

# Character 파일 입출력

경북대학교 스마트IT과

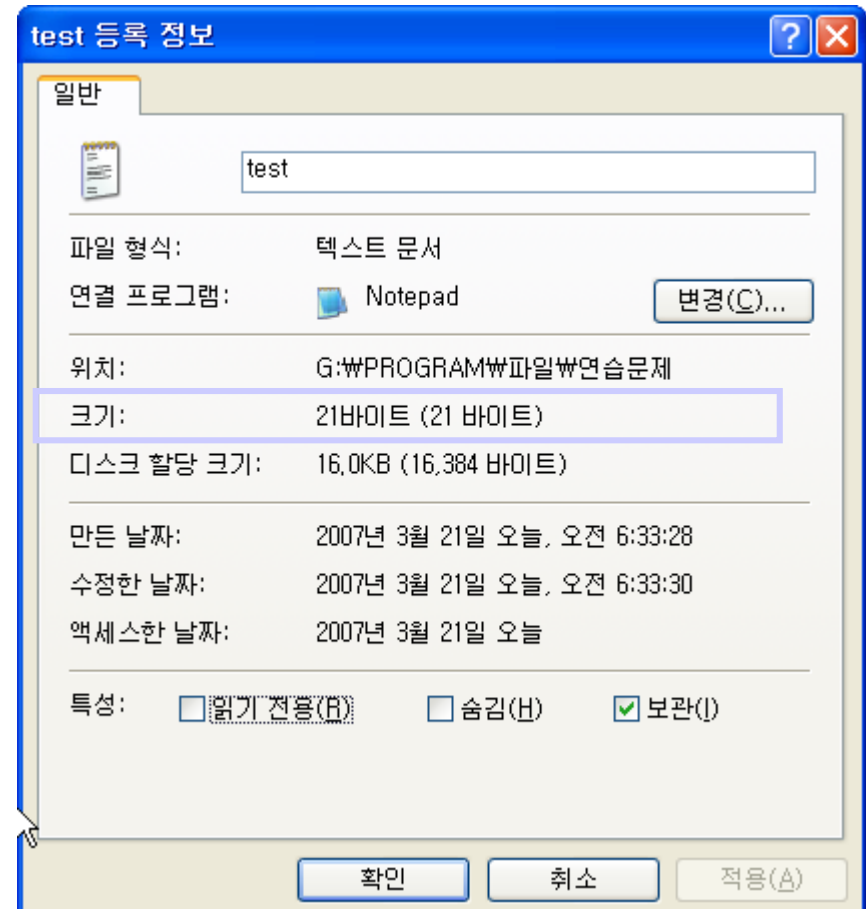
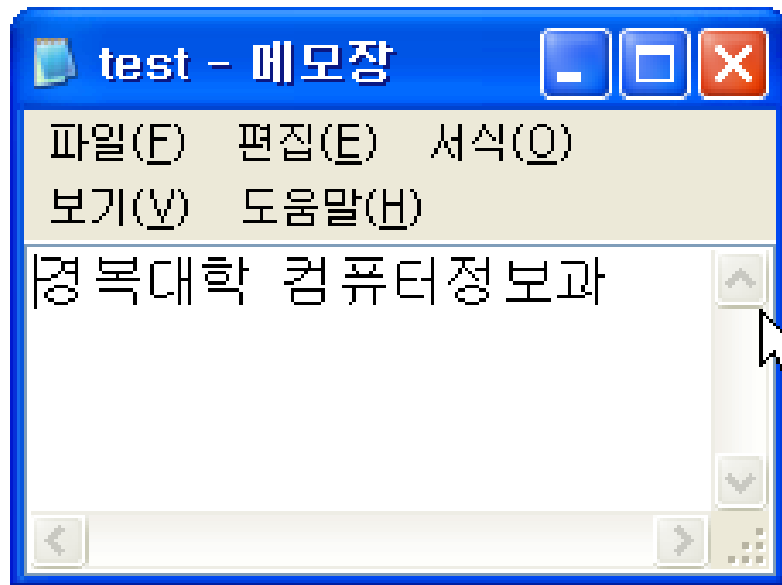
배희호 교수

010-2369-4112

031-570-9600

hhbae@kbu.ac.kr

# 파일의 크기를 구하기



# 파일의 크기를 구하기

```
public static void main(String[] args) throws IOException {  
    String source = "../data/test.txt";  
    int data;  
    int tot = 0; // 읽어온 바이트 수  
  
    File file = new File(source);  
    if (file.exists()) {  
        FileInputStream fis = new FileInputStream(file);  
  
        byte[] buf = new byte[1024];  
        while ((data = fis.read(buf)) != -1) {  
            tot += data;  
        }  
        System.out.println(String.format("%s 파일 : %,d Bytes", source, tot));  
        fis.close();  
    } else {  
        System.err.println("입력 파일이 존재하지 않아요");  
    }  
}
```

