



# 자료구조 7주차

## 재귀함수2, 순차 리스트

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 (GCD)

Package Name : recursion2

Class Name : GCD

- 재귀함수를 이용해 최대공약수를 구하는 메소드 구현

```
public class GCD {
```

```
    public static void main(String[] args) {  
        System.out.println("36, 12 : " + gcd(36, 12));  
    }
```

```
    public static int gcd(int a, int b) {  
  
    }
```

```
<terminated> GCD [Java Application]  
36, 12 : 12
```

```
}
```

## 확인문제 2 (이진수 변환)

Package Name : recursion2

Class Name : IntToBinary

```
public class IntToBinary {  
    public static void main(String[] args) {  
        System.out.println(intToBinary(100));  
    }  
  
    private static String intToBinary(int n) {  
    }  
}
```

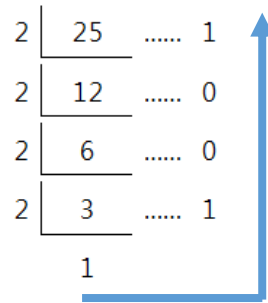
<terminated> IntToBinary

1100100

## 확인문제 2 (이진수 변환)

Package Name : recursion2

Class Name : IntToBinary



<그림1 십진수에서 이진수 변환 방법>

- 이진수로 변환하기 위해서는 나누기 연산과 나머지 연산을 잘 사용해야 합니다.
- 값을 2로 더 이상 나눌 수 없을 때 까지 나눈 후, 나머지의 값을 **역순**으로 이어 붙이면 됩니다.
- 25를 이진수로 변환하면 11001이 됩니다.

## 확인문제 3 (순차 리스트)

Package Name : list  
Class Name : LinearList

- 메소드 구현

```
public class LinearList {
    private char array[];
    private int size;
    public static int MAX = 100;

    public LinearList() {
        size = 0;
        array = new char[MAX];
    }

    public boolean isEmpty(){
        return (size == 0);
    }

    public int length(){
        return size;
    }

    public char retrieve(int i) {
        // 해당 인덱스의 원소를 리턴하는 메소드
    }

    public void replace(int i, char k) {
        // 인덱스 i 번째 원소를 k로 대체하는 메소드
    }

    public void replaceByElem(char c1, char c2) {
        // 리스트 내부의 c1 원소를 c2로 대체하는 메소드
    }

    public void insert(int i, char k){
        // 리스트에서 i 번째에 k 원소를 삽입하는 메소드
    }

    public void print() {
        // 리스트의 문자를 전부 출력하는 메소드
    }
}
```

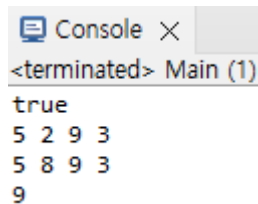


## 확인문제 3 (순차 리스트)

Package Name : list

Class Name : Main

```
public class Main {  
    public static void main(String[] args) {  
        LinearList a = new LinearList();  
  
        System.out.println(a.isEmpty());  
  
        a.insert(0, '5');  
        a.insert(1, '2');  
        a.insert(2, '9');  
        a.insert(3, '3');  
        a.print();  
        a.replaceByElem('2', '8');  
        a.print();  
        System.out.println(a.retrieve(2));  
    }  
}
```



```
Console ×  
<terminated> Main (1)  
true  
5 2 9 3  
5 8 9 3  
9
```

## 과제 1 (순차 리스트)

Package Name : list

Class Name : LinearList

- 확인문제 3에서 구현한 클래스에 delete, deleteByElem 메소드를 구현

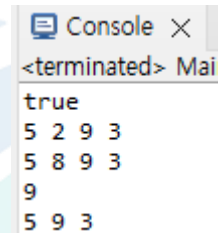
```
public void delete(int i){  
    // 리스트의 i 번째 원소를 삭제하는 메소드  
}  
public void deleteByElem(char c) {  
    // 리스트에서 c에 해당하는 원소를 삭제하는 메소드  
}
```

## 과제 1 (순차 리스트)

Package Name : list

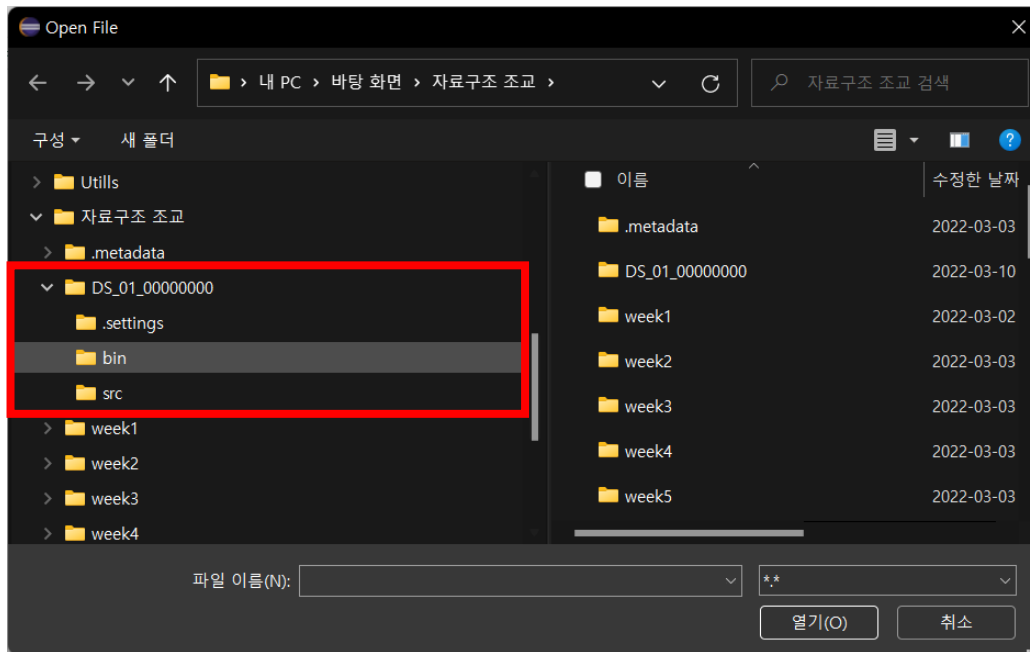
Class Name : Main

```
public class Main {  
    public static void main(String[] args) {  
        LinearList a = new LinearList();  
  
        System.out.println(a.isEmpty());  
  
        a.insert(0, '5');  
        a.insert(1, '2');  
        a.insert(2, '9');  
        a.insert(3, '3');  
        a.print();  
        a.replaceByElem('2', '8');  
        a.print();  
        System.out.println(a.retrieve(2));  
  
        a.deleteByElem('8');  
        a.print();  
    }  
}
```



```
Console X  
<terminated> Mai  
true  
5 2 9 3  
5 8 9 3  
9  
5 9 3
```

