

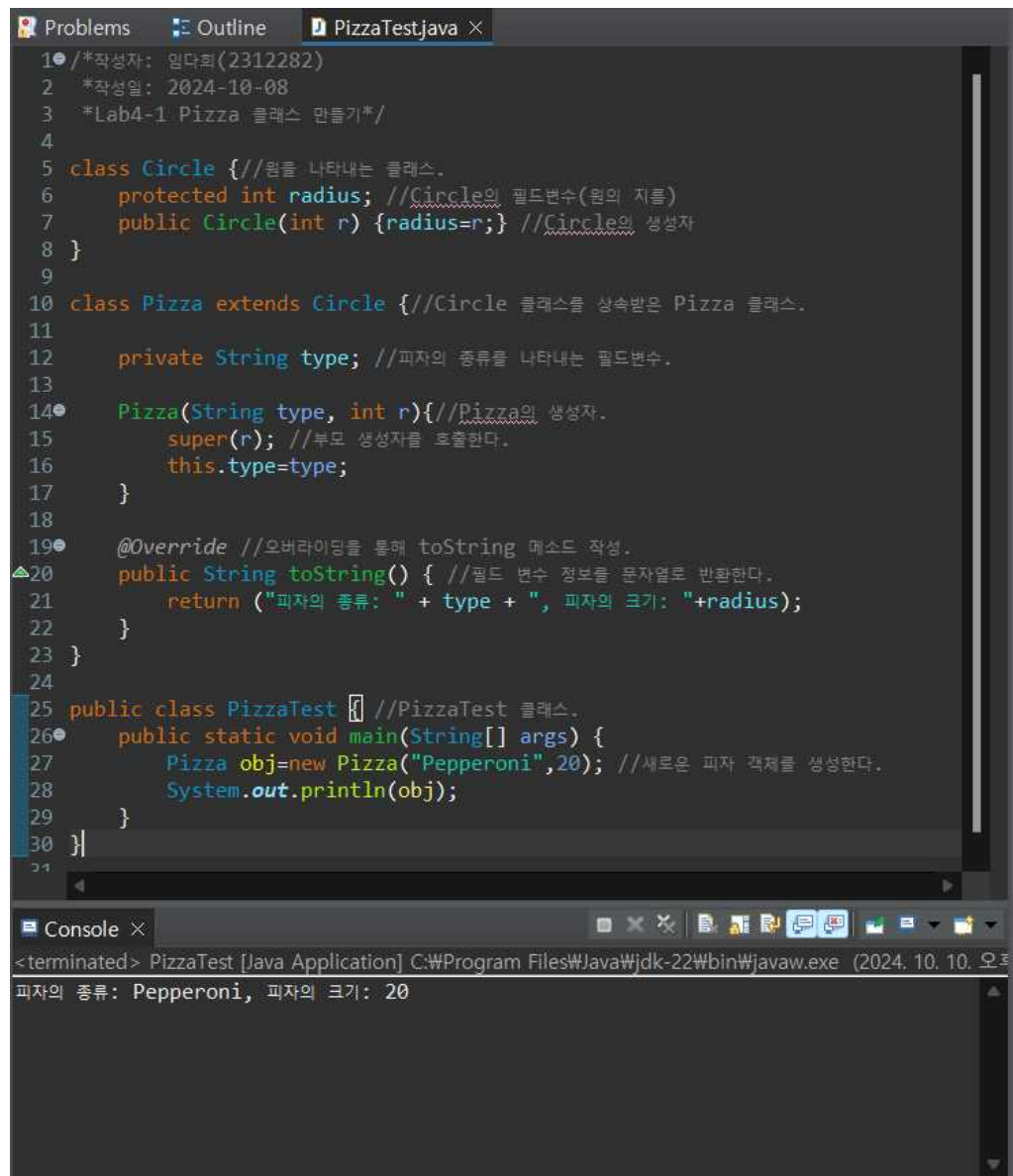
학과: 소프트웨어학부 컴퓨터과학전공

학번: 2312282

이름: 임다희

<실습 1> Pizza 클래스 작성

-코드 및 실행결과



```
1  /*작성자: 임다희(2312282)
2   *작성일: 2024-10-08
3   *Lab4-1: Pizza 클래스 만들기*/
4
5  class Circle { //원을 나타내는 클래스.
6      protected int radius; //Circle의 필드변수(원의 지름)
7      public Circle(int r) {radius=r;} //Circle의 생성자
8  }
9
10 class Pizza extends Circle { //Circle 클래스를 상속받은 Pizza 클래스.
11
12     private String type; //피자의 종류를 나타내는 필드변수.
13
14     Pizza(String type, int r){ //Pizza의 생성자.
15         super(r); //부모 생성자를 호출한다.
16         this.type=type;
17     }
18
19     @Override //오버라이딩을 통해 toString 메소드 작성.
20     public String toString() { //필드 변수 정보를 문자열로 반환한다.
21         return ("피자의 종류: " + type + ", 피자의 크기: "+radius);
22     }
23 }
24
25 public class PizzaTest { //PizzaTest 클래스.
26     public static void main(String[] args) {
27         Pizza obj=new Pizza("Pepperoni",20); //새로운 피자 객체를 생성한다.
28         System.out.println(obj);
29     }
30 }
```

Console ×

<terminated> PizzaTest [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe (2024. 10. 10. 오후 10:00)

피자의 종류: Pepperoni, 피자의 크기: 20

<실습 2> Magazine 만들기

-코드

```
Problems Outline ColorPointTest.java StudentTest.java *Magazine.java X
10 /*작성자: 임다희(2312282)
2  *작성일: 2024-10-08
3  *Lab4-2 Magazine 클래스 만들기*/
4
5 class MyBook { //Mybook 클래스.
6
7     protected String title, author; //필드 변수 정의. (책의 제목, 저자)
8     protected int page; //책의 페이지 수
9     protected static int count; //static 변수 정의. (생성된 MyBook타입 객체 개수)
10
11     MyBook(String title, int page, String author){
12         //MyBook의 생성자.
13         this.title=title;
14         this.page=page;
15         this.author=author;
16         //객체를 하나 생성할 때마다 static 변수의 크기를 1씩 증가시킨다.
17         count++;
18     }
19
20     //각 필드에 대한 접근자와 생성자.
21     public String getTitle() {
22         return title;
23     }
24
25     public void setTitle(String title) {
26         this.title = title;
27     }
28
29     public String getAuthor() {
30         return author;
31     }
32
33     public void setAuthor(String author) {
34         this.author = author;
35     }
36
37     public int getPage() {
38         return page;
39     }
40
41     public void setPage(int page) {
