# 개념 정리

## 파라미터 전달

메소드 파라미터로 변수를 전달했을 경우에, 그 변수의 값이 변경될 수 있는가?

예:

|  |
| --- |
| public static void main(String[] args) {  int a = 3;  method1(a);  System.out.print(a);  } |

위 코드의 실행 결과 출력은 method1의 구현과 무관하게 언제나 3 인가?

## 파라미터 전달

|  |
| --- |
| public static void main(String[] args) {  Person p = new Person();  p.age = 3;  method2(p);  System.out.print(p.age);  } |

위 코드의 실행 결과 출력은 method2의 구현과 무관하게 언제나 3 인가?

## 파라미터 전달

|  |
| --- |
| public static void main(String[] args) {  String s = "hello";  method3(s);  System.out.print(s);  } |

위 코드의 실행 결과 출력은 method3의 구현과 무관하게 언제나 hello 인가?

## equality와 identity

|  |
| --- |
| public static void main(String[] args) {  String s1 = "hello";  String s2 = "HELLO";  String s3 = s2.toLowerCase();  System.out.println(s1 == s3); // (가)  System.out.println(s1.equals(s3)); // (나)  } |

(가)에서 출력이 false 인 이유를 설명하라

(나)에서 출력이 true 인 이유를 설명하라.

# 구현 실습

## String.format 메소드

|  |
| --- |
| int double String  ---------- ---------- ----------  345 23.240 one  12 301.234 two  478 3.124 three  1003 98.024 four |

실행 결과 출력이 위와 같은 형태로 출력될 수 있도록, 아래 코드를 완성하라.

기존의 코드를 수정하지 말고, 새 코드만 추가하라.

String.format 메소드를 활용하라

|  |
| --- |
| package e2018.exer01;  public class Example01 {  public static void main(String[] args) {  int[] a1 = new int[] { 345, 12, 478, 1003 };  double[] a2 = new double[] { 23.24, 301.234, 3.1235, 98.0238 };  String[] a3 = new String[] { "one", "two", "three", "four" };  System.out.println(" int double String");  System.out.println("---------- ---------- ----------");  for (int i = 0; i < a1.length; ++i) {  }  }  } |

## String.charAt 메소드

|  |
| --- |
| One 2  Two 1  Three 2  Four 2  Five 2  hello world 3 |

실행 결과 출력이 위와 같은 형태로 출력될 수 있도록, getVowelCount 메소드를 구현하라.

String.charAt 메소드를 활용하라

|  |
| --- |
| package e2018.exer01;  public class Example02 {  static int getVowelCount(String s) {  }  public static void main(String[] args) {  String[] a = new String[] { "One", "Two", "Three", "Four", "Five", "hello world" };  for (String s : a) {  int count = getVowelCount(s);  System.out.printf("%s %d\n", s, count);  }  }  } |

## 구현

|  |
| --- |
| lecture.docx, lecture.docx  mainpage.html, mainpage.html  javac.exe, javac.exe |

실행 결과 출력이 위와 같은 형태로 출력될 수 있도록, 아래 코드를 완성하라.

파라미터로 전달된 path에서 파일 이름만 리턴하도록 getFileName1, getFileName2 메소드를 구현하라.

getFileName1 메소드 구현 방법: String.lastIndexOf 메소드, String.substring 메소드를 사용하여 구현

getFileName2 메소드 구현 방법: String.split 메소드를 사용하여 구현

|  |
| --- |
| package e2018.exer01;  public class Example03 {  static String getFileName1(String path) {    }  static String getFileName2(String path) {    }  public static void main(String[] args) {  String[] a = { "c:/data/student/lecture.docx",  "c:/www/mainpage.html",  "c:/program files/java/javac.exe" };  for (String s : a) {  String fileName1 = getFileName1(s);  String fileName2 = getFileName2(s);  System.out.printf("%s, %s\n", fileName1, fileName2);  }  }  } |

## String.replace 메소드

|  |
| --- |
| ne  Tw  Three  Fur  Five  hell wrld  yah |

실행 결과 출력이 위와 같은 형태로 출력될 수 있도록, 아래 코드를 완성하라.

