

# Java Team Project

## 쇼핑몰 사이트 구현

---

작성자 김주현

작성날짜 2021.01

문서분류 포트폴리오

# 목차

## 1 프로젝트 개요

1-1 주제 선정

1-2 맡은 역할

## 2 프로젝트 설계

2-1 개발 환경

2-2 데이터베이스 설계

## 3 구현 화면 (UI)

3-1 메인 화면

3-2 회원가입 화면

3-3 상품목록 확인

3-4 장바구니 / 결제내역 화면

## 4 코드 리뷰

4-1 DB연결

4-2 DTO

4-3 DAO

## 5 정리

5-1 소감 및 기록

## Chapter 1

# 프로젝트 개요

# 1. 프로젝트 개요

## 주제 선정

쇼핑몰 사이트 구현

- 회원가입 및 로그인 기능
- 장바구니 기능
- 선택한 상품의 총 금액 계산 기능

## 맡은 역할

설계 및 구현

- 데이터베이스 설계
- eclipse와 오라클 데이터베이스 연동, SQL문 작성
- 최종 발표

## Chapter 2

# 프로젝트 설계

## 2. 프로젝트 설계

### 개발 환경

#### 언어

- Java

#### 도구

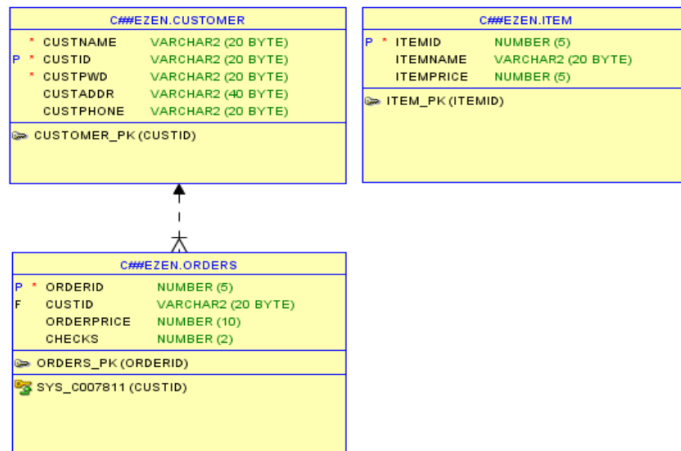
- Eclipse
- Oracle Database

## 2. 프로젝트 설계

### 데이터베이스 설계

#### DB 구조

- 테이블 3개 : 고객, 상품, 주문 관리
- 시퀀스 1개 : PK 중복방지를 위해 사용



```

CREATE TABLE CUSTOMER(
    custname varchar2(20) NOT NULL,
    custid varchar2(20) PRIMARY KEY,
    custpwd varchar2(20) NOT NULL,
    custaddr varchar2(40),
    custphone varchar2(20));

CREATE TABLE ITEM(
    itemid number(5) PRIMARY KEY,
    itemname varchar2(20),
    itemprice number(5));

CREATE TABLE ORDERS(
    orderid NUMBER(5) PRIMARY KEY,
    custid varchar2(20),
    orderprice NUMBER(10),
    checks NUMBER(2),
    FOREIGN KEY(custid) REFERENCES CUSTOMER(custid));

CREATE SEQUENCE order_seq
INCREMENT BY 1
START WITH 1
MINVALUE 1 MAXVALUE 1000
NOCYCLE;
  
```

## Chapter 3

# 구현 화면 (UI)



### 3. 구현 화면 (UI)

#### 메인 화면

로그인 전 / 후



## 3. 구현 화면 (UI)

### 회원가입 화면

아이디, 이름, 비밀번호는 필수입력사항  
중복확인 버튼을 통해 사용 가능한 아이디인지 확인

회원가입 창

아이디(\*)  중복확인

이름(\*)

비밀번호(\*)

