**고급 JAVA 프로그래밍 보고서**

21.11.04

1826074 오현진

[3장. Generic & Functional Interface & Regular Expression]

< Generic >

1번 예제 코드

**public** **class** GenericEx<T> {

T val;

**void** set(T a) {

val = a;

}

T get() {

**return** val;

}

**public** **static** **void** main(String[] args) {

GenericEx<String> s = **new** GenericEx<String>();

s.set("hello");

System.***out***.println(s.get());

GenericEx<Integer> n = **new** GenericEx<Integer>();

n.set(5);

System.***out***.println(n.get());

}

}

1번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

2번 예제 코드

**class** GStack<T>

{

**int** tos;

Object [] stck;

**public** GStack()

{

tos = 0;

stck = **new** Object [10];

}

**public** **void** push(T item)

{

**if**(tos == 10)

**return**;

stck[tos] = item;

tos++;

}

**public** T pop()

{

**if**(tos == 0)

**return** **null**;

tos--;

**return** (T)stck[tos];

}

}

**public** **class** GenericEx {

**public** **static** **void** main(String[] args) {

GStack<String> stringStack = **new** GStack<String>();

stringStack.push("Seoul");

stringStack.push("busan");

stringStack.push("LA");

**for**(**int** n=0; n<3; n++)

System.***out***.println(stringStack.pop());

GStack<Integer> intStack = **new** GStack<Integer>();

intStack.push(1);

intStack.push(3);

intStack.push(5);

**for**(**int** n=0; n<3; n++)

System.***out***.println(intStack.pop());

}

}

2번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

3번 예제 코드

**class** MyUtil

{

**public** **static** <T **extends** String> **int** compare(T s1, T s2)

{

**return** s1.compareTo(s2);

}

**public** **static** <T **extends** Number> **int** compare(T n1, T n2)

{

**return** Double.*compare*(n1.doubleValue(), n2.doubleValue());

}

}

**public** **class** BoundedEx {

**public** **static** **void** main(String[] args) {

**int** result2 = MyUtil.*compare*(10, 20);

**int** result3 = MyUtil.*compare*(5.14, 4.5);

**int** result4 = MyUtil.*compare*(20, 20.5);

**int** result5 = MyUtil.*compare*(30L, 30L);

System.***out***.println("result2 = " + result2);

System.***out***.println("result3 = " + result3);

System.***out***.println("result4 = " + result4);

System.***out***.println("result5 = " + result5);

}

}

3번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

4번 예제 코드

**class** Person {}

**class** Worker **extends** Person {}

**class** Student **extends** Person {}

**class** HighStudent **extends** Student {}

**class** Course<T>

{

**private** String name;

**private** T[] students;

**public** Course(String name, **int** num)

{

**this**.name = name;

students = (T[])(**new** Object[num]);

}

**public** **void** add(T student) {

**for**(**int** i=0; i<students.length; i++) {

**if**(students[i] == **null**) {

students[i] = student;

**break**;

}

}

}

**public** String getName() { **return** name; }

**public** T[] getStudents() { **return** students; }

}

**public** **class** WildcardEx {

**public** **static** **void** registerCourse(Course<?> course) {}

**public** **static** **void** registerStudentCourse(Course<? **extends** Student> course) {}

**public** **static** **void** registerWorkerCourse(Course<? **super** Worker> course) {}

**public** **static** **void** main(String[] args) {

Course<Person> course\_P = **new** Course<Person>("일반인과정", 5);

Course<Worker> course\_W = **new** Course<Worker>("직장인과정", 5);

Course<Student> course\_S = **new** Course<Student>("대학생과정", 5);

Course<HighStudent> course\_H = **new** Course<HighStudent>("고등학생과정", 5);

*registerCourse*(course\_P);

*registerCourse*(course\_W);

*registerCourse*(course\_S);

*registerCourse*(course\_H);

*registerStudentCourse*(course\_S);

*registerStudentCourse*(course\_H);

*registerWorkerCourse*(course\_P);

*registerWorkerCourse*(course\_W);

}

}

< Functional Interface >

5번 예제 코드

**import** java.util.function.\*;

**public** **class** FuncInterfaceEx {

**public** **static** **void** main(String[] args) {

Consumer<String> stringConsumer = str -> System.***out***.println(str);

BiConsumer<String, String> biConsumer = (t,u) -> System.***out***.println(t+u);

DoubleConsumer doubleConsumer = d -> System.***out***.println("java" + d);

ObjDoubleConsumer<String> objDoubleConsumer = (t,d) -> System.***out***.println(t+d);

stringConsumer.accept("Hello!");

biConsumer.accept("Hello ", "Java!");

doubleConsumer.accept(8.0);

objDoubleConsumer.accept("Java ", 8.0);

