
윈도우 프로그래밍 실습 결과 보고서

연구과제명 | 2048



학과 | 컴퓨터정보과

담당교수 | 임성택

제작자 | 정의창, 정인환

작성년월 | 2019년 06월 26일

작 품 설 명 서

작 품 명	2048		
학 과	컴퓨터정보과	지 도 교 수	임성택
제 작 자	정의창, 정인환		
작 품 개 요	현재 존재하는 '2048' 게임을 플레이해 게임의 원리와 구조를 파악하여 코딩을 함으로써 코드 구상력을 높이기 위해 시작한 프로젝트		
동 작 기 능	Form	DB	
	<ul style="list-style-type: none">- JFrame을 이용한 폼- 각 컨트롤을 이용한 이벤트	<ul style="list-style-type: none">- 유저 정보 저장- 회원 정보 저장- 게임 정보 저장- 랭킹 정보 저장- 쿼리문을 이용한 정보 관리	
활 용 방 법	방향키를 누르면서 같은 숫자블록끼리 합쳐지게 하여 최대한 높은 숫자블록을 만드는 게임		
제 작 기 술	JAVA, MySQL		

목 차

I. 서 론

- 가. 세부 계획표
- 나. 프로젝트 제작 동기

II. 동작원리 및 관계이론

- 가. DB 설계
- 나. DB 연동
- 다. 쿼리문
- 라. JFrame

III. 작품구성

- 가. 2048 DataBase
- 나. 화면구성 및 주요기능
 - 1. 시작화면
 - 2. 회원가입
 - 3. 게임화면
 - 4. 랭킹

IV. 기타

- 가. 기대효과 및 활용방안
- 나. 보완해야 할 점
- 다. 참조 문헌

I. 서론

가. 세부 계획표

1. 기존 2048 게임 조사 및 분석을 통한 구조 확립
 - 방향키를 움직여서 블록을 움직이는 기능을 구현하려면 어떤 구조로 코드를 구성해야 하는지, 또한 어떤 기능이 필요한지 기본적인 구조를 설계하였다.
2. 시간을 분배를 통한 기간선정 및 효율성 추구
 - 정해진 기간에 맞게 적절한 분배를 하고, 설계하여 주어진 과제를 효율적으로 진행할 수 있도록 하였다.
3. 언어 환경, 버전 확인 및 DB 설계
 - 개발자가 사용하고 있는 JAVA, JDK 및 SQL 등 환경과 버전에 따라 오류가 발생할 수 있다. 따라서 개발 전 환경에 맞게 준비하고, 프로그램에 필요한 DB를 설계하였다.
4. 해당 작품 JFrame설계
 - UI화면을 설계하기 위한 환경으로는 JFrame을 사용하였다.
5. 디자인 완성 및 소스 입력
 - 2048 프로젝트에 맞는 디자인, 기능 등을 고려하여 코드를 구성하였다.

나. 프로젝트 제작 동기

'2048'게임의 구조를 파악해서 다소 복잡한 코드를 직접 구상하고, 짜며 코드 구상력을 높이기 위해 기획하였다. 기존에 있던 '2048'은 단순하게 플레이기능과 점수를 출력해주는 기능만 있지만, 이 프로젝트는 '2048'이 가지고 있는 모든 기능들을 바탕으로 유저가 중간에 플레이하던 정보를 저장해주는 기능과 게임오버시 점수를 저장하는 기능을 추가하기 위해 유저들을 관리하는 유저관리기능을 추가하여 제작하게 되었다.

II. 동작원리 및 관계이론

가. DB 설계

원하는 정보를 DB를 사용할 프로그램에 맞게 DB를 설계하고, 생성하는 작업이다.

행 이름	데이터 타입	길이	디폴트 (초기화)	PK?	Null이 아님?	부호 없이?	자동 증가?	Zerofill?
user_ID	varchar	10		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
user_PW	varchar	10		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

그림 1. 유저의 정보를 저장하는 테이블

나. DB 연동

JAVA에서 MySQL을 사용하기 위해서는 JAVA과 MySQL의 커넥터인 'MySQL Connector/J'를 설치해야 한다. 아래는 DB연동을 위한 코드이다.

```
static final String JDBC_DRIVER = "com.mysql.jdbc.Driver"; //jdbc 드라이버 주소
static final String DB_URL = "jdbc:mysql://127.0.0.1:3306/java";
//jdbc:mysql://IP:포트 /DB이름

static final String USERNAME = "admin";
static final String PASSWORD = "root";

static Connection conn = null;
static Statement stmt = null;
static PreparedStatement st = null;
```

