

멘토시리즈 파이썬2

(패키지)

Chapter 02

패키지

Chapter02의 학습목표

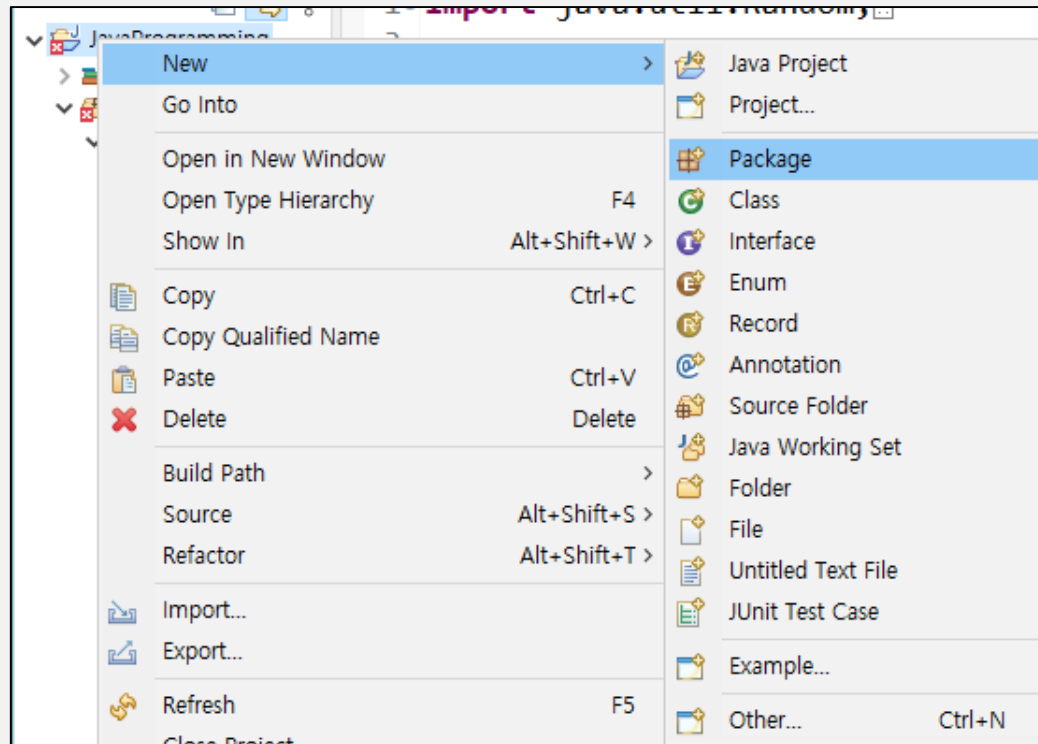
- 패키지의 생성과 사용법에 대해 학습한다.
- 자바에서 쓰이는 패키지와 클래스에 대해 알아본다.

패키지

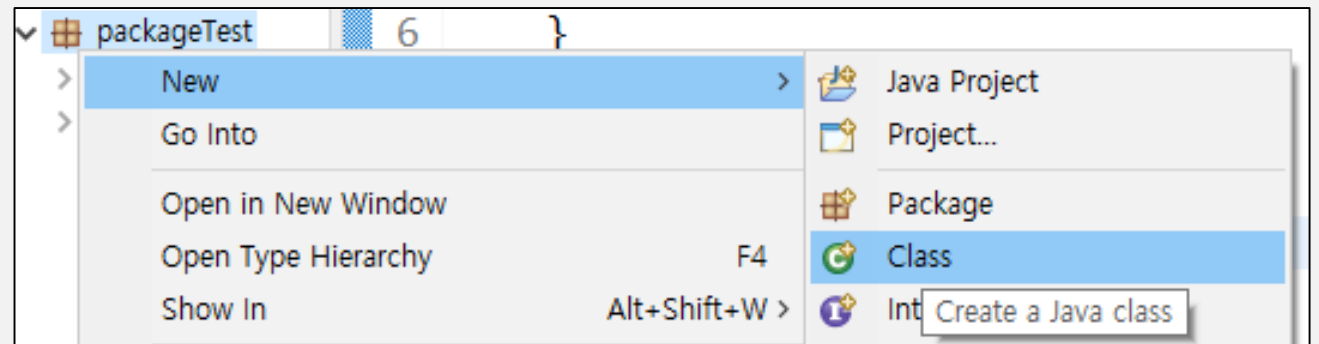
패키지(Package)

