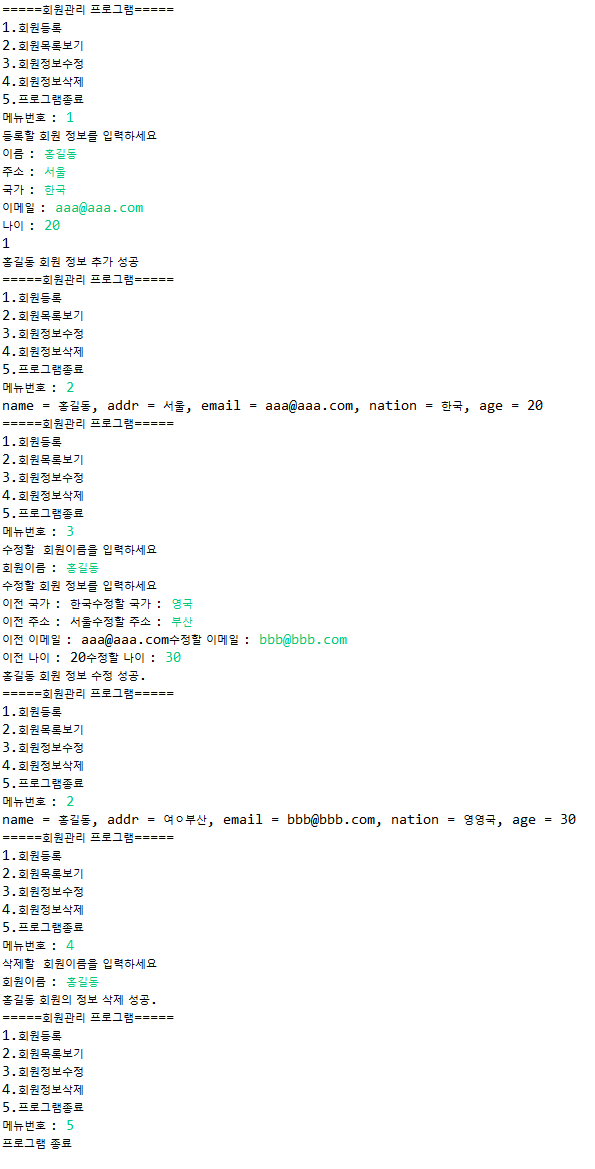
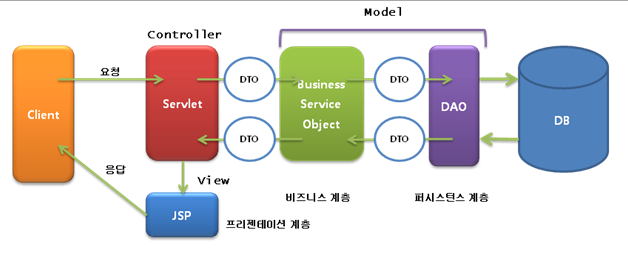
**Proj :: 회원관리**

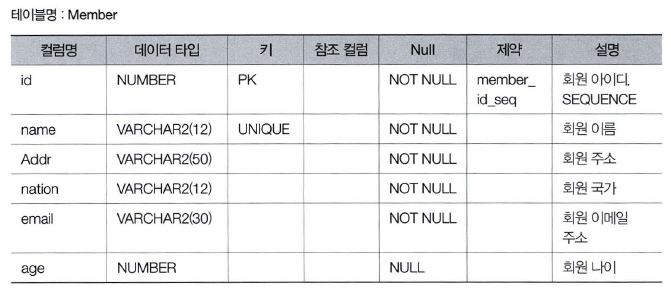
**(Java+Oracle)**

[결과]



