

프로그래밍 언어 응용

chapter15

자바 입출력

제공된 자료는 훈련생의 수업을 돕기 위한 것으로, 타인과 공유하시면 안됩니다.

Contents

part.1

자바 입출력과 스트림

part.2

표준 입출력

part.3

바이트 단위 스트림

part.4

문자 단위 스트림

part.5

보조 스트림

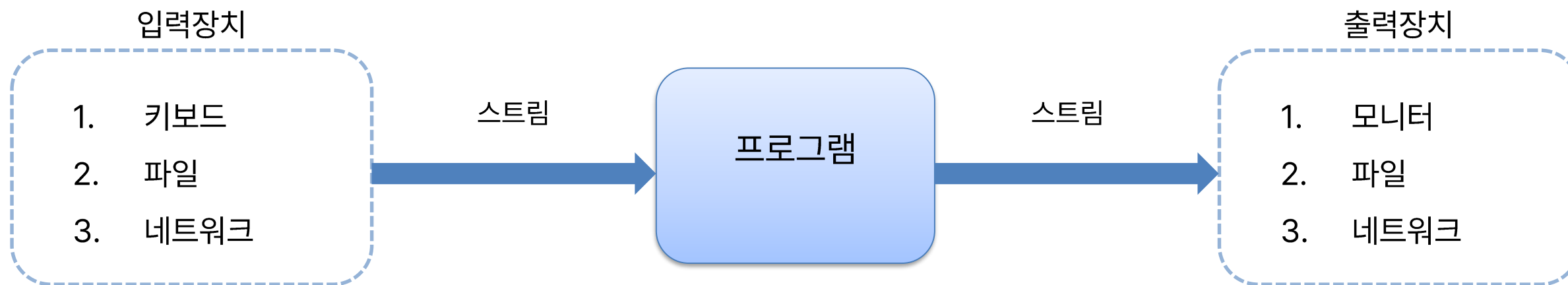
자바 입출력과 스트림 스트림이란?

스트림이란?

스트림은 프로그램으로 데이터가 들어오고 나가는 흐름이다

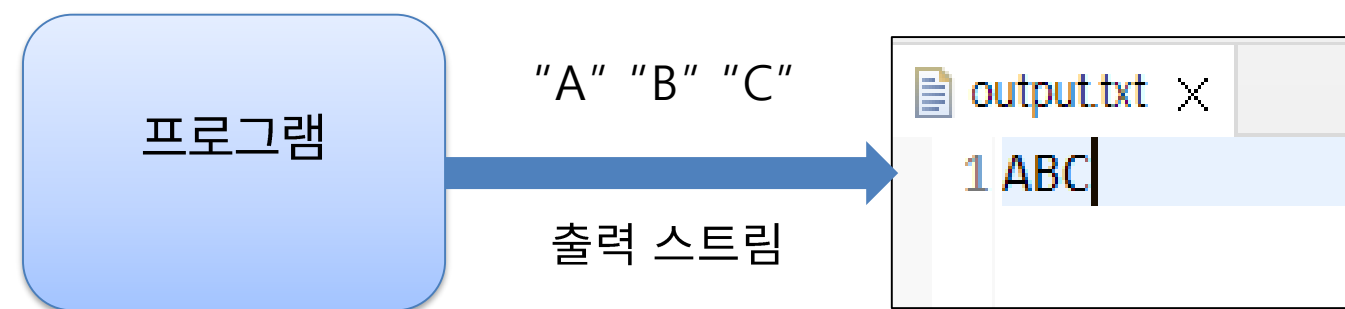
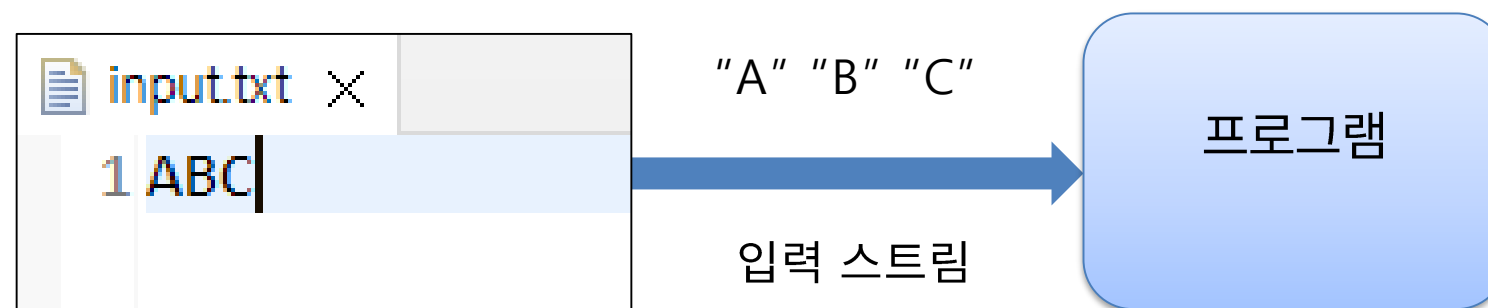
스트림의 특징

- 파일, 키보드, 모니터, 네트워크 등 다양한 입출력 장치가 있다.
- 연결되는 입출력 장치에 따라 다른 클래스를 제공한다.



자바 입출력과 스트림 입력 스트림과 출력 스트림

스트림은 단방향으로 데이터가 이동되어, 입력과 출력을 동시에 할 수 없다.
예를 들어 입력 스트림은 파일의 내용을 읽어오는데 사용되고,
반대로 출력 스트림은 파일에 글자를 저장하는데 사용된다.