42         this.page = page;
43     }
44     //책의 개수에 대한 접근자 메소드.
45     public static int getCount() {
46         return count;
47     }
48 }
49
50 public class Magazine extends MyBook {
51     //MyBook 클래스를 상속받은 클래스 Magazine.
52     private String date;
53     //추가 속성 정보(발매일 정보)
54
55     Magazine(String title, int page, String author, String date){
56         super(title,page,author); //생성자 내에서 부모 생성자를 호출한다.
57         this.date=date;
58     }
59
60     //date 에 대한 생성자와 접근자.
61     public String getDate() {
62         return date;
63     }
64
65     public void setDate(String date) {
66         this.date = date;
67     }
68
69 }
```

<실습 3> ColorPoint 클래스 작성

-코드 및 실행결과

```
Problems Outline Magazine.java PizzaTest.java ColorPointTest.java ×
1  /* 작성자: 임다희(2312282)
2   * 작성일: 2024-10-08
3   * Lab4-3 ColorPoint 클래스 만들기 */
4
5  class Point { //Point 클래스 작성.
6      private int x,y; //점의 좌표를 나타내는 필드 변수.
7      public Point(int x, int y) { //Point의 생성자.
8          this.x=x;
9          this.y=y;
10     }
11
12     //각 필드에 대한 접근자.
13     public int getX() {return x;}
14     public int getY() {return y;}
15     public void setXY(int x,int y) { //x,y값을 동시에 지정하는 메소드.
16         this.x=x;
17         this.y=y;
18     }
19 }
20
21 class ColorPoint extends Point { //Point 클래스를 상속받은 ColorPoint 클래스.
22
23     private String color; //색을 나타내는 문자열 필드 변수.
24
25     ColorPoint(int x, int y, String color) { //ColorPoint의 생성자.
26         super(x, y); //부모 생성자 호출.
27         this.color=color;
28     }
29
30     public void setColor(String color) { //color에 대한 생성자.
31         this.color=color;
32     }
33
34     public String toString() { //필드 변수 정보를 문자열로 반환하는 메소드.
35         return (color + "색의" + "(" + getX() + "," + getY() + ")의 점");
36     }
37 }
38
39
40 public class ColorPointTest { //ColorPointTest 메소드.
41
42     public static void main(String[] args) {
43         ColorPoint cp=new ColorPoint(5,5,"YELLOW");
44         //(5,5)의 위치값과 노란색의 color값을 가지는 ColorPoint 객체 생성.
45         cp.setXY(10,20); //setXY 메소드(부모의 메소드)를 통해 객체의 위치값 변경.
46         cp.setColor("RED"); //setColor 메소드(자식의 메소드)를 통해 객체 색 변경.
47         System.out.println(cp.toString()+"입니다. "); //변경된 필드값 정보 출력.
48     }
49 }
50
```