그림 2. DB접속코드

다. 쿼리문

데이터 관리를 위해서 INSERT, SELECT, DELETE, UPDATE 등과 같은 쿼리문을 이용해서 정보들을 입력, 수정, 삭제, 불러오기 등의 기능을 수행한다.

```
String sel;  
sel = "SELECT * FROM rank ORDER BY jumsu DESC LIMIT 10";  
String del;  
del = "DELETE FROM save WHERE user_ID='"+ nowid + "'";  
String ins;  
ins = "INSERT INTO user VALUES(?,?)";
```

그림 3. 쿼리문 예시

라. JFrame

사용자가 컨트롤을 사용해서 원하는 화면을 구성하고 이벤트 발생시 처리하고자 하는 작업을 기술하는 방식으로 프로그래밍이 진행된다.

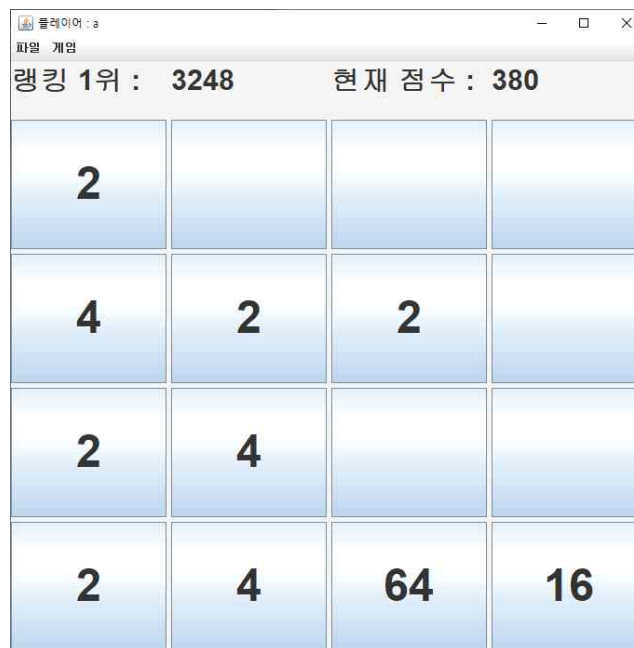
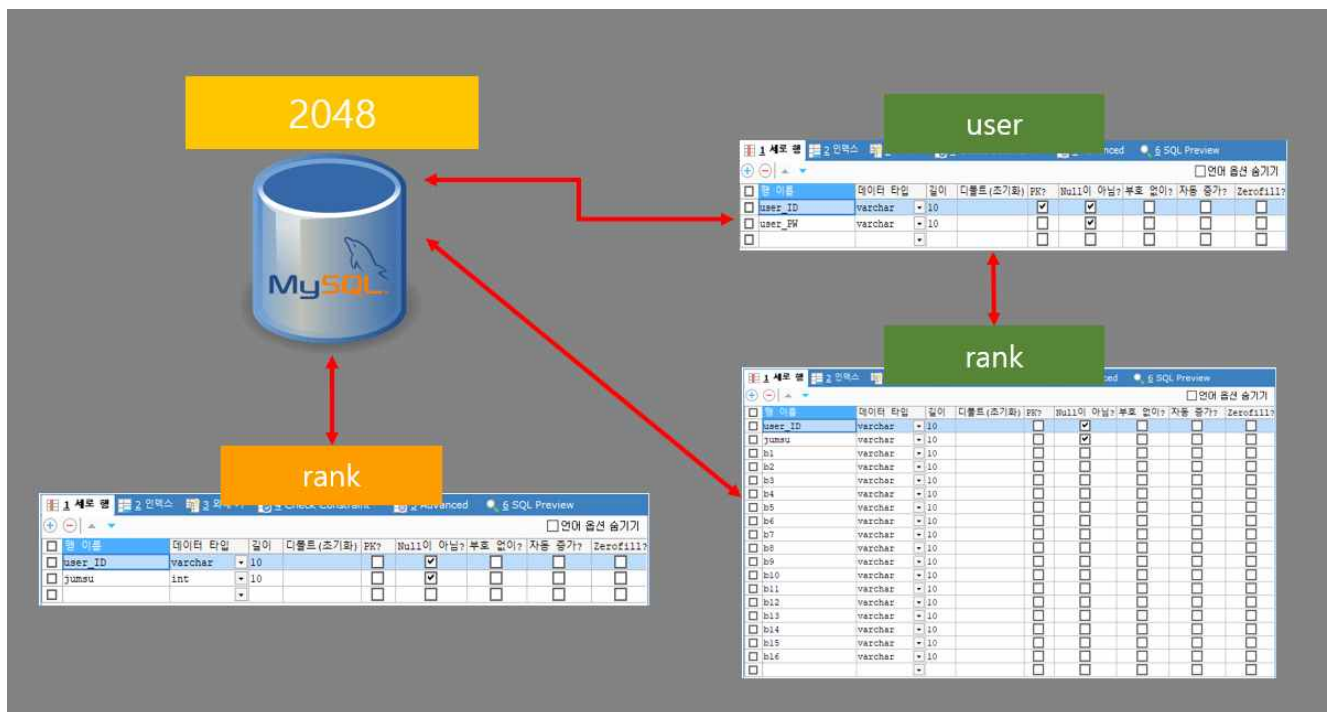