파라미터로 전달된 s에서 o 문자와 O 문자를 제거하고 리턴하도록 removeO 메소드를 구현하라.

|  |
| --- |
| package e2018.exer01;  public class Example04 {  static String removeO(String s) {  }  public static void main(String[] args) {  String[] a = new String[] { "One", "Two", "Three", "Four", "Five", "hello world", "yahoo" };  for (String s : a) {  String temp = removeO(s);  System.out.println(temp);  }  }  } |

## String trim, length 메소드

|  |
| --- |
| the 3  world 5  war 3 |

실행 결과 출력이 위와 같은 형태로 출력될 수 있도록, 아래 코드를 완성하라.

|  |
| --- |
| package e2018.exer01;  public class Example06 {  public static void main(String[] args) {  String[] a = { " the", "world\t ", " \nwar " };  for (String s : a) {  System.out.printf("%s %d\n", );  }  }  } |

## String.split 메소드

|  |
| --- |
| [One]  [Two]  [Three]  [Four]  [Five]  [One]  [Two]  [Three]  [Four]  [Five] |

실행 결과 출력이 위와 같은 형태로 출력될 수 있도록, split1, split2 메소드를 구현하라.

split1 메소드 구현 방법: String.replace 메소드와 String.split 메소드를 사용하여 구현

split2 메소드 구현 방법: String.split 메소드와 String.trim 메소드를 사용하여 구현

|  |
| --- |
| package e2018.exer01;  public class Example05 {  static String[] split1(String s) {    }  static String[] split2(String s) {    }  public static void main(String[] args) {  String s = "One, Two ,Three , Four,Five";  String[] a1 = split1(s);  for (String t : a1)  System.out.printf("[%s]\n", t);  String[] a2 = split2(s);  for (String t : a2)  System.out.printf("[%s]\n", t);  }  } |

## equals 메소드 재정의

|  |
| --- |
| true  false  false  false  false |

실행 결과 출력이 위와 같은 형태로 출력될 수 있도록, Data 클래스의 equals 메소드를 재정의하라

|  |
| --- |
| package e2018.exer01;  import java.util.Date;  import java.util.GregorianCalendar;  import java.util.Objects;  class Data {  double weight;  String name;  boolean enabled;  Date startDate;  public Data(double weight, String name, boolean enabled, Date startDate) {  this.weight = weight;  this.name = name;  this.enabled = enabled;  this.startDate = startDate;  }  @Override  public boolean equals(Object obj) {  }  }  public class Example06 {  static Date createDate(int year, int month, int day) {  GregorianCalendar calendar = new GregorianCalendar(year, month, day);  return calendar.getTime();  }  public static void main(String[] args) {  Data d1 = new Data(10.3, "data", true, createDate(2018, 9, 10));  Data d2 = new Data(10.3, "data", true, createDate(2018, 9, 10));  Data d3 = new Data(11.9, "data", true, createDate(2018, 9, 10));  Data d4 = new Data(10.3, "datt", true, createDate(2018, 9, 10));  Data d5 = new Data(10.3, "data", false, createDate(2018, 9, 10));  Data d6 = new Data(10.3, "data", true, createDate(2018, 9, 11));  System.out.println(d1.equals(d2));  System.out.println(d1.equals(d3));  System.out.println(d1.equals(d4));  System.out.println(d1.equals(d5));  System.out.println(d1.equals(d6));  }  } |

## String 클래스 trim, length 메소드 활용

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | package finalexam;  public class Example01 {    public static boolean isNullEmptyBlank(String s) {  }  public static void main(String[] args) {  String[] a = { null, "", " ", " ", " \t ", " \t\n", " . "};    for (String s : a)  System.out.println(isNullEmptyBlank(s));  }  } |

isNullEmptyBlank 메소드를 구현하라.

사용자 입력을 검사할 때, 필수 입력 항목을 입력했는지 검사해야 한다.

이때 사용할 isNullEmptyBlank 메소드를 구현하라.

이 메소드는 파라미터 s 문자열이 null 이거나, 빈 문자열이거나, 공백만 들어있는 경우에

true를 리턴해야 한다.

공백에는 스페이스(space) 문자뿐만 아니라, 탭(tab) 문자, 줄바꿈(new line) 문자도 포함된다.

위 예제의 실행 결과 출력은 다음과 같아야 한다.

|  |
| --- |
| true  true  true  true  true  true  false |

isNullEmptyBlank 메소드 내부만 구현하고 다른 부분을 수정하지 말라.