}

}

5번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

6번 예제 코드

**import** java.util.function.IntBinaryOperator;

**class** Calc

{

**public** **static** **int** add(**int** x, **int** y) { **return** x+y; }

}

**public** **class** LambdaEx {

**public** **static** **void** main(String[] args) {

IntBinaryOperator op1, op2;

op1 = (x,y) -> Calc.*add*(x, y);

op2 = Calc :: *add*;

System.***out***.println(op1.applyAsInt(3,7));

System.***out***.println(op2.applyAsInt(3,7));

}

}

6번 예제 실행창



7번 예제 코드

**import** java.util.function.BiFunction;

**import** java.util.function.Function;

**class** Person2

{

**private** String name;

**private** **int** age;

**public** Person2(String name) {

**this**.name = name;

}

**public** Person2(String name, **int** age)

{

**this**(name);

**this**.age = age;

}

}

**public** **class** LambdaEx2 {

**public** **static** **void** main(String[] args) {

Function<String, Person2> func1 = Person2 :: **new**;

Person2 man = func1.apply("Kim");

BiFunction<String, Integer, Person2> func2 = Person2 :: **new**;

Person2 woman = func2.apply("Kim", 24);

}

}

< Regular Expression >

8번 예제 코드

**public** **class** RegularEx {

**public** **static** **void** main(String[] args) {

String str = "Hello Java!";

System.***out***.println(str.matches("He"));

System.***out***.println(str.matches("(.\*)He(.\*)"));

}

}

8번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

9번 예제 코드

**public** **class** RegularEx2 {

**public** **static** **void** main(String[] args) {

String str = "The cat sat on the mat. And the cat is sleeping.";

String result1 = str.replaceAll("[Tt]he", "#");

String result2 = str.replaceAll("^[Tt]he", "#");

String result3 = str.replaceAll("cat", "#");

String result4 = str.replaceAll(".$", "#");

String result5 = str.replaceAll("\\bis\\b", "#");

System.***out***.println(result1);

System.***out***.println(result2);

System.***out***.println(result3);

System.***out***.println(result4);

System.***out***.println(result5);

}

}

9번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

10번 예제 코드

**public** **class** RegularEx0 {

**public** **static** **void** main(String[] args) {

String str = "123A99B987C77";

String[] result = str.split("[a-zA-Z]+");

**for**(String s : result)

System.***out***.println(s);

}

}

10번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

11번 예제 코드

**import** java.util.regex.Matcher;

**import** java.util.regex.Pattern;

**public** **class** RegularEx3 {

**public** **static** **void** main(String[] args) {

String sample = "The cat sat on the mat. And the cat is sleeping";

Pattern p = Pattern.*compile*("[Tt]he");

Matcher m = p.matcher(sample);

**while**(m.find())

{

System.***out***.println(m.group());

System.***out***.println("start index = " + m.start());

System.***out***.println("end index = " + Integer.*toString*(m.end()-1));

System.***out***.println();

}

}

}

11번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

12번 예제 코드

**import** java.util.regex.Pattern;

**public** **class** RegularEx4 {

**public** **static** **void** main(String[] args) {

String name = "홍길동";

String tel = "010-1234-5678";

String email = "test@naver.com";

String p1 = "^[가-힣]\*$";

String p2 = "^01(?:0|1|[6-9])-(?:\\d{3}|\\d{4})-\\d{4}$";

String p3 = "\\w+@\\w+\\.\\w+(\\.\\w+)?";

**boolean** result1 = Pattern.*matches*(p1, name);

**boolean** result2 = Pattern.*matches*(p2, tel);

**boolean** result3 = Pattern.*matches*(p3, email);

System.***out***.println("이름 : " + result1);

System.***out***.println("전화번호 : " + result2);

System.***out***.println("이메일 : " + result3);

}

}

12번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

[4장. Class와 Interface 고급]

< Reflection >

1번 예제 코드

**import** java.lang.reflect.Method;

**class** Example

{

**private** **void** print(**int** p, **int** q) { System.***out***.println(p+q); }

}

**public** **class** ReflectEx {

**public** **static** **void** main(String[] args) **throws** ClassNotFoundException {

Class a = Class.*forName*("Example");

Method[] methods = a.getDeclaredMethods();

**for**(Method method : methods)

{

System.***out***.println(method.getName());

Class[] params = method.getParameterTypes();

**for**(**int** i=0; i<params.length; i++)

System.***out***.println(params[i]);

System.***out***.println(method.getReturnType());

}

}

}

1번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

2번 예제 코드

**import** java.lang.reflect.\*;

**class** Example2

{

**public** Example2(**int** x) {}

**public** **void** print(**int** p, **int** q) {System.***out***.println(p+q);}

}

**public** **class** ReflectEx2

{

**public** **static** **void** main(String[] args)

{

**try** {

Class a = Class.*forName*("Example2");

