



자료구조 11주차

연결리스트

MMC 연구실

석사 과정 강민제

조교 소개

- 강민제
- 컴퓨터공학과 대학원 석사과정
- MMC연구실 (A1406)
- rkdalswp29@gmail.com



실습 수업 진행 방식

- 확인 문제 풀이
- 확인 문제를 해결한 학생은 검사 받고 퇴실



과제 설명

- 자료구조 수업은 Eclipse를 사용하여 코드를 작성합니다.
- 확인 문제 및 과제를 전부 해결하여 제출해주세요.
- 과제 제출 시 **프로젝트 폴더를 압축**해서 제출합니다.
- 과제의 채점은 프로젝트의 실행 결과를 기준으로 점수를 매깁니다.

확인문제 1 (연결 리스트)

Package Name : list_2

Class Name : ListNode

- 지난 실습 과제의 ListNode와 동일



확인문제 1 (연결 리스트)

Package Name : list_2

Class Name : LinkedList

```
public static LinkedList concatList(LinkedList list1, LinkedList list2) {  
    // 전달받은 리스트 두 개를 하나로 연결하여 반환하는 메소드  
}  
  
public void append(String name) {  
    // 전달받은 name을 이용해 새로운 노드를 생성하고, 리스트의 맨 뒤에 추가하는 메소드  
}  
  
public void deleteLastNode() {  
    // 리스트의 맨 마지막 노드를 삭제하는 메소드  
}
```

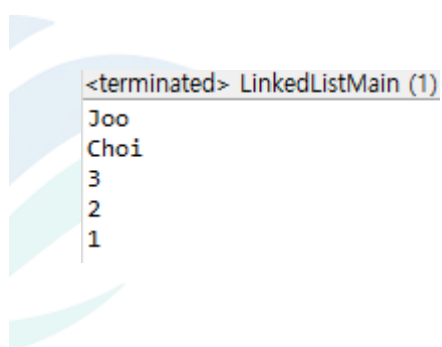
list_2/LinkedList.txt 파일 참고
- 지난 주 실습한 메소드 포함

확인문제 1 (연결 리스트)

Package Name : list_2

Class Name : LinkedListMain

```
public class LinkedListMain {  
    public static void main(String[] args) {  
        LinkedList list1 = new LinkedList();  
        list1.addFirst("Kim");  
        list1.addFirst("Choi");  
        list1.addFirst("Joo");  
        list1.deleteLastNode();  
  
        LinkedList list2 = new LinkedList();  
        list2.addFirst("1");  
        list2.addFirst("2");  
        list2.addFirst("3");  
  
        LinkedList.concatList(list1, list2).print();  
    }  
}
```



```
<terminated> LinkedListMain (1)  
Joo  
Choi  
3  
2  
1
```

list_2/LinkedListMain.txt 파일 참고

과제 1 (원형 연결 리스트)

Package Name : circular

Class Name : ListNode

- 지난 실습 과제의 ListNode와 동일



과제 1 (원형 연결 리스트)

Package Name : circular
Class Name : CircularList

circular/CircularList.txt 파일 참고

```
package circular;

public class CircularList {
    private ListNode tail;

    public CircularList() {
        tail = null;
    }

    public int size() {
        ListNode p;
        if(tail == null)
            return 0;
        int len = 1;
        p = tail.getLink();
        while(p != tail) {
            len++;
            p = p.getLink();
        }
        return len;
    }

    public void addFirst(String name) {
        // 리스트의 맨 앞에 새로운 노드 삽입
    }

    public ListNode listSearch(String targetName) {
        // 리스트에서 targetName을 가진 Node를 반환
    }

    public void delete(ListNode target) {
        // target Node 뒤의 Node를 삭제
    }

    public void print() {
        ListNode target;
        if(tail != null) {
            target = tail.getLink();
            while(target != tail) {
                System.out.print(target.getName() + ", ");
                target = target.getLink();
            }
            System.out.println(target.getName());
        }
    }
}
```

과제 1 (원형 연결 리스트)

Package Name : circular

Class Name : CircularMain

```
1 package circular;
2
3 public class CircularMain {
4
5     public static void main(String[] args) {
6         CircularList cl = new CircularList();
7         cl.addFirst("3");
8         cl.addFirst("2");
9         cl.addFirst("1");
10        cl.print();
11        cl.delete(cl.listSearch("1"));
12        cl.print();
13    }
14 }
```

```
Console X
<terminated> CircularV
1, 2, 3
1, 3
```