# 파일의 크기를 구하기

```
public static void main(String[] args) {  
    final String filename = ".\\data\\test.txt";  
    Path path = Paths.get(filename);  
    if (Files.exists(path)) {  
        try {  
            byte[] bytes = Files.readAllBytes(path);  
            System.out.printf("%s File : %,d Bytes", filename, bytes.length);  
        } catch (IOException e) {  
            System.err.println(e.getMessage());  
        }  
    } else {  
        System.out.printf("\n %s파일이 존재하지 않습니다.", filename);  
    }  
}
```

# Character 클래스

- Character 클래스는 기본형 중 char의 값을 객체로 포장 (Wrapping)
  - Character 형태의 객체에는 char의 단일 필드가 들어감
  - 이 클래스는 문자의 형태를 판별해서 문자를 대문자로부터 소문자(또는 그 반대)로 변환하기 위한 각종 메소드도 제공
- 생성자

생성자	설명
Character(char value)	Character 객체를 구축해 기본형 value 인수를 표현할 수 있도록 초기화

# Character 클래스

## ■ 메소드

메소드	설명
char charValue()	Character 오브젝트의 값을 반환
int compareTo (Character anotherCharacter)	2개의 Character를 수치로서 비교
int compareTo(Object o)	Character 객체를 지정된 객체와 비교
static int digit (char ch, int radix)	지정된 기수에서의 문자 ch의 수치로서의 값을 반환
boolean equals(Object o)	현재 객체와 지정된 객체가 같은지 비교
static char forDigit (int digit, int radix)	지정된 기수에서의 지정된 숫자의 문자 표현을 판정
static int getNumericValue (char ch)	Unicode 문자를 정수로 반환
static int getType(char ch)	문자의 카테고리를 나타내는 값을 반환 한글은 5를 반환
int hashCode()	Character의 해시 코드를 반환

# Character 클래스

## ■ 메소드

메소드	설명
static boolean isDefined(char ch)	문자가 Unicode로 정의되어 있는지를 판별
static boolean isDigit(char ch)	지정된 문자가 숫자인지를 판별
static boolean isIdentifierIgnorable(char ch)	지정된 문자가 Java 식별자 또는 Unicode 식별자 내 무시할 수 있는 문자인지 판별
static boolean isISOControl(char ch)	지정된 문자가 ISO 제어 문자인지를 판별
static boolean isJavaIdentifierPart(char ch)	지정된 문자가 Java 식별자의 선두 이외의 일부로서 사용할 수 있을지 판별
static boolean isJavaIdentifierStart(char ch)	지정된 문자가 Java 식별자의 선두의 문자로서 사용할 수 있을지 판별
static boolean isLetter(char ch)	지정된 문자가 범용 문자인지를 판별
static boolean isLetterOrDigit(char ch)	지정된 문자가 범용 문자인지 숫자인지를 판별

# Character 클래스

## ■ 메소드

메소드	설명
static boolean isLowerCase(char ch)	지정된 문자가 소문자인지를 판별
static boolean isSpaceChar(char ch)	지정된 문자가 Unicode의 스페이스 문자인지를 판별
static boolean isTitleCase (char ch)	지정된 문자가 타이틀 케이스 문자인지를 판별
static boolean isUnicodeIdentifierPart(char ch)	지정된 문자가 Unicode 식별자의 선두 이외의 일부로서 사용할 수 있을지 판별
static boolean isUnicodeIdentifierStart(char ch)	지정된 문자가 Unicode 식별자의 선두 문자로서 사용할 수 있을지 판별
static boolean isUpperCase(char ch)	지정된 문자가 대문자인지를 판별
static boolean isWhitespace(char ch)	지정된 문자가 Java의 기분에 따른 공백인지를 판별



# Character 클래스

## ■ 메소드

메소드	설명
static char toLowerCase(char ch)	지정된 문자를 대응하는 소문자에 매핑
String toString()	문자의 값을 표현하는 String 객체를 반환
static char toTitleCase(char ch)	문자의 인수를 대응하는 타이틀 케이스로 변환
static char toUpperCase (char ch)	문자의 인수를 대응하는 대문자로 변환

# Character 예제

- 다음과 같이 제공하는 파일을 이용하여 그 파일에 포함된 문자 종류별 수를 세는 프로그램을 작성해보자.

test.txt

This text is a  
test file.

1 2 3 4 5

경북대학교

# Character 예제

## ■ 실행 결과

입력 파일 이름 : *test.txt*

This text is a  
test file.

1 2 3 4 5

경북대학교

4 개 라인 파일

38 개 문자 파일

19 개 영문자 포함

5 개 숫자 포함

5 개 한글 포함

8 스페이스 포함

종료 코드 0(으)로 완료된 프로세스

■