Class[] paramTypes = **new** Class[2];

paramTypes[0] = Integer.***TYPE***;

paramTypes[1] = Integer.***TYPE***;

Method method = a.getMethod("print", paramTypes);

Example2 obj = **new** Example2(0);

Object[] argList = **new** Object[2];

argList[0] = **new** ~~Integer~~(3);

argList[1] = **new** ~~Integer~~(5);

Object rtn = method.invoke(obj, argList);

}

**catch**(Exception e) {System.***out***.print("오류");}

}

}

2번 예제 실행창



3번 예제 코드

**import** java.lang.reflect.\*;

**class** Example3

{

**public** Example3(**int** x) {}

}

**public** **class** ReflectEx3 {

**public** **static** **void** main(String[] args) **throws** ClassNotFoundException {

Class a = Class.*forName*("Example3");

Constructor[] cons = a.getDeclaredConstructors();

**for**(Constructor con : cons)

{

System.***out***.println(con.getName());

Class[] params = con.getParameterTypes();

**for**(**int** i=0; i<params.length; i++)

System.***out***.println(params[i]);

}

}

}

3번 예제 실행창



4번 예제 코드

**import** java.lang.reflect.\*;

**class** Example4

{

**public** Example4(**int** x, **int** y)

{

System.***out***.println("Constructor!!");

}

}

**public** **class** ReflectEx4 {

**public** **static** **void** main(String[] args) {

**try** {

Class a = Class.*forName*("Example4");

Class[] paramTypes = **new** Class[2];

paramTypes[0] = Integer.***TYPE***;

paramTypes[1] = Integer.***TYPE***;

Constructor cs = a.getConstructor(paramTypes);

Object[] argList = **new** Object[2];

argList[0] = **new** ~~Integer~~(3);

argList[1] = **new** ~~Integer~~(5);

Object rtn = cs.newInstance(argList);

}

**catch**(Exception e) {}

}

}

4번 예제 실행창



5번 예제 코드

**import** java.lang.reflect.\*;

**class** Example5

{

**private** **double** data = 3.14;

**public** Example5 (**int** x) {}

**private** **void** print(**int** p , **int** q) { System.***out***.print(p+q); }

}

**public** **class** ReflectEx5 {

**public** **static** **void** main(String[] args) **throws** ClassNotFoundException, IllegalArgumentException, IllegalAccessException {

Class a = Class.*forName*("Example5");

Field[] fields = a.getDeclaredFields();

**for**(Field field : fields)

{

System.***out***.println(field.getName());

System.***out***.println(field.getType());

field.setAccessible(**true**);

Example5 obj = **new** Example5(0);

System.***out***.println(field.get(obj));

field.setDouble(obj, 2.75);

System.***out***.println(field.get(obj));

}

}

}

5번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

< Annotation >

6번 예제 코드

**import** java.lang.annotation.\*;

@Inherited

@Documented

@Retention(RetentionPolicy.***RUNTIME***)

@Target({

ElementType.***PACKAGE***,

ElementType.***TYPE***,

ElementType.***CONSTRUCTOR***,

ElementType.***FIELD***,

ElementType.***METHOD***,

ElementType.***ANNOTATION\_TYPE***,

ElementType.***LOCAL\_VARIABLE***,

ElementType.***PARAMETER***,

ElementType.***TYPE\_PARAMETER***,

ElementType.***TYPE\_USE***

})

**public** **@interface** AnnotationEx

{

**public** **enum** DAY

{

***MON***, ***TUE***, ***WED***, ***TUR***, ***FRI***, ***SAT***, ***SUN***

}

String today() **default** "SUN";

**int** count() **default** 7;

DAY getday() **default** DAY.***SUN***;

}

7번 예제 코드

( 어노테이션 정의 - 필드 )

**import** java.lang.annotation.\*;

@Target(ElementType.***FIELD***)

@Retention(RetentionPolicy.***RUNTIME***)

**public** **@interface** InsertInt

{

**int** data() **default** 0;

}

( 클래스 정의 및 값 출력 – 필드 )

**import** java.lang.reflect.\*;

**class** AnnotationEx2

{

@InsertInt

**private** **int** myAge;

@InsertInt(data = 30)

**private** **int** age;

**public** **int** getAge() { **return** age; }

}

**public** **class** Annotation\_FieldEX {

**public** **static** **void** main(String[] args) {

AnnotationEx2 obj = **new** AnnotationEx2();

Field[] fields = obj.getClass().getDeclaredFields();

**for**(Field field : fields)

{

InsertInt an = field.getAnnotation(InsertInt.**class**);