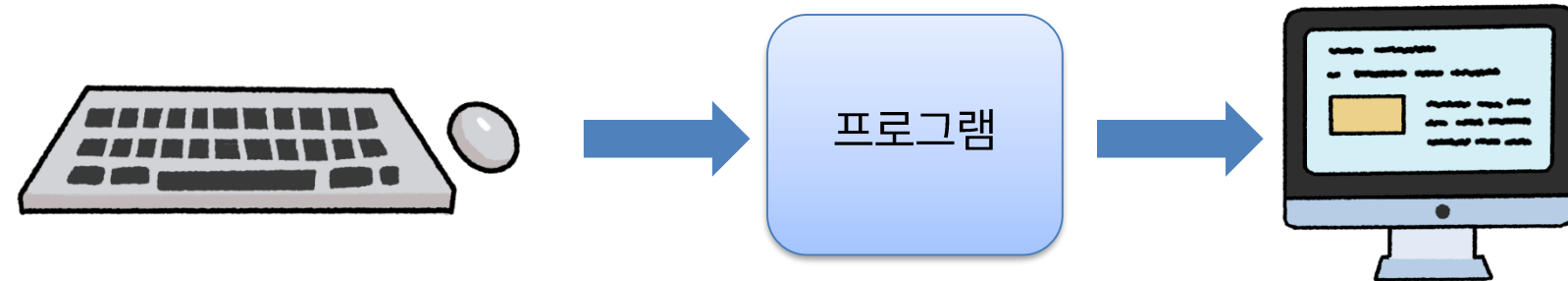


표준 입출력

표준 입출력 스트림

표준 입출력 스트림이란?

- 시스템이 제공하는 표준 입출력 스트림이다.
- 표준 출력 장치: 모니터
- 표준 입력 장치: 키보드



자료형	변수 이름	설명
static PrintStream	out	표준 출력 스트림
static InputStream	in	표준 입력 스트림
static OutputStream	err	표준 오류 출력 스트림

System.out

- 표준 출력 스트림으로 내용을 모니터에 출력한다.

System. + out. + println()



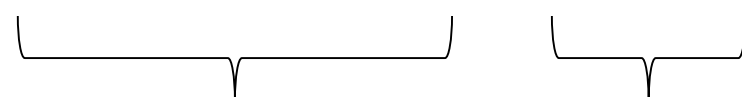
출력장치

내용을 출력

System.in

- 표준 입력 스트림으로 키보드에서 데이터를 입력 받는다.
- 데이터를 바이트 단위로 입력 받아, 아스키 코드 값으로 저장한다.

System. + in. + read()



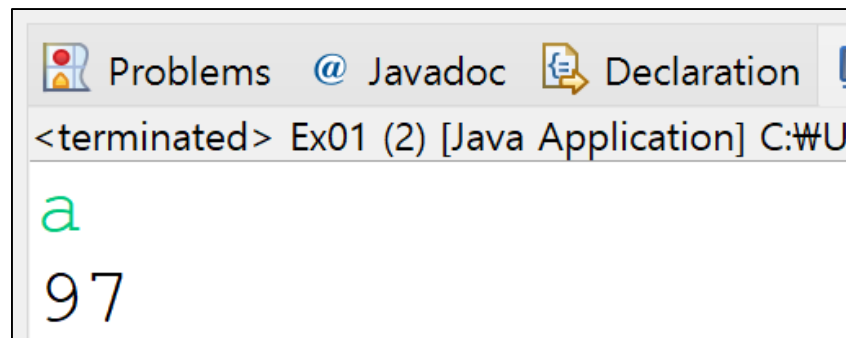
입력장치

데이터를 입력 받음

표준 입력 스트림으로 문자 입력 받기

1. 입력스트림을 이용하여 값을 입력 받는 코드를 작성한다.
2. 프로그램을 실행한다.
3. 콘솔에서 문자를 입력하고 엔터키(enter)를 누른다.

```
System.out.println("알파벳 하나를 쓰고 [Enter]를 누르세요");  
  
int i = System.in.read();
```



97	0x61	a
98	0x62	b
99	0x63	c
100	0x64	d
101	0x65	e
102	0x66	f
103	0x67	g

아스키코드 표

Scanner 클래스

- 문자, 정수, 실수 등 다양한 자료형을 읽을 수 있다.
- 표준 입력을 매개변수로 받아서 데이터를 받아 온다.
- 표준 입력보다 더 다양한 기능을 제공한다.

생성자	설명
Scanner(File source)	파일을 매개변수로 받아 Scanner를 생성한다
Scanner(InputStream source)	바이트 스트림을 매개변수로 받아 Scanner를 생성한다
Scanner(String source)	문자열을 매개변수로 받아 Scanner를 생성한다

1번.

5개 크기의 정수형 배열을 생성하세요

Scanner를 사용하여 배열에 숫자 5개를 입력 받으세요.

그리고 배열의 모든 요소의 합을 구하세요

```

Problems @ Javadoc Declaration Console × Terminal
<terminated> Quiz33 [Java Application] C:\Users\wimjiyeon\p2\pool\plugins
숫자를 입력하세요.
1
2
3
4
5
배열: [1, 2, 3, 4, 5]
숫자 5개의 합은 15입니다
  
```

2번.

Scanner를 사용하여 두자리 정수(10~99)를 입력 받으세요.

그리고 십의자리와 일의자리의 합을 구하세요.

```

Problems @ Javadoc Declaration Console × Terminal History
<terminated> Quiz022 (1) [Java Application] C:\Users\wimjiyeon\p2\pool\plugins\org
55
십의자리와 일의자리의 합은 10입니다
  
```

3번.