Console ×

<terminated> ColorPointTest [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe (2024. 10. 1
RED색의(10,20)의 점입니다.

<실습 4> 학생 만들기

-코드 및 실행결과

```
1  /*작성일: 임다희(2312282)
2   *작성일: 2024-10-08
3   *Lab4-4 학생 만들기*/
4
5  import java.util.Scanner;
6
7  class Human { //Human 클래스 작성.
8
9      protected String name; //이름, 나이 값을 나타내는 필드 변수.
10     protected int age;
11
12     public Human(String name, int age) { //Human의 생성자.
13         this.name=name;
14         this.age=age;
15     }
16
17     //name, age에 대한 생성자와 접근자.
18     public String getName() {
19         return name;
20     }
21     public void setName(String name) {
22         this.name = name;
23     }
24     public int getAge() {
25         return age;
26     }
27     public void setAge(int age) {
28         this.age = age;
29     }
30
31     @Override //오버라이딩을 통해 필드 변수값을 문자열로 반환하는 메소드 작성.
32     public String toString() {
33         return "이름: " + name + ", 나이: " + age;
34     }
35 }
36
37 class Student extends Human { //Human 클래스를 상속받은 Student 클래스.
38
39     private String major; //전공, 학번 값을 나타내는 필드 변수.
40     private int sID;
41
42     public Student(String name, int age, String major, int sID) {
43         //Student의 생성자.
44         super(name,age); //부모의 생성자를 호출한다.
45         this.major=major;
46         this.sID=sID;
47     }
48
49     //전공 필드에 대한 접근자와 생성자.
50     public String getMajor() {
51         return major;
52     }
53     public void setMajor(String major) {
54         this.major = major;
55     }
56 }
```

```

57  @Override
58  public String toString() {
59      //오버라이딩을 통해 필드 변수값을 문자열로 반환하는 메소드 작성.
60      //super.toString()을 통해 부모의 toString 메소드를 호출해 연결한다.
61      return ("[학생 정보]" + super.toString() + ", 전공:" + major + ", 학번:" + sID);
62  }
63  }
64
65  public class StudentTest { //StudentTest 클래스 작성.
66      public static void main(String[] args) {
67
68          Scanner sc = new Scanner(System.in);
69          //크기 3의 Human 객체와 Student 객체 배열 생성.
70          Human[] Humans = new Human[3];
71          Student[] Students = new Student[3];
72
73          //사용자로부터 이름과 나이를 입력받아 3명의 Human 객체를 생성한다.
74          for(int i=0; i<Humans.length; i++) {
75              int index=i+1;
76              System.out.print("[ "+index+" ] Human 입력:");
77              Humans[i]=new Human(sc.next(), sc.nextInt());
78          }
79          //사용자로부터 이름과 나이, 전공, 학번을 입력받아 3명의 Student 객체를 생성한다.
80          for(int i=0; i<Students.length; i++) {
81              int index=i+1;
82              System.out.print("[ "+index+" ] Student 입력:");
83              Students[i]=new Student(sc.next(), sc.nextInt(), sc.next(), sc.next());
84          }
85
86          //각 객체참조변수 이름으로 객체 정보를 나타내는 출력문을 출력한다.
87          for(int i=0; i<Humans.length; i++) {
88              System.out.println(Humans[i]);
89          }
90          for(int i=0; i<Students.length; i++) {
91              System.out.println(Students[i]);
92          }
93      }
94  }
95