주소

전화번호

회원가입 뒤로가기

회원가입

## 3. 구현 화면 (UI)

### 상품목록 화면

상품 우측의 체크박스를 사용하여 선택한 상품을 장바구니에 담을 수 있음



### 3. 구현 화면 (UI)

#### 장바구니 / 결제내역 화면

장바구니에서 주문번호를 선택한 후, 결제버튼을 누르면 해당 내역이 결제내역으로 이동

장바구니 창

주문번호	가격
2	50000
3	80000
4	12000

결제 삭제 뒤로가기

결제내역 창

주문번호	가격
3	80000

삭제 뒤로가기

## Chapter 4

# 코드 리뷰

## 4. 코드 리뷰

### DB 연결

DBConnection.java

- getConnection() : 데이터베이스 연결 함수
- close() : 자원 정리 함수

```
Connection conn = null;
String url = "jdbc:oracle:thin:@localhost:1521:orcl";
String id = "ezen"; //DB접속 아이디
String pw = "1234"; //DB접속 비밀번호
String driver = "oracle.jdbc.driver.OracleDriver";

try {
    Class.forName(driver);
    conn = DriverManager.getConnection(url, id, pw);
    System.out.println("DB연결 성공");
} catch (Exception e) {
    e.printStackTrace();
    System.out.println("DB연결 에러");
}

return conn;
```

## 4. 코드 리뷰

### DTO

MemberDTO.java

- private 변수 : 사용자 정보를 담는 변수
- get(), set() : private 변수에 접근하기 위한 함수

```
private String custid;    //아이디
private String custname;  //이름
private String custpwd;   //비밀번호
private String custaddr;  //주소
private String custphone; //전화번호
```

```
public String getCustid() {
    return custid;
}
public void setCustid(String custid) {
    this.custid = custid;
}
```

## 4. 코드 리뷰

### DAO (1)

MemberDAO.java

- private 변수 : 데이터베이스 연결에 사용되는 변수
- MemberDAO타입의 변수 및 함수 : 싱글톤을 위한 변수
- release() : 자원 정리 함수

```
private Connection conn;  
private PreparedStatement ps;  
private ResultSet rs;
```

```
private static MemberDAO instance = new MemberDAO();  
public static MemberDAO getInstance() { return instance; }
```



## 4. 코드 리뷰

### DAO (2)

MemberDAO.java

- joinMember() : 회원가입을 처리하는 함수

```
conn = DBConnection.getConnection();

try {
    ps = conn.prepareStatement(
        "INSERT INTO Customer(custid,custname,custpwd,custaddr,custphone)"
        + " VALUES(?,?,?,?,?)");
    ps.setString(1, member.getCustid());
    ps.setString(2, member.getCustname());
    ps.setString(3, member.getCustpwd());
    ps.setString(4, member.getCustaddr());
    ps.setString(5, member.getCustphone());

    int result = ps.executeUpdate();
}
```

## 4. 코드 리뷰

### DAO (3)

MemberDAO.java

- findByUsernameAndPassword() : 사용자 정보를 조회하는 함수

```
conn = DBConnection.getConnection();

try {
    ps = conn.prepareStatement(
        "SELECT * FROM Customer WHERE custid = ? AND custpwd = ?");
    ps.setString(1, username);
    ps.setString(2, password);

    rs = ps.executeQuery();
```

## 4. 코드 리뷰

### DAO (4)

MemberDAO.java

- loginMember() : 로그인한 사용자 정보를 가져오는 함수

```
conn = DBConnection.getConnection();

try {
    ps = conn.prepareStatement(
        "SELECT * FROM Customer WHERE custid = ? AND custpwd = ?");
    ps.setString(1, member.getCustid());
    ps.setString(2, member.getCustpwd());
    rs = ps.executeQuery();

    if (rs.next()) {
        member.setCustname(rs.getString("custname"));
        member.setCustaddr(rs.getString("custaddr"));
        member.setCustphone(rs.getString("custphone"));
    }
}
```

## Chapter 5

# 정리

## 5. 정리

### 프로젝트를 통해 배운 점

eclipse와 오라클 데이터베이스 연동 방법

데이터베이스 설계 방법 및 시퀀스의 사용법

DTO(Data Transfer Object)와 DAO(Data Access Object)의 개념과 사용법

### 소감

데이터베이스 설계는 팀원 모두와 논의하였으며 큰 문제 없이 마침.

자바 Swing을 처음 사용해보았기 때문에 페이지 디자인은 다른 팀원의 도움을 받음.

구현 과정에서 어려움이 있을 때는 구글 검색을 통해 해결함. → 블로그, Stack Overflow 사이트 참고

변수, 함수 이름은 일정한 규칙을 통해 정해야 함. → 설계 시 논의가 필요

좀 더 완성도 있는 프로그램으로 발전하기 위해서는 어떤 것이 필요한지 고민을 함. → 이후 Spring 공부

감사합니다.