Scanner를 사용하여 문자열을 계속 입력 받으세요.

while 루프를 사용하여 입력을 받다가, 입력된 문자열이 "STOP"이면 루프를 종료하세요. 단, "STOP"의 대소문자는 구분할 필요가 없습니다.

```

Problems @ Javadoc Declaration Console × Terminal
<terminated> Quiz03 (3) [Java Application] C:\Users\wimjiyeon\p2\p
문자를 입력하세요:
aa
문자를 입력하세요:
bb
문자를 입력하세요:
stop
프로그램이 종료되었습니다
  
```

메소드	설명
<code>boolean nextBoolean()</code>	boolean 자료를 읽는다
<code>byte nextByte()</code>	byte 자료를 읽는다
<code>short nextShort()</code>	short 자료를 읽는다
<code>int nextInt()</code>	int 자료를 읽는다
<code>long nextLong()</code>	long 자료를 읽는다
<code>float nextFloat()</code>	float 자료를 읽는다
<code>double nextDouble()</code>	double 자료를 읽는다
<code>String nextLine()</code>	문자열 자료를 읽는다

바이트 단위 스트림

스트림 종류

스트림의 종류

- 바이트 단위 스트림: 그림, 동영상, 음악, 문자 등 모든 데이터 입출력 가능
- 문자 단위 스트림: 문자만 입출력 가능 (한글 처리 가능)

바이트 단위 스트림은 1바이트 단위로 데이터를 읽고 쓴다.

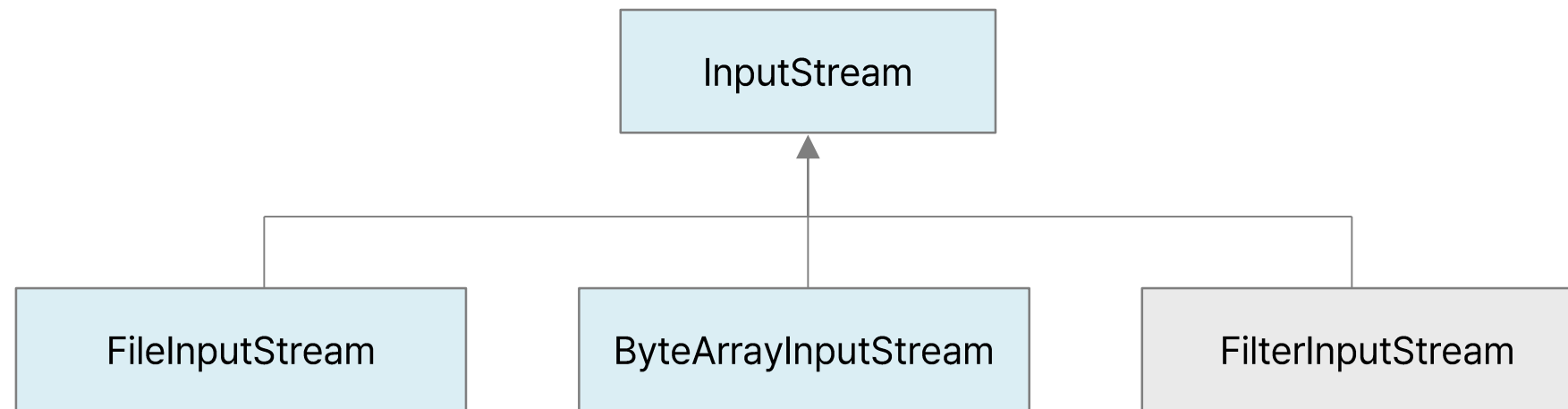
구분	입력 스트림	출력 스트림
최상위 클래스	InputStream	OutputStream
하위 클래스	XXXInputStream	XXXOutputStream

문자 단위 스트림을 2바이트 단위로 데이터를 읽고 쓴다.

구분	입력 스트림	출력 스트림
최상위 클래스	Reader	Writer
하위 클래스	XXXReader	XXXWriter

InputStream

- 바이트 단위로 데이터를 읽는 스트림이다.



스트림 클래스	설명	입력 대상
FileInputStream	파일에서 바이트 단위로 자료를 읽는다	파일
ByteArrayInputStream	Byte 배열 메모리에서 바이트 단위로 자료를 읽는다	byte 배열
FilterInputStream	(보조 스트림) 모든 보조 스트림의 부모이다	

메소드	설명
int read()	입력스트림으로부터 1바이트 자료를 읽는다. 읽은 자료의 바이트수를 반환한다. 1byte
int read(byte[] b)	입력스트림으로부터 읽은 바이트를 배열 b에 저장하고, 읽은 바이트 수를 반환한다.
int read(byte[] b, int off, int len)	입력 스트림으로부터 off 위치에서 len만큼 자료를 읽어서 배열 b에 저장한다. 읽은 바이트 수를 반환한다.
void close()	사용한 시스템 자원을 반납하고 입력스트림을 닫는다.

자료를 읽는 과정

1. input.txt파일과 연결된 FileInputStream 객체를 생성한다.
2. read 메소드를 이용해서 파일의 내용을 한 바이트씩 읽어 들인다.

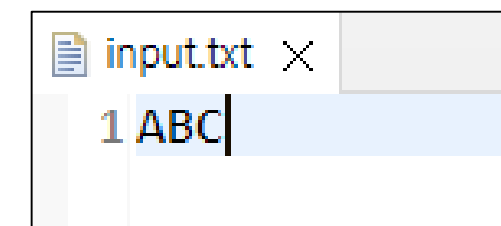
```
fis = new FileInputStream("input.txt");  
fis.read();  
fis.read();  
fis.read();
```

