

11장 기본 API 클래스



자바 API

◆ 자바 API (Application Programming Interface)

API는 라이브러리(library) 라고 부르며, 프로그램 개발에 자주 사용되는 클래스 및 인터페이스의 모음이다.

`$(JAVA_HOME)\jre\lib\rt.jar` 파일



자바 API

◆ java.lang 패키지

자바 프로그램의 기본적인 클래스들의 집합이다. import 없이 사용할 수 있다.

클래스		용도
Object		자바 클래스의 최상위 클래스
System		표준 입력 장치(키보드)로 부터 데이터를 입력. 표준 출력 장치(모니터)로 출력. 자바 가상 기계 종료. 현재 시스템 시간 조회. 시스템 프로퍼티 및 환경변수 조회.
Class		클래스의 정보를 담고 있는 클래스
String		문자열을 저장하고 문자열 관련 정보를 제공
StringBuffer, StringBuilder		문자열을 저장하고 내부 문자열 조작 기능을 제공
Math		수학 함수를 제공
Wrapper	Byte, Short, Character , Integer, Floate, Dou ble, Boolean, Long	기본 타입의 데이터를 갖는 객체를 만들 때 사용 문자열을 기본 타입으로 변환 입력값 검사에 사용

자바 API

◆ java.util 패키지

자바 프로그램의 Utility 클래스들의 집합이다.

클래스	용도
Arrays	배열을 조작(비교, 복사, 정렬, 찾기) 등의 기능 제공
Calendar	시스템의 날짜와 시간 정보를 제공
Date	날짜와 시간 정보를 저장하는 클래스
Objects	객체 비교, 널(null) 여부 등을 조사
StringTokenizer	특정 문자로 구분된 문자열을 추출하는 기능 제공
Random	난수를 생성

자바 API

◆ Object 클래스

자바의 모든 클래스는 명시되어 있지 않지만 Object 클래스를 상속받는다.

모든 클래스의 최상위 클래스로, 참조값에 연결된 메모리와 관련있는 메서드들로 이루어져 있다.

Object 클래스의 메서드	설 명
Object clone()	객체 자신의 복사본을 반환한다.
boolean equals(Object obj)	객체 자신과 객체 obj가 같은 객체인지 알려준다.
void finalize()	객체가 소멸될 때 가비지 컬렉터에 의해 자동으로 호출된다. 이때 수행되어야 하는 코드가 있을 때 오버라이딩 한다.
Class getClass()	객체 자신의 클래스 정보를 담고 있는 Class 인스턴스를 반환한다.
int hashCode()	객체 자신의 해시코드를 반환한다.
String toString()	객체 자신의 정보를 문자열로 반환한다.
void notify()	객체 자신을 사용하려고 기다리는 스레드를 하나만 깨운다.
void notifyAll()	객체 자신을 사용하려고 기다리는 모든 스레드를 깨운다.
void wait()	다른 스레드가 notify(), notifyAll()을 호출할 때까지 현재 스레드를 무한히 또는 지정된 시간 동안 기다리게 한다.

자바 API

◆ System 클래스

운영체제의 일부 기능을 이용할 수 있다.

프로그램 종료, 키보드 입력, 모니터 출력, 메모리 정리, 현재 시간 읽기, 시스템 프로퍼티 읽기, 환경 변수 읽기 등이 가능하다.

메서드	설명
exit()	프로세스 강제 종료. System.exit(0);
gc()	쓰레기 수집기 실행. 메모리 정리
currentTimeMillis()	시스템의 현재 시각을 밀리세컨드(1/1000초) 단위로 읽기.
nanoTime()	시스템의 현재 시각을 나노세컨드(1/10^9초) 단위로 읽기.
getProperty(String key)	시스템 프로퍼티 읽기
getenv(String name)	환경 변수 읽기

자바 API

◆ System 클래스

운영체제의 일부 기능을 이용할 수 있다.

프로그램 종료, 키보드 입력, 모니터 출력, 메모리 정리, 현재 시간 읽기, 시스템 프로퍼티 읽기, 환경 변수 읽기 등이 가능하다.