**if**(an != **null**)

{

System.***out***.println(an.data());

}

}

}

}

7번 예제 실행창



8번 예제 코드

( 어노테이션 정의 – 메소드 )

**import** java.lang.annotation.\*;**abstract**

@Target(ElementType.***METHOD***)

@Retention(RetentionPolicy.***RUNTIME***)

**public** **@interface** InsertInt2 {

**int** data() **default** 0;

}

( 클래스 정의 및 값 출력 – 메소드 )

**import** java.lang.reflect.\*;

**class** AnnotationEx3

{

**private** **int** age;

@InsertInt2(data = 30)

**public** **int** getAge() { **return** age; }

}

**public** **class** Annotation\_MethodEX {

**public** **static** **void** main(String[] args) {

AnnotationEx3 obj = **new** AnnotationEx3();

Method[] methods = obj.getClass().getDeclaredMethods();

**for**(Method method : methods)

{

InsertInt2 an = method.getAnnotation(InsertInt2.**class**);

**if**(an != **null**)

{

System.***out***.println(an.data());

}

}

}

}

8번 예제 실행창



< Serialization >

9번 예제 코드

**import** java.io.Serializable;

**public** **class** SerializeEx **implements** Serializable {

**private** **static** **final** **long** ***serialVersionUID*** = 1L;

**private** String name;

**private** **int** age;

**public** SerializeEx(String name, **int** age)

{

**this**.name = name;

**this**.age = age;

}

**public** String toString()

{

**return** String.*format*("%s, %d", name, age);

}

}

10번 예제 코드

**import** java.io.\*;

**public** **class** SerializationEx {

**public** **static** **void** main(String[] args) **throws** Exception {

SerializeEx obj = **new** SerializeEx("홍길동", 24);

String path = "c:\\temp\\serialTest.dat";

FileOutputStream fos = **new** FileOutputStream(path);

ObjectOutputStream out = **new** ObjectOutputStream(fos);

out.writeObject(obj);

System.***out***.println("객체저장 완료");

}

}

10번 예제 실행창



11번 예제 코드

**import** java.io.\*;

**public** **class** DeSerializationEx {

**public** **static** **void** main(String[] args) **throws** Exception {

String path = "c:\\temp\\serialTest.dat";

FileInputStream fis = **new** FileInputStream(path);

ObjectInputStream in = **new** ObjectInputStream(fis);

SerializeEx temp = (SerializeEx)in.readObject();

System.***out***.println(temp);

}

}

11번 예제 실행창



[5장. Stream과 병렬처리]

< Stream >

1번 예제 코드

import java.util.\*;

import java.util.stream.Stream;

public class StreamEx {

public static void main(String[] args) {

List<String> list = Arrays.asList("이순신", "유관순","홍길동");

// Iterator 이용(JDK 7까지만..)

Iterator<String> it = list.iterator();

while(it.hasNext())

{

String name = it.next();

System.out.println(name);

}

// Stream 이용(JDK 8부터!)

Stream<String> stream = list.stream();

stream.forEach(name -> System.out.println(name));

}

}

1번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

2번 예제 코드

import java.util.\*;

import java.util.stream.Stream;

public class StreamEx2 {

public static void main(String[] args) {

List<String> list1 = Arrays.asList("98.3", "67.2", "50.0", "78");

List<Double> list2 = Arrays.asList(98.3, 67.2, 50.0, 78.0);

double avg1 = list1.stream()

.mapToDouble(num -> Double.parseDouble(num))

.average()

.getAsDouble();

double avg2 = list2.stream()

.mapToDouble(Double::doubleValue)

.average()

.getAsDouble();

System.out.println("평균점수 : " + avg1);

System.out.println("평균점수 : " + avg2);

}

}

2번 예제 실행창



3번 예제 코드

**import** java.util.\*;

**public** **class** FilteringEx {

**public** **static** **void** main(String[] args) {

List<String> list = Arrays.*asList*("이순신", "김철수", "홍길동", "김봉근", "김기자", "김봉근");

list.stream()

.distinct()

.forEach(n -> System.***out***.println(n));

System.***out***.println();

list.stream()

.filter(n -> n.startsWith("김"))

.forEach(n -> System.***out***.println(n));

System.***out***.println();

list.stream()

.distinct()

.filter(n -> n.startsWith("김"))

.forEach(n -> System.***out***.println(n));

}

}

3번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

4번 예제 코드

**import** java.util.\*;

**public** **class** flatMapEx {

**public** **static** **void** main(String[] args) {

List<String> inputList1 = Arrays.*asList*("Java lambda", "stream mapping example");

List<String> inputList2 = Arrays.*asList*("10 20 30", "40 50 60 70");

inputList1.stream()

