

웹 기초 프로그래밍

[실행화면]

- 상품 목록

```
작업 메뉴를 선택하세요.
[A] 상품 추가  [R] 상품 삭제  [U] 가격 변경
[L] 전체 상품 목록  [S] 상품 조회  [E] 종료
선택 >>> 1

      상품 리스트
0:  Product(pid=1212, title=LG그램, price=345600.0)
1:  Product(pid=2499, title=LG스탠바이, price=190100.0)
2:  Product(pid=1213, title=삼성OLED, price=554500.0)
3:  Product(pid=4546, title=바디프랜드, price=632000.0)
4:  Product(pid=7890, title=삼성 갤럭시북, price=452000.0)
```

- 상품 추가

```
작업 메뉴를 선택하세요.
[A] 상품 추가  [R] 상품 삭제  [U] 가격 변경
[L] 전체 상품 목록  [S] 상품 조회  [E] 종료
선택 >>> a
      상품 제목 입력 >>> 삼성 갤럭시북
      상품 가격 입력 >>> 452000
상품 등록 을(를) 실행할까요?(확인:엔터, 취소:N)
등록 확인 : Product(pid=7890, title=삼성 갤럭시북, price=452000.0)
```

- 상품 조회

```
작업 메뉴를 선택하세요.
[A] 상품 추가  [R] 상품 삭제  [U] 가격 변경
[L] 전체 상품 목록  [S] 상품 조회  [E] 종료
선택 >>> s
      찾는 상품 제목 단어 입력 >>> LG
0:  Product(pid=1212, title=LG그램, price=345600.0)
1:  Product(pid=2499, title=LG스탠바이, price=190100.0)
```

```
작업 메뉴를 선택하세요.
[A] 상품 추가  [R] 상품 삭제  [U] 가격 변경
[L] 전체 상품 목록  [S] 상품 조회  [E] 종료
선택 >>> s
      찾는 상품 제목 단어 입력 >>> 삼성
2:  Product(pid=1213, title=삼성OLED, price=554500.0)
```

- 가격 변경

```
작업 메뉴를 선택하세요.
[A] 상품 추가  [R] 상품 삭제  [U] 가격 변경
[L] 전체 상품 목록  [S] 상품 조회  [E] 종료
선택 >>> u
      수정할 인덱스 입력 >>>>2
      판매 가격 입력 >>> 500000
포인트 변경 을(를) 실행할까요?(확인:엔터, 취소:N)
업데이트 결과 : Product(pid=1213, title=삼성OLED, price=500000.0)
```

- 상품 삭제

```
작업 메뉴를 선택하세요.
[A] 상품 추가  [R] 상품 삭제  [U] 가격 변경
[L] 전체 상품 목록  [S] 상품 조회  [E] 종료
선택 >>> r
      삭제할 인덱스 입력 >>>>a
숫자만 입력하세요.
```

```
작업 메뉴를 선택하세요.
[A] 상품 추가  [R] 상품 삭제  [U] 가격 변경
[L] 전체 상품 목록  [S] 상품 조회  [E] 종료
선택 >>> r
```

```
작업 메뉴를 선택하세요.
[A] 상품 추가  [R] 상품 삭제  [U] 가격 변경
[L] 전체 상품 목록  [S] 상품 조회  [E] 종료
선택 >>> r
      삭제할 인덱스 입력 >>>>2
삭제 을(를) 실행할까요?(확인:엔터, 취소:N)  n
삭제를 취소했습니다.
```

1. 다음 다이어그램에 따라 클래스를 작성하세요.

```
+-----+
      클래스(1) : Product
+-----+
속성 변수
- pid: int
- title: String
- price: double
+-----+
메소드
+ toString, AllArgsConstructor
+ 그 외는 문제 참고
+-----+

+-----+
      클래스명(2) : ProductManagerApp
+-----+
static 메소드
+ void initProductList(List<Product> list)
+ void addProduct(List<Product> list)
+ boolean confirm(String task)
+-----+
```

✓ 프로젝트를 새로 만드세요.
프로젝트 이름 : **java_test** , 패키지 이름 : **test**
파일명 : **ProductManagerApp.java**
 Product.java

2. 실행화면과 요구사항에 따라 상품 관리 프로그램을 만듭니다.

