

# 기본 패키지

**java.util**

# 패키지

- 상호 관련 있는 클래스와 인터페이스를 한곳에 묶어 놓은 것
- 자바는 개발자가 편리하게 프로그래밍할 수 있도록 기본 패키지를 제공

| 패키지         | 설명                            |
|-------------|-------------------------------|
| java.awt    | 그래픽을 처리하는 API                 |
| java.io     | 입출력을 스트림하는 API                |
| java.lang   | 자바 프로그램의 필수 API               |
| java.math   | 수학과 관련된 API                   |
| java.net    | 네트워크를 처리하는 API                |
| java.text   | 날짜, 시간, 문자열 등 지역화를 처리하는 API   |
| java.time   | JDK 8이 지원하는 날짜 및 시간을 처리하는 API |
| java.util   | 날짜, 리스트, 벡터 등 유틸리티 API        |
| javax.swing | 스윙 컴포넌트 API                   |

# 패키지

## 주요기본 패키지

| 패키지         | 설명                            |
|-------------|-------------------------------|
| java.awt    | 그래픽을 처리하는 API                 |
| java.io     | 입출력을 스트림하는 API                |
| java.lang   | 자바 프로그램의 필수 API               |
| java.math   | 수학과 관련된 API                   |
| java.net    | 네트워크를 처리하는 API                |
| java.text   | 날짜, 시간, 문자열 등 지역화를 처리하는 API   |
| java.time   | JDK 8이 지원하는 날짜 및 시간을 처리하는 API |
| java.util   | 날짜, 리스트, 벡터 등 유틸리티 API        |
| javax.swing | 스윙 컴포넌트 API                   |

# java.util 패키지

- 날짜, 시간, 리스트, 벡터, 해시 테이블, 컬렉션 등 다양한 유틸리티 클래스와 인터페이스를 제공
- java.util 패키지가 제공하는 주요 클래스

| 클래스             | 설명                           |
|-----------------|------------------------------|
| Arrays          | 배열을 비교, 복사, 정렬 등 조작할 때 사용한다. |
| Calendar        | 날짜와 시간 정보가 필요할 때 사용한다.       |
| Date            | 밀리초 단위의 현재 시각이 필요할 때 사용한다.   |
| StringTokenizer | 특정 문자로 구분된 문자열을 뽑아낼 때 사용한다.  |
| Random          | 난수가 필요할 때 사용한다.              |

# java.util 패키지

## Calendar 클래스

- 지역이나 문화에 따라 달력을 표시하는 방식이 다르기 때문에 추상 클래스로 되어 있음
- 표준 달력을 사용 객체 생성

```
Calendar now = Calendar.getInstance();
```

```
1 package sec03;
2
3 import java.util.Calendar;
4
5
6 public class CalendarDemo {
7     public static void main(String[] args) {
8         Date now = new Date();
9         System.out.println(now);
10
11         Calendar c = Calendar.getInstance();
12         System.out.println(c);
13
14         System.out.println(c.get(Calendar.YEAR));
15         System.out.println(c.get(Calendar.MONTH) + 1);
16
17         System.out.println(c.get(Calendar.DAY_OF_MONTH));
18         System.out.println(c.get(Calendar.DAY_OF_WEEK));
19         System.out.println(c.get(Calendar.WEEK_OF_YEAR));
20         System.out.println(c.get(Calendar.WEEK_OF_MONTH));
21
22         System.out.println(c.get(Calendar.HOUR));
23         System.out.println(c.get(Calendar.HOUR_OF_DAY));
24         System.out.println(c.get(Calendar.MINUTE));
25     }
26 }
```

Problems @ Javadoc Declaration Console

<terminated> CalendarDemo [Java Application] /Library/Java/JavaVirtualMachines/zulu-16

Tue Nov 30 19:07:50 KST 2021

java.util.GregorianCalendar[time=1638266870322,areFieldsSet=true,2021

11

30

3

49

5

7

19

7



# java.util 패키지

## StringTokenizer 클래스

- 문자열을 토큰으로 분리하는 데 사용
- 토큰은 공백이나 줄 바꿈 등 구분자를 사용해 문자열을 분리
- StringTokenizer 클래스의 주요 생성자

| 생성자                                     | 설명                                                |
|-----------------------------------------|---------------------------------------------------|
| StringTokenizer(String s)               | 주어진 문자열을 기본 구분자로 파싱한 StringTokenizer 객체를 생성한다.    |
| StringTokenizer(String s, String delim) | 주어진 문자열을 delim 구분자로 파싱한 StringTokenizer 객체를 생성한다. |