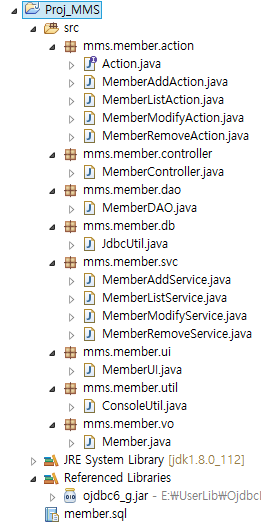


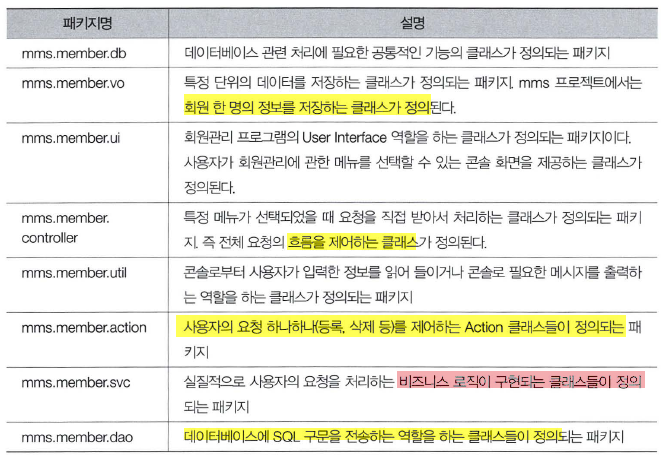
**1. 테이블 정의 및 생성**



|  |
| --- |
| **CREATE** **TABLE** mms\_member(    ); |

**2. Package 분류**





**3. 클래스 구현**

|  |
| --- |
| **package** mms.member.db;  //1. DB관련 공통 기능 클래스  **import** java.sql.Connection;  **import** java.sql.DriverManager;  **import** java.sql.ResultSet;  **import** java.sql.SQLException;  **import** java.sql.Statement;  **public** **class** JdbcUtil {  **static**{  //클래스가 로딩될 때 단 한번 호출되는 영역  //Class.forName : 특정 클래스를 메모리로 로딩하는 메소드      }    **public** **static** Connection getConnection(){  //디비 작업에 필요한 Connection 객체를 생성해 주는 메소드    }    **public** **static** **void** close(Connection con){  **try**{  con.close();  }  **catch**(SQLException e){  e.printStackTrace();  }  }      //transaction 처리 메소드  } |

|  |
| --- |
| **package** mms.member.vo;  //2. 회원 한명의 정보를 저정하는 클래스  **public** **class** Member {    **private** **int** id;  **private** String name; //식별값  **private** String addr;  **private** String nation;  **private** String email;  **private** **int** age;    } |

|  |
| --- |
| **package** mms.member.ui;  //3. 사용자가 메뉴를 선택할수 있는 화면을 구성해 주는 클래스  **import** java.util.Scanner;  **import** mms.member.action.Action;  **import** mms.member.action.MemberAddAction;  **import** mms.member.action.MemberListAction;  **import** mms.member.action.MemberModifyAction;  **import** mms.member.action.MemberRemoveAction;  **import** mms.member.controller.MemberController;  **public** **class** MemberUI {  **public** **static** **void** main(String[] args) {  **boolean** isStop = **false**;  MemberController memberController = **new** MemberController();  Scanner sc = **new** Scanner(System.***in***);  **do** {  } **while** (!isStop);  }  } |

|  |
| --- |
| **package** mms.member.action;  //4. 각 요청을 제어하는 Action 클래스 객체들을 공통적으로 접근할수 있는 인터페이스 정의  **import** java.util.Scanner;  **public** **interface** Action {  **void** execute(Scanner sc) **throws** Exception;  } |

|  |
| --- |
| **package** mms.member.util;  //5. 사용자로부터의 입력을 콘솔에서 읽어 들이거나 콘솔로 메시지를 출력하는 역할을 하는 클래스 구현  **import** java.util.ArrayList;  **import** java.util.Scanner;  **import** mms.member.vo.Member;  **public** **class** ConsoleUtil {  **public** Member getNewMember(Scanner sc) {  }  **public** **void** printAddSuccessMessage(Member newMember) {  System.***out***.println(newMember.getName() + " 회원 정보 추가 성공");  }    **public** **void** printAddFailMessage(Member newMember) {  System.***out***.println(newMember.getName() + " 회원 정보 추가 실패");  }  **public** **void** printMemberList(ArrayList<Member> memberList) {  //size() : 컬렉션에 저장되어 있는 요소의 개수를 리턴하는 메소드  }  **public** String getName(String msgKind, Scanner sc) {  }  **public** Member getUpdateMember(Scanner sc, Member oldMember) {    }  **public** **void** printModifySuccessMessage(Member updateMember) {  System.***out***.println(updateMember.getName() + " 회원 정보 수정 성공.");  }    **public** **void** printModifyFailMessage(Member updateMember) {  System.***out***.println(updateMember.getName() + " 회원 정보 수정 실패.");  }  **public** **void** printRemoveSuccessMessage(String name) {  System.***out***.println(name + " 회원의 정보 삭제 성공.");  }  **public** **void** printRemoveFailMessage(String name) {  System.***out***.println(name + " 회원의 정보 삭제 실패.");  }  } |

|  |
| --- |
| **package** mms.member.controller;  //6. 사용자의 전체 요청을 제어하는 클래스 구현  **import** java.util.Scanner;  **import** mms.member.action.Action;  **public** **class** MemberController {    **public** **void** processRequest(Action action,Scanner sc){  **try**{  action.execute(sc);  //인터페이스의 레퍼런스 변수가 인터페이스를 구현한 클래스 객체를 참조하며  //인테페이스에 정의된 메소드를 호출하면 실질적으로 각 클래스에서 구현된 메소드가 호출된다.  }  **catch**(Exception e){  e.printStackTrace();  //e.printStackTrace() : 예외 클래스 에서 제공되는 메소드 이며 스택에 저장되어 있는 메소드 호출 기록을 추적하는 메소드 주로 디버깅시 사용됨.  }  }  } |