```

Console ×

<terminated> StudentTest (1) [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe (2024. 10. 10.)

```

[1] Human 입력:준향 18
[2] Human 입력:몽룡 21
[3] Human 입력:사또 50
[1] Student 입력:미진 21 컴퓨터 112345
[2] Student 입력:가현 22 경영 123456
[3] Student 입력:용준 24 경제 143215
이름: 준향, 나이: 18
이름: 몽룡, 나이: 21
이름: 사또, 나이: 50
[학생 정보]이름: 미진, 나이: 21, 전공:컴퓨터, 학번:112345]
[학생 정보]이름: 가현, 나이: 22, 전공:경영, 학번:123456]
[학생 정보]이름: 용준, 나이: 24, 전공:경제, 학번:143215]

```


<실습 5> Buyer 클래스 작성

-실습 5 코드

```
Problems Outline ColorPointTest.java StudentTest.java BuyerTest.java X
1  /*작성자: 임다희(2312282)
2   *작성일: 2024-10-08
3   *Lab4-5 BuyerTest 클래스*/
4
5  import java.util.Scanner;
6
7  class Item { //Item 클래스.
8      protected int price; //가격, 이름을 나타내는 필드 값.
9      protected String name;
10 }
11
12 class Food extends Item{ //Item 클래스를 상속받은 Food 클래스.
13
14     Food(String name, int price){ //Food의 생성자.
15         super.name=name; //super.부모 필드변수명.을 통해 부모의 필드값을 지정한다.
16         super.price=price;
17     }
18     @Override //오버라이딩을 통해 필드 정보를 문자열로 출력하는 메소드를 작성한다.
19     public String toString() {
20         return "[Food]" + name ;
21     }
22 }
23
24 class Book extends Item{ //Item 클래스를 상속받은 Book 클래스.
25     private String author; // Book의 필드 변수. (저자 정보를 나타냄)
26
27     Book(String name, int price,String author){ //Book의 생성자.
28         super.name=name; //부모 필드변수(이름, 가격) 지정
29         super.price=price;
30         this.author=author; //자신의 필드변수 지정
31     }
32
33     @Override //오버라이딩을 통해 필드 정보를 문자열로 출력하는 메소드를 작성한다.
34     public String toString() {
35         return "[Book]" + name + ", 저자: " + author;
36     }
37 }
38
39 class Movie extends Item{ //Item 클래스를 상속받은 Movie 클래스.
40     private String director; // Item의 필드 변수(감독 정보를 나타냄)
41
42     Movie(String name, int price, String director){ //Movie의 생성자.
43         super.name=name; //부모 필드변수(이름, 가격) 지정.
44         super.price=price;
45         this.director=director; //자신의 필드변수 지정.
46     }
47
48     public String toString() {
49         //오버라이딩을 통해 필드 정보를 문자열로 출력하는 메소드를 작성한다.
50         return "[Movie]" + name + ", 감독: " + director;
51     }
52 }
53
54 class Buyer{ //Buyer 클래스.
55     private int money; //현재 가지고 있는 돈을 나타내는 int 타입 필드변수.
56
57     Buyer(int money){ //Buyer 클래스의 생성자.
58         this.money=money;
59     }
60
61     public void buy(Item t, int n) { //물건의 이름, 수량을 받아 구매하는 buy 메소드.
62         if(money>t.price*n) { //물건의 가격*수량이 현재 보유한 돈보다 작은지 검사.
63             money-=t.price*n; //현재 보유한 돈이 충분할 경우 총 가격만큼을 차감한다.
64             System.out.println(t.toString()+"=>"+"n+"개 구매");
65             System.out.println("남은 돈: "+money);
66             //구매 정보와 남은 돈의 액수를 출력한다.
67         }
68
69         else {System.out.println("돈이 부족합니다.");
70             //돈이 충분하지 않을 경우 메시지와 남은 돈의 액수를 출력한다.
71             System.out.println("남은 돈: "+money);
72         }
73     }
74 }
75
```

```

76 public class BuyerTest{ //BuyerTest 클래스.
77     public static void main(String[] args) {
78         Scanner scan=new Scanner(System.in);
79         System.out.print("소지금액을 입력하세요: ");
80         int money=scan.nextInt(); //사용자에게 현재 소지금액을 입력받는다.
81         Buyer buyer=new Buyer(money); //입력받은 금액을 바탕으로 Buyer 객체 생성.
82         Item[] items=new Item[5]; //Item 형태를 받는 크기 5의 객체 배열을 생성한다.
83         items[0]=new Food("비빔밥", 9000);
84         items[1]=new Food("라면", 6000);
85         items[2]=new Food("김밥", 5000);
86         items[3]=new Book("자바의 정석", 20000, "남궁성");
87         items[4]=new Movie("부산행",15000, "연상호");
88
89         while(true) { //종료 조건 전까지 반복해서 상품 정보를 보여줌.
90             System.out.println("구매할 물건을 선택하세요.");
91             System.out.println("0. [Food] 비빔밥");
92             System.out.println("1. [Food] 라면");
93             System.out.println("2. [Food] 김밥");
94             System.out.println("3. [Book] 자바의 정석, 저자: 남궁성");
95             System.out.println("4. [Movie] 부산행, 감독: 연상호");
96             System.out.print("선택 : ");
97             int choice=scan.nextInt(); //사용자에게 구매할 물건과 수량을 입력받는다.
98             System.out.print("수량 : ");
99             int num=scan.nextInt();
100             //사용자 입력을 바탕으로 buy 메소드를 실행한다.
101             buyer.buy(items[choice], num);
102
103             System.out.print("계속 구매하시겠습니까?(y/n): ");
104             String answer=scan.next();
105             //물건 구매 후 계속 구매할지 여부를 사용자에게 입력받는다.
106             if (answer.equals("n")) {
107                 break; //사용자가 n을 입력하면 while루프 탈출해 실행 종료한다.
108             }
109         }
110     }
111 }