그림 4. 2048 프로젝트 게임 진행화면

Ⅲ. 작품구성

가. 2048 DataBase

*2048에서 사용하는 테이블



user : 유저정보가 들어있는 테이블

save : 유저가 저장한 게임진행정보가 들어있는 테이블

rank : 게임오버된 유저들의 아이디와 점수가 들어있는 테이블

나. 화면구성 및 주요기능

1. 시작화면

*프로젝트 실행 후 시작화면

회원가입과, 로그인을 할 수 있는 창.



*핵심 코드

```
String sql;
//현재 입력된 아이디와 동일한 아이디가 있는지 검사
sql = "SELECT * FROM user WHERE user_ID =" + id.getText() +
      "' AND user_PW =" + pw.getText() + "'";
```

2. 회원가입

*시작화면에서 회원가입버튼을 통해 생성되는 창

유저의 정보를 생성할 수 있는 창



*핵심 코드

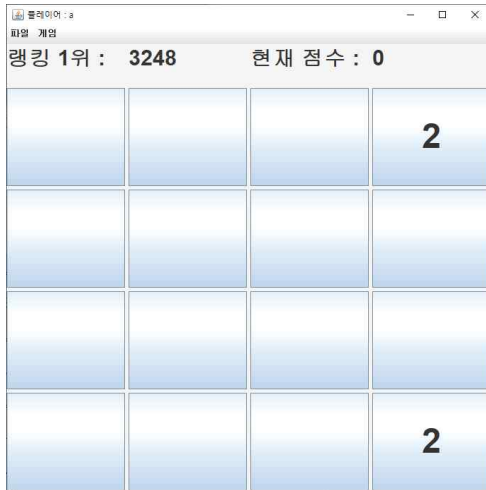
```
if(id.getText().equals("")||pw.getText().equals(""))
{
    JOptionPane.showMessageDialog(null, "아이디와 패스워드를 입력해주세요!");
}
else
{
    String sql1;
    sql1 = "INSERT INTO user VALUES (?,?)";
    st = conn.prepareStatement(sql1);
    st.setString(1, id.getText());
    st.setString(2, pw.getText());
    st.executeUpdate();

    JOptionPane.showMessageDialog(null, "회원가입 완료!! 환영합니다~!");
    this.dispose();
}
```


3.게임화면

*로그인 성공 후 생성되는 게임을 진행할 수 있는 창

저장, 랭킹을 볼 수 있는 메뉴바와, 방향키를 이용해 게임을 진행



*핵심 코드(↑ 방향키를 입력했을 때 실행되는 코드)

```
case KeyEvent.VK_UP:
    for(int j =4; j <= 7; j++)
    {
        for(int i = j; i <= j+8; i=i+4)
        {
            if(!g.btn[i].getText().equals(""))
            {
                if(g.btn[i-4].getText().equals(""))
                {
                    g.btn[i-4].setText(g.btn[i].getText());
                    g.btn[i].setText("");
                    i = j-4;
                }
                else
                {
                    if(g.btn[i-4].getText().equals(g.btn[i].getText()))
                    {
                        g.btn[i-4].setText(Integer.toString(Integer.parseInt(g.btn[i-4].getText())*2));
                        g.jumsu.setText(Integer.toString(Integer.parseInt(g.jumsu.getText())+Integer.parseInt(g.btn[i-4].getText())));
                        g.btn[i].setText("");
                        i = i-4;
                    }
                }
            }
        }
    }
}
```