InputStream

바이트

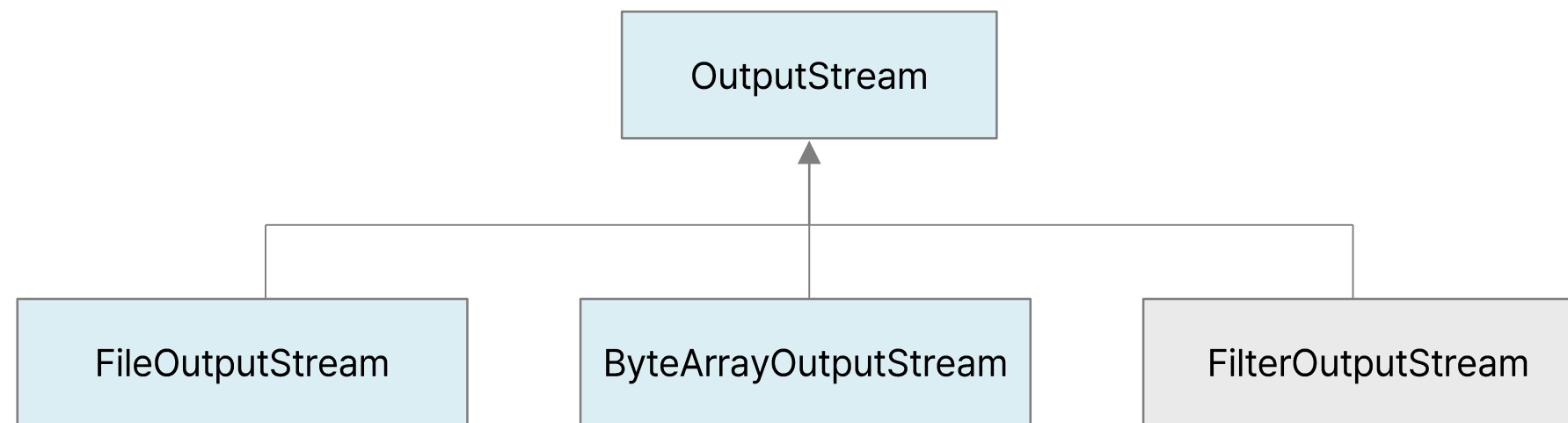
바이트

바이트



OutputStream

- 바이트 단위로 출력하는 스트림 중 최상위 스트림이다.
- 출력 스트림의 계층 구조는 다음과 같다.



스트림 클래스	설명	출력 대상
FileOutputStream	파일에서 바이트 단위로 자료를 쓴다.	파일
ByteArrayOutputStream	Byte 배열 메모리에서 바이트 단위로 자료를 쓴다.	byte 배열
FilterOutputStream	(보조 스트림) 모든 보조 스트림의 부모이다	

메소드	설명
<code>void write(int b)</code>	1바이트를 출력한다.
<code>void write(byte[] b)</code>	b배열에 있는 자료를 출력한다.
<code>void write(byte[] b, int off, int len)</code>	b배열에 있는 자료의 off 위치에서 len만큼 자료를 출력한다.
<code>void flush()</code>	출력스트림은 효율을 높이기 위해서 버퍼에 데이터를 모았다가 한번에 출력한다. 버퍼가 꽉 차면 자동으로 출력된다. 그전에 데이터를 출력하고 싶으면 flush() 메소드를 사용한다.
<code>void close()</code>	사용한 시스템 자원을 반납하고 출력스트림을 닫는다.

자료를 쓰는 과정

1. output.txt파일과 연결된 FileOutputStream 클래스를 생성한다.
2. write 메소드를 이용해서 파일에 바이트코드를 출력한다.
3. FileOutputStream은 숫자를 해당 아스키코드의 문자로 변환하여 저장한다

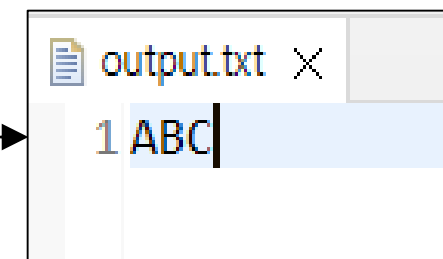
```
FileOutputStream fos = new FileOutputStream("output.txt");  
  
byte a = 65;  
byte b = 66;  
byte c = 67;  
  
fos.write(a);  
fos.write(b);  
fos.write(c);
```

OutputStream

바이트

바이트

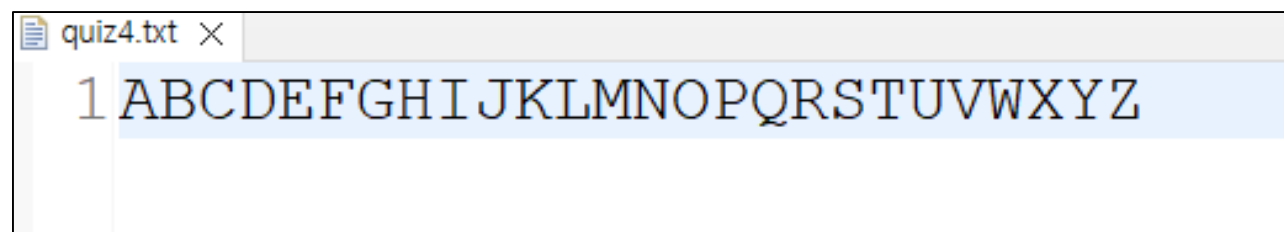
바이트



4번.

FileOutputStream을 사용하여 텍스트파일에 문자 A-Z까지 출력하세요.

결과

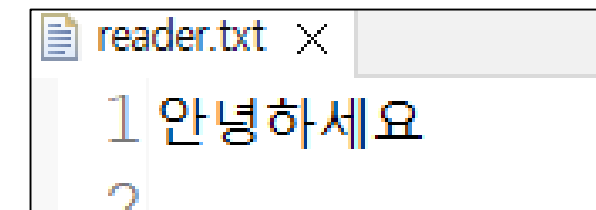
A screenshot of a text editor window titled 'quiz4.txt'. The first line of the file contains the text '1 ABCDEFGHIJKLMNOPQRSTUVWXYZ', which is highlighted in blue. The rest of the file is empty.

```
1 ABCDEFGHIJKLMNOPQRSTUVWXYZ
```

Reader

- 데이터를 문자 단위로 읽는 스트림이다.

