

# 문장

Practice 4

# 문제 1

- 소수(Prime number)란 1과 자기 자신 외에는 약수를 갖지 않는 수이다. 100이하의 소수를 구하는 프로그램을 작성하고 결과를 출력하시오. 여기서 약수란 어떤 수를 나누어 떨어지게 하는 수를 의미한다.  
예) 8의 약수: 1,2,4,8

## 문제 2

- 완전수(Perfect number)란 자기 자신을 제외한 약수의 합이 자기 자신과 같은 수이다. 예)  $6=1+2+3$ . 1부터 500 사이의 완전수를 구하는 프로그램을 작성하고 결과물을 출력하시오.

## 문제 3

- 회문수(Palindromic number)란 숫자를 역순으로 쓴 수와 같은 수를 말한다. 즉, 121 또는 526625와 같은 수이다. 정수를 읽어서 회문수인지 아닌지 판별하는 자바 프로그램을 작성하고 다음 입력에 대한 출력을 제출하시오.

121, 526625, 362341, 36763, 478230

# File I/O

- 파일을 복사 및 변조하는 자바 프로그램을 구현하고 이를 이용하여 파일을 복사해보자.

```
import java.io.FileInputStream;  
import java.util.Scanner;
```

System.in : 표준 입력  
Ex) 키보드

```
Scanner sc = new Scanner(System.in);  
Scanner file = new Scanner(new FileInputStream("File dir/File name.txt"));
```

```
sc.close();  
file.close();
```

입출력 종료 후 Stream 닫기!

파일 입력 클래스 FileInputStream 사용  
→ 해당 파일을 입력으로 사용!

# File I/O

```
6 import java.io.BufferedReader;  
7 import java.io.BufferedWriter;  
8  
9 import java.io.FileReader;  
10 import java.io.FileWriter;  
11
```

```
BufferedReader br = new BufferedReader(new FileReader("File dir/File name.txt"));  
BufferedWriter wr = new BufferedWriter(new FileWriter("File dir/File name.txt"));  
  
br.close();  
wr.close();
```

FileReader와 FileWriter 는  
각각 파일에서 문자스트림을 읽고  
쓰는데 사용되는 클래스이다.

파일을 읽고 쓰는데 조금더 효율적으로 하기  
위해 버퍼를 사용하도록 해주는 클래스  
(한꺼번에 읽고 쓰는것을 가능하게 해주어  
디스크의 IO시간을 줄여줌)

# File I/O

```
public class Testing
{
    public static void main(String args[])
    {
        BufferedReader br;
        BufferedWriter wr;

        // to do Something
        try
        {
            br = new BufferedReader(new FileReader("File dir/File name.txt"));
            wr = new BufferedWriter(new FileWriter("File dir/File name.txt"));

            br.close();
            wr.close();
        }

        // catch the error
        catch (IOException e)
        {
            e.printStackTrace();
        }

        finally
        {
            System.out.println("Program ends.");
        }
    }
}
```

Try : 정상적으로 실행되는 경우!

Catch : 해당 에러 발생시!

Finally : 무조건 실행!

# File I/O

Throws : 자동으로 에러 발생시 미리  
정의된 내용 수행!

```
public static void main(String args[]) throws IOException
```

내부 내용은 Try와 같은 정상 작동 내용



## 문제 4

- BufferedReader, FileReader, BufferedWriter, FileWriter 만을 이용하여 파일을 읽고 쓰는 연습을 해보자.
- 아래와 같은 파일을 하나 생성한뒤 이를 읽어서 그대로 복사하는 프로그램을 만들어보자. 이때 파일을 줄 단위로 읽어보자.

```
Hi this is JAVA  
What is your name?  
1 2 3 4 5 6 7  
1,2,3,4,5,6,7
```

## 문제 5

- 입력 파일을 줄 단위로 읽어 각 줄을 토큰으로 분리해서(스페이스와 콤마 , 마다 토큰으로 나누어지도록) 오른쪽과 같은 출력파일이 생성되도록 프로그램을 작성해보자.

```
Hi  
this  
is  
JAVA  
What  
is  
your  
name?  
1  
2  
3  
4  
5  
6  
7  
1  
2  
3  
4  
5  
6  
7
```