객체지향프로그래밍 및 실습

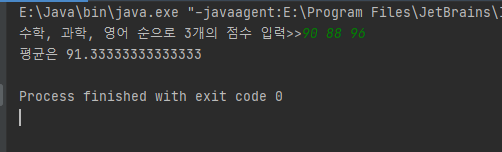
20192875

조성현

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

2.

|  |
| --- |
| import java.util.Scanner; class Grade {  private int math;  private int science;  private int english;   Grade(int math, int science, int english) {  this.math = math;  this.science = science;  this.english = english;  }   double average() {  return (math + science + english) / 3.0;  } } public class HW\_4\_2 {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.*in*);   System.*out*.print("수학, 과학, 영어 순으로 3개의 점수 입력>>");  int math = scanner.nextInt();  int science = scanner.nextInt();  int english = scanner.nextInt();  Grade me = new Grade(math, science, english);  System.*out*.println("평균은 " + me.average());   scanner.close();  } |

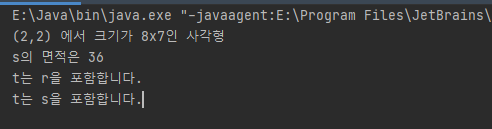


부연 설명

Grade 클래스를 만들고 평균을 구하는 함수를 만들어서 평균을 구함

4.

|  |
| --- |
| class Rectangle {  private int x;  private int y;  private int width;  private int height;   Rectangle(int x, int y, int width, int height) {  this.x = x;  this.y = y;  this.height = height;  this.width = width;  }   int square() {  return width \* height;  }   void show() {  System.*out*.println("(" + x + "," + y + ") 에서 크기가 " + width + "x" + height + "인 사각형");  }   boolean contains(Rectangle r) {  if (x <= r.x && x + width >= r.x + r.width) {  if (y <= r.y && y + height >= r.y + r.height) return true;  }  return false;  } } public class HW\_4\_4 {  public static void main(String[] args) {  Rectangle r = new Rectangle(2, 2, 8, 7);  Rectangle s = new Rectangle(5, 5, 6, 6);  Rectangle t = new Rectangle(1, 1, 10,10);   r.show();  System.*out*.println("s의 면적은 " + s.square());  if(t.contains(r)) System.*out*.println("t는 r을 포함합니다.");  if(t.contains(s)) System.*out*.println("t는 s을 포함합니다.");  } } |

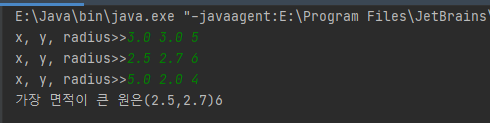


부연 설명

Rectangle 클래스를 만들고 넓이를 구하는 메소드, 출력해주는 메소드, 포함여부를 구하는 메소드를 만들어서 해결함

6.

|  |
| --- |
| import java.util.Scanner; class Circle{  private double x, y;  private int radius;  public Circle(double x, double y, int radius){  this.x=x;  this.y=y;  this.radius=radius;  }  public void show(){  System.*out*.println("("+x+","+y+")"+radius);  }  public int square(){return radius\*radius;} } public class HW\_4\_6 {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.*in*);  Circle c [] = new Circle[3];  for(int i=0;i<c.length;i++){  System.*out*.print("x, y, radius>>");  double x=scanner.nextDouble();  double y=scanner.nextDouble();  int r = scanner.nextInt();  c[i]=new Circle(x,y,r);  }  int max=0;  int tmp=0;  for(int i=0;i<c.length;i++){  if(max<c[i].square()) {  max = c[i].square();  tmp=i;  }  }  System.*out*.print("가장 면적이 큰 원은");  c[tmp].show();  scanner.close();  } } |