.flatMap(data -> Arrays.*stream*(data.split(" ")))

.forEach(word -> System.***out***.println(word));

System.***out***.println();

inputList2.stream()

.flatMapToInt(data -> {

String[] strArr = data.split(" |, ");

**int**[] intArr = **new** **int**[strArr.length];

**for**(**int** i=0; i<strArr.length; i++)

intArr[i] = Integer.*parseInt*(strArr[i].trim());

**return** Arrays.*stream*(intArr);

})

.forEach(number -> System.***out***.println(number));

}

}

4번 예제 실행창

테이블이(가) 표시된 사진

자동 생성된 설명

5번 예제 코드

**import** java.util.Arrays;

**import** java.util.List;

**class** Student {

**private** String name;

**private** **int** score;

**public** Student(String name, **int** score) {

**this**.name = name;

**this**.score = score;

}

**public** **int** getScore() { **return** score; }

}

**public** **class** MapEx {

**public** **static** **void** main(String[] args) {

List<Student> list = Arrays.*asList*( **new** Student("홍길동", 90),

**new** Student("유관순", 80), **new** Student("이순신", 95));

list.stream()

.mapToInt(Student::getScore)

.forEach(score -> System.***out***.println(score));

}

}

5번 예제 실행창



6번 예제 코드

**import** java.util.\*;

**import** java.util.stream.\*;

**public** **class** MappingEX {

**public** **static** **void** main(String[] args) {

**int**[] intArr = { 1, 2, 3, 4, 5 };

IntStream intStream = Arrays.*stream*(intArr);

intStream

.asDoubleStream()

.forEach(data -> System.***out***.println(data));

intStream = Arrays.*stream*(intArr);

intStream

.boxed()

.forEach(obj -> System.***out***.print(obj.intValue()));

}

}

6번 예제 실행창

**텍스트이(가) 표시된 사진

자동 생성된 설명**

7번 예제 코드

**import** java.util.\*;

**import** java.util.stream.\*;

**class** Studente **implements** Comparable<Studente>

{

**private** String name;

**private** **int** score;

**public** Studente(String name, **int** score)

{

**this**.name = name;

**this**.score = score;

}

**public** **int** getScore() { **return** score; }

@Override

**public** **int** compareTo(Studente obj)

{

**return** Integer.*compare*(score, obj.getScore());

}

}

**public** **class** SortedEx {

**public** **static** **void** main(String[] args) {

IntStream intStream = Arrays.*stream*(**new** **int**[] {5, 3, 4, 1, 2});

intStream

.sorted()

.forEach(n -> System.***out***.print(n + " "));

System.***out***.println();

List<Studente> list = Arrays.*asList*(**new** Studente("홍길동", 90),

**new** Studente("유관순", 80), **new** Studente("이순신", 95));

list.stream()

.sorted()

.forEach(n -> System.***out***.print(n.getScore() + " "));

System.***out***.println();

list.stream()

.sorted(Comparator.*reverseOrder*())

.forEach(n -> System.***out***.print(n.getScore() + " "));

}

}

7번 예제 실행창

**텍스트이(가) 표시된 사진

자동 생성된 설명**

8번 예제 코드

**import** java.util.Optional;

**public** **class** OptionalEx {

**public** **static** **void** main(String[] args) {

String str = "test";

Optional<String> optStr1 = Optional.*ofNullable*(str);

optStr1.ifPresent(s -> System.***out***.println(s.charAt(0)));

Optional<String> optStr2 = Optional.*ofNullable*(**null**);

optStr2.ifPresent(s -> System.***out***.println(s.charAt(0)));

Optional<String> optStr = Optional.*ofNullable*(**null**);

String result = optStr.orElse("test");

System.***out***.println(result);

}

}

8번 예제 실행창

****

9번 예제 코드

**import** java.util.\*;

**public** **class** AggregateEx {

**public** **static** **void** main(String[] args) {

**long** count = Arrays.*stream*(**new** **int**[] {1, 2, 3, 4, 5})

.filter(n -> n%2==0).count();

System.***out***.println("2의 배수 개수 : " + count);

**long** sum = Arrays.*stream*(**new** **int**[] {1, 2, 3, 4, 5})

.filter(n -> n%2==0).sum();

System.***out***.println("2의 배수 개수 : " + sum);

**double** avg = Arrays.*stream*(**new** **int**[] {1, 2, 3, 4, 5})

.filter(n -> n%2==0).average().getAsDouble();

System.***out***.println("2의 배수 개수 : " + avg);

**int** max = Arrays.*stream*(**new** **int**[] {1, 2, 3, 4, 5})

.filter(n -> n%2==0).max().getAsInt();

System.***out***.println("2의 배수 개수 : " + max);

