|  |  |  |
| --- | --- | --- |
| **2019\_2\_java2\_10** | **학번 : 20165614** | **이름 : 지현한** |

* **내용 점검**

1. 패키지 사용 - StringTokenizer 클래스

|  |
| --- |
| **//문자열로부터 토큰 분리**  import java.util.\*;  public class StringTokenizerExample {  public static void main(String args[]) {  StringTokenizer stok = new StringTokenizer("사과 배 복숭아");  System.out.println("토큰 갯수 : " + stok.countTokens());  while (stok.hasMoreTokens()) { **//토큰이 있는 동안만**  String str = stok.nextToken(); **//토큰을 추출하여**  System.out.println(str); **//추출된 토큰 출력**  }  }  } |
| **[실행 결과]** |

1. 패키지 사용 - Calendar 클래스

|  |
| --- |
| import java.util.Calendar;  public class CalendarTest {  public static void main(String[] args) {  Calendar now = Calendar.getInstance(); **//Calendar 객체 생성**  System.out.println(now.getTime());  int year = now.get(Calendar.YEAR); **//년**  int month = now.get(Calendar.MONTH) + 1; **//월 시작이 0**  int date = now.get(Calendar.DATE); **//일**  int hour = now.get(Calendar.HOUR\_OF\_DAY); **//시**  int minute = now.get(Calendar.MINUTE); **//분**  int second = now.get(Calendar.SECOND); **//초**  System.out.println(year + "년 " + month + "월 " + date + "일");  System.out.println(hour + "시 " + minute + "분 " + second + "초");  }  } |
| **[실행 결과]** |

1. 패키지 사용 - Arrays 클래스

* 배열 항목 정렬

|  |
| --- |
| public class Member implements Comparable<Member> {  String name;  Member(String name) {  this.name = name;  }  @Override  public int compareTo(Member o) {  return name.compareTo(o.name);  }  }  import java.util.Arrays;  public class SortExample {  public static void main(String[] args) {  int[] scores = { 99, 97, 98 };  Arrays.sort(scores);  for(int i=0; i<scores.length; i++) {  System.out.println("scores[" + i + "]=" + scores[i]);  }  System.out.println();    String[] names = { "홍길동", "박동수", "김민수" };  Arrays.sort(names);  for(int i=0; i<names.length; i++) {  System.out.println("names[" + i + "]=" + names[i]);  }  System.out.println();    Member m1 = new Member("홍길동");  Member m2 = new Member("박동수");  Member m3 = new Member("김민수");  Member[] members = { m1, m2, m3 };  Arrays.sort(members);  for(int i=0; i<members.length; i++) {  System.out.println("members[" + i + "].name=" + members[i].name);  }  }  } |
| **[실행 결과]** |

1. 제네릭 클래스

|  |
| --- |
| **class** Box<T> { // T는 데이터 타입을 의미, < >안에 표기, generic  **private** T data; // T 타입을 자료형으로 갖는 data 필드  **public** Box(T data) { // T 타입을 형식매개변수로 사용  **this**.data = data;  }  **public** **void** set(T data) {  **this**.data = data;  }  **public** T get() {  **return** data;  }  }  **public** **class** BoxTest {  **public** **static** **void** main(String[] args) {  // 다이아 몬드 생성자 호출 시 타입 인수 생략 가능 SE7 버전부터 적용  Box<String> sg = **new** Box<>("Generic"); // 문자열을 저장하는 Box 클래스의 객체 생성  Box<Integer> ig = **new** Box<Integer>(340); // 정수를 저장하는 Box 클래스의 객체  System.***out***.println("<String>" + sg.get());  System.***out***.println("<Integer>" + ig.get());  }  } |
| **[실행결과]** |

1. 제네릭 메소드

|  |
| --- |
| **class** MyArrayAlg {  **public** **static** <T> **void** swap(T[] a, **int** i, **int** j) {  T tmp = a[i];  a[i] = a[j];  a[j] = tmp;  }  }  **public** **class** MyArrayAlgTest {  **public** **static** **void** main(String[] args) {  String[] language = { "C++", "C#", "JAVA" };  Integer[] integer= {4,2,7,30,55,21};  MyArrayAlg.*swap*(language, 1, 2);  MyArrayAlg.*swap*(integer, 3, 5);  **for** (String value : language)  System.***out***.println(value);  **for** (Integer value : integer)  System.***out***.println(value);  }  } |
| **[실행결과]** |

* **내용 점검**

1. 아래의 Director 클래스를 movie 패키지에 속하게 하려면 어떤 문장을 추가하여야 하는가?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

public class Director {

...