```

-실습 5 실행결과

```
81 Buyer buyer=new Buyer(money); //입력받은 금액을 바탕으로 Buyer 객체 생성.
82 Item[] items=new Item[5]; //Item 항목을 받는 크기 5의 객체 배열을 생성한다.
83

Console X
<terminated> BuyerTest [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe (2024. 10. 11. 오후
소지금액을 입력하세요 : 70000
구입할 물건을 선택하세요.
0. [Food] 비빔밥
1. [Food] 라면
2. [Food] 김밥
3. [Book] 자바의 정석, 저자: 남궁성
4. [Movie] 부산행, 감독: 연상호
선택 : 3
수량 : 1
[Book]자바의 정석, 저자: 남궁성=>1개 구매
남은 돈: 50000
계속 구매하시겠습니까?(y/n): y
구입할 물건을 선택하세요.
0. [Food] 비빔밥
1. [Food] 라면
2. [Food] 김밥
3. [Book] 자바의 정석, 저자: 남궁성
4. [Movie] 부산행, 감독: 연상호
선택 : 0
수량 : 3
[Food]비빔밥=>3개 구매
남은 돈: 23000
계속 구매하시겠습니까?(y/n): y
구입할 물건을 선택하세요.
0. [Food] 비빔밥
1. [Food] 라면
2. [Food] 김밥
3. [Book] 자바의 정석, 저자: 남궁성
4. [Movie] 부산행, 감독: 연상호
선택 : 4
수량 : 1
[Movie]부산행, 감독: 연상호=>1개 구매
남은 돈: 8000
계속 구매하시겠습니까?(y/n): y
구입할 물건을 선택하세요.
0. [Food] 비빔밥
1. [Food] 라면
2. [Food] 김밥
3. [Book] 자바의 정석, 저자: 남궁성
4. [Movie] 부산행, 감독: 연상호
선택 : 4
수량 : 1
돈이 부족합니다.
남은 돈: 8000
계속 구매하시겠습니까?(y/n): y
구입할 물건을 선택하세요.
0. [Food] 비빔밥
1. [Food] 라면
2. [Food] 김밥
3. [Book] 자바의 정석, 저자: 남궁성
4. [Movie] 부산행, 감독: 연상호
선택 : 1
수량 : 2
돈이 부족합니다.
남은 돈: 8000
계속 구매하시겠습니까?(y/n): y
구입할 물건을 선택하세요.
0. [Food] 비빔밥
1. [Food] 라면
2. [Food] 김밥
3. [Book] 자바의 정석, 저자: 남궁성
4. [Movie] 부산행, 감독: 연상호
선택 : 1
수량 : 1
[Food]라면=>1개 구매
남은 돈: 2000
계속 구매하시겠습니까?(y/n): n
```