**int** first = Arrays.*stream*(**new** **int**[] {1, 2, 3, 4, 5})

.filter(n -> n%2==0).findFirst().getAsInt();

System.***out***.println("2의 배수 개수 : " + first);

}

}

9번 예제 실행창

**텍스트이(가) 표시된 사진

자동 생성된 설명**

< Parallel Processing >

10번 예제 코드

**import** java.util.\*;

**import** java.util.stream.\*;

**public** **class** StreamEx3 {

**public** **static** **void** print(String str)

{

System.***out***.println(str + " : " + Thread.*currentThread*().getName());

}

**public** **static** **void** main(String[] args) {

List<String> list = Arrays.*asList*("이순신", "유관순", "홍길동", "김봉근", "이기자", "차차차");

Stream<String> stream = list.stream();

stream.forEach(str -> *print*(str));

System.***out***.println();

Stream<String> pstream = list.parallelStream();

pstream.forEach(str -> *print*(str));

}

}

10번 예제 실행창

**텍스트이(가) 표시된 사진

자동 생성된 설명**

[6장. 날짜조작]

< 기본적인 날짜 다루기 >

1번 예제 코드

**import** java.util.\*;

**public** **class** DnTEx {

**public** **static** **void** main(String[] args) {

Date date1 = **new** Date();

System.***out***.println(date1);

Calendar calendar1 = Calendar.*getInstance*();

Date date2 = calendar1.getTime();

System.***out***.println(date2);

Calendar calendar2 = Calendar.*getInstance*(Locale.***KOREA***);

TimeZone krZone = TimeZone.*getTimeZone*("Asia/Seoul");

Calendar calendar3 = Calendar.*getInstance*(krZone);

}

}

1번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

2번 예제 코드

**import** java.util.\*;

**public** **class** DnTEx {

**public** **static** **void** main(String[] args) {

Calendar calendar = Calendar.*getInstance*();

**int** year = calendar.get(Calendar.***YEAR***);

**int** month = calendar.get(Calendar.***MONTH***) + 1;

**int** day = calendar.get(Calendar.***DAY\_OF\_MONTH***);

**int** hour = calendar.get(Calendar.***HOUR***);

**int** minute = calendar.get(Calendar.***MINUTE***);

**int** second = calendar.get(Calendar.***SECOND***);

**int** date = calendar.get(Calendar.***DAY\_OF\_WEEK***);

String strDate = **null**;

**switch**(date) {

**case** Calendar.***MONDAY***: strDate = "월"; **break**;

**case** Calendar.***TUESDAY***: strDate = "화"; **break**;

**case** Calendar.***WEDNESDAY***: strDate = "수"; **break**;

**case** Calendar.***THURSDAY***: strDate = "목"; **break**;

**case** Calendar.***FRIDAY***: strDate = "금"; **break**;

**case** Calendar.***SATURDAY***: strDate = "토"; **break**;

**case** Calendar.***SUNDAY***: strDate = "일"; **break**;

}

System.***out***.printf("%d-%02d-%02d(%s) %02d:%02d:%02d",

year, month, day, strDate, hour, minute, second);

}

}

2번 예제 실행창



3번 예제 코드

**import** java.text.ParseException;

**import** java.text.SimpleDateFormat;

**import** java.util.\*;

**public** **class** DnTEx {

**public** **static** **void** main(String[] args) {

SimpleDateFormat sdf1 = **new** SimpleDateFormat("yyyy'/'MM'/'dd HH':'mm");

**try** {

Date date = sdf1.parse("2020/07/22 13:11");

System.***out***.println(date);

} **catch**(ParseException e) {}

SimpleDateFormat sdf2 = **new** SimpleDateFormat("yyyy'년'MM'월'dd'일' E, a KK':'mm':'ss");

SimpleDateFormat sdf3 = **new** SimpleDateFormat("yyyy'/'MM'/'dd'('EEEE')', HH':'mm");

String str2 = sdf2.format(**new** Date());

String str3 = sdf3.format(**new** Date());

System.***out***.println(str2);

System.***out***.println(str3);

}

}

3번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

4번 예제 코드

**import** java.text.ParseException;

**import** java.text.SimpleDateFormat;

**import** java.util.\*;

**public** **class** DnTEx {

**public** **static** **void** main(String[] args) {

Calendar calendar = Calendar.*getInstance*();

Date date1 = calendar.getTime();

System.***out***.println(date1);

calendar.add(Calendar.***DATE***, 32);

Date date2 = calendar.getTime();

System.***out***.println(date2);

calendar.add(Calendar.***DATE***, -3);

Date date3 = calendar.getTime();

System.***out***.println(date3);

