

MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

■ 프로젝트의 개요

- 이번에 진행할 프로젝트는 쇼핑몰 등에서 사용할 수 있는 상품 관리 프로그램이다. 상품 정보 등록, 등록된 상품 현황 조회, 정보 수정 및 삭제 기능으로 구성한다

표 10-3 상품 관리 프로젝트 클래스

클래스	설명	비고
AppMain	프로그램의 메인 클래스로, 화면 구성과 이벤트 처리를 담당한다.	View, Controller
Product	상품 정보를 표현하는 클래스로, product 테이블과 구조가 동일하다.	Data Object
ProductDAO	데이터베이스와 연동하는 데 필요한 클래스로, 데이터베이스 연결 및 입력, 수정, 삭제 등 실제 처리 기능을 제공한다.	Data Access Object

MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

■ 프로젝트의 개요

- 화면 왼쪽에는 상품 정보를 입력하거나 수정할 수 있는 양식이, 화면 오른쪽에는 등록된 상품을 모두 보여 주는 목록을 배치

그림 10-27 프로그램 동작 화면

The screenshot shows a Java Swing window titled "Product Manager Application V1.0". Inside the window, there is a message at the top: "## 메시지: 상품정보를 가져왔습니다." Below this, on the left side, is a form with four input fields: "관리번호" (Management Number) with a dropdown menu showing "2", "상품명" (Product Name) with a text box containing "인공지능 청소기", "단가" (Unit Price) with a text box containing "560000", and "제조사" (Manufacturer) with a text box containing "LG전자". On the right side, there is a table displaying a list of products. The table has four columns: "관리번호" (Management Number), "상품명" (Product Name), "단가" (Unit Price), and "제조사" (Manufacturer). The table contains three rows of data. At the bottom of the window, there are three buttons: "등록" (Register), "조회" (Search), and "삭제" (Delete).

관리번호	상품명	단가	제조사
1	60인치 스마트TV	1500000	삼성전자
2	인공지능 청소기	560000	LG전자
3	블루투스헤드셋	80000	소니

MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

■ 프로젝트의 개요

표 10-4 프로그램 주요 기능

주요 기능	설명	비고
상품 등록	상품 정보 입력 후 [등록] 버튼을 눌러 등록한다.	등록 후에는 오른쪽 목록이 갱신되어 새로 등록한 내용을 확인할 수 있다.
상품 수정	관리번호 콤보박스에서 원하는 상품 번호를 선택하고 [조회] 버튼을 눌러 정보를 로딩한다. 내용을 수정한 후에는 [등록] 버튼을 눌러 업데이트한다.	전체를 선택할 때는 전체 목록을 다시 출력한다.
상품 삭제	관리번호 콤보박스에서 원하는 상품 정보를 불러온 후 [삭제] 버튼을 눌러 삭제한다.	삭제 후 전체 목록을 다시 출력한다.
전체 보기	관리번호 콤보박스에서 전체를 선택하고 [조회] 버튼을 누르면 전체 목록을 다시 출력한다.	프로그램의 시작은 전체 목록 보기에서 시작한다.

MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

■ 테이블 생성

- 테이블은 product 이름으로 생성하며, MySQL Workbench를 이용하여 SQL 문을 입력하거나 테이블 생성 메뉴를 이용하여 입력 방식으로 생성할 수 있다.

```
CREATE TABLE product (  
    prcode int(11) NOT NULL AUTO_INCREMENT,  
    prname varchar(45) NOT NULL,  
    price int(11) NOT NULL,  
    manufacture varchar(20) NOT NULL,  
    PRIMARY KEY(prcode)  
) ENGINE = InnoDB AUTO_INCREMENT = 15 DEFAULT CHARSET = utf8;
```

MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

■ 테이블 생성

표 10-5 product 테이블 컬럼의 용도

컬럼	데이터 타입	용도
prcode	int(11)	상품 관리번호[주 키]
prname	varchar(45)	상품 이름
price	int(11)	상품 가격
manufacture	varchar(20)	제조사

MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

■ 데이터베이스 연동 클래스 구현 : 상품 정보 데이터베이스 처리 클래스 구현

- 상품 정보 데이터베이스와 연동하여 데이터를 처리하는 데 필요한 ProductDAO 클래스를 구현해 본다.