- 클래스를 모아 놓은 일종의 집합
- 유지보수 및 관리를 용이하게 하기 위해서 관련 있는 클래스끼리 묶어서 분리하는 일종의 디렉터리

새로운 package 생성



새로운 package안에 새 class 생성



패키지

Import

- 다른 패키지에 있는 클래스를 사용하고 싶을 때는 클래스의 이름으로 참조

default 패키지의 Test클래스

```
import packageTest.A_Class;
import packageTest.B_Class;

public class Test{
    public static void main(String[] args) {
        A_Class a_Class = new A_Class();
        a_Class.Print_A();

        B_Class b_Class = new B_Class();
        b_Class.Print_B();
    }
}
```

package 패키지의 Test클래스

```
package packageTest;

public class A_Class {
    public void Print_A() {
        System.out.println("A-class");
    }
}
```

```
package packageTest;

public class B_Class {
    public void Print_B() {
        System.out.println("B-class");
    }
}
```

패키지

Java에서 지원하는 Package

- 자바의 기본 패키지는 java로 시작하며 확장 패키지는 javax로 시작

패키지	설명
java.applet	애플릿을 생성하는데 필요한 클래스
java.awt	그래픽과 이미지를 위한 클래스
java.beans	자바빈즈 구조에 기초한 컴포넌트를 개발하는데 필요한 클래스
java.io	입력과 출력 스트림을 위한 클래스
java.lang	자바 프로그래밍 언어에 필수적인 클래스
java.math	수학에 관련된 클래스
java.net	네트워킹 클래스
java.nio	새로운 네트워킹 클래스
javax.xml	XML을 지원하는 패키지
java.security	보안 프레임워크를 위한 클래스와 인터페이스
java.sql	데이터베이스에 저장된 데이터를 접근하기 위한 클래스
java.util	날짜, 난수 생성기 등의 유틸리티 클래스
javax.swing	스윙 컴포넌트를 위한 클래스
javax.imageio	자바 이미지 I/O API

패키지

System 클래스

System

메소드명	설명	사용 예
currentTimeMills()	1970년 1월 1일을 기준으로 현재까지의 시간을 1/1000(ms)초 단위로 환산해서 반환 (long)	System.currentTimeMills()
nanoTime()	1970년 1월 1일을 기준으로 현재까지의 시간을 1/1,000,000,000(ns)초 단위로 환산해서 반환 (long)	System.nanoTime()
gc()	JVM에서 Garbage Collector를 실행시켜 사용하지 않는 객체를 제거하여 메모리 공간을 확보	System.gc()

패키지

java.time 패키지

- 기존의 Date, Calendar 클래스를 보완하는 패키지
- 편리하게 시간과 날짜를 표현가능

java.time

java.time의 하위 패키지	설명
java.time	날짜와 시간을 나타내는 LocalDate, LocalTime등을 포함한 패키지
java.time.format	날짜와 시간을 파싱하고 포매팅하는 API
java.time.chrono	여러 가지 달력 시스템을 사용할 수 있는 API
java.time.temporal	날짜와 시간을 연산하기 위한 API
java.time.zone	타임존을 지원하는 API

패키지

java.time 패키지

클래스 명	설명	
LocalDate	날짜 정보를 저장	LocalDate lD = LocalDate.now(); LocalDate mld = LocalDate.of(int year, int month, int dayOfMonth);
LocalTime	시간 정보를 저장	LocalTime lt = LocalTime.now(); LocalTime mlt = LocalTime.of(int hour, int minute, int second, int nanoOfSecond);
LocalDateTime	날짜 정보와 시간 정보 모두 저장	LocalDateTime lt = LocalDateTime.now(); LocalDateTime mlt = LocalDateTime.of(int hour, int minute, int second, int nanoOfSecond);

```
LocalDate ld = LocalDate.now();
LocalDate new_ld = LocalDate.of(2012, 3, 5);

System.out.println(ld);
System.out.println(new_ld);
```

```
LocalTime lt = LocalTime.now();
LocalTime new_lt = LocalTime.of(17, 55, 12);

System.out.println(lt);
System.out.println(new_lt);
```


패키지

java.time 패키지

시간과 날짜를 가져오기 위한 메소드

클래스 명	리턴 타입	메소드 명	설명
LocalDate	int	getYear()	년도
	Month	getMonth()	월(글자)
	int	getMonthValue()	월(숫자)
	int	getDayOfYear()	1년의 몇 번째 일
	int	getDayOfMonth()	월의 몇 번째 일
	DayOfWeek()	getDayOfWeek()	요일
	boolean	isLeapYear()	윤년 여부
LocalTime	int	getHour()	시간
	int	getMinute()	분
	int	getSecond()	초
	int	getNano()	나노초