}

정답】 package movie;

1. import 문장의 도움없이 Director 클래스를 다른 패키지에서 사용하려면 어떻게 하여야 하는가?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_.Director d = new \_\_\_\_\_\_\_\_\_\_\_\_\_\_.Director();

정답】 movie.Director d = new movie.Director();

1. import 문장을 사용하여 movie 패키지를 포함시켜보자.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_;

...

Director d = new Director();

정답】 import movie.\*;

1. 다음 설명과 같은 문장을 작성하시오.

(1) 1부터 10까지의 난수를 발생하는 문장

정답】

Random random = new Random();

int a= random.nextInt(9)+1

(2) 문자열 “By doubting we come at the truth"을 단어로 분리하는 문장

정답】

String str = “By doubting we come at the truth”;

String[] words = str.split(“ “);

(3) Calendar 객체를 사용하여 어제 날짜를 출력하는 문장

정답】

Calendar cld = Calendar.getInstance();

System.out.println(cld.get(Calendar.DATE)-1)

(4) 문자열 “12345”를 정수로 변환하는 문장

정답】 int a = Integer.parseInt(“12345”);

1. 다음 문장에 오류가 있으면 지적하고 수정하시오.
2. Calendar d = new Calendar();

정답】 Calendar d = Calendar.getInstance();

Calendar는 싱글톤이기 때문에 객체생성이 불가하고, getInstance로 객체를 받아와야 한다

1. int i = new Integer(0);

정답】 오류 없음

1. Integer i = new Integer("100");

정답】 오류 없음

1. int i = Integer.parseInt("100a");

정답】 문자열내에 숫자가 아닌 a가 포함 돼있어서 오류

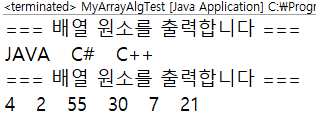
int i = Integer.parseInt("100");

(5) Double s = Double.parseDouble("345.12");

정답】 오류 없음

* **실습 과제**

1. 제시된 부분 소스를 함고하여 내용 점검 5번 문제의 MyArrayAlg 클래스를 제네릭 클래스로 수정하시오



**public** **class** MyArrayAlgTest {

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

String[] language = { "C++", "C#", "JAVA" };

Integer[] integer= {4,2,7,30,55,21};

MyArrayAlg<String> str=**new** MyArrayAlg<>();

MyArrayAlg<Integer> digit=**new** MyArrayAlg<>();

str.swap(language, 0,2);

str.write(language);

digit.swap(integer, 2, 4);

digit.write(integer);

}

}

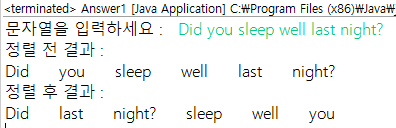
|  |
| --- |
| **[프로그램 소스]**  **class** MyArrayAlg<T> {  **public** **static** <T> **void** swap(T[] a, **int** i, **int** j) {  T tmp = a[i];  a[i] = a[j];  a[j] = tmp;  }  **public** **static** <T>**void** write(T[] arr) {  System.***out***.println("=== 배열 원소를 출력합니다 ===");  **for**(T temp:arr)  System.***out***.print(temp+" ");  System.***out***.println();  }  } |
| **[실행 결과]** |

1. 다음과 같이 문자열을 처리하는 프로그램을 작성하시오.

(1) StringTokenizer 클래스를 이용하여서 사용자로부터 받은 문자열을 단어로 분리한다. 단어들의 개수를 출력한다. **단, 공백이 포함된 문자열 입력은 nextLine() 사용**

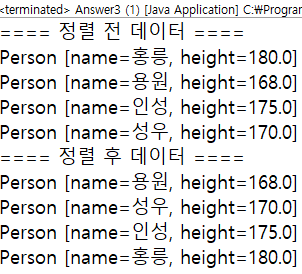
(2) 단어들을 문자열 배열에 넣고 이 배열을 Arrays의 sort() 메소드를 이용하여 정렬한다.

(3) mypackage 패키지를 생성한다



|  |
| --- |
| **[프로그램 소스]**  **package** mypackage;  **import** java.util.Arrays;  **import** java.util.Scanner;  **public** **class** main2 {  **public** **static** **void** main(String[] args) {  Scanner scan = **new** Scanner(System.***in***);  System.***out***.print("문자열을 입력하세요: ");  String str = scan.nextLine();  String[] strs = str.split(" ");  System.***out***.println("정렬 전 결과: ");  **for** (String temp : strs)  System.***out***.print(temp + " ");  System.***out***.println();  Arrays.*sort*(strs);  System.***out***.println("정렬 후 결과: ");  **for** (String temp : strs)  System.***out***.print(temp + " ");  }  } |
| **[실행결과]** |

1. 키를 기준으로 정렬하여 출력하는 프로그램을 완성하시오. 단, 정렬과정은 Arrays 클래스의 sort()를 사용.