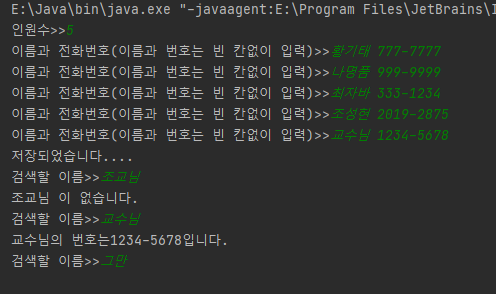


부연설명

Circle 클래스를 만들고 넓이를 구하는 메소드를 만들어 넓이를 비교하여 해결함

8.

|  |
| --- |
| import java.util.Scanner; class Phone{  String name=null;  String p\_number=null;   Phone(String name, String p\_number){  this.name = name;  this.p\_number= p\_number;  } }  public class HW\_4\_8 {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.*in*);  System.*out*.print("인원수>>");   int people=scanner.nextInt();  Phone [] p= new Phone [people];  for(int i=0;i<people;i++){  System.*out*.print("이름과 전화번호(이름과 번호는 빈 칸없이 입력)>>");  String name=scanner.next();  String p\_number=scanner.next();  p[i] = new Phone(name, p\_number);  }  System.*out*.println("저장되었습니다....");   while(true){  System.*out*.print("검색할 이름>>");  String search=scanner.next();  if(search.equals("그만")) break;  for(int i = 0;i<=people;i++){  if(i==people) {  System.*out*.println(search+" 이 없습니다.");  break;  }  if(p[i].name.equals(search)) {  System.*out*.println(search +"의 번호는"+ p[i].p\_number+"입니다.");  break;  }  }   }  } } |

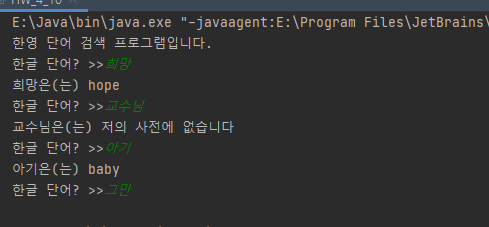


부연 설명

Phone 클래스를 만들고 메인함수에서 Phone 객체 배열을 만들어 이름과 전화번호를 저장하여 해결함

10.

|  |
| --- |
| import java.util.Scanner; class Dictionary {  private static String [] *kor* = {"사랑", "아기", "돈", "미래", "희망"};  private static String [] *eng* = {"love", "baby", "money", "future", "hope"};  public static String kor2Eng(String word){  for(int i=0;i<5;i++){  if(*kor*[i].equals(word)) return *eng*[i];  }  return "저의 사전에 없습니다";  } } public class HW\_4\_10 {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.*in*);  System.*out*.println("한영 단어 검색 프로그램입니다.");  while (true) {  System.*out*.print("한글 단어? >>");  String word = scanner.next();  if (word.equals("그만")) break;  System.*out*.println(word + "은(는) " + Dictionary.*kor2Eng*(word));  }  } } |

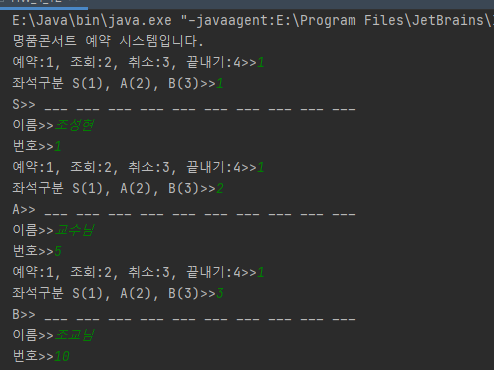
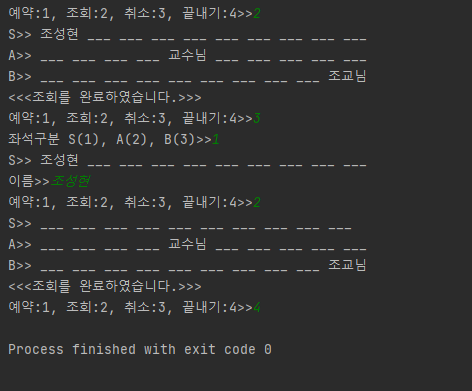