}

}

4번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

5번 예제 코드

**import** java.util.\*;

**public** **class** DnTEx {

**public** **static** **void** main(String[] args) {

Calendar calendar = Calendar.*getInstance*();

calendar.set(2016, Calendar.***FEBRUARY***, 1);

**int** day1 = calendar.getActualMaximum(Calendar.***DATE***);

System.***out***.println(day1);

calendar.set(2017, Calendar.***FEBRUARY***, 1);

**int** day2 = calendar.getActualMaximum(Calendar.***DATE***);

System.***out***.println(day2);

}

}

5번 예제 실행창



< Date & Time API >

6번 예제 코드

**import** java.time.\*;

**public** **class** DnTEx1 {

**public** **static** **void** main(String[] args) {

LocalDate date = LocalDate.*now*();

LocalTime time = LocalTime.*now*();

LocalDateTime datetime = LocalDateTime.*now*();

OffsetTime offTime = OffsetTime.*now*();

OffsetDateTime offDatetime = OffsetDateTime.*now*();

ZonedDateTime zoned1 = ZonedDateTime.*now*();

ZonedDateTime zoned2 = ZonedDateTime.*now*(ZoneId.*of*("Asia/Aden"));

System.***out***.println(date);

System.***out***.println(time);

System.***out***.println(datetime);

System.***out***.println(offTime);

System.***out***.println(offDatetime);

System.***out***.println(zoned1);

System.***out***.println(zoned2);

}

}

6번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

7번 예제 코드

**import** java.time.\*;

**public** **class** DntEx2 {

**public** **static** **void** main(String[] args) {

LocalDate date = LocalDate.of(2019, 3, 15);

LocalTime time = LocalTime.of(12, 25);

LocalDateTime datetime1 = LocalDateTime.of(date, time);

LocalDateTime datetime2 = LocalDateTime.of(2019, 3, 15, 12, 25);

OffsetDateTime offDatetime = OffsetDateTime.of(2019, 3, 15, 12, 25, 30, 0, ZoneOffset.ofHours(-9));

ZonedDateTime zoned = ZonedDateTime.of(2019, 3, 15, 12, 25, 30, 0, ZoneId.systemDefault());

System.out.println(date);

System.out.println(time);

System.out.println(datetime1);

System.out.println(datetime2);

System.out.println(offDatetime);

System.out.println(zoned);

}

}

7번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

8번 예제 코드

**import** java.time.\*;

**public** **class** DnTEx3 {

**public** **static** **void** main(String[] args) {

LocalDate date = LocalDate.*of*(2019, 3, 15);

LocalTime time = LocalTime.*of*(12, 25);

LocalDateTime datetime = LocalDateTime.*of*(date, time);

**int** year = datetime.getYear();

**int** month = datetime.getMonthValue();

**int** day = datetime.getDayOfMonth();

**int** hour = datetime.getHour();

**int** minute = datetime.getMinute();

**int** second = datetime.getSecond();

**int** nanoSecond = datetime.getNano();

**int** dayOfWeek = datetime.getDayOfWeek().getValue();

System.***out***.println(year);

System.***out***.println(month);

System.***out***.println(day);

System.***out***.println(hour);

System.***out***.println(minute);

System.***out***.println(second);

System.***out***.println(nanoSecond);

System.***out***.println(dayOfWeek);

}

}

8번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

9번 예제 코드

**import** java.time.LocalDate;

**import** java.time.LocalDateTime;

**import** java.time.LocalTime;

**public** **class** DnTEx4 {

**public** **static** **void** main(String[] args) {

LocalDate date = LocalDate.*of*(2019, 3, 15);

LocalTime time = LocalTime.*of*(12, 25);

LocalDateTime datetime = LocalDateTime.*of*(date, time);

LocalDateTime result1 = datetime

.withYear(2020)

.withMonth(2)

.withDayOfMonth(25);

LocalDateTime result2 = datetime

.withHour(11)

.withMinute(1)

.withSecond(1)

.withNano(999999);

System.***out***.println(result1);

System.***out***.println(result2);

}

}

9번 예제 실행창



10번 예제 코드

**import** java.time.\*;

**import** java.time.format.DateTimeFormatter;

**public** **class** DnTEx5 {

**public** **static** **void** main(String[] args) {

LocalDateTime datetime = LocalDateTime.*now*();

String format1 = datetime.format(DateTimeFormatter.***BASIC\_ISO\_DATE***);

String format2 = datetime.format(DateTimeFormatter.***ISO\_DATE\_TIME***);

String format3 = datetime.format(DateTimeFormatter.***ISO\_DATE***);

String format4 = datetime.format(DateTimeFormatter.***ISO\_TIME***);

ZonedDateTime zonedDatetime = ZonedDateTime.*now*();

