

KINECT PROGRAMMING

청기백기



GAME START!

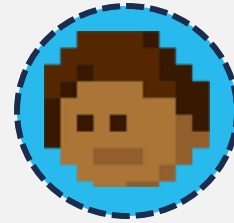


팀원소개



컴퓨터정보과
2-C
202044095

고원영



컴퓨터정보과
2-C
201844071

김규준



컴퓨터정보과
2-C
202044070

이아름별



컴퓨터정보과
2-C
202044096

임병준

목차

- Hi! 1. 프로젝트 설명
- Hi! 2. 회원가입 및 로그인
- Hi! 3. 게임 조건 및 실행
- Hi! 4. 랭킹

Chapter 01

프로젝트 설명

Level Up!



현재, 날이 갈수록 청년층은 감소하고, 노년층의 비율은 증가하는
고령화가 빠른 속도로 진행되고 있다.

이에 따른 현상으로 노년층 치매 환자들도 더욱이 증가하는 추세이다.

그로 인해 치매 예방의 중요성이 대두되는 시대이므로

이번 프로젝트에서는 치매 예방 애플리케이션을 설계하고 구현한다.

해당 애플리케이션은 키넥트 센서의 관절 인식 기능과 음성 기능을 이용하여
실행되게 한다.

Chapter 02. 회원가입 및 로그인

회원가입

아이디

비밀번호

이름

생년월일(6자리)

이메일

완료

▶ 회원가입 화면

LOGIN

아이디

비밀번호

LOGIN 회원가입

비밀번호 찾기

▶ 로그인 화면

Chapter 02. 회원가입 및 로그인

로그인 성공 시



로그인 실패 시



소스 코드

```
1 using System;
2 using System.Collections.Generic;
3 using System.Data;
4 using System.Linq;
5 using System.Text;
6 using System.Threading.Tasks;
7 using System.Windows;
8 using System.Windows.Controls;
9 using System.Windows.Data;
10 using System.Windows.Documents;
11 using System.Windows.Input;
12 using System.Windows.Media;
13 using System.Windows.Media.Imaging;
14 using System.Windows.Navigation;
15 using System.Windows.Shapes;
16 using MySql.Data.MySqlClient;

17
18 namespace Login2
19 {
20     /// <summary>
21     /// Interacção lógica para MainWindow.xaml
22     /// </summary>
23     참조 6개
24     public partial class MainWindow : Window
25     {
26         RegisForm regisform;
27         Game game;
28         //Game game;
29         참조 0개
30         public MainWindow()
31         {
32             InitializeComponent();
33             regisform = new RegisForm(this);
34             game = new Game(this);
35         }
36     }
37 }
```

```
37
38
39 private void LoginButton_Click(object sender, RoutedEventArgs e)
40 {
41     if (checkId(this.dataId.Text) && checkPw(this.dataPw.Password))
42     {
43         //form2.myPicture.Image = Image.FromFile(form2.getPicture());
44         //form2.textId.Text = this.dataId.Text;
45         string username = checkName(this.dataId.Text);
46         MessageBox.Show(username + "님 환영합니다");
47         changeToGame();
48     }
49     else
50     {
51         MessageBox.Show("올바른 아이디 또는 패스워드가 아닙니다.");
52     }
53 }
54
55 public bool checkId(string id) //id check
56 {
57     string mysqldb = "Server = localhost; Port = 3306; Database = kinect; Uid = root; Pwd = rootpw";
58     MySqlConnection conn = new MySqlConnection(mysqldb);
59     conn.Open();
60
61     string sql = "select id from base where id='" + this.dataId.Text + "'";
62     MySqlCommand cmd = new MySqlCommand(sql, conn);
63     cmd.CommandType = CommandType.Text;
64
65     MySqlDataReader dr = cmd.ExecuteReader();
66     bool result = false;
67     if (dr.HasRows) result = true;
68     else result = false;
69
70     conn.Close();
71     conn.Dispose();
72
73     return result;
74 }
```

