# 구현 문제

## equals

Person 클래스의 equals 메소드를 구현(override) 하시오

### Person.java

|  |
| --- |
| package e2018.exam1;  import java.util.Objects;  public class Person {  String name;  String no;  int year;  double grade;  public Person(String name, String no,int year, double grade) {  this.name = name;  this.no = no;  this.year = year;  this.grade = grade;  }  @Override  public boolean equals(Object obj) {  if (obj instanceof Person == false) return false;  Person p = (Person)obj;  return Objects.equals(this.name, p.name) &&  Objects.equals(this.no, p.no) &&  this.year == p.year &&  this.grade == p.grade;  }  } |

### Exam01.java

|  |
| --- |
| package e2018.exam1;  public class Exam01 {  public static void main(String[] args) {  Person p1 = new Person("홍길동", "201132050", 2, 4.1);  Person p2 = new Person("홍길동", null, 2, 4.1);  Person p3 = new Person("홍길동", "201132050", 2, 4.1);  System.out.println(p1 == p2);  System.out.println(p1.equals(p2));  System.out.println(p1.equals(p3));  }  } |

### 실행결과 출력

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

## equality & identity



객체 구조가 위와 같도록 아래 구현을 완성하시오.

### Exam02.java

|  |
| --- |
| package e2018.exam1;  public class Exam02 {  public static void main(String[] args) {  Person[] a = new Person[2];  a[0] = new Person("홍길동", "201132050", 2, 4.1);  a[1] = new Person("홍길동", "201132050", 2, 4.1);  System.out.println(a[0] == a[1]);  System.out.println(a[0].equals(a[1]));  }  } |

### 실행 결과 출력

|  |
| --- |
| false  true |

## equality & identity



객체 구조가 위와 같도록 아래 구현을 완성하시오.

### Exam03.java

|  |
| --- |
| package e2018.exam1;  public class Exam03 {  public static void main(String[] args) {  // 구현할 부분  Person[] a = new Person[2];  Person[] b = a;  a[0] = new Person("홍길동", "201132050", 2, 4.1);  a[1] = a[0];  System.out.println(a == b);  System.out.println(a[0] == b[1]);  System.out.println(a[0].equals(b[1]));  }  } |

### 실행 결과 출력

|  |
| --- |
| true  true  true |

## String

|  |
| --- |
| (h) (e) (l) (l) (o)  (w) (o) (r) (l) (d)  (h) (e) (l) (l) (o) ( ) (w) (o) (r) (l) (d) |

실행 결과 출력이 위와 같도록 doSomething 메소드를 구현하시오.

### Exam04.java

|  |
| --- |
| package e2018.exam1;  public class Exam04 {  public static void doSomething(String s) {  // 구현할 부분  for (int i = 0; i < s.length(); ++i)  System.out.printf("(%c) ", s.charAt(i));  System.out.println();  }  public static void main(String[] args) {  String[] a = { "hello", "world", "hello world" };  for (String s : a)  doSomething(s);  }  } |

## format

|  |
| --- |
| 00002  00034  00256  01980 |

실행 결과 출력이 위와 같도록 doSomething 메소드를 구현하시오.

### Exam05.java

|  |
| --- |
| package e2018.exam1;  public class Exam05 {  public static void doSomething(int i) {  // 구현할 부분  System.out.printf("%05d\n", i);  }  public static void main(String[] args) {  int[] a = { 2, 34, 256, 1980 };  for (int i : a)  doSomething(i);  }  } |

## 확장자

|  |
| --- |
| .docx  .html  .exe |

실행 결과 출력이 위와 같도록 getExtension 메소드를 구현하시오.

### Exam06.java

|  |
| --- |
| package e2018.exam1;  public class Exam06 {  static String getExtension(String path) {  int index = path.lastIndexOf(".");  return path.substring(index);  }  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 ext = getExtension(s);  System.out.println(ext);  }  }  } |

## array

|  |
| --- |
| [c:, data, student, lecture.docx]  [c:, www, mainpage.html]  [c:, program files, java, javac.exe] |

실행 결과 출력이 위와 같도록 convertArray 메소드를 구현하시오.

convertArray 메소드는 파라미터 문자열을 배열로 쪼개서 리턴해야 한다.

