

<실습문제>

제품 관리 프로그램을 만들어 보자

1) DB 구축하기

오라클 'JDBC' 계정에 생성

아래의 테이블 참고하여 본인이 직접 테이블 구성

테이블 생성 및 하단의 샘플데이터 삽입 쿼리문은 직접 기술하시오.

(단, 테스트데이터의 INSERT 가 다 끝나면 반드시 COMMIT 까지 진행할 것!)

<PRODUCT> 테이블

PRODUCT_ID(PK)	PRODUCT_NAME(NN)	PRICE(NN)	DESCRIPTION	STOCK(NN)
nb_ss7	삼성노트북	1570000	시리즈7	10
nb_ama4	맥북에어	1200000	xcode4	20
pc_ibm	ibmPC	750000	windows10	5

- PRODUCT_ID : 상품아이디, VARCHAR2 (고유한 아이디므로 PRIMARY KEY 제약조건)
- PRODUCT_NAME : 상품명, VARCHAR2 (값이 무조건 있어야 되므로 NOT NULL 제약조건)
- PRICE : 상품가격, NUMBER (값이 무조건 있어야 되므로 NOT NULL 제약조건)
- DESCRIPTION : 상품상세정보, VARCHAR2
- STOCK : 재고, NUMBER (값이 무조건 있어야 되므로 NOT NULL 제약조건)

2) 프로젝트 만들기

✓ 프로젝트 명 : 05_JDBC_MiniProject

✓ 패키지 및 클래스 구성

com.kh.model.vo.Product

- DB에 생성 한 테이블의 컬럼과 일치하게 필드 생성.
- 생성자, getter&setter 작성, toString() 메소드 오버라이딩

com.kh.run.Run

public static void main(String[] args)

- 프로그램 실행 클래스. main 메소드 포함.
- ProductView 클래스의 mainMenu() 실행

com.kh.view.ProductView

public void mainMenu()

- 화면에 메뉴를 출력. 아래에 제시된 메뉴로 구성하고 각 메뉴를 선택하면 해당 기능 실행
- 반복문을 활용하여 반복하게 작성하며 선택한 메뉴는 switch 문을 통해 각 기능별 메소드를 실행

✓ 메뉴 구성

1. 상품 전체 조회 하기
2. 상품 추가 하기
3. 상품명 검색 하기 (상품 이름으로 키워드 검색)
4. 상품 정보 수정 하기 (상품 id로 조회하고 수정)
5. 상품 삭제 하기 (상품 id로 조회해서 삭제)
0. 프로그램 종료하기

- 서비스 요청 처리 후 사용자가 보게 되는 응답 화면에 해당하는 메소드는 자유롭게 구현

com.kh.common.JDBCTemplate

public static Connection getConnction() : 접속 시 Properties 를 이용하여 동적 코딩 할 것

public static void close(Connection conn)

public static void close(Statement stmt)

public static void close(ResultSet rset)

public static void commit(Connection conn)

public static void rollback(Connection conn)

com.kh.controller.ProductController

```
public void selectAll()
```

```
public void insertProduct(String productId, String productName, int price, String description, int stock)
```

```
public void selectByProductName(String keyword)
```

```
public void updateProduct(String productId, int price, String description, int stock)
```

```
public void deleteProduct(String productId)
```

- 입력 받은 데이터의 수가 많을 경우에는 Product 객체에 담아 ProductService 클래스로 넘긴다.

com.kh.model.service.ProductService

```
public ArrayList<Product> selectAll()
```

```
public int insertProduct(Product p)
```

```
public ArrayList<Product> selectByProductName(String keyword)
```

```
public int updateProduct(Product p)
```

```
public int deleteProduct(String productId)
```

- JDBCTemplate 클래스의 getConnection() 메소드를 호출하여 커넥션 객체를 생성 후 ProductDao 의 메소드 호출 시 매개변수로 사용하며 처리 후 JDBCTemplate 클래스의 close() 메소드 호출을
- commit 필요시 결과값에 따라 JDBCTemplate 클래스의 commit() 또는 rollback() 메소드 호출 처리
- JDBCTemplate 클래스의 close() 메소드 호출을 통해 커넥션 객체 자원 반납

com.kh.model.dao.ProductDao

```
public ArrayList<Product> selectAll(Connection conn)
```

```
public int insertProduct(Connection conn, Product p)
```

```
public ArrayList<Product> selectByProductName(Connection conn, String keyword)
```

```
public int updateProduct(Connection conn, Product p)
```

```
public int deleteProduct(Connection conn, String productId)
```

- 공통 기능: db에 접근하여 각 기능 별 쿼리 실행 후 결과 처리 및 결과값 리턴