1 Charater

• Wrapper class 중에 하나이며 한 문자를 관리하는 클래스이다.

리턴값	메소드	설명
int	compareTo (Character)	메소드를 호출하는 개체와 인수의 개체가 보유하고 값을 비교 값이 동일한 경우는 0을 돌려줌 메소드 호출 개체가 더 큰 값을 보유하고 있는 경우는 정의 정수를 돌려줌 인수에 주어진 객체가 더 작은 값을 보관 유지하고 있는 경우는 음의 정수 반환
boolean	equals(Object)	메소드를 호출하는 개체와 인수의 객체 비교 두 객체의 값이 동일한 경우에 true를 돌려줌
String	toString( )	Character 객체를 문자열로 변환하는 방법 인수의 객체는 길이 1개체의 값을 유지 ⇒ 한 문자열로 변환
char	charValue( )	Character 개체가 보유하고 값을 char 형으로 변환하여 반환

2 String

String 클래스는 자바에서 문자열을 관리하는 클래스이다.
중요 멤버로는 문자열을 소문자, 대문자로 변환한다거나 소문자, 대문자 여부, char[]로 리턴 하는 등의 메소드를 가진다.

반환 형식	메소드	개요
char	charAt (int index)	지정된 인덱스 위치에 있는 문자 돌려줌
boolean	equals (Object anObject)	이 캐릭터 라인과 지정된 오브젝트 비교
byte []	getBytes ()	String를 플랫폼의 디폴트의 문자 인코딩에 따라 바이트로 변환해서 결과를 새로운 바이트 배열에 저장
void	getChars (int srcBegin, int srcEnd, char [] dst, int dstBegin)	이 캐릭터 라인으로부터 카피 처의 문자 배열에 문자 카피
int	indexOf (int ch)	이 캐릭터 라인 내에서 지정된 문자가 최초로 출현하는 위치의 인덱스 돌려줌
int	indexOf (String str)	이 캐릭터 라인 내에서 지정된 부분 캐릭터 라인이 최초로 출현하는 위치의 인덱스 돌려줌
int	length ()	이 문자열의 길이 반환
String	replace (char oldChar, char newChar)	이 캐릭터 라인 내에 있는 모든 oldChar를 newChar에 치환한 결과 생성되는 새로운 캐릭터 라인 돌려줌

2 String

String 클래스는 자바에서 문자열을 관리하는 클래스이다.
중요 멤버로는 문자열을 소문자, 대문자로 변환한다거나 소문자, 대문자 여부, char[]로 리턴 하는 등의 메소드를 가진다.

반환 형식	메소드	개요
String	substring (int beginIndex, int endIndex)	이 캐릭터 라인의 부분 캐릭터 라인인 새로운 캐릭터 라인 돌려줌
char []	toCharArray ()	이 문자열을 새로운 문자 배열로 변환
String	toLowerCase ()	Locale.getDefault에 의해 돌려 주어지는 디폴트 로케일의 규칙을 사용해서 이 String 내의 모든 문자를 소문자로 변환
String	toString ()	이 오브젝트 (벌써 캐릭터 라인이다) 자신이 돌려줌
String	toUpperCase ()	Locale.getDefault에 의해 돌려 주어지는 디폴트 로케일의 규칙을 사용해서 이 String 내의 모든 문자를 대문자로 변환
String	trim ()	이 문자열의 끝에서 공백 제거

3 StringBuffer

String클래스는 인스턴스를 생성할 때 지정된 문자열을 변경할 수 없지만 StringBuffer클래스는 변경이 가능하다.

반환형	메소드	설명
int	length ()	문자의 길이 리턴
int	capacity ()	문자열 버퍼 수 (메모리 할당 영역) 리턴
int	charAt (int)	지정된 인덱스 번호 위치에 있는 문자 리턴
String String	substring (int) substring (int, int)	지정된 인덱스 번호 위치에서 끝까지의 문자열을 리턴, 두 번째 인수가 지정되는 경우, 제 1인수에서 두 번째 인수까지의 문자열 반환
StringBuffer	append (boolean) append (char) append (char []) append (char [], int, int) append (double) append (float) append (int) append (long) append (Object) append (String)	전화 StringBuffer 객체의 끝에 인수로 지정된 데이터를 추가 데이터는 문자열로 변환되고 나서 추가 char 형 배열의 두 번째 인자는 배열의 오프셋을 제 3인수는 오프셋 길이를 나타냄

3 StringBuffer

String클래스는 인스턴스를 생성할 때 지정된 문자열을 변경할 수 없지만 StringBuffer클래스는 변경이 가능하다.