**class** Person {

**private** String name;

**private** **double** height;

**public** String toString() {

**return** "Person [name=" + name + ", height=" + height + "]";

}

**public** Person(String name, **double** height) {

**this**.name = name;

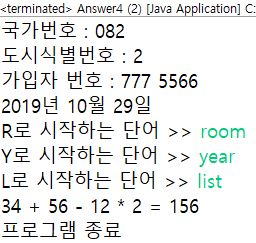
**this**.height = height;

}

}

|  |
| --- |
| **[프로그램 소스]**  **class** Person **implements** Comparable<Person> {  **private** String name;  **private** **double** height;  **public** String toString() {  **return** "Person [name=" + name + ", height=" + height + "]";  }  **public** Person(String name, **double** height) {  **this**.name = name;  **this**.height = height;  }  @Override  **public** **int** compareTo(Person o) {  **return** Double.*compare*(height, o.height);  }  }  **import** java.util.Arrays;  **public** **class** PersonTest {  **public** **static** **void** main(String[] args) {  Person[] persons = { **new** Person("홍릉", 180.0), **new** Person("용원", 168.0), **new** Person("인성", 175.0),  **new** Person("성우", 170.0) };  System.***out***.println("=== 정렬 전 데이터 ===");  **for** (Person temp : persons)  System.***out***.println(temp);  Arrays.*sort*(persons);  System.***out***.println("=== 정렬 후 데이터 ===");  **for** (Person temp : persons)  System.***out***.println(temp);  }  } |
| **[실행결과]** |

1. 여러 가지 유틸리티 라이브러리를 사용하여 제시된 결과처럼 출력하는 프로그램을 작성하시오.
2. 국제 전화 번호 "082-2-777-5566"에서 국가 번호, 도시 식별 번호, 가입자 번호를 추출하여 보자. StringTokenizer를 사용한다.
3. 오늘 날짜를 출력한다
4. 3개의 영문자를 무작위로 생성하고, 생성된 영문자로 시작하는 단어를 입력한다.
5. 수식 문자열을 계산한 결과를 출력한다



**public** **class** Answer4 {

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

Manage manage = **new** Manage();

manage.splitString(**new** String[] { "국가번호", "도시식별번호", "가입자 번호" }, "082-2-777-5566");

manage.dateWrite();

manage.startChar();

System.***out***.println("34 + 56 - 12 \* 2 = " + manage.calChar("34 + 56 - 12 \* 2"));

System.***out***.println("프로그램 종료");

}

}

|  |
| --- |
| **[프로그램 소스]**  **import** java.util.Calendar;  **import** java.util.Random;  **import** java.util.Scanner;  **public** **class** Manage {  **public** **void** splitString(String[] str1, String str2) {  String[] str2s = str2.split("-");  System.***out***.println(str1[0] + ": " + str2s[0]);  System.***out***.println(str1[1] + ": " + str2s[1]);  System.***out***.println(str1[2] + ": " + str2s[2] + " " + str2s[3]);  }  **public** **void** dateWrite() {  Calendar now = Calendar.*getInstance*();  **int** year = now.get(Calendar.***YEAR***);  **int** month = now.get(Calendar.***MONTH***) + 1;  **int** date = now.get(Calendar.***DATE***);  System.***out***.println(year + "년 " + month + "월 " + date + "일");  }  **public** **void** startChar() {  Scanner scan = **new** Scanner(System.***in***);  Random random = **new** Random();  **int** a = random.nextInt(26);  System.***out***.print((**char**) (a + 65) + "로 시작하는 단어 >> ");  scan.next();  **int** b = random.nextInt(26);  System.***out***.print((**char**) (b + 65) + "로 시작하는 단어 >> ");  scan.next();  **int** c = random.nextInt(26);  System.***out***.print((**char**) (c + 65) + "로 시작하는 단어 >> ");  scan.next();  }  **public** **int** calChar(String str) {  String[] strs = str.split(" ");  **int** result = Integer.*parseInt*(strs[0]);  **for** (**int** i = 1; i < strs.length; i += 2) {  **char** ch = strs[i].charAt(0);  **int** num = Integer.*parseInt*(strs[i + 1]);  **if** (ch == '+') {  result += num;  } **else** **if** (ch == '-') {  result -= num;  } **else** **if** (ch == '\*') {  result \*= num;  } **else** **if** (ch == '/') {  result /= num;  }  }  **return** result;  }  } |
| **[실행 결과]** |