메서드	설명
exit()	프로세스 강제 종료. System.exit(0);
gc()	쓰레기 수집기 실행. 메모리 정리
currentTimeMillis()	시스템의 현재 시각을 밀리세컨드(1/1000초) 단위로 읽기.
nanoTime()	시스템의 현재 시각을 나노세컨드(1/10^9초) 단위로 읽기.
getProperty(String key)	시스템 프로퍼티 읽기
getenv(String name)	환경 변수 읽기

자바 API

◆ StringBuffer & StringBuilder 클래스

String 클래스의 인스턴스는 값을 변경할 수 없는 불변객체(immutable object)

StringBuffer 클래스는 버퍼 공간을 가지고 문자열을 변경, 조작할 수 있는 기능을 제공한다.

메소드		기능
StringBuffer	append(String str)	문자열 뒤에 str을 덧붙임
StringBuffer	insert(int offset, String str)	문자열의 offset 위치에 str을 삽입
StringBuffer	delete(int start, int end)	Start부터 end - 1까지의 부분 문자열을 삭제
StringBuffer	deleteCharAt(int index)	Index 위치에 있는 하나의 문자를 삭제
int	capacity()	버퍼 크기를 리턴
void	ensureCapacity(int miniumCapacity)	버퍼 크기를 늘리는 기능
void	trimToSize()	버퍼 크기를 줄이는 기능

자바 API

◆ 날짜와 시간에 관련된 클래스들

Date 클래스 : 날짜를 표현하는 클래스

Calender 클래스 : 현재 시스템의 날짜와 시간을 관리하는 클래스

SimpleDateFormat 클래스 : 날짜와 시간을 포맷하는 기능을 가진 클래스

```
public static void main(String[] args){
    SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
    Date date = new Date();
    String time = sdf.format(date);
    System.out.println(time);
}
```

```
public static void main(String[] args){
    SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
    Calendar calender = Calendar.getInstance();
    String time = sdf.format(calender.getTime());
    System.out.println(time);
}
```

패턴	의미	출력
G	연대(BC, AD)	AD
y	년도	2017
M	월(1~12)	12
w	해당 년도의 몇 번째 주(1~53)	52
W	해당 월의 몇 번째 주(1~5)	5
D	해당 연도의 몇 번째 일(1~366)	364
d	해당 월의 몇 번째 일(1~31)	30
F	해당 월의 몇 번째 요일(1~5)	5
E	요일(월~일)	Sat
a	오전/오후(AM, PM)	PM
H	시간(0~23)	21
h	시간(1~12)	9
K	시간(0~11)	9
k	시간(1~24)	21
m	분(0~59)	31
s	초(0~59)	8
S	1/1000초(0~999)	297
Z	타임존	+0900
z	타임존(RFC 822)	KST

자바 API

◆ 숫자 형식 클래스

DecimalFormat 클래스

기호	의미	패턴 예	1234567.89 변환결과
0	10 진수(빈자리는 0으로 채움)	0 0.0 0000000000.00000	1234568 1234567.9 0001234567.89000
#	10 진수(빈자리는 채우지 않음)	# #.# #####.#####	1234568 1234567.9 1234567.89
.	소수점	#.0	1234567.9
-	음수 기호	+#.0 -#.0	+1234567.9 -1234567.9
,	단위 구분	#,###.0	1,234,567.9
E	지수 문자	0.0E0	1.2E6
;	양수와 음수의 패턴을 모두 기술할 경우, 패턴 구분자	+#,###;-#,###	+1,234,568(양수일 때) -1,234,568(음수일 때)
%	%문자	## %	123456789 %
\u00A4	통화 기호	\u00A4 #,###	\ 1,234,568

자바 API

◆ 문자 형식 클래스

MessageFormat 클래스

: 문자열에 데이터가 들어갈 자리를 표시해두고, 프로그램이 실행하면서 동적으로 데이터를 삽입해 문자열을 완성시킬 수 있다.

```
public static void main(String[] args){
    String message = "회원 ID : {0} \n회원 이름: {1} \n회원 전화: {2}";
    String result = MessageFormat.format(message, "a001", "정찬웅", "010-7398-7332");
    System.out.println(result);
}
```

```
public static void main(String[] args){
    String message = "회원 ID : {0} \n회원 이름: {1} \n회원 전화: {2}";
    Object[] arguments = {"a001", "정찬웅", "010-7398-7332"};
    String result = MessageFormat.format(message, arguments);
    System.out.println(result);
}
```