부연 설명

Dictionary 클래스를 만들고 while문을 메인함수에 넣어 반복 실행할 수 있도록 함

12.

|  |
| --- |
| import java.util.Scanner; class Concert{  Scanner scanner = new Scanner(System.*in*);   String [] seatS={"\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_"};  String [] seatA={"\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_"};  String [] seatB={"\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_","\_\_\_"};   boolean reserve(){ //예약시스템  System.*out*.print("좌석구분 S(1), A(2), B(3)>>");  int seatCase=scanner.nextInt();  show(seatCase);  System.*out*.print("이름>>");  String name=scanner.next();  System.*out*.print("번호>>");  int seatNumber=scanner.nextInt();  switch (seatCase){  case 1:  if(!seatS[seatNumber-1].equals("\_\_\_")) return false;  seatS[seatNumber-1]=name;  return true;  case 2:  if(!seatA[seatNumber-1].equals("\_\_\_")) return false;  seatA[seatNumber-1]=name;  return true;  case 3:  if(!seatB[seatNumber-1].equals("\_\_\_")) return false;  seatB[seatNumber-1]=name;  return true;  }  return false;  }   void show(int n){ //좌석표 출력을 도와줌  switch (n){  case 0: //전체 출력  System.*out*.print("S>>");  for(int i=0;i<10;i++) System.*out*.print(" "+seatS[i]);  System.*out*.println();  System.*out*.print("A>>");  for(int i=0;i<10;i++) System.*out*.print(" "+seatA[i]);  System.*out*.println();  System.*out*.print("B>>");  for(int i=0;i<10;i++) System.*out*.print(" "+seatB[i]);  System.*out*.println();  System.*out*.println("<<<조회를 완료하였습니다.>>>");  break;  case 1: //S출력  System.*out*.print("S>>");  for(int i=0;i<10;i++) System.*out*.print(" "+seatS[i]);  System.*out*.println();  break;  case 2: //A출력  System.*out*.print("A>>");  for(int i=0;i<10;i++) System.*out*.print(" "+seatA[i]);  System.*out*.println();  break;  case 3: //B출력  System.*out*.print("B>>");  for(int i=0;i<10;i++) System.*out*.print(" "+seatB[i]);  System.*out*.println();  break;  }  }   boolean cancle(){ //취소시스템  System.*out*.print("좌석구분 S(1), A(2), B(3)>>");  int seatCase=scanner.nextInt();  show(seatCase);  System.*out*.print("이름>>");  String name=scanner.next();  switch (seatCase){  case 1:  for(int i=0;i<10;i++){  if(seatS[i].equals(name)){  seatS[i]="\_\_\_";  return true;  }  }  return false;  case 2:  for(int i=0;i<10;i++){  if(seatA[i].equals(name)){  seatA[i]="\_\_\_";  return true;  }  }  return false;  case 3:  for(int i=0;i<10;i++){  if(seatB[i].equals(name)){  seatB[i]="\_\_\_";  return true;  }  }  return false;  }  return false;  } } public class HW\_4\_12 {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.*in*);  System.*out*.println("명품콘서트 예약 시스템입니다.");  Concert concert = new Concert();  int key=0;  while(key!=4){  System.*out*.print("예약:1, 조회:2, 취소:3, 끝내기:4>>");  key=scanner.nextInt();  switch (key){  case 1:  if(!concert.reserve()) System.*out*.println("예약 실패!");  break;  case 2:  concert.show(0);  break;  case 3:  if(!concert.cancle()) System.*out*.println("취소 실패!");  break;  case 4:  key=4;  break;  }  }  } } |



부연설명

Concert 클래스를 만들고 각 좌석 이름을 String 배열로 설정하고 초기값을 “\_\_\_”으로 함. 그리고 reserve() 메소드, show()메소드, cancel()메소드를 만들었고 이 메소드들로 각 좌석들을 수정하고 출력함. 만약 정상적으로 예약이나 취소가 안될 경우 false를 반환하여 확인함 메인 함수에서는 while문을 이용하여 반복 실행하게 함.