*핵심 코드(게임오버되는 조건)

```
if (up+down+left+right==4)
{
    try{
        Class.forName(JDBC_DRIVER) ;//JDBC드라이버 로드
        conn = DriverManager.getConnection(DB_URL, USERNAME, PASSWORD) ;//DB연결
        stmt = conn.createStatement();

        String gameover;
        gameover = "INSERT INTO rank VALUES(?,?)";

        st = conn.prepareStatement(gameover);
        st.setString(1, nowid);
        st.setInt(2, Integer.parseInt(g.jumsu.getText()));
        st.executeUpdate();

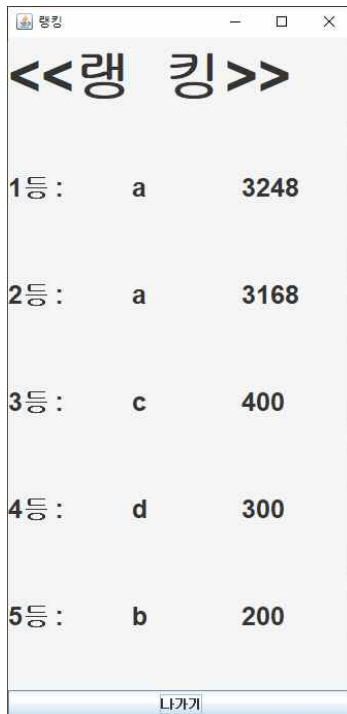
        String savecheck;
        savecheck = "SELECT * FROM save WHERE user_ID =' " + nowid + " " ;

        ResultSet rs = stmt.executeQuery(savecheck);
        int s=0;
        while(rs.next())
        {
            s++;
        }
        rs.close();
    }
```

4.랭킹

*유저들의 점수를 출력해주는 창

게임오버시 점수를 DB에 저장해 오름차순으로 정렬하여 이름과 점수 출력



*핵심 코드

```
while(rs.next())
{
    stack++;
}
rs.close();
String username1;
username1 = "SELECT * FROM rank ORDER BY jumsu DESC LIMIT 10";
ResultSet rs1 = stmt.executeQuery(username1);
int i = 0;
while(rs1.next()){
    rank[i] = new JLabel();
    rank[i].setText(Integer.toString(i+1)+"등:");
    rank[i].setFont(f);
    p.add(rank[i]);

    name[i] = new JLabel();
    name[i].setText(" "+rs1.getString(1));
    name[i].setFont(f);
    p.add(name[i]);

    jumsu[i] = new JLabel();
    jumsu[i].setText(Integer.toString(rs1.getInt(2)));
    jumsu[i].setFont(f);
    p.add(jumsu[i]);
    i++;
    if(stack == i) break;
}
```

IV. 기타

가. 기대효과 및 활용방안

2048프로젝트를 이용해 간단한 플레이 방법과, 조작법을 통해서 누구나 즐길 수 있는 게임 프로젝트이다. 또한 생각하는 방식에 따라서 다양한 방법으로 플레이 할 수 있고, 랭킹을 통해서 경쟁심리도 자극한다. 네트워크 기능을 추가한다면 온라인으로 플레이하여 친구들 및 다른 유저들과 플레이 하여 더 재밌는 게임이 될 수 있다.

나. 보완해야 할 점

디자인 : 시간이 부족해 디자인이 매우 미흡

기능 : 비회원도 플레이할 수 있는 기능추가필요(저장 안됨), TCP/IP를 이용해 네트워크를 연동하여
온라인으로 플레이 할수 있는 기능추가필요

다. 참조문헌

MySQL DB생성 및 테이블 관리 : 은노기의 JSP 웹 프로그래밍 입문

JAVA와 MySQL연동 : <https://blog.naver.com/opgj123/221318458638>

SELECT 쿼리문(LIMIT) :

<http://blog.naver.com/PostView.nhn?blogId=50after&logNo=220939872934&categoryNo=13&parentCategoryNo=0&viewDate=¤tPage=1&postListTopCurrentPage=1&from=postView>

JAVA 메시지 박스 : <https://hallang.tistory.com/136>