자바 API

◆ 수학적 계산에 사용되는 클래스

Math 클래스 : 수학에서 사용되는 여러가지 함수들을 메서드로 제공하는 클래스

Random 클래스 : 난수를 발생시키는 클래스

```
public static void main(String[] args) {
    Random random = new Random(); // 랜덤 객체 생성(디폴트 시드값 : 현재시간)
    random.setSeed(73987332); // 시드값 설정을 따로 할수도 있음

    System.out.println("0~n 까지의 랜덤 정수 리턴 : " + random.nextInt(10));
    System.out.println("무작위 int 리턴 : " + random.nextInt());
    System.out.println("무작위 boolean 값 : " + random.nextBoolean());
    System.out.println("무작위 long 값 : " + random.nextLong());
    System.out.println("무작위 float 값 : " + random.nextFloat());
    System.out.println("무작위 double 값 : " + random.nextDouble());
    System.out.println("무작위 정규 분포의 난수 값 : " + random.nextGaussian());
}
```

시드값

난수를 만드는 알고리즘에 사용되는 값

자바 API

◆ Math 클래스

메소드	설명	예제 코드	리턴 값
int abs(int a) double abs(double a)	절대값	int v1 = Math.abs(-5); double v2 = Math.abs(-3.14);	v1 = 5; v2 = 3.14
double ceil(double a)	올림값	double v3 = Math.ceil(5.3); double v4 = Math.ceil(-5.3);	v3 = 6.0; v4 = -5.0;
double floor(double a)	버림값	double v5 = Math.floor(5.3); double v6 = Math.floor(-5.3);	v5 = 5.0; v6 = -6.0;
int max(int a,int b)	최대값	int v7 = Math.max(5,9); double v8 = Math.max(5.3, 2.5);	v7 = 9; v8 = 5.3;
int min(int a, int b)	최소값	int v9 = Math.min(5,9);	v9 = 5;
double random()	랜덤값	double v10 = Math.random();	0.0<= v10 <1.0
double rint(double a)	가까운 정수 의 실수값	double v11 = Math.rint(5.3);	v11 = 5.0;
long round(double a)	반올림값	long v12 = Math.round(5.3);	v12 = 5;

자바 API

◆ 정규 표현식과 Pattern 클래스

Pattern 클래스 : 문자열을 정규 표현식으로 검증하는 기능

기호	설명
[]	[abc] a, b, c 중 하나의 문자
	[^abc] a, b, c 이외의 하나의 문자
	[a-zA-Z] a~z, A~Z 중 하나의 문자
\d	한 개의 숫자. [0-9]와 동일
\s	공백
\w	한 개의 알파벳 또는 한 개의 숫자 [a-zA-Z_0-9]와 동일
?	없음 또는 한 개
*	없음 또는 한 개 이상
+	한 개 이상
{n}	정확히 n개
{n,}	최소한 n개
{n, m}	n개에서부터 m개까지
()	<u>그룹핑</u>

자바 API

◆ 정규 표현식과 Pattern 클래스

기호	설명
^	문자열의 시작
\$	문자열의 종료
.	임의의 한 문자
*	앞 문자가 없을 수도, 무한정 많을 수도 있음
+	앞 문자가 하나 이상
?	앞 문자가 없거나 하나있음
[]	문자의 집합, 범위를 나타냄. 두 문자 사이는 -기호로 범위를 나타낸다. []내에 ^가 선행하여 존재하면 not을 나타낸다.
{}	횟수 또는 범위를 나타냄
()	소괄호 안의 문자를 하나의 문자로 인식
	패턴 안에서 or 연산을 수행할 때 사용
\s	공백 문자
\S	공백 문자가 아닌 나머지 문자
\w	알파벳이나 숫자
\W	알파벳이나 숫자를 제외한 문자
\d	숫자 [0-9]와 동일
\D	숫자를 제외한 모든 문자