텍스트 파일



```
reader.txt ×  
1 안녕하세요  
2
```

바이트 단위 스트림을 사용하여 파일 읽기

```
FileInputStream fis = new FileInputStream("reader.txt");  
while (true) {  
    int i = fis.read();  
    if (i == -1) {  
        break;  
    }  
    System.out.print((char) i);  
}
```

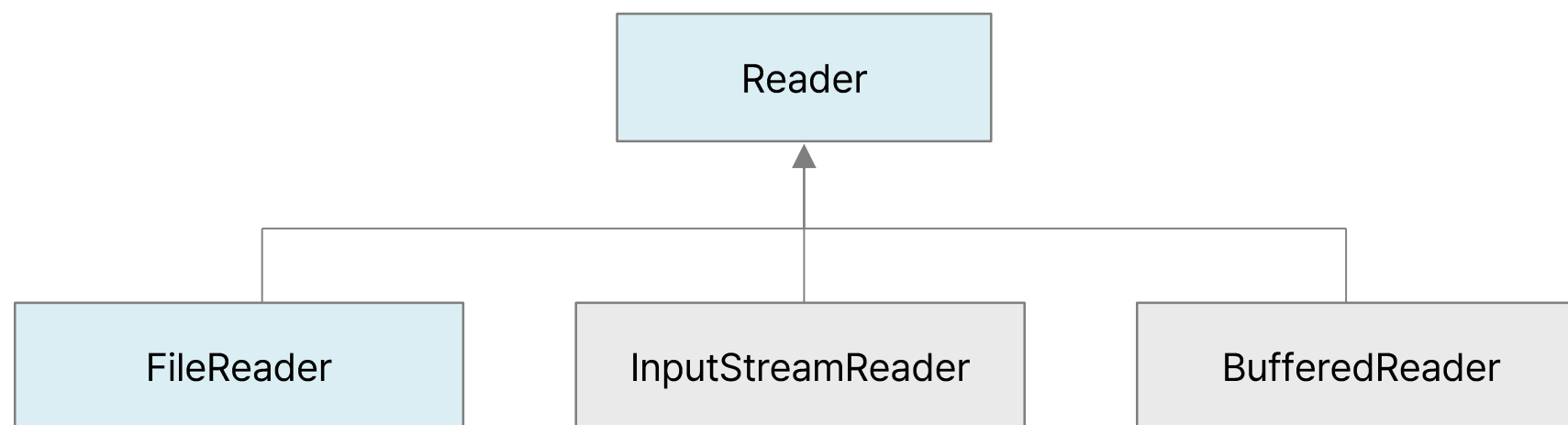
→ ìëíì,ì

문자 단위 스트림을 사용하여 파일 읽기

```
FileReader fr = new FileReader("reader.txt");  
while (true) {  
    int i = fr.read();  
    if (i == -1) {  
        break;  
    }  
    System.out.print((char) i);  
}
```

→ 안녕하세요

입력 스트림의 계층 구조는 다음과 같다.

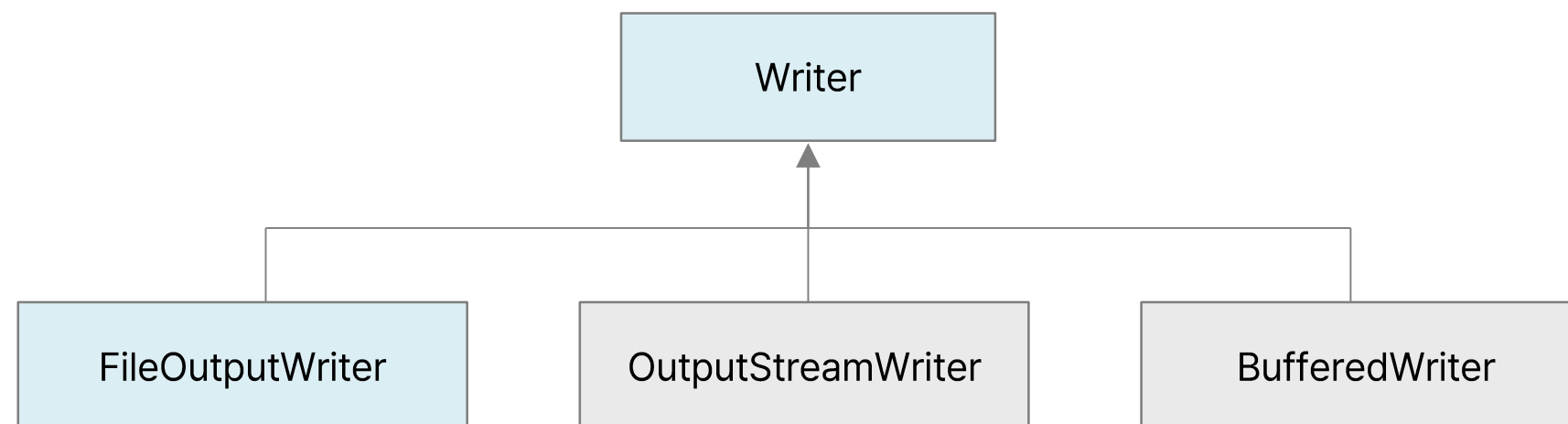


스트림 클래스	설명	입력 대상
FileReader	파일에서 문자 단위로 자료를 읽는다	파일
InputStreamReader	(보조스트림) 바이트 단위로 읽은 자료를 문자로 변환한다	
BufferedReader	(보조스트림) 배열로 문자를 한번에 읽는다	

메소드	설명
int read()	입력스트림으로부터 한 문자(2바이트)를 읽고, 읽은 값을 반환한다. <div>1byte1byte</div>
int read(char[] buf)	입력스트림으로부터 읽은 문자를 buf 배열에 저장한다.
int read(char[] buf, int off, int len)	입력스트림으로부터 len개 문자를 읽고, 읽은 문자를 buf 배열의 off위치에서부터 len 개까지 저장한다.
void close()	사용한 시스템 자원을 반납하고 입력스트림을 닫는다.

