() 메서드기						
				/// 메르의 단독	그저는 이에서 시작하는 없으로 :	호텔을 구 없다. 그대지 get() 메지드기	r accr.					
	*Interator 패턴 2										 	
	interator Me -	48 ÷		Iterator(반복자)	if(hadNext())							
		4	→	GIOLEI		, MOISI THE	B 다른 객체에게 맡긴다.					
		ArrayList	→	조회 전문가	← next();	1. 데이디 소의1	다른 역체에게 열긴다.					
		LinkedList		소의 전문가	next();	2. 데이터 조회	* 11 5 5 01 81 51					
		LinkedList			W W W.		5억을 등일었다.					
					if(hadNext())	>> 데이터 조회						
		Stack	→	데이터	-							
				조희 전문가	next();	>> 유지 보수가	입 다.					
		Queue										
				BIOLEI	if(hadNext())							
		HashSet	→	조회 전문가	-							
					next();							
		데이터 조회 코드를										
		>자료구조에 상관(없이 일관된 방법	받으로 데이터 조회	수행							
		>객체화 → 다양한	조회 방식을 제	공/교체								
					- hasNext() : boolean							
				Iterator	- next(): 해당 객체를 리턴							
				Δ								
				1								
			至者									
		LinkedList	♦→	LinkedLi	stiterator ←	if(hasNext();						
						next();						
	*Interator 사용법	1										
		1. iterator()	-		LinkedList							
			←									
		3.Iterator 구현체 i	리턴	2. 생성								
					1 1	사용						
			200번지	1								
		4. hasNext()	-		LinkedListIterator							
		5. next()	-									
		메서드 / operater		new ArrayList()	;							
		i.toArray()										
				heap								
			add()									
		i i	insert()	40	arr size	ArrayList의 인스턴스	값을 담는 변수 - 인스턴스					
			get()	1	200 0							
			size()									
		r	remove()									
		-										



-									
1									
1									

1									

1									

1									
1									
l									
l									
l									
l									