String format5 = zonedDatetime.format(DateTimeFormatter.***ISO\_DATE\_TIME***);

String format6 = zonedDatetime.format(DateTimeFormatter.***ISO\_DATE***);

String format7 = zonedDatetime.format(DateTimeFormatter.***ISO\_TIME***);

System.***out***.println(format1);

System.***out***.println(format2);

System.***out***.println(format3);

System.***out***.println(format4);

System.***out***.println(format5);

System.***out***.println(format6);

System.***out***.println(format7);

}

}

10번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

11번 예제 코드

**import** java.time.\*;

**import** java.time.format.DateTimeFormatter;

**import** java.time.format.FormatStyle;

**public** **class** DnTEx7 {

**public** **static** **void** main(String[] args) {

DateTimeFormatter[] formatter = **new** DateTimeFormatter[8];

LocalDateTime datetime = LocalDateTime.*now*();

formatter[0] = DateTimeFormatter.*ofLocalizedDate*(FormatStyle.***FULL***);

formatter[1] = DateTimeFormatter.*ofLocalizedDate*(FormatStyle.***LONG***);

formatter[2] = DateTimeFormatter.*ofLocalizedDate*(FormatStyle.***MEDIUM***);

formatter[3] = DateTimeFormatter.*ofLocalizedDate*(FormatStyle.***SHORT***);

formatter[4] = DateTimeFormatter.*ofLocalizedTime*(FormatStyle.***MEDIUM***);

formatter[5] = DateTimeFormatter.*ofLocalizedTime*(FormatStyle.***SHORT***);

formatter[6] = DateTimeFormatter.*ofLocalizedDateTime*(FormatStyle.***MEDIUM***);

formatter[7] = DateTimeFormatter.*ofLocalizedDateTime*(FormatStyle.***SHORT***);

**for**(**int** i=0; i<formatter.length; i++)

System.***out***.println(datetime.format(formatter[i]));

}

}

11번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

12번 예제 코드

**import** java.time.\*;

**import** java.time.format.DateTimeFormatter;

**public** **class** DnTEx6 {

**public** **static** **void** main(String[] args) {

LocalDateTime datetime1 = LocalDateTime.*parse*("2019-12-24T11:59:30");

DateTimeFormatter formatter = DateTimeFormatter.*ofPattern*("yyyy/MM/dd HH:mm");

LocalDateTime datetime2 = LocalDateTime.*parse*("2020/12/31 12:30", formatter);

LocalDate date1 = LocalDate.*parse*("2019-08-15");

DateTimeFormatter dformatter = DateTimeFormatter.*ofPattern*("yyyy-MM-dd");

LocalDate date2 = LocalDate.*parse*("2019-08-15", dformatter);

LocalTime time1 = LocalTime.*parse*("12:30:20");

DateTimeFormatter tformatter = DateTimeFormatter.*ofPattern*("HH-mm-ss");

LocalTime time2 = LocalTime.*parse*("11-20-45", tformatter);

System.***out***.println(datetime1);

System.***out***.println(datetime2);

System.***out***.println(date1);

System.***out***.println(date2);

System.***out***.println(time1);

System.***out***.println(time2);

}

}

12번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명

13번 예제 코드

**import** java.time.\*;

**public** **class** DnTEx8 {

**public** **static** **void** main(String[] args) {

LocalDate date1 = LocalDate.*of*(2014, 8, 15);

LocalDate date2 = LocalDate.*of*(2018, 10, 25);

Period period = Period.*between*(date1, date2);

**int** years = period.getYears();

**int** months = period.getMonths();

**int** days = period.getDays();

System.***out***.println(years);

System.***out***.println(months);

System.***out***.println(days);

}

}

13번 예제 실행창



14번 예제 코드

**import** java.util.\*;

**import** java.time.\*;

**public** **class** DnTEx8 {

**public** **static** **void** main(String[] args) {

Date today = **new** Date();

Instant instant = today.toInstant();

LocalDateTime datetime1 = LocalDateTime.*ofInstant*(instant, ZoneId.*systemDefault*());

ZonedDateTime datetime2 = ZonedDateTime.*ofInstant*(instant, ZoneId.*of*("Asia/Seoul"));

System.***out***.println(datetime1);

System.***out***.println(datetime2);

LocalDateTime now = LocalDateTime.*now*();

ZonedDateTime zoned = ZonedDateTime.*now*();

Instant instant1 = now.toInstant(ZoneOffset.*ofHours*(9));

Instant instant2 = zoned.toInstant();

Date today1 = Date.*from*(instant1);

Date today2 = Date.*from*(instant2);

System.***out***.println(today1);

System.***out***.println(today2);

}

}

14번 예제 실행창

텍스트이(가) 표시된 사진

자동 생성된 설명