Week 10 : I/O Library

Part1. File Class

File class object show a path of the file or directory

- File Class' Constructor

Method	Description
File(File parent, String child)	Generate a file object represented as a file or directory of the child's name in a directory of the parent
File(String pathname)	Generate a file object represented as a pathname
File(String parent, String child)	return a string of file or directory name represented as a file object's abstract pathname represented as a directory or file of child's name in a directory of the parent
File(URI uri)	Change a file:URI to an abstract pathname and generate a file object

- File class의 주요 메소드

메소드	설명
boolean mkdir()	Generate a new directory represented as an abstract pathname
String[] list()	return a string array of a file and directory's name represented as an abstract pathname.
String[] listFiles()	return a string array of a file name in the directory represented as an abstract pathname
boolean renameTo(File dest)	Change a file name to an abstract pathname defined as dest
boolean delete()	Delete a file or directory represented as an abstract pathname
long length()	return a file size represented as an abstract pathname
String getPath()	Return a string of the abstract pathname
String getName()	Return a string of the file or directory name represented as an abstract pathname
boolean isFile()	Return a true value, when an abstract pathname is a file

boolean isDirectory()	Return a true value, when an abstract pathname is a directory
long lastModified()	Return a time that a file change finally
boolean exists()	Return a true when a file or directory represented as an abstract pathname exists.

Part2. Byte Stream

InputStream/OutputStream

a super class of every classes for processing a byte input and output stream.

```java
import java.io.BufferedOutputStream;
import java.io.IOException;
import java.io.InputStreamReader;




public class File {


        public static void main(String[] args) throws IOException {
        InputStreamReader in = new InputStreamReader(System.in);
        BufferedOutputStream out = new BufferedOutputStream(System.out,5);


        int c;
        while((c=in.read())!=-1){
        out.write(c);
        }
        out.flush();
        in.close();
        out.close();


        }

}
```

☞ Get input value from the buffer, then print out 5 characters from the buffer. Over the 5 characters print out when you press

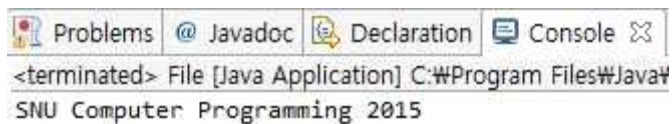ctrl+z.

FileInputStream/FileOutputStream

file input and output stream. Read a byte data or save a byte data from a file at the file system.

```java
import java.io.FileInputStream;
import java.io.IOException;


public class File {

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

        FileInputStream fin = new FileInputStream("c:₩₩io.txt");
        int c;
        while((c=fin.read())!=-1)
        {
        System.out.print((char)c);
        }
        }

}
```

```
Problems  @ Javadoc  Declaration  Console 

<terminated> File [Java Application] C:₩Program Files₩Java₩
SNU Computer Programming 2015
```
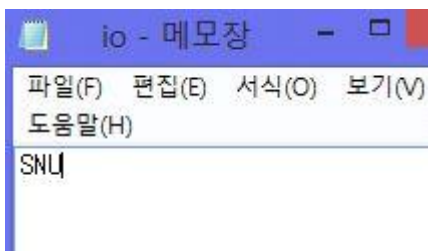
```java
import java.io.FileOutputStream;
import java.io.IOException;


public class File {

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

        FileOutputStream fout = new
```

```
FileOutputStream("c:₩₩Users₩₩mk₩₩Desktop₩₩io.txt");
        char c[]={'S','N','U'};
        for(int i=0; i<c.length; i++)
        {
        fout.write(c[i]);
        }
        }


}
```


DataInputStream/DataOutputStream

Input and output of the binary value or string about basic data type of the Java

FileWriter/FileReader

read and write to the file.

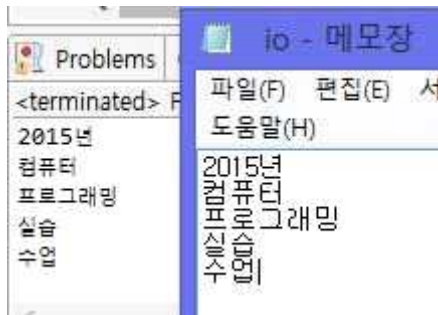## BufferedReader/BufferedWriter

Java class about character buffer stream.

- Example

```
public class File {

        public static void main(String[] args) throws IOException {
        FileReader fr = new FileReader("c:₩₩Users₩₩mk₩₩Desktop₩₩io.txt");
        BufferedReader br = new BufferedReader(fr);
        StringBuffer sb=new StringBuffer();
        String s = "";
```

```
        while((s=br.readLine())!=null){
        sb.append(s+"\n");
        }
        s=sb.toString();
        System.out.println(s);
        br.close();
        fr.close();


        }

    }
```


[Exercise]

1. Make a text copy program with exception handling using try-catch.

- hint

File type : define a location of file

FileReader/FileWriter : connect File input/output character streams.

BufferedReader/BufferedWriter : connect buffer input/ouput streams.

2. Attach the WeeklyCalendar Code file. Then, write a code about 'saveWeekPlan' method, 'loadWeekPlan' method which are located in HCalendar class. Just making a I/O code about Monday section is ok.