

기본 API

▶ String 관련 클래스

✓ String 클래스

문자열 값 수정 불가능, immutable(불변)

수정 시 수정된 문자열이 새로 할당 되어 새 주소를 넘김

✓ StringBuffer 클래스

문자열 값 수정 가능, mutable(가변)

수정, 삭제 등이 기존 문자열에 수정되어 적용

기본 16문자 크기로 지정된 버퍼를 이용하며 크기 증가 가능

쓰레드 safe기능 제공(성능 저하 요인)

✓ StringBuilder 클래스

StringBuffer와 동일하나 쓰레드 safe기능을 제공하지 않음

▶ Wrapper 클래스

Primitive Data Type을 객체화 해주는 클래스

Primitive Data Type	Wrapper Class
boolean	Boolean
byte	Byte
char	Character
short	Short
int	Integer
long	Long
float	Float
double	Double

▶ Wrapper 클래스

✓ String을 기본 자료형으로 바꾸기

```
byte b = Byte.parseByte("1");  
short s = Short.parseShort("2");  
int i = Integer.parseInt("3");  
long l = Long.parseLong("4");  
float f = Float.parseFloat("0.1");  
double d = Double.parseDouble("0.2");  
boolean bool = Boolean.parseBoolean("true");  
  
char c = "abc".charAt(0);
```

▶ Wrapper 클래스

✓ 기본 자료형을 String으로 바꾸기

```
String b = Byte.valueOf((byte)1).toString();  
String s = Short.valueOf((short)2).toString();  
String i = Integer.valueOf(3).toString();  
String l = Long.valueOf(4L).toString();  
String f = Float.valueOf(0.1f).toString();  
String d = Double.valueOf(0.2).toString();  
String bool = Boolean.valueOf(true).toString();  
String ch = Character.valueOf('a').toString();
```

▶ 날짜 관련 클래스

✓ Date 클래스

시스템으로부터 현재 날짜, 시간 정보를 가져와서 다룰 수 있게 만들어진 클래스
생성자 2개와 몇 개의 메소드만 사용 가능하고 나머지는 모두 deprecated
Calendar 클래스 혹은 GregorianCalendar 클래스 사용 권장

✓ 예시

```
Date today = new Date();  
//시스템으로부터 현재 날짜, 시간 정보를 가져와 기본 값으로 사용  
  
Date when = new Date(123456798L);  
//long형 정수 값을 가지고 날짜 시간 계산  
//1970년 1월 1일 0시 0분 0초를 기준으로 함
```

▶ 날짜 관련 클래스

✓ Calendar 클래스

Calendar클래스는 생성자가 protected이기 때문에 new연산자를 통해 객체 생성 불가능
getInstance() 메소드를 통해서 객체 생성

✓ GregorianCalendar 클래스

GregorianCalendar클래스는 Calendar클래스의 후손 클래스
년, 월, 일, 시, 분, 초 정보를 필드를 이용하여 다룰 수 있음

▶ Format 관련 클래스

✓ SimpleDateFormat 클래스

Date의 날짜, 시간 정보를 원하는 format으로 출력하는 기능 제공
java.text 패키지에 속해있음

✓ 예시

```
Date today = new Date();  
  
SimpleDateFormat ft = new SimpleDateFormat("yyyy-MM-dd");  
  
String ftToday = ft.format(today);  
//today에 포맷을 적용한 결과를 문자열로 리턴
```