표 10-6 ProductDAO 클래스 주요 메서드

메서드	설명
void connectDB()	DB 연결 메서드
void closeDB()	DB 연결 종료 메서드
ArrayList<Product> getAll()	전체 Product로 구성된 ArrayList를 리턴
Product getProduct(int pcode)	파라미터의 pcode에 해당하는 상품을 리턴
boolean newProduct(Product product)	파라미터의 Product 객체의 내용을 DB에 저장
boolean delProduct(int pcode)	파라미터의 pcode에 해당하는 상품을 삭제
boolean updateProduct(Product product)	파라미터의 Product 객체의 내용으로 업데이트
Vector<String> getItems()	콤보박스용 관리번호 목록을 리턴

MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

■ 데이터베이스 연동 클래스 구현 : 상품 정보 데이터베이스 처리 클래스 구현

▶ *getAll() 메서드 구현하기

getAll()은 전체 상품 목록을 가져오는 메서드이다.

[getAll() 메서드의 앞부분]

```
public ArrayList<Product> getAll() {  
    connectDB();  
    sql = "select * from product";  
  
    // 전체 검색 데이터를 전달하는 ArrayList  
    ArrayList<Product> datas = new ArrayList<Product>();  
  
    // 관리번호 콤보박스 데이터를 위한 벡터 초기화  
    items = new Vector<String>();  
    items.add("전체");  
    .....
```

MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

■ 데이터베이스 연동 클래스 구현 : 상품 정보 데이터베이스 처리 클래스 구현

▶ `getItems()` 메서드 구현하기

`getItems()`는 사용자 편의를 위해 관리번호 콤보박스에 들어갈 목록을 제공하는 메서드이다. 벡터 데이터는 `getAll()`에서 추가하며, 여기서는 단순히 `items` 변수를 리턴한다.

MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

■ GUI 처리 클래스 구현 : AppMain 클래스 개요

- GUI는 AppMain 클래스에서 구현한다.

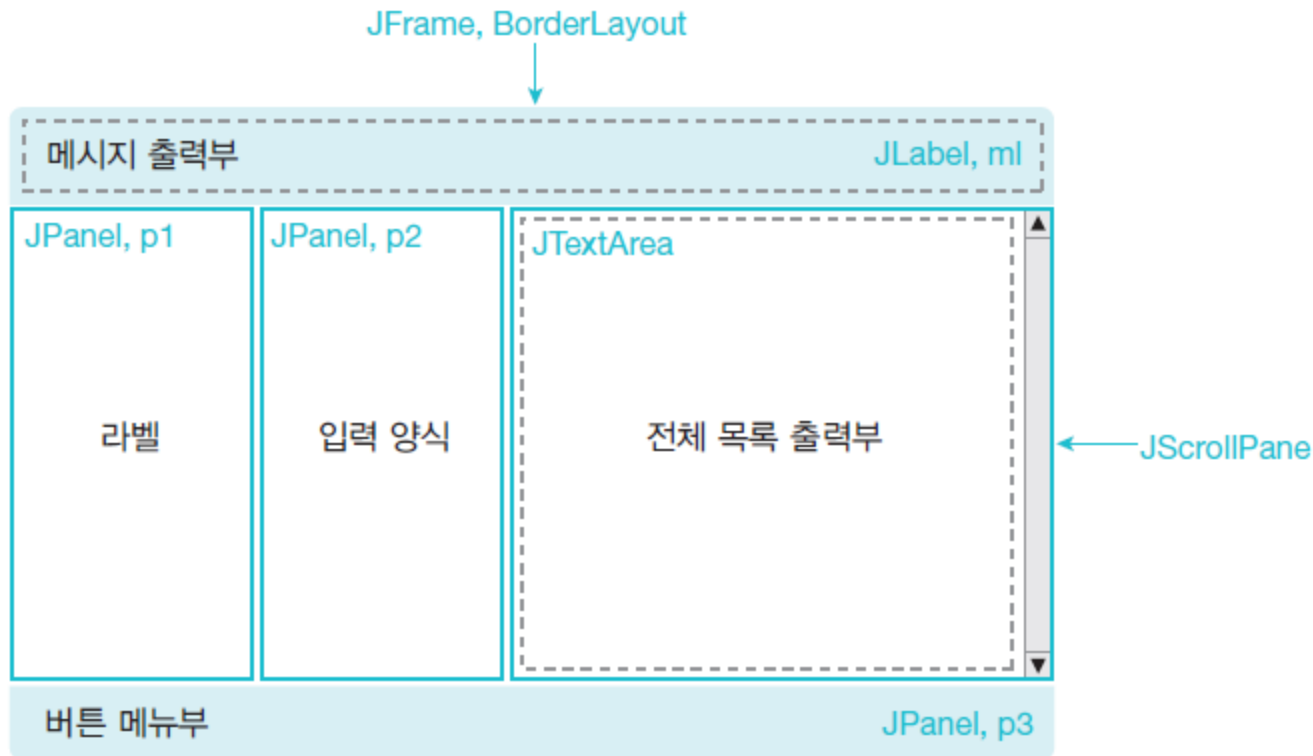
[기본 구성]

