객제지향프로그래밍 2장 실습 문제 보고서

글로벌미디어학부20192722 이지희

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 2번 | 4번 | 6번 | 8번 | 10번 | 12-1번 | 12-2번 |
| O | O | O | O | O | O | O |

2.

|  |
| --- |
| import java.util.Scanner;  public class OPP2 {  public static void main(String[] args) {  Scanner scanner= new Scanner(System.*in*);  int input;  System.*out*.print("2자리수 정수 입력(10~99)>>");  input=scanner.nextInt();  if(input/10==input%10)  {  System.*out*.println("Yes! 10의 자리와 1의 자리가 같습니다.");  }  else  {  System.*out*.println("No! 10의 자리와 1의 자리가 다릅니다.");  }  scanner.close();   } } |

* 실행 화면

텍스트이(가) 표시된 사진

자동 생성된 설명

* 부연 설명

int형 변수 input을 선언한 후Scanner 객체를 이용해서 input의 값을 입력 받았다. 입력 받은 정수를 10으로 나눠서 십의 자릿수를 구하고 %(나머지) 연산을 통해 일의 자리수를 구했다. 이 둘이 같다는 문장이 true 이면 “Yes! 10의 자리와 1의 자리가 같습니다."를 출력하고, false이면 “No! 10의 자리와 1의 자리가 다릅니다” 를 출력하게 하였다.

4.

|  |
| --- |
| import java.util.Arrays; import java.util.Scanner;  public class OPP4 {  public static void main(String[] args) {   int [] num = new int[3];  Scanner scanner=new Scanner(System.*in*);  System.*out*.print("정수 3개를 입력하시오.>> ");   for(int i=0;i< num.length;i++){  num[i]=scanner.nextInt();  }   Arrays.*sort*(num);   System.*out*.println(num[1]);  scanner.close();   } } |

* 실행화면

텍스트이(가) 표시된 사진

자동 생성된 설명

* 부연설명

정수를 입력 받을 배열 num을 만들고 ( scanner 이용하여 입력 받을 때와 정렬할 때 편의를 위해 배열로 선언함) for문을 이용해 각 배열 요소에 Scanner로 정수를 입력 받아 저장하였다. Java.util.Arrays의 sort() 메서드를 이용해서 배열 요소들을 정렬하였다. 정렬 후에 println()함수의 인자로 num[1]를 통해 배열에 접근하여 두번째로 큰 수를 출력하도록 하였다.

6.

|  |
| --- |
| import java.util.Scanner;  public class OPP6 {  public static void main(String[] args) {  int num ;  String game369="박수";  Scanner scanner=new Scanner(System.*in*);   num=scanner.nextInt();   for(int i=num; i>0; i/=10){  int temp=num%10;  if(temp%3==0){  game369+="짝";  }  }   if(game369=="박수") System.*out*.println(num);  else System.*out*.println(game369);  scanner.close();  } } |

* 실행 화면

|  |  |
| --- | --- |
| 3,6,9가 들어가는 수일 경우 | 3,6,9가 들어가지 않을 경우 |
| 텍스트이(가) 표시된 사진  자동 생성된 설명 | 텍스트이(가) 표시된 사진  자동 생성된 설명 |

* 부연설명

int형 변수 num을 선언 후에 Scanner 객체를 이용해서 숫자를 입력 받는다. 출력할 문자열을 String형 변수 game369로 만들고 “박수”라는 문자열을 미리 저장해둔다. 그 뒤에 for문을 이용해서 각 자리수에 3,6,9가 들어가 있는지 확인 후 들어가 있으면 “짝” 이라는 문자열을 game369에 하나씩 더한다.

실제 게임과 같이 3,6,9가 나오면 박수를 출력하고 아닌 경우에는 입력받은 숫자를 출력하도록 if문을 이용해서 문자열 game369에 “짝”이 한번도 더해지지 않은 경우(각 자리에 3,6,9가 들어가지 않는 경우)에는 num을 그대로 출력하고 else로 “짝”이 한번이라도 더해지면 game369를 출력하도록 했다.

8.

|  |
| --- |
| import java.util.Scanner;  public class OPP8 {  public static void main(String[] args) {  int[][] rect = new int[2][2];  Scanner scanner= new Scanner(System.*in*);  for(int i=0;i<rect.length;i++){  for(int j=0;j<rect[i].length;j++){  rect[i][j]=scanner.nextInt();  }  }  if(*inRect*(rect[0][0],rect[0][1],100,100,200,200)||*inRect*(rect[1][0],rect[1][1],100,100,200,200)){  System.*out*.println("사각형이 충돌합니다.");  }  else System.*out*.println("사각형이 충돌하지 않습니다.");  scanner.close();   }   public static boolean inRect(int x,int y, int rectx1, int recty1, int rectx2, int recty2){  if((x>=rectx1 && x<=rectx2) && (y>=recty1 && y<=recty2)){  return true;  }  else return false;   } } |

* 실행화면

텍스트이(가) 표시된 사진

자동 생성된 설명

* 부연 설명

두 직사각형이 충돌하는 경우는 직사각형을 결정하는 두 점 중 한 점이 (100,100),(200,200)의 안에 있는 경우이다. 한 점이 사각형 안에 있는 경우는 입력받은 x의 범위가 100<x<200 이고 y의 범위가 100<y<200를 만족해야 한다.