|  |
| --- |
| package e2018.exam1;  import java.util.Arrays;  public class Exam07 {  static String[] convertArray(String path) {  // 구현할 부분  return path.split("/");  }  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[] temp = convertArray(s);  System.out.println(Arrays.toString(temp));  }  }  } |

## 2차원 배열 합계 계산

|  |
| --- |
| 46  46 |

실행 결과 출력이 위와 같도록 sum 메소드를 구현하시오.

sum 메소드는 2차원 배열의 원소 값을 모두 더한 합계를 리턴해야 한다.

|  |
| --- |
| package e2018.exam1;  public class Exam08 {  static int sum(int[][] a) {  // 구현할 부분  int sum = 0;  for (int r = 0; r < a.length; ++r)  for (int c = 0; c < a[r].length; ++c)  sum = sum + a[r][c];  return sum;  }  public static void main(String[] args) {  int[][] a1 = { {1, 3, 6}, {2, 6, 3}, {6, 9, 10} };  int[][] a2 = { {5, 4, 1}, {10, 8, 3}, {7, 2, 6} };  System.out.println(sum(a1));  System.out.println(sum(a2));  }  } |

## 문자열 뒤집기

|  |
| --- |
| olleh  dlrow  dlrow olleh |

실행 결과 출력이 위와 같도록 reverse 메소드를 구현하시오.

reverse 메소드는 파라미터 문자열을 거꾸로 뒤집어서 리턴해야 한다.

|  |
| --- |
| package e2018.exam1;  public class Exam09 {  static String reverse(String path) {  // 구현할 부분  StringBuilder builder = new StringBuilder(path);  builder.reverse();  return builder.toString();  }  public static void main(String[] args) {  String[] a = { "hello", "world", "hello world" };  for (String s : a)  System.out.println(reverse(s));  }  } |

## 부분 문자열 제거

|  |
| --- |
| ho  wod  he hello |

실행 결과 출력이 위와 같도록 removeSubstring 메소드를 구현하시오.

removeSubstring 메소드는 파라미터 문자열에서 fromIndex와 toIndex 사이의 문자를 제거한 문자열을 리턴해야 한다. (제거할 문자의 위치: fromIndex <= 인덱스 < toIndex)

|  |
| --- |
| package e2018.exam1;  public class Exam10 {  static String removeSubstring(String s, int fromIndex, int toIndex) {  // 구현할 부분  StringBuilder builder = new StringBuilder(s);  builder.delete(fromIndex, toIndex);  return builder.toString();  }  public static void main(String[] args) {  String[] a = { "hello", "world", "hello hello" };  System.out.println(removeSubstring(a[0], 1, 4));  System.out.println(removeSubstring(a[1], 2, 4));  System.out.println(removeSubstring(a[2], 2, 5));  }  } |

## 정렬

### Product.java

|  |
| --- |
| package e2018.exam1;  public class Product {  int id;  String name;  int price;  public Product(int id, String name, int price) {  this.id = id;  this.name = name;  this.price = price;  }  @Override  public String toString() {  // 구현할 부분  return String.format("{%s, %s, %d}", id, name, price);  }  } |

### ProductComparator.java

|  |
| --- |
| package e2018.exam1;  import java.util.Comparator;  public class ProductComparator implements Comparator<Product> {  @Override  public int compare(Product p1, Product p2) {  int r = p1.name.compareTo(p2.name);  if (r != 0) return r;  r = p1.price - p2.price;  if (r != 0) return r;  return p1.id - p2.id;  }  } |

### Exam11.java

|  |
| --- |
| package e2018.exam1;  import java.util.Arrays;  public class Exam11 {  public static void main(String[] args) {  Product[] a = new Product[] {  new Product(1, "맥주", 3000),  new Product(2, "맥주", 2000),  new Product(3, "맥주", 3000),  new Product(4, "막걸리", 1000),  new Product(5, "소주", 2000)  };  Arrays.sort(a, new ProductComparator());  for (Product p : a)  System.out.println(p);  }  } |

### 실행 결과 출력

|  |
| --- |
| {4, 막걸리, 1000}  {2, 맥주, 2000}  {1, 맥주, 3000}  {3, 맥주, 3000}  {5, 소주, 2000} |

실행 결과 출력이 위와 같도록 ProductComparator 클래스를 구현하라.

그리고 Product 클래스의 toString 메소드를 구현하라.