|  |
| --- |
| **package** mms.member.action;  //7-1 회원등록 요청 처리하는 Action 클래스 구현  **import** java.util.Scanner;  **import** mms.member.svc.MemberAddService;  **import** mms.member.util.ConsoleUtil;  **import** mms.member.vo.Member;  //각 요청을 제어하는 클래스  //주로 실질적인 비지니스 로직을 처리하기 전에 필요한 선행작업들을 처리함.  //만약, 웹에서 회원등록 페이지에 정보를 입력하고 회원등록 요청이 넘어왔을 때 페이지에서 넘어오는 정보를 받는다든지......  **public** **class** MemberAddAction **implements** Action {  @Override  **public** **void** execute(Scanner sc) **throws** Exception {  } |

|  |
| --- |
| **package** mms.member.action;  //7-2 회원등록 보기 요청 처리하는 Action 클래스 구현  **import** java.util.ArrayList;  **import** java.util.Scanner;  **import** mms.member.svc.MemberListService;  **import** mms.member.util.ConsoleUtil;  **import** mms.member.vo.Member;  **public** **class** MemberListAction **implements** Action {  @Override  **public** **void** execute(Scanner sc) {    }  } |

|  |
| --- |
| **package** mms.member.action;  //7-3 회원 정보 수정 요청을 처리하는 Action 클래스 구현  **import** java.util.Scanner;  **import** mms.member.svc.MemberModifyService;  **import** mms.member.util.ConsoleUtil;  **import** mms.member.vo.Member;  **public** **class** MemberModifyAction **implements** Action {  @Override  **public** **void** execute(Scanner sc) {    }  } |

|  |
| --- |
| **package** mms.member.action;  //7-4 회원 정보 수정 요청을 처리하는 Action 클래스 구현  **import** java.util.Scanner;  **import** mms.member.svc.MemberRemoveService;  **import** mms.member.util.ConsoleUtil;  **public** **class** MemberRemoveAction **implements** Action {  @Override  **public** **void** execute(Scanner sc) {      }  } |

|  |
| --- |
| **package** mms.member.svc;  //8-1. 회원등록 Business Logic이 구현되는 Service 클래스 구현  //트렌젝션 처리는 비지니스 단 에서 해 주는 것이 Convention 이다.  //즉, Connection 객체는 비지니스 단 에서 생성해서 사용하는 것이 일반적이다.  **import** **static** mms.member.db.JdbcUtil.\*;  **import** java.sql.Connection;  **import** mms.member.dao.MemberDAO;  **import** mms.member.vo.Member;  **public** **class** MemberAddService {  **public** **boolean** addMember(Member newMember) **throws** Exception{    }  } |

|  |
| --- |
| **package** mms.member.svc;  //8-2. 회원목록 보기 요청을 처리하는 Business Logic이 구현되는 Service 클래스 구현  **import** java.sql.Connection;  **import** java.util.ArrayList;  **import** **static** mms.member.db.JdbcUtil.\*;  **import** mms.member.dao.MemberDAO;  **import** mms.member.vo.Member;  **public** **class** MemberListService {  **public** ArrayList<Member> getMemberList() {      }  } |

|  |
| --- |
| **package** mms.member.svc;  //8-3. 회원정보 수정 요청을 처리하는 Business Logic이 구현되는 Service 클래스 구현  **import** java.sql.Connection;  **import** **static** mms.member.db.JdbcUtil.\*;  **import** mms.member.dao.MemberDAO;  **import** mms.member.vo.Member;  **public** **class** MemberModifyService {  **public** Member getOldMember(String name) {      }  **public** **boolean** modifyMember(Member updateMember) {      }  } |

|  |
| --- |
| **package** mms.member.svc;  //8-4. 회원정보 삭제 요청을 처리하는 Business Logic이 구현되는 Service 클래스 구현  **import** **static** mms.member.db.JdbcUtil.\*;  **import** java.sql.Connection;  **import** mms.member.dao.MemberDAO;  **public** **class** MemberRemoveService {  **public** **boolean** removeMember(String name) {      }  } |

|  |
| --- |
| **package** mms.member.dao;  //9. Oracle DB로 필요한 SQL구문을 전송하는 클래스 구현  **import** **static** mms.member.db.JdbcUtil.\*;  **import** java.sql.Connection;  **import** java.sql.PreparedStatement;  **import** java.sql.ResultSet;  **import** java.sql.SQLException;  **import** java.util.ArrayList;  **import** mms.member.vo.Member;  **public** **class** MemberDAO {    Connection con;  **public** MemberDAO(Connection con) {  **this**.con = con;  }    **public** **int** insertNewMember(Member newMember) {      }    **public** ArrayList<Member> selectMemberList() {      }    **public** Member selectOldMember(String name) {      }    **public** **int** updateMember(Member updateMember) {      }    **public** **int** deleteMember(String name) {      }  } |