

Binary File Copy 프로그램

경북대학교
소프트웨어융합과
배희호 교수
010-2369-4112
031-570-9600
hhbae@kbu.ac.kr

Binary File 복사

- Byte 단위 Stream을 이용하여 Binary File을 복사하는 Program을 작성하라.

Binary File 복사

- Binary File 복사 방법
 - 전통적인 방법
 - FileInputStream, FileOutputStream
 - BufferedInputStream, BufferedOutputStream
 - Files.copy()
 - FileChannel
- FileUtils 방식
 - Apache Commons IO

Binary File 복사(I)

```
public static void main(String[] args) {
    String inputfile = ".\\data\\notepad.exe";
    String outputfile = ".\\data\\note.exe";

    File file = new File(inputfile);    // 소스 파일
    if (file.exists()) {
        try {
            FileInputStream input = new FileInputStream(file);
            FileOutputStream output = new FileOutputStream(outputfile);
            int ch;
            int size = 0;
            while ((ch = input.read()) != EOF) {
                output.write(ch);
                size++;
            }
            input.close();
            output.close();
            System.out.printf("%n %d Bytes 파일 복사 성공", size);
        } catch (IOException e) {
            System.err.println(e.getMessage());
        }
    }
}
```

Binary File 복사(I)

```
} else {  
    System.out.printf("\n 입력파일 : %s가 없습니다", inputfile);  
}  
}
```

Binary File 복사(I-1)

```
public static void main(String[] args) {  
    String inputfile = ".\\data\\notepad.exe";  
    String outputfile = ".\\data\\note.exe";  
  
    File file = new File(inputfile);    // 소스 파일  
    if (file.exists()) {  
        try {  
            FileInputStream input = new FileInputStream(file);  
            FileOutputStream output = new FileOutputStream(outputfile);  
            byte[] buffer = new byte[1024];  
            int size;  
            int count = 0;  
            while ((size = input.read(buffer)) != EOF) {  
                output.write(buffer, 0, size);  
                count += size;  
            }  
            input.close();  
            output.close();  
            System.out.printf("%n %d Bytes 파일 복사 성공", count);  
        }  
    }  
}
```

Binary File 복사(I-1)

```
    } catch (IOException e) {  
        System.err.println(e.getMessage());  
    }  
} else {  
    System.err.printf("\n 입력파일 : %s가 없습니다", inputfile);  
}  
}
```

Binary File 복사(II)

```
public static void main(String[] args) {
    String inputfile = ".\\data\\notepad.exe";
    String outputfile = ".\\data\\note.exe";
    File file = new File(inputfile);    // 소스 파일
    if (file.exists()) {
        try {
            BufferedInputStream in = new BufferedInputStream(
                                    new FileInputStream(file));
            BufferedOutputStream out = new BufferedOutputStream(
                                    new FileOutputStream(outputfile));

            int ch;
            while ((ch = in.read()) != EOF) {
                out.write(ch);
            }
            in.close();
            out.close();
            System.out.println("파일 복사 성공");
        } catch (IOException e) {
            System.err.println(e.getMessage());
        }
    }
}
```


Binary File 복사(II)

```
}else {  
    System.out.printf("\n 입력파일 : %s가 없습니다", inputfile);  
}  
}
```

Binary File 복사(II -1)

```
public static void main(String[] args) {
    String inputfile = ".\\data\\notepad.exe";
    String outputfile = ".\\data\\note.exe";
    File file = new File(inputfile);    // 소스 파일
    if (file.exists()) {
        int size;
        try {
            BufferedInputStream in = new BufferedInputStream(
                                                new FileInputStream(file));
            BufferedOutputStream out = new BufferedOutputStream(
                                                new FileOutputStream(outputfile));

            byte[] buffer = new byte[1024];
            while ((size = in.read(buffer)) != EOF) {
                out.write(buffer, 0, size);
            }
            in.close();
            out.close();
            System.out.println("파일 복사 성공");
        } catch (IOException e) {
            System.err.println(e.getMessage());
        }
    }
}
```

Binary File 복사(III)

```
public static void main(String[] args) throws IOException {  
    String inputfile = ".\\data\\notepad.exe";  
    String outputfile = ".\\data\\note.exe";  
  
    Path source = Paths.get(inputfile);  
    if (Files.exists(source)) {  
        Path target = Paths.get(outputfile);  
  
        Files.copy(source, target, StandardCopyOption.REPLACE_EXISTING);  
    } else {  
        System.err.printf("입력파일 : %s가 없습니다", inputfile);  
    }  
}
```