Writer

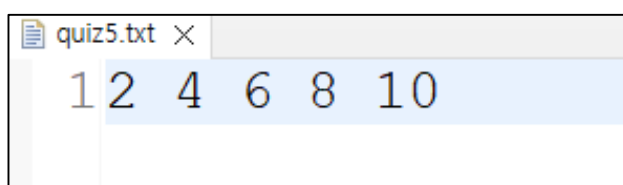
- 문자 단위로 출력하는 스트림 중 최상위 스트림이다.
- 출력 스트림의 계층 구조는 다음과 같다.



스트림 클래스	설명	출력 대상
FileOutputStreamWriter	파일에서 문자 단위로 자료를 출력한다	파일
OutputStreamWriter	(보조스트림) 바이트 단위로 출력한 자료를 문자로 변환한다	
BufferedWriter	(보조스트림) 배열로 문자를 한번에 출력한다	

5번.

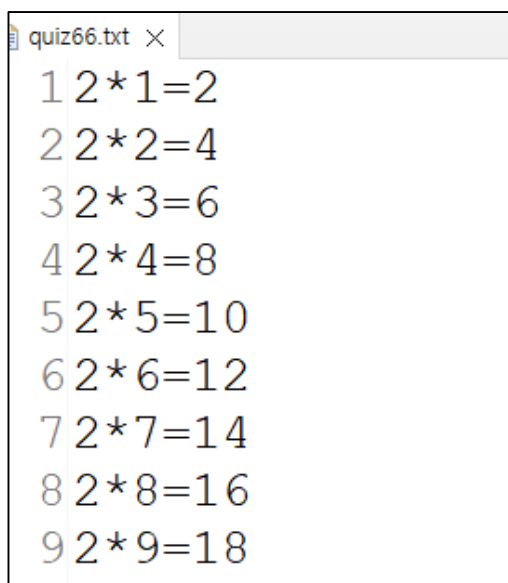
FileWriter를 사용하여 텍스트파일에 숫자 1부터 10 중에서 짝수를 출력하세요.



```
1 2 4 6 8 10
```

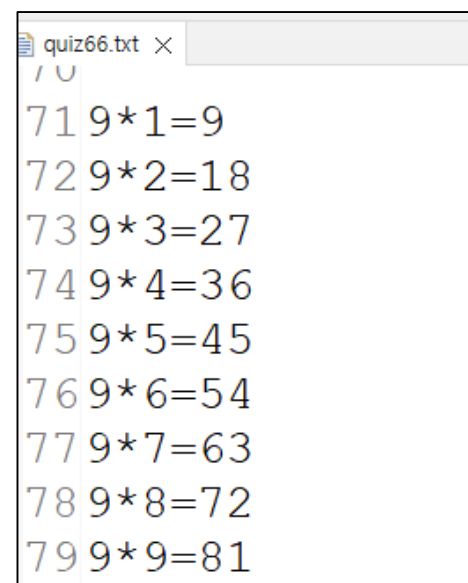
6번.

FileWriter를 사용하여 파일에 구구단을 출력하세요.



```
1 2*1=2  
2 2*2=4  
3 2*3=6  
4 2*4=8  
5 2*5=10  
6 2*6=12  
7 2*7=14  
8 2*8=16  
9 2*9=18
```

.....



```
1 9  
7 1 9*1=9  
7 2 9*2=18  
7 3 9*3=27  
7 4 9*4=36  
7 5 9*5=45  
7 6 9*6=54  
7 7 9*7=63  
7 8 9*8=72  
7 9 9*9=81
```

메소드	설명
<code>void write(int c)</code>	한 문자(2바이트)를 출력한다
<code>void write(char[] buf)</code>	buf배열의 내용을 출력한다
<code>void write(char[] buf, int off, int len)</code>	buf배열의 off 위치부터 len개까지 문자를 출력한다
<code>void write(String str)</code>	문자열을 출력한다
<code>void write(String str, int off, int len)</code>	문자열의 off위치부터 len개까지 문자를 출력한다
<code>void flush()</code>	출력스트림은 효율을 높이기 위해서 버퍼에 데이터를 모았다가 한번에 출력한다. 버퍼가 꽉 차면 자동으로 출력된다. 그전에 데이터를 출력하고 싶으면 flush() 메소드를 사용한다.
<code>void close()</code>	사용한 시스템 자원을 반납하고 출력 스트림을 닫는다.

Q. 이름을 보고 스트림의 종류를 유추하여 메모장에 적어보세요.

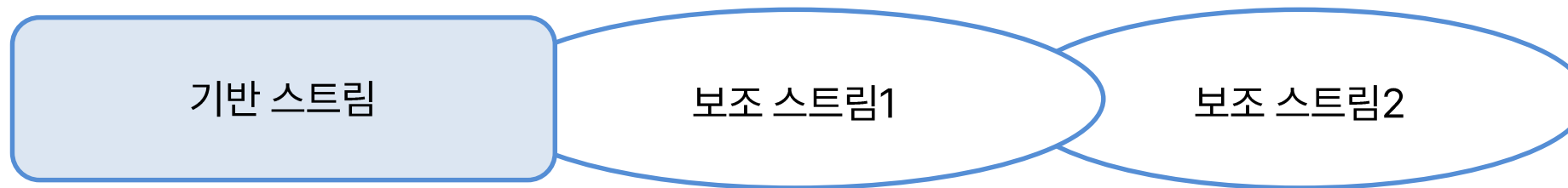
스트림	입력/출력	바이트/문자
FileInputStream		
FileOutputStream		
FileReader		
FileOutputStream		

보조 스트림

기반스트림과 보조스트림

스트림은 데이터를 직접 처리하는 스트림과 부가 기능을 제공하는 스트림으로 구분 한다.