반환형	메소드	설명
StringBuffer	delete (int, int)	전화 StringBuffer 객체의 문자열을 인수로 지정된 범위에서 삭제
StringBuffer	insert (int, boolean) insert (int, char) insert (int, char []) insert (int, char [], int, int) insert (int, double) insert (int, float) insert (int, int) insert (int, long) insert (int, Object) insert (int, String)	전화 StringBuffer 객체의 문자열에 두 번째 인수로 지정된 데이터를 삽입. 제 1인수로 삽입 위치를 지정 지정된 인덱스 번호 앞에 데이터가 삽입 선두에 문자를 삽입하려면 0을 지정 데이터 삽입 될 때 문자열로 변환 된 후 삽입 char 형 배열의 제 3인수는 배열의 오프셋을 제 4인수는 오프셋 길이를 나타냄
StringBuffer	replace (int, int, String)	StringBuffer 객체의 문자열을 인수로 지정된 범위에서 문자열로 바꿈
void	setLength (int)	문자열 버퍼 수 (메모리 할당 영역)을 설정

4 StringBuild

스레드에 안전하지 않기 때문에 단일 스레드 환경에서만 사용을 하며 StringBuffer는 다중 스레드에 사용을 한다.

리턴형	메소드	설명
StringBuilder	append (String s)	끝에 s 추가
StringBuilder	insert (int x, String s)	a 번째로 s 삽입
StringBuilder	replace (int a, int b, String s)	a 번째 ~ b 번째 문자 s로 대체
void	setCharAt (int a, char s)	a 번째 문자를 s로 대체
StringBuilder	delete (int a, int b)	a 번째 ~ b 번째 문자 삭제
StringBuilder	deleteCharAt (int a)	a 번째 문자 삭제
char	charAt (int a)	a 번째 문자를 char 형으로 얻음
int	indexOf (String s)	s가 가장 먼저 출현 인덱스 번호를 반환 (없으면 -1을 반환)
int	indexOf (String s, int a)	s가 a 번째부터 가장 먼저 출현, 인덱스 번호 반환 (없으면 -1을 반환)
int	lastIndexOf (String s)	s가 마지막에 출현하는 인덱스 번호 반환 (없으면 -1을 반환)

4 StringBuild

스레드에 안전하지 않기 때문에 단일 스레드 환경에서만 사용을 하며 StringBuffer는 다중 스레드에 사용을 한다.

리턴형	메소드	설명
int	lastIndexOf (String s, int a)	s가 a 번째보다 앞에서 가장 먼저 출현 인덱스 번호 반환 (없으면 -1을 반환)
int	length()	문자열의 길이 (문자 수)를 반환
StringBuilder	reverse()	문자열의 순서를 역순으로 리턴
String	substring (int a)	a 번째 이상을 String 형으로 반환
String	substring (int a, int b)	a 번째 ~ b 번째를 String 형으로 반환
void	toString ()	String 형으로 반환

5 문자열 분할

StringTokenizer 클래스, String의 split 메소드 또는 java.util.regex 패키지 (정규 표현식)를 이용해서 문자열을 분할할 수 있다.

① StringTokenizer 클래스 :

문자열을 지정된 구분 기호로 분리 할 수 있는 클래스.
java.util 팩키지에 있는 클래스
특히 CSV 데이터를 처리할 때 등 매우 편리함

② java.util.regex :

정규식은 문자, 기호를 이용하여 특정 문자 패턴을 표현하는 것을 말한다.
문자열이 패턴과 일치하는지 확인하거나 문자열의 문자 패턴과 일치하는 부분을 변경할 경우에 사용한다.

③ Pattern 클래스 :

compile 메소드 인수에 정규식을 지정하고 정규 표현식을 컴파일한다.
matcher 메소드 인수에 정규 표현식 처리의 대상이 되는 문자열을 지정하고 Matcher 클래스의 객체를 생성한다. split 메소드 인수에 지정된 문자열을 정규식에 따라 분할하고 분할된 문자열을 배열로 리턴 한다.

5 문자열 분할

StringTokenizer 클래스, String의 split 메소드 또는 java.util.regex 패키지 (정규 표현식) 을 이용해서 문자열을 분할할 수 있다.

④ Matcher 클래스 :

replaceAll 메소드 정규식과 일치하는 문자열을 인수로 지정한 문자열에 대체한다.
matches 메소드 문자열이 정규식과 일치하는지 확인한다.
find 메소드 문자열이 정규식과 일치하거나 순서대로 확인한다.