Image File 복사

- Image File 복사 방법
 - FileInputStream/FileOutputStream 방식
 - 가장 고전적인 방식으로 InputStream, OutputStream을 사용하여 Byte 단위로 복사하는 방식
 - Files.Copy 방식
 - java.nio Package를 사용하여 쉽게 복사하는 방식
 - 기본적으로 nio 패키지가 io 패키지를 개선하기 위해 나왔고, Code 양도 대폭 줄일 수 있음
 - FileUtils 방식
 - Apache의 Commons-io Library를 사용하여 구현하는 방식
 - 별도의 Library를 추가해야 함

Image File 복사(I)

```
public static void main(String[] args) throws IOException {  
    String inputfile = ".\\data\\kbu.jpg";  
    String outputfile = ".\\data\\test1.jpg";  
  
    File file = new File(inputfile);  
    if (file.exists()) {  
        InputStream input = new FileInputStream(inputfile);  
        OutputStream output = new FileOutputStream(outputfile);  
        int data;  
        int count = 0;  
        while ((data = input.read()) != EOF) {  
            output.write(data);  
            count++;  
            System.out.print(".");  
        }  
        input.close();  
        output.close();  
        System.out.printf("%,d Bytes 복사", count);  
    } else {  
        System.out.printf("입력 파일 %s가 없습니다.\n", inputfile);  
    }  
}
```

Image File 복사(II)

```
public static void main(String[] args) throws IOException {
    String inputfile = ".\\data\\kbu.jpg";
    String outputfile = ".\\data\\test.jpg";

    File file = new File(inputfile);
    if (file.exists()) {
        FileInputStream input = new FileInputStream(inputfile);
        FileOutputStream output = new FileOutputStream(outputfile);
        int size = input.available();
        byte[] buf = new byte[size];
        int readCount = input.read(buf);
        output.write(buf, 0, readCount);
        System.out.println(inputfile + "을 " + outputfile + "로 복사하였다.");
        System.out.println(String.format("%,d Bytes 복사", size));
        input.close();
        output.close();
    } else {
        System.out.printf("입력 파일 %s가 없습니다.\n", inputfile);
    }
}
```

Image File 복사(III)

```
public static void main(String[] args) throws IOException {
    String inputfile = ".\\data\\kbu.jpg";
    String outputfile = ".\\data\\test.jpg";

    File file = new File(inputfile);
    if (file.exists()) {
        BufferedInputStream input = new BufferedInputStream(
                                    new FileInputStream(file));
        BufferedOutputStream output = new BufferedOutputStream(
                                    new FileOutputStream(outputfile));

        int ch;
        int count = 0;
        while ((ch = input.read()) != EOF) {
            count++;
            output.write(ch);
        }
    }
}
```

Image File 복사(III)

```
System.out.println(inputfile + "을 " + outputfile + "로 복사하였다. ");  
System.out.println(String.format("%,d bytes", count));  
input.close();  
output.close();  
} else {  
    System.out.printf("입력 파일 %s가 없습니다.\n", inputfile);  
}  
}
```


Image File 복사(IV)

```
public static void main(String[] args) {  
    String inputfile = "http://news.kyungbok.ac.kr/images/  
                        0609669218_041/2017.09.05.jpg";  
    String outputfile = ".\\data\\test.jpg";  
  
    try {  
        URL url = new URL(inputfile);  
        BufferedInputStream input = new BufferedInputStream(url.openStream());  
        BufferedOutputStream output = new BufferedOutputStream(  
                                        new FileOutputStream(outputfile));  
  
        int ch;  
        int count = 0;
```

Image File 복사(IV)

```
while ((ch = input.read()) != EOF) {  
    count++;  
    output.write(ch);  
}  
System.out.println(inputfile + "을 " + outputfile + "로 복사하였다. ");  
System.out.println(String.format("%,d bytes", count));  
input.close();  
output.close();  
} catch (IOException e) {  
    System.err.println(e.getMessage());  
}  
}
```

Image File 복사(V)

```
public static void main(String[] args) {  
    String inputfile = ".\\data\\kbu.jpg";  
    String outputfile = ".\\data\\test2.jpg";  
  
    File input = new File(inputfile);  
    File output = new File(outputfile);  
    try {  
        Files.copy(input.toPath(), output.toPath(),  
                    StandardCopyOption.REPLACE_EXISTING);  
        System.out.println("File Copy 완료");  
    } catch (IOException e) {  
        System.out.println(e.getMessage());  
    }  
}
```

Image File 복사(VI)

```
public static void main(String[] args) {  
    String inputfile = ".\\data\\kbu.jpg";  
    String outputfile = ".\\data\\test1.jpg";  
  
    File orgFile = new File(inputfile);  
    File outFile = new File(outputfile);  
    try {  
        FileUtils.copyFile(orgFile, outFile);  
        System.out.println("File Copy 완료");  
    } catch (IOException e) {  
        System.out.println(e.getMessage());  
    }  
}
```