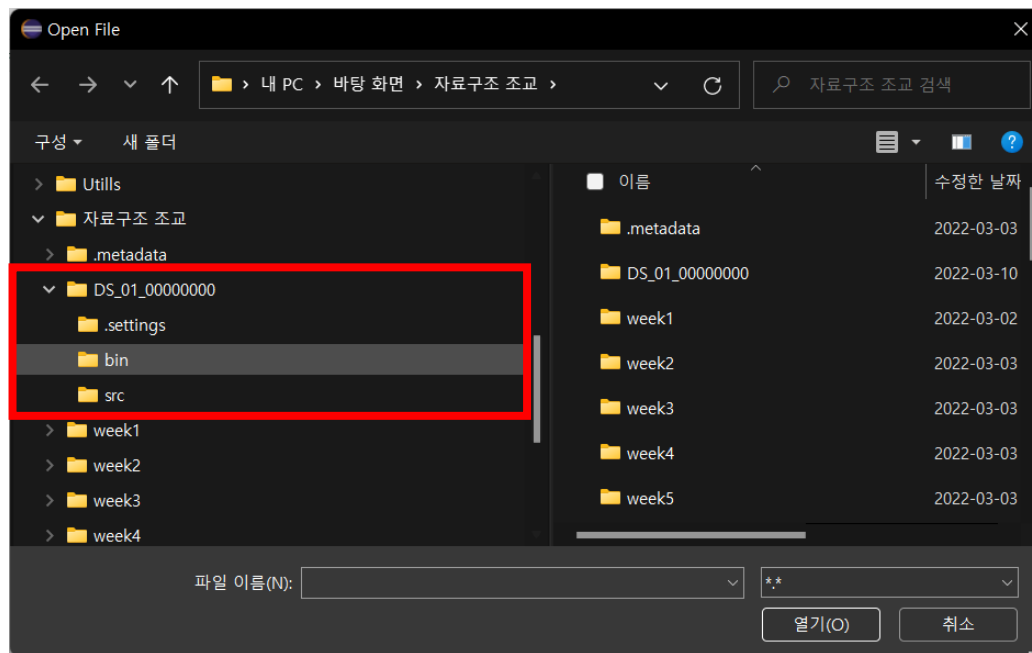
circular /CircularMain.txt 파일 참고

확인문제 및 과제 파일 구조

- ▼ 11week_test
 - > JRE System Library [jre]
 - ▼ circular
 - > CircularList.java
 - > CircularMain.java
 - > ListNode.java
 - ▼ list_2
 - > LinkedList.java
 - > LinkedListMain.java
 - > ListNode.java



과제 제출 방법



- 프로젝트 폴더를 압축하여 제출

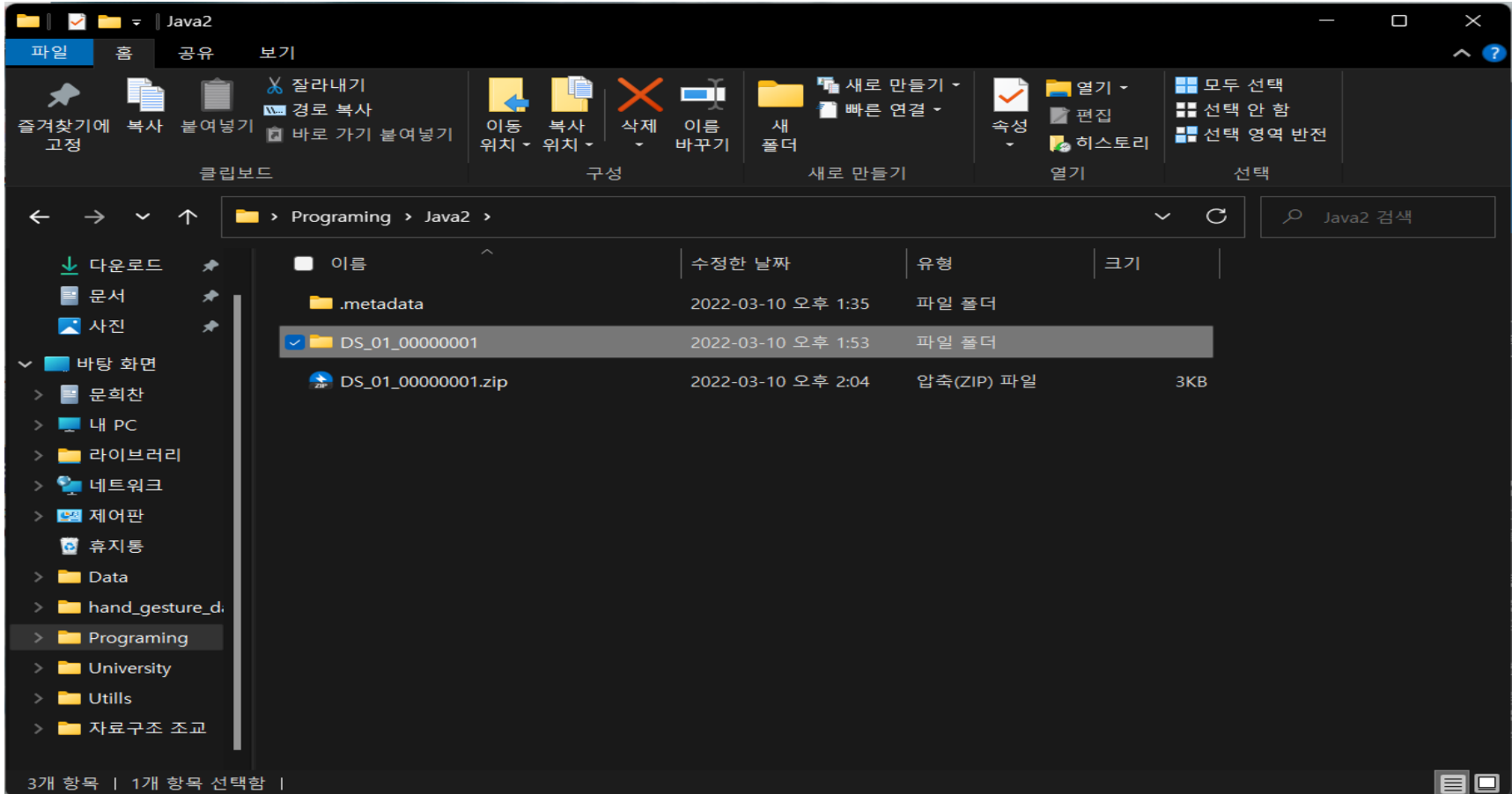
- 프로젝트이름 : DS_(주차)_(학번)

예) DS_11_00000000

- *.java파일만 제출하면 안됩니다.

제출양식을 반드시 지켜주세요

과제 제출 방법



- 반드시 **프로젝트 폴더를 압축**하여 제출