패키지

java.time 패키지

시간과 날짜를 더하기/빼기 위한 메소드

메소드	설명	메소드	설명
minusYears(long)	년도 빼기	plusWeeks(long)	주 더하기
minusMonth(long)	월 빼기	minusHours(long)	시간 빼기
minusDays(long)	일 빼기	minusMinutes(long)	분 빼기
minusWeeks(long)	주 빼기	minusSeconds(long)	초 빼기
plusYears(long)	년도 더하기	minusNanos(long)	나노 초 빼기
plusMonths(long)	월 더하기	plusHours(long)	시간 더하기
plusDays(long)	일 더하기	plusMinutes(long)	분 더하기

시간과 날짜를 변경하기 위한 메소드

메소드	설명	메소드	설명
withYears(long)	년도 변경	withHours(long)	시간 변경
withMonth(long)	월 변경	withMinutes(long)	분 변경
withDayOfMonth(int)	월의 일 변경	withSeconds(long)	초 변경
withDayOfYear(int)	년의 일 변경	withNanos(long)	나노 초 변경

출처 : <https://programmers.co.kr/learn/courses/30/lessons/12901?language=java>

java.time 패키지 실전 문제

2016년

문제 설명

2016년 1월 1일은 금요일입니다. 2016년 a월 b일은 무슨 요일일까요? 두 수 a ,b를 입력받아 2016년 a월 b일이 무슨 요일인지 리턴하는 함수, solution을 완성하세요. 요일의 이름은 일요일부터 토요일까지 각각 `SUN,MON,TUE,WED,THU,FRI,SAT`

입니다. 예를 들어 a=5, b=24라면 5월 24일은 화요일이므로 문자열 "TUE"를 반환하세요.

제한 조건

- 2016년은 윤년입니다.
- 2016년 a월 b일은 실제로 있는 날입니다. (13월 26일이나 2월 45일같은 날짜는 주어지지 않습니다)

입출력 예

a	b	result
5	24	"TUE"

패키지

Math 클래스

- 수학에 관련된 기능이 정의되어 있는 클래스

Math

메소드 명	반환 타입	설명
random()	double	0.0 이상 1.0 미만의 범위에서 임의의 double형 값을 하나 생성하여 반환
abs(기본자료형 a)	double	전달된 값의 절대값을 반환
ceil(double a)	double	전달된 double형 값의 소수 부분이 존재하면 소수 부분을 무조건 올리고 반환
floor(double a)	double	전달된 double형 값의 소수 부분이 존재하면 소수 부분을 무조건 버리고 반환
round(double a) round(float a)	long, int	전달된 값을 소수점 첫째 자리에서 반올림한 정수를 반환
max(기본자료형 a, 기본자료형 b)	전달된 두 값의 타입과 같은 타입	전달된 두 값을 비교하여 큰 값을 반환
min(기본자료형 a, 기본자료형 b)	전달된 두 값의 타입과 같은 타입	전달된 두 값을 비교하여 작은 값을 반환
pow(double a, double b)	double	a의 b제곱 값을 반환
sqrt(double a)	double	전달된 double형 값의 제곱근 값을 반환

패키지

BigDecimal 클래스

- Java에서 숫자를 정밀하게 저장하고 표현할 수 있는 방법
- 정확한 소수점 연산을 해야 할 때 사용

java.math.BigDecimal

종류	종류 및 설명	사용 예
정의	BigDecimal 객체 = new BigDecimal(String);	BigDecimal big1 = new BigDecimal("0.01");
비교 연산	객체.equals(객체2)	big1.equals(big2);
사칙 연산	객체.add(객체2) //더하기 객체.subtract(객체2) //빼기 객체.multiply(객체2) //곱하기 객체.divide(객체2, 자리값, 반올림정책) //나누기	big1.add(big2); big1.subtract(big2); big1.multiply(big2); big1.divide(big2, 6, RoundingMode.HALF_EVEN)
상수	BigDecimal.ZERO BigDecimal.ONE BigDecimal.TEN	-

패키지

Format 클래스

- 숫자를 형식화하는 클래스
- 원하는 패턴을 지정하여 값을 표현

java.text.DecimalFormat

기호	의미
#	있으면 출력
0	없어도 0으로 채움
.	소수점
-	음수 기호를 붙임
,	콤마를 넣음
E	지수기호
%	퍼센트

```
import java.text.DecimalFormat;

public class Test{
    public static void main(String[] args) throws InterruptedException {
        //프로그램이 종료될 때 객체가 사라지도록 true로 설정
        String[] pattern = {
            "###.###",
            "000,000",
            "-###,###",
            "000000.00%"
        };

        double[] arr = {1.3, 3.33, 124.243, 242};
        for (int i = 0; i < pattern.length; i++) {
            DecimalFormat d = new DecimalFormat(pattern[i]);
            for (int j = 0; j < arr.length; j++) {
                System.out.println(d.format(arr[j]));
            }
        }
    }
}
```

수고하셨습니다.