- 기반 스트림: 입출력 대상에 연결되어 읽고 쓸 수 있음
- 보조 스트림: 기반 스트림에 붙어서 부가기능을 제공



종류	예시
기반스트림	FileInputStream, FileOutputStream, FileReader, FileWriter 등
보조스트림	InputStreamReader, OutputStreamWriter, BufferedInputStream, BufferedOutputStream 등

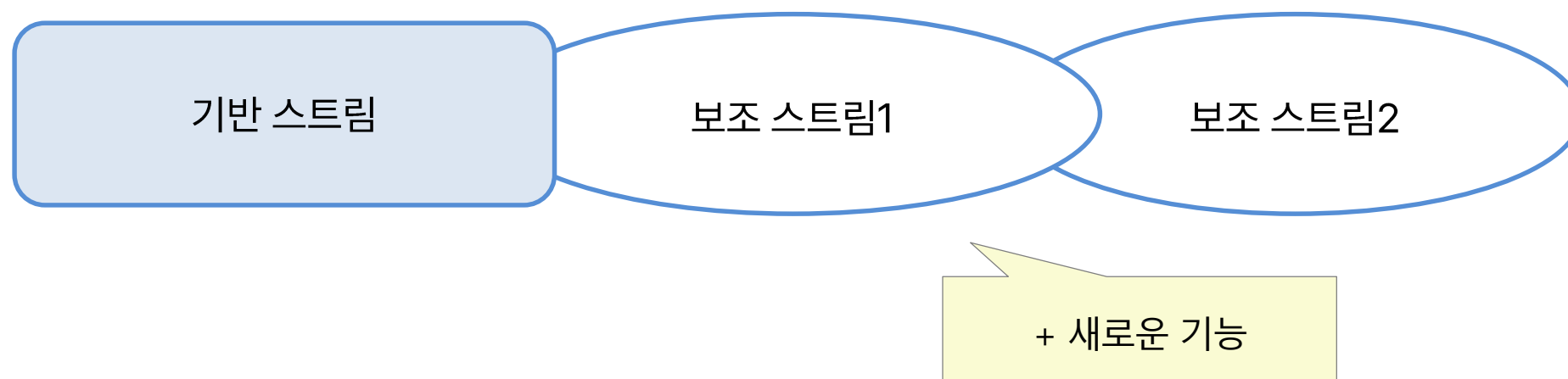
보조스트림이란?

- 기반스트림에 연결되어 편리한 기능을 제공하는 스트림이다.

보조스트림의 특징

- 기반스트림에 연결하려면 생성자 함수의 매개변수로 기반스트림을 입력해야 한다.
- 보조 스트림은 여러 번 사용할 수 있다.

보조스트림 변수 = new 보조스트림(**입출력스트림**)



보조 스트림

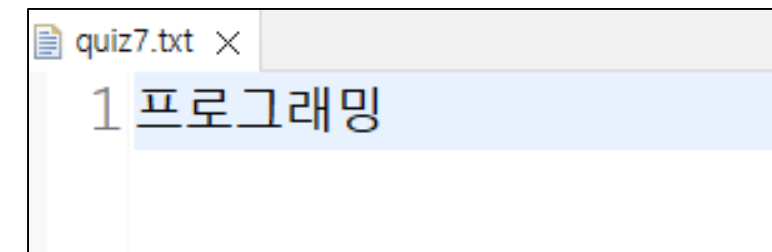
보조 스트림 클래스의 종류

구분	입력 보조 스트림	출력 보조 스트림
최상위 클래스	FilterInputStream	FilterOutputStream

하위 클래스	설명
InputStreamReader OutputStreamWriter	바이트 단위 스트림으로 한글문자를 읽거나 쓰면 깨진다. 해당 스트림은 바이트 단위 자료를 문자로 변환하는 기능을 제공한다
BufferedInputStream BufferedOutputStream	입출력스트림은 1바이트 또는 2바이트 단위로 자료를 읽고 쓴다. 용량이 큰 자료에 접근하면 그만큼 여러 번 입출력이 오가기 때문에 수행속도가 느려진다. 해당 스트림은 8kb 크기의 자료를 한번에 읽고 쓰는 기능을 제공한다.

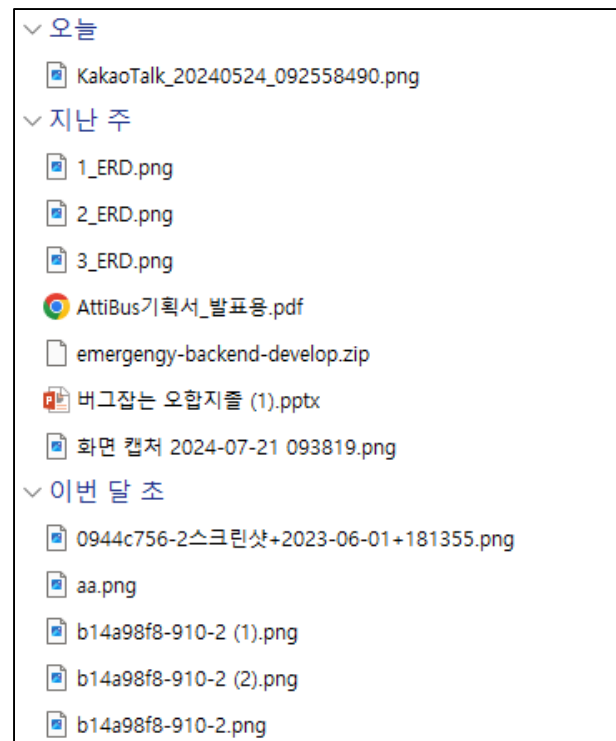
7번.