# 과제 제출 방법



- 프로젝트 폴더를 압축하여 제출

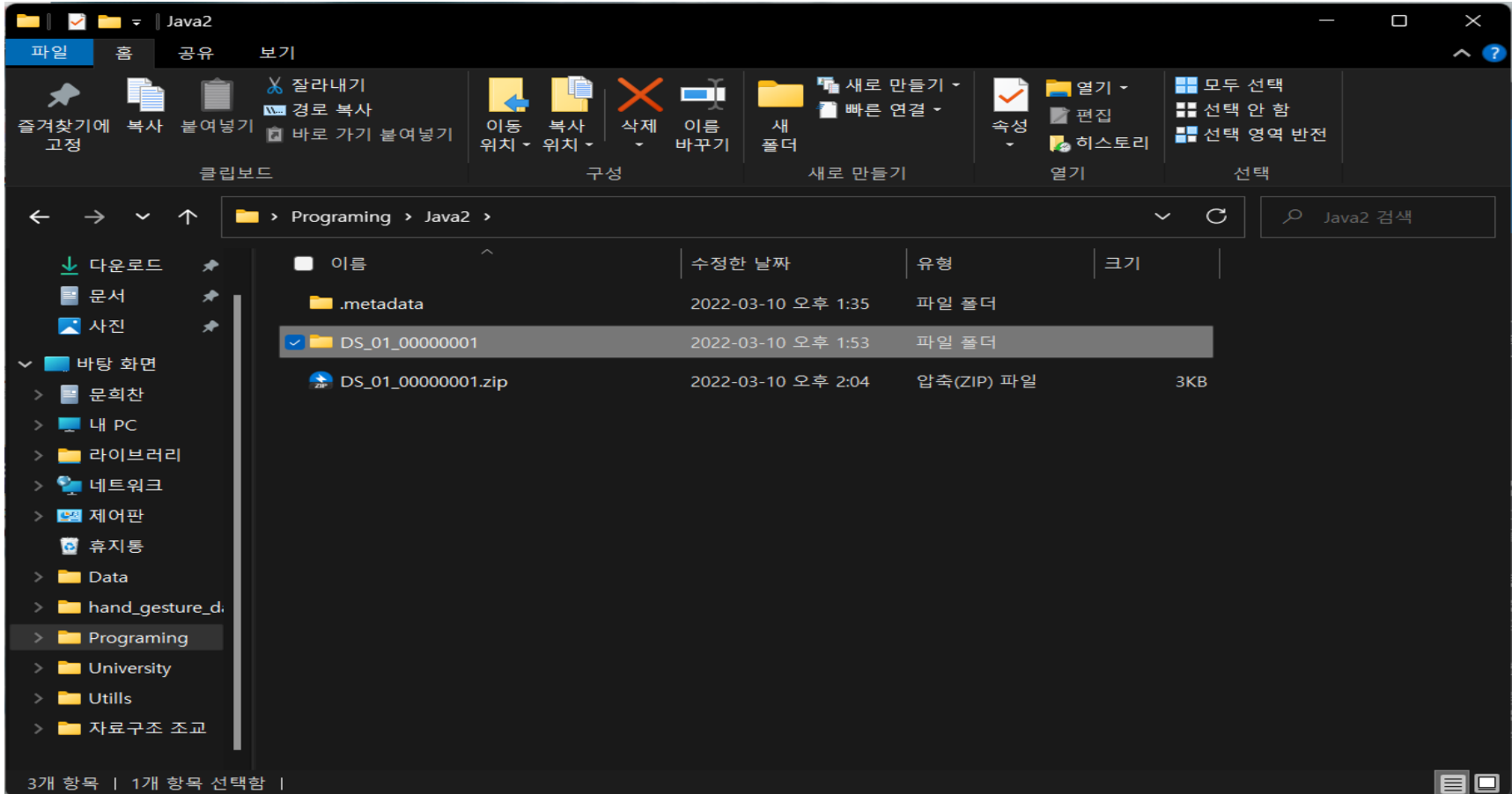
- 프로젝트이름 : DS\_(주차)\_(학번)

예) DS\_07\_00000000

- \*.java파일만 제출하면 안됩니다.

**제출양식을 반드시 지켜주세요**

# 과제 제출 방법



- 반드시 **프로젝트 폴더를 압축**하여 제출