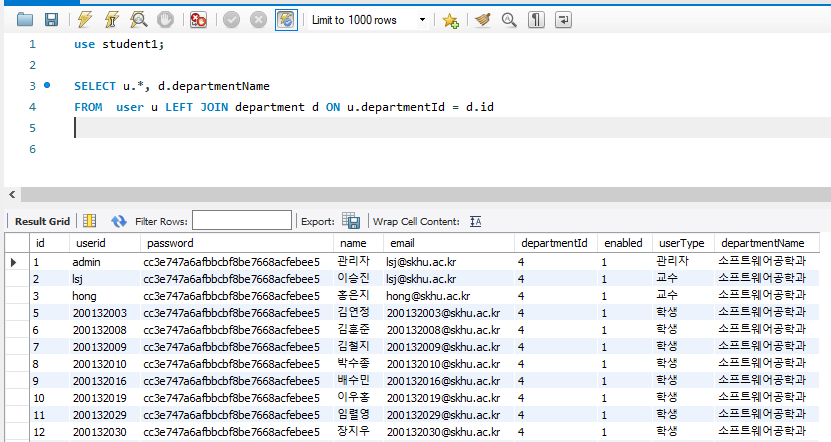
## SQL



## User.java

|  |
| --- |
| package lecture1.jdbc1;  public class User {  int id;  String userid;  String password;  String name;  String email;  int departmentId;  String departmentName;  boolean enabled;  String userType;  public int getId() {  return id;  }  public void setId(int id) {  this.id = id;  }  public String getUserid() {  return userid;  }  public void setUserid(String userid) {  this.userid = userid;  }  public String getPassword() {  return password;  }  public void setPassword(String password) {  this.password = password;  }  public String getName() {  return name;  }  public void setName(String name) {  this.name = name;  }  public String getEmail() {  return email;  }  public void setEmail(String email) {  this.email = email;  }  public int getDepartmentId() {  return departmentId;  }  public void setDepartmentId(int departmentId) {  this.departmentId = departmentId;  }  public String getDepartmentName() {  return departmentName;  }  public void setDepartmentName(String departmentName) {  this.departmentName = departmentName;  }  public boolean isEnabled() {  return enabled;  }  public void setEnabled(boolean enabled) {  this.enabled = enabled;  }  public String getUserType() {  return userType;  }  public void setUserType(String userType) {  this.userType = userType;  }  } |

멤버 변수 값을 리턴하는 메소드의 이름은 get으로 시작해야 한다.

그런데, 멤버 변수의 타입이 boolean인 경우에는 (예: boolean enabled

get으로 시작해도 되고 is로 시작해도 된다. (예: getEnabled(), isEnabled()

is로 시작하는 것이 좀더 낫다.

## UserDAO.java

|  |
| --- |
| package lecture1.jdbc1;  import java.sql.Connection;  import java.sql.PreparedStatement;  import java.sql.ResultSet;  import java.util.ArrayList;  import java.util.List;  import lecture1.DB;  public class UserDAO {  public static List<User> findAll() throws Exception {  String sql = "SELECT u.\*, d.departmentName " +  "FROM user u LEFT JOIN department d ON u.departmentId = d.id";  try (Connection connection = DB.getConnection("student1");  PreparedStatement statement = connection.prepareStatement(sql);  ResultSet resultSet = statement.executeQuery()) {  ArrayList<User> list = new ArrayList<User>();  while (resultSet.next()) {  User user = new User();  user.setId(resultSet.getInt("id"));  user.setUserid(resultSet.getString("userid"));  user.setName(resultSet.getString("name"));  user.setPassword(resultSet.getString("password"));  user.setEmail(resultSet.getString("email"));  user.setDepartmentId(resultSet.getInt("departmentId"));  user.setDepartmentName(resultSet.getString("departmentName"));  user.setEnabled(resultSet.getBoolean("enabled"));  user.setUserType(resultSet.getString("userType"));  list.add(user);  }  return list;  }  }  } |

## userList.jsp

|  |
| --- |
| <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>  <%@ page import="java.util.List, lecture1.jdbc1.\*" %>  <%  List<User> list = UserDAO.findAll();  %>  <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>  <style>  body { font-family: 굴림체; }  thead th { background-color: #eee; }  table.table { width: 700px; }  </style>  </head>  <body>  <div class="container">  <h1>학생목록</h1>  <table class="table table-bordered table-condensed">  <thead>  <tr>  <th>아이디</th>  <th>이름</th>  <th>이메일</th>  <th>학과</th>  <th>사용자유형</th>  </tr>  </thead>  <tbody>  <% for (User usre : list) { %>  <tr>  <td><%= usre.getUserid() %></td>  <td><%= usre.getName() %></td>  <td><%= usre.getEmail() %></td>  <td><%= usre.getDepartmentName() %></td>  <td><%= usre.getUserType() %></td>  </tr>  <% } %>  </tbody>  </table>  </div>  </body>  </html> |