힌트로 주어진 메서드를 이용하여 프로그램을 작성하면, 첫번째 점이 inRect함수의 매개변수로 들어가서 true를 반환하거나 두번째 점이 매개변수가 되었을 때 true를 반환해야 한다.

10

|  |
| --- |
| import java.util.Scanner;  public class OPP10 {  public static void main(String[] args) {  int[][] circle = new int[2][3];  Scanner scanner=new Scanner(System.*in*);  System.*out*.print("첫번째 원의 중심과 반지름 입력>>");  for(int i=0;i<circle[0].length;i++){  circle[0][i]=scanner.nextInt();  }  System.*out*.print("두번째 원의 중심과 반지름 입력>>");  for(int j=0;j<circle[1].length;j++){  circle[1][j]=scanner.nextInt();  }   double d=*distance*(circle[0][0],circle[0][1],circle[1][0],circle[1][1]);  if((circle[0][2]+circle[1][2])>d)  System.*out*.println("두 원은 서로 겹친다.");  else  System.*out*.println("두 원은 서로 겹치지 않는다.");  scanner.close();  }  public static double distance(int x1, int y1, int x2, int y2){  return Math.*sqrt*((x1-x2)\*(x1-x2)+(y1-y2)\*(y1-y2));  } } |

* 실행화면

텍스트이(가) 표시된 사진

자동 생성된 설명

* 부연설명

두 원의 중심점의 x,y 좌표와 반지름을 입력 받을 2차원 배열 circle을 선언하고 for문과 Scanner 객체를 이용하여 각각의 값을 입력 받는다.

두 원이 충돌하는 경우(겹치는 경우)는 두 원의 반지름을 더한 값이 중심거리보다 커야 한다. 따라서, 첫번째 원의 반지름이 r이고 두번째 원의 반지름이 r', 두 중심 사이 거리가 d일 때 r + r' > d을 만족해야 한다.

두 점 사이의 거리를 구하기 위해 distance 메서드를 만들었다. 두 점 사이 거리를 구하는 공식인 을 이용하였다. 이때, Math 클래스의 sqrt 함수를 사용하였다.

12.(1)

|  |
| --- |
| import java.util.Scanner;  public class OPP12\_1 {  public static void main(String[] args) {  System.*out*.print("연산>>");  Scanner scanner= new Scanner(System.*in*);  double op1 = scanner.nextInt();  String op = scanner.next();  double op2 = scanner.nextInt();  double result=0;  if(op.equals("+")) result=op1+op2;  else if(op.equals("-")) result=op1-op2;  else if(op.equals("\*")) result=op1\*op2;  else if(op.equals("/")) {  if(op1==0 || op2==0) {  System.*out*.println("0으로 나눌 수 없습니다.");  scanner.close();  return;  }  result=op1/op2;  }  System.*out*.println(op1+op+op2+"의 계산결과는"+result);  scanner.close();  } } |

* 실행화면

텍스트이(가) 표시된 사진

자동 생성된 설명

* 부연설명 : 연산의 피연산자 부분과 연산자 부분을 나누어서 각 변수에 Scanner 객체를 이용하여 입력받았다. 그 뒤 else if문을 이용하여 각 사칙연산의 케이스를 나누고 equals 메서드를 이용하여 string형 변수 op와 비교하였다. 0으로 나누는 경우는 ("0으로 나눌 수 없습니다."를 출력하고 프로그램을 종료하도록 했다.

12(2)

|  |
| --- |
| import java.util.Scanner;  public class OPP12\_2 {  public static void main(String[] args) {  System.*out*.print("연산>>");  Scanner scanner= new Scanner(System.*in*);  double op1 = scanner.nextInt();  String op = scanner.next();  double op2 = scanner.nextInt();  double result=0;  switch (op){  case "+":  result=op1+op2;  break;  case "-":  result=op1-op2;  break;  case "/":  if(op2==0) {  System.*out*.println("0으로 나눌 수 없습니다.");  scanner.close();  return;  }  result=op1/op2;  break;  case "\*":  result=op1\*op2;  break;  }  System.*out*.println(op1+op+op2+" 의 계산결과는 "+result);  } } |

* 실행화면

텍스트이(가) 표시된 사진

자동 생성된 설명

* 부연설명

12와 풀이 과정은 동일하다. If else 구문을 switch로 바꾸어서 op를 비교하여 각각 연산을 진행한다.