- 화면 구성을 위한 컴포넌트 선언 및 초기화
- 생성자에서 화면 크기 등 기본값 설정
- startUI() 메서드에서 화면 구성
- actionPerformed() 메서드에서 이벤트 처리
- refreshData() 메서드에서 화면 데이터 로딩 및 재로딩 처리
- clearField() 메서드에서 입력 양식 초기화

MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

■ GUI 처리 클래스 구현 : AppMain 클래스 개요

그림 10-28 전체 화면 컴포넌트와 레이아웃



MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

■ GUI 처리 클래스 구현 : 레이아웃을 이용한 패널 배치

[배치의 최종적인 BorderLayout 코드]

```
add(m1, BorderLayout.PAGE_START);  
add(p1, BorderLayout.LINE_START);  
add(p2, BorderLayout.CENTER);  
add(scroll, BorderLayout.LINE_END);  
add(p3, BorderLayout.PAGE_END);
```

MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

■ GUI 처리 클래스 구현 : 관리번호를 선택하는 콤보박스 구성

[콤보박스 생성 코드]

```
cb = new JComboBox();  
    ta = new JTextArea(10, 40);  
    JScrollPane scroll = new JScrollPane(ta, JScrollPane.VERTICAL_SCROLLBAR_  
ALWAYS, JScrollPane.HORIZONTAL_SCROLLBAR_ALWAYS);
```

MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

- GUI 처리 클래스 구현 : refreshData() 메서드 구현

- refreshData()는 콤보 박스 데이터 및 전체 목록 데이터를 그때그때 최신 데이터로 업데이트하는 메서드

예제 10-3 refreshData() 메서드 구현하기

```
01 public void refreshData() {  
02     ta.setText("");  
03     clearField();  
04     editmode = false;  
  
05  
06     ta.append("관리번호\t상품명\t\t단가\t제조사\n");  
07     datas = dao.getAll();  
  
08  
09     // 데이터를 변경하면 콤보박스 데이터 갱신  
10     cb.setModel(new DefaultComboBoxModel(dao.getItems()));  
11 }
```

MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

```
12         if(datas != null) {
13             // ArrayList의 전체 데이터를 형식에 맞춰 출력
14             for(Product p : datas) {
15                 StringBuffer sb = new StringBuffer();
16                 sb.append(p.getPrcode() + "\t");
17                 sb.append(p.getPrname() + "\t\t");
18                 sb.append(p.getPrice() + "\t");
19                 sb.append(p.getManufacture() + "\n");
20                 ta.append(sb.toString());
21             }
22         }
23         else {
24             ta.append("등록된 상품이 없습니다. !!\n상품을 등록해 주세요 !!");
25         }
26     }
```

MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

■ 이벤트 처리

- 프로그램 시작
- [등록] 버튼 클릭
- [조회] 버튼 클릭
- [삭제] 버튼 클릭

MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

■ 프로그램 실행 및 테스트

그림 10-29 최초 실행 화면

Product Manager Application V1.0

메시지: 프로그램이 시작되었습니다.

관리번호: 전체

상품명:

단가:

제조사:

관리번호	상품명	단가	제조사
------	-----	----	-----

등록 조회 삭제

MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

■ 프로그램 실행 및 테스트

그림 10-30 등록 후 관리번호 선택

The screenshot shows a Java Swing window titled "Product Manager Application V1.0". At the top, a message box says "## 메시지: 상품정보를 가져왔습니다." Below this, on the left, are four labels: "관리번호", "상품명", "단가", and "제조사". To the right of these labels are input fields. The "관리번호" field is a dropdown menu currently showing "전체". The "상품명" field shows a list with items "전체", "1", "2", and "3". The "단가" field shows "1", "2", and "3". The "제조사" field is empty. On the right side of the window is a table with four columns: "관리번호", "상품명", "단가", and "제조사". The table contains three rows of data. At the bottom of the window are three buttons: "등록", "조회", and "삭제".

관리번호	상품명	단가	제조사
1	스마트TV	1500000	삼성전자
2	인공지능 청소기	580000	LG전자
3	블루투스 헤드셋	80000	소니

MVC 기반의 JDBC를 이용한 상품 관리 프로그램 작성

■ 프로그램 실행 및 테스트

그림 10-31 두 번 상품 조회

Product Manager Application V1.0
메시지: 상품정보를 가져왔습니다.##

관리번호	상품명	단가	제조사
1	스마트TV	1500000	삼성전자
2	인공지능 청소기	580000	LG전자
3	블루투스 헤드셋	80000	소니

관리번호: 2
상품명: 인공지능 청소기
단가: 580000
제조사: LG전자

등록 **조회** 삭제