- 주요 메서드

| 메서드                     | 설명                      |
|-------------------------|-------------------------|
| int countTokens()       | 남아 있는 토큰의 개수를 반환한다.     |
| boolean hasMoreTokens() | 남아 있는 토큰이 있는지 여부를 반환한다. |
| String nextToken()      | 다음 토큰을 꺼내 온다.           |

```
1 package sec03;
2
3 import java.util.StringTokenizer;
4
5 public class StringTokenizerDemo {
6     public static void main(String[] args) {
7         String s = "of the people, by the people, for the people";
8
9         StringTokenizer st = new StringTokenizer(s, " ,");
10
11         System.out.println(st.countTokens());
12
13         while (st.hasMoreTokens()) {
14             System.out.print "[" + st.nextToken() + " ] ");
15         }
16     }
17 }
```

Problems Javadoc Declaration Console

<terminated> StringTokenizerDemo [Java Application] /Library/Java/JavaVirtualMachines/zulu-16.jdk/Contents/

9

[of] [the] [people] [by] [the] [people] [for] [the] [people]



```

1-
2
3
4
5 public class MyAge {
6
7- public static int getAge(String sBirthYear){
8     Calendar current = Calendar.getInstance();
9
10    int birthYear = 
11    int currentYear = current.get(Calendar.YEAR);
12    int age = currentYear - birthYear;
13    return age;
14 }
15
16- public static void main(String[] args) {
17    // TODO Auto-generated method stub
18
19    int count = 0;
20    int myAge = 0;
21
22    Scanner sc = 
23    System.out.print("input your birthday : ");
24    String str = sc.next();
25
26    StringTokenizer st = new 
27    while (st.hasMoreTokens()) {
28        String tok = st.nextToken();
29        if (count == 0)
30            count++;
31    }
32    System.out.println( str + " : " + myAge);
33 }
34 }
35 }
36

```

 Problems
  Javadoc
  Declaration
  Console
 

```

<terminated> MyAge [Java Application] /Library/Java/JavaVirtualMachines/zulu-16.jdk/Conter
input your birthday : 1967/01/16
1967/01/16 : 55

```



# java.util 패키지

## Random 클래스

- Random 클래스의 주요 생성자

| 생성자                 | 설명                            |
|---------------------|-------------------------------|
| Random( )           | Random 객체를 생성한다.              |
| Random(long seed)   | 주어진 시드를 사용하는 Random 객체를 생성한다. |
| String nextToken( ) | 다음 토큰을 꺼내 온다.                 |

- 주요 메서드

| 메서드                     | 설명                                   |
|-------------------------|--------------------------------------|
| boolean nextBoolean( )  | 논리 타입 난수를 발생시킨다.                     |
| double nextDouble( )    | 0.0~1.0 사이의 double 타입 난수를 발생시킨다.     |
| float nextFloat( )      | 0.0~1.0 사이의 float 타입 난수를 발생시킨다.      |
| double nextGaussian( )  | 평균, 표준편차가 0.0 및 1.0인 정규분포 난수를 발생시킨다. |
| int nextInt( )          | int 타입의 난수를 발생시킨다.                   |
| int nextInt(int bound)  | 0~n 사이의 int 타입 난수를 발생시킨다.            |
| long nextLong( )        | long 타입의 난수를 발생시킨다.                  |
| void setSeed(long seed) | 시드 값을 설정한다.                          |

```
1 package sec03;
2
3 import java.util.Random;
4
5 public class RandomDemo {
6     public static void main(String[] args) {
7         Random r = new Random();
8
9         for (int i = 0; i < 5; i++)
10             System.out.print(r.nextInt(100) + " ");
11     }
12 }
```

Problems @ Javadoc Declaration Console ✕

<terminated> RandomDemo [Java Application] /Library/Java/JavaVirtualMachines/zulu-11.0.2-b12.1-jdk-8.0.520-b12.1

66 80 9 87 83