[요구 사항]

- `initProductList` 메소드 실행하여 주어진 4개의 **Product** 객체로 **list** 를 초기화 합니다.
- 새로운 상품 등록은 `addProduct` 메소드로 합니다.
→ 상품의 **pid** 는 4자리 정수 **1111 ~ 9999** 범위 랜덤값으로 초기화 하고 상품제목과 가격은 콘솔 입력을 받습니다.
- **list** 에 저장된 전체 상품 목록 출력과 상품명 검색을 구현합니다.
- **list** 에서 상품 삭제를 구현합니다. 삭제 대상은 인덱스로 하며 콘솔 입력 받습니다.
- **list** 에 저장된 상품을 인덱스로 지정하여 가격 수정을 구현합니다.
- 인덱스와 회원**id** 입력값의 `NumberFormatException` 예외를 처리합니다. (삭제 실행화면 참고)

✓ 프로젝트 폴더를 압축해서 파일 제출하세요.

```

package test;

import java.util.ArrayList;
import java.util.List;
import java.util.Random;

public class ProductManagerApp {
    public static void main(String[] args) {
        List<Product> list = new ArrayList<>();
        ✓_1
        System.out.println("□□□□ 상품 관리 프로그램입니다. □□□□");
        boolean status = true;
        while (status) {
            System.out.println("\n작업 메뉴를 선택하세요.");
            System.out.println("[A] 상품 추가    [R] 상품 삭제    [U] 가격 변경    \n"
                + "[L] 전체 상품 목록    [S] 상품 조회    [E] 종료    ");
            String menu = System.console().readLine("선택 >>> ");
            try {
                switch (menu.toUpperCase()) {
                    case "A":
                        addProduct(list);
                        break;
                    case "R":
                        int index = System.console().readLine("\t삭제할 인덱스 입력 >>>>");
                        ✓수정_2
                        if (confirm("삭제")) {
                            ✓_3
                            System.out.println("\t삭제 데이터 : " + p);
                        } else {
                            System.out.println("\t삭제를 취소했습니다.");
                        }
                        break;
                }
            }
        }
    }
}

```

```

case "U":
    index = Integer.parseInt(System.console().readLine("\t수정할 인덱스 입력 >>>>"));
    System.console().readLine("\t판매 가격 입력 >>> ");    √수정 _4
    if (confirm("포인트 변경")) {
        √_5
        System.out.println("\t업데이트 결과 : " );    √수정 _6
    } else {
        System.out.println("\t포인트 변경을 취소했습니다.");
    }
    break;
case "L":
    System.out.println("\t\t상품 리스트");
    for (int i = 0;    √_7    ; i++)
        System.out.println("\t" + i + ": " + list.get(i));
    break;
case "S":
    System.console().readLine("\t찾는 상품 제목 단어 입력 >>> ");    √수정 _8
    for (int i = 0; i < list.size(); i++){
        √_9
        if(p.getTitle().contains(find)) {
            System.out.println("\t" + i + ": " + p);
        }
    }
    break;
case "E":
    √_10
    break;
default:
    System.out.println("잘못된 메뉴 선택입니다.");
    break;

```

```

        } // switch end
    } √ _11수정
    System.out.println("숫자만 입력하세요.");
}
} // while end
} // main end

private static void initProductList(List<Product> list) {

    Product p1 = new Product(1212, "LG그램", 345600.0);
    Product p2 = new Product(2499, "LG스탠바이", 190100.0);
    Product p3 = new Product(1213, "삼성OLED", 554500.0);
    Product p4 = new Product(4546, "바디프랜드", 632000.0);
    √ _12
}

private static void addProduct(List<Product> list) {
    √ _13
    String title = System.console().readLine("\t상품 제목 입력 >>> ");
    double price = Double.parseDouble(System.console().readLine("\t상품 가격 입력 >>> "));
    if (confirm("상품 등록")) {
        Product p = new Product(id, title, price);
        √ _14
        System.out.println("\t등록 확인 : " + p);
    } else {
        System.out.println("\t등록 취소 했습니다.");
    }
}
}

```

```

private static boolean confirm(String task) {
    String yn = System.console().readLine(task + " 을(를) 실행할까요? (확인:엔터, 취소:N) ");
    return yn.length() != 0 && yn.toUpperCase().equals("N") ? false : true;
}
}

```

```

package test;

```

```

public class Product {
    private int pid;
    private String title;
    private double price;
    public Product(int pid, String title, double price) {
        this.pid = pid;
        this.title = title;
        this.price = price;
    }
}

```

√ _14 꼭 필요한 메소드 1개 추가

```

public void setPrice(double price) {
    this.price = price;
}

@Override
public String toString() {
    return "Product [pid=" + pid + ", title=" + title + ", price=" + price + "]";
}
}

```