FileOutputStream와 보조스트림을 사용하여 파일에 "프로그래밍"을 출력하세요.



8번.

다운로드 폴더에 있는 파일의 목록을 읽어서 텍스트 파일에 출력하세요.



다운로드 폴더



텍스트 파일

9번.

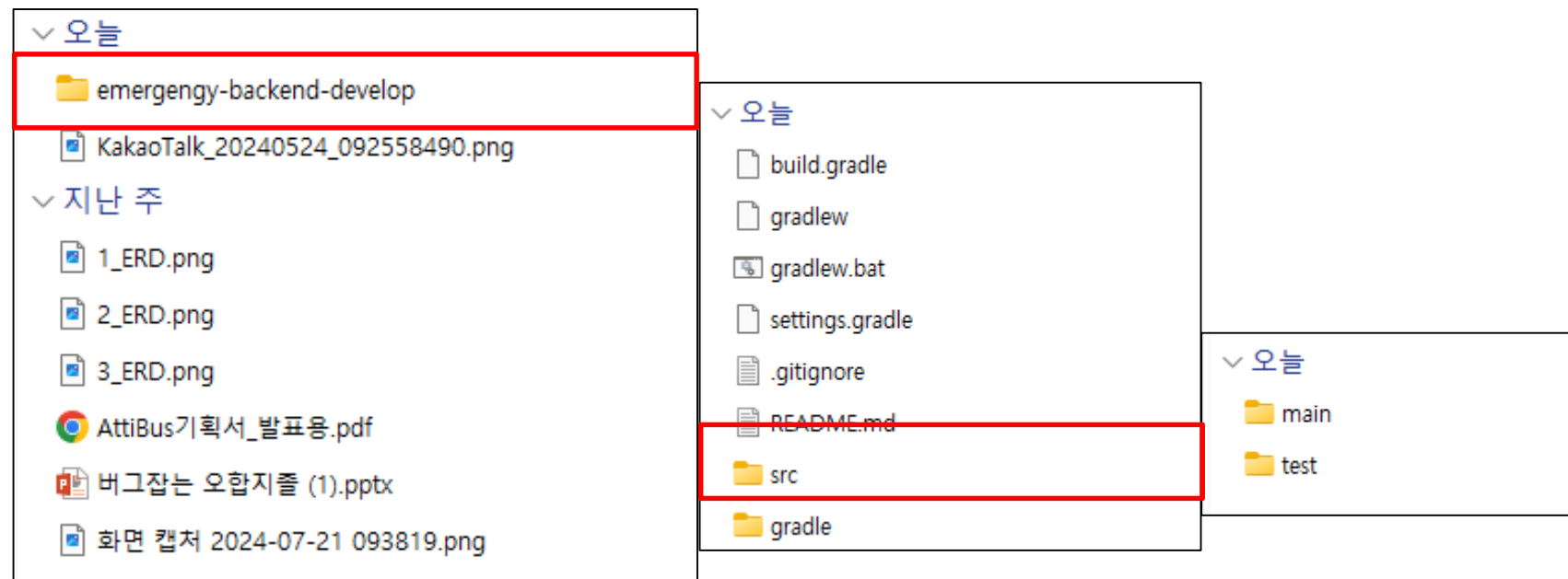
이전 문제를 함수 형태로 변경하세요.

10번.

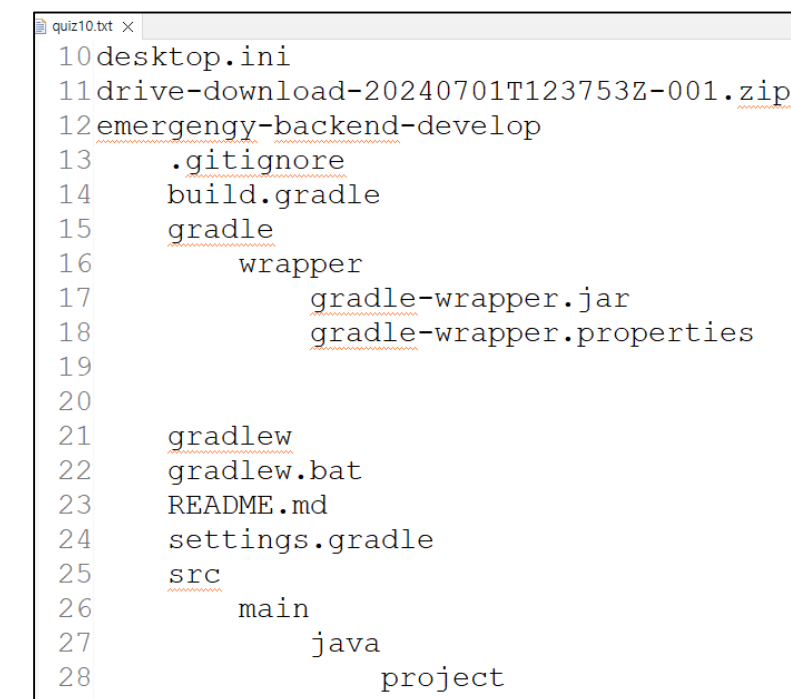
다운로드 폴더에 있는 파일의 목록을 읽어서 텍스트 파일에 출력하세요.

만약 하위 폴더가 있다면, 하위폴더의 파일목록도 출력하세요.

(Hint. 재귀함수를 활용하세요)



다운로드 폴더



텍스트 파일