소스 코드

```
72 public bool checkPW(string pw) //pwd check
73 {
74     string mysqldb = "Server = localhost; Port = 3306; Database = kinect; Uid = root; Pwd = rootpw";
75     MySqlConnection conn = new MySqlConnection(mysqldb);
76     conn.Open();
77
78     string sql = "select pwd from base where id=@'" + this.dataId.Text + "'";
79     MySqlCommand cmd = new MySqlCommand(sql, conn);
80     cmd.CommandType = CommandType.Text;
81
82     MySqlDataReader dr = cmd.ExecuteReader();
83
84     bool result = false;
85     dr.Read();
86     if (pw == dr.GetString(0)) result = true;
87     else result = false;
88
89
90     conn.Close();
91     conn.Dispose();
92
93     return result;
94 }
```


소스 코드

```
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125
```

```
public void changeToGame()  
{  
    game.Show();  
    this.Hide();  
}  
  
참조 1개  
public void usingSQL(string sql)  
{  
    string mysqldb = "Server = localhost; Port = 3306; Database = kinect; Uid = root; Pwd = rootpw";  
    MySqlConnection conn = new MySqlConnection(mysqldb);  
    conn.Open();  
  
    MySqlCommand cmd = new MySqlCommand(sql, conn);  
    cmd.CommandType = CommandType.Text;  
  
    MySqlDataReader dr = cmd.ExecuteReader();  
    while (dr.Read())  
    {  
        Console.WriteLine(dr[0].ToString() + "\n");  
        Console.WriteLine(dr.GetString(1).ToString() + "\n");  
    }  
    conn.Close();  
    conn.Dispose();  
}  
  
참조 1개  
private void RegisButton_Click(object sender, RoutedEventArgs e)  
{  
    regisform = new RegisForm(this);  
    changeToRegisform();  
}
```

소스 코드

```
127 public void changeToregistform()  
128 {  
129     regisform.Show();  
130     this.Hide();  
131 }  
132  
참조 1개  
133 public string checkName(string id)  
134 {  
135     string mysqlDb = "Server = localhost; Port = 3306; Database = kinect; Uid = root; Pwd = rootpw";  
136     MySqlConnection conn = new MySqlConnection(mysqlDb);  
137     conn.Open();  
138  
139     string sql = "select name from base where id='" + id + "'";  
140     MySqlCommand cmd = new MySqlCommand(sql, conn);  
141     cmd.CommandType = CommandType.Text;  
142  
143     MySqlDataReader dr = cmd.ExecuteReader();  
144     dr.Read();  
145     string result = dr.GetString(0);  
146     conn.Close();  
147     conn.Dispose();  
148  
149     return result;  
150 }  
151 }  
152 }  
153 }
```

Chapter 03. 게임 조건 및 실행

- 사람이 있는 지부터 우선적으로 인식 되어야 함
- 우측 손, 좌측 손이 모두 인식 되어야 함
- 게임 시작은 두 손이 모두 올라간 경우 시작됨
- 랭킹은 제한 시간 내에 성공한 횟수에 따라 차등 분배
- 오른손은 청기, 왼손은 백기로 세팅
- 화면에 랜덤으로 청기 혹은 백기를 올리거나 내리라는 메시지를 출력해준다
- 손의 관절 좌표를 활용하여 어느 손이 올라갔는지를 인식
- 60초의 시간 제한
- 조건대로 실행이 잘 되었는지 확인하여 성공 여부를 출력해주고 점수를 저장해준다

초기 화면

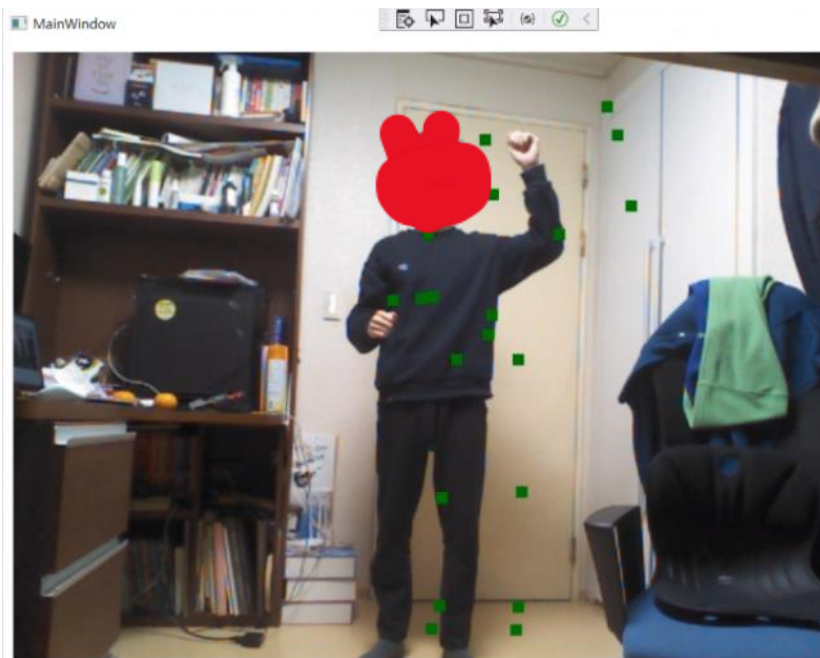


게임시작

순위

게임하기 페이지

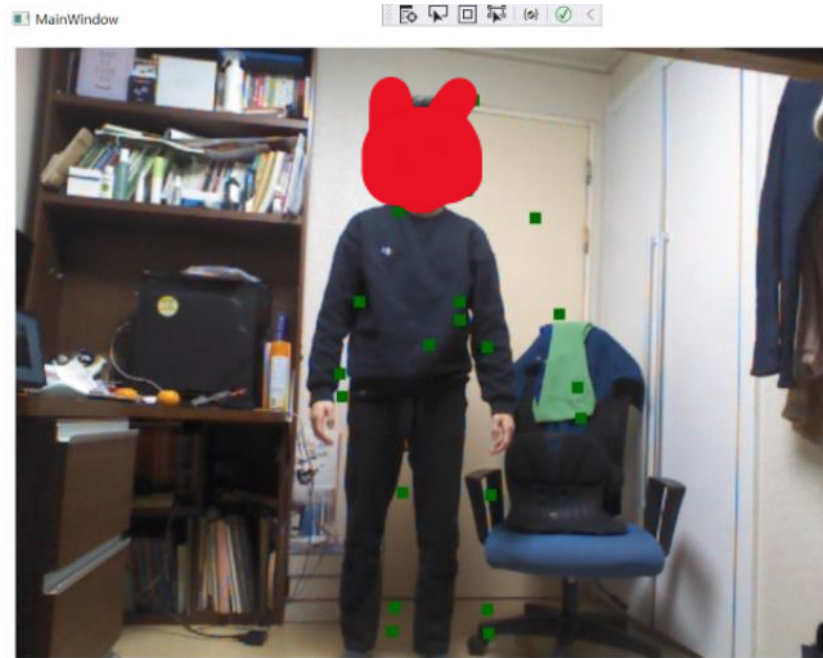
성공했을 경우



남은시간 : 4 **성공** 성공 횟수 : 1

캠킹보기

실패했을 경우



남은시간 : 6 **실패** 성공 횟수 : 0

캠킹보기

소스 코드

```
void nui_AllFramesReady(object sender, AllFramesReadyEventArgs e)
{
    SkeletonFrame sf = e.OpenSkeletonFrame();
    if (sf == null) return;
    Skeleton[] skeletonData = new Skeleton[sf.SkeletonArrayLength];
    sf.CopySkeletonDataTo(skeletonData);
    using (DepthImageFrame depthImageFrame = e.OpenDepthImageFrame())
    {
        if (depthImageFrame != null)
        {
            foreach (Skeleton sd in skeletonData)
            {
                if (sd.TrackingState == SkeletonTrackingState.Tracked)
                {
                    int temp;
                    int nMax = 20;
                    Joint[] joints = new Joint[nMax];
                    for (int j = 0; j < nMax; j++)
                    {
                        joints[j] = sd.Joints[(JointType)j];
                    }

                    Point[] points = new Point[nMax];
                    for (int j = 0; j < nMax; j++)
                    {
                        DepthImagePoint depthPoint;
                        depthPoint = depthImageFrame.MapFromSkeletonPoint(joints[j].Position);
                        points[j] = new Point((int)(image1.Width * depthPoint.X / depthImageFrame.Width),
                                                (int)(image1.Height * depthPoint.Y / depthImageFrame.Height));
                    }

                    for (int j = 0; j < nMax; j++)
                    {
                        m_rect[j].Visibility = Visibility.Visible;
                        Canvas.SetTop(m_rect[j], points[j].Y - (m_rect[j].Height / 2));
                        Canvas.SetLeft(m_rect[j], points[j].X - (m_rect[j].Width / 2));
                    }
                }
            }
        }
    }
}
```

```

if (check2)
{
    ThisMoment = DateTime.Now;
    duration = new TimeSpan(0, 0, 0, 0, 59000);
    AfterWards = ThisMoment.Add(duration);
    check2 = false;
}
ThisMoment = DateTime.Now;
time = (AfterWards - ThisMoment).Seconds;
if (time == 0) chk++;
if (chk==1)
{
    chk++;
    System.Windows.MessageBox.Show("게임이 종료되었습니다.");
    newRank newrecpage = new newRank(count, id);
    this.Close();
}
retime.Text = "남은시간 : " + (AfterWards - ThisMoment).Seconds.ToString();

```

```

if (!check)
{
    if (check3)
    {
        color.Text = "";
        tempThisMoment = DateTime.Now;
        duration = new TimeSpan(0, 0, 0, 0, 3000);
        tempAfterWards = tempThisMoment.Add(duration);
        check3 = false;
    }

    //System.Windows.Forms.Application.DoEvents();
    ThisMoment = DateTime.Now;

    time = (tempAfterWards - ThisMoment).Seconds;
    if (time <= 0) check = true;
}
else
{
    result.Text = ""; //성공실패 결과
    time = move(points);
    if (time >= 0)
    {
        if (time > 0)
        {
            result.Text = "성공";
            count++;
        }
        else
        {
            result.Text = "실패";
            if (count > 0) count--;
        }
        countb.Text = "성공 횟수 : " + count.ToString();
        check = false;
        check3 = true;
        check4 = true;
    }
}

```

```

int move(Point[] p)
{
    if (check4)
    {
        Random rand = new Random();
        updown = rand.Next(0, 7);
    }

    if (updown == 0) //청기올려
    {
        color.Text = "청기올려";
        if (check4) ss.SpeakAsync("청기올려");
        if (p[11].Y < p[2].Y && p[7].Y > p[2].Y)
        {
            return 1;
        }
        else if (p[7].Y > p[0].Y || p[11].Y > p[0].Y || p[7].Y < p[2].Y)
        {
            return 0;
        }
    }
    else if (updown == 1) //백기올려
    {
        color.Text = "백기올려";
        if (check4) ss.SpeakAsync("백기올려");
        if (p[11].Y > p[2].Y && p[7].Y < p[2].Y)
        {
            return 1;
        }
        else if (p[11].Y > p[0].Y || p[11].Y < p[2].Y || p[7].Y > p[0].Y)
        {
            return 0;
        }
    }
}

```

```

    else if (updown == 2) //청기내려
    {
        color.Text = "청기내려";
        if (check4) ss.SpeakAsync("청기내려");
        if (p[11].Y > p[0].Y && p[7].Y < p[0].Y)
        {
            return 1;
        }
        else if (p[11].Y < p[2].Y || p[7].Y < p[2].Y || p[7].Y > p[0].Y)
        {
            return 0;
        }
    }
    else if (updown == 3) //백기내려
    {
        color.Text = "백기내려";
        if (check4) ss.SpeakAsync("백기내려");
        if (p[11].Y < p[0].Y && p[7].Y > p[0].Y)
        {
            return 1;
        }
    }
    else if (updown == 4) //청기백기올려
    {
        color.Text = "청기올리고 백기올려";
        if (check4) ss.SpeakAsync("청기올리고 백기올려");
        if (p[11].Y < p[2].Y && p[7].Y < p[2].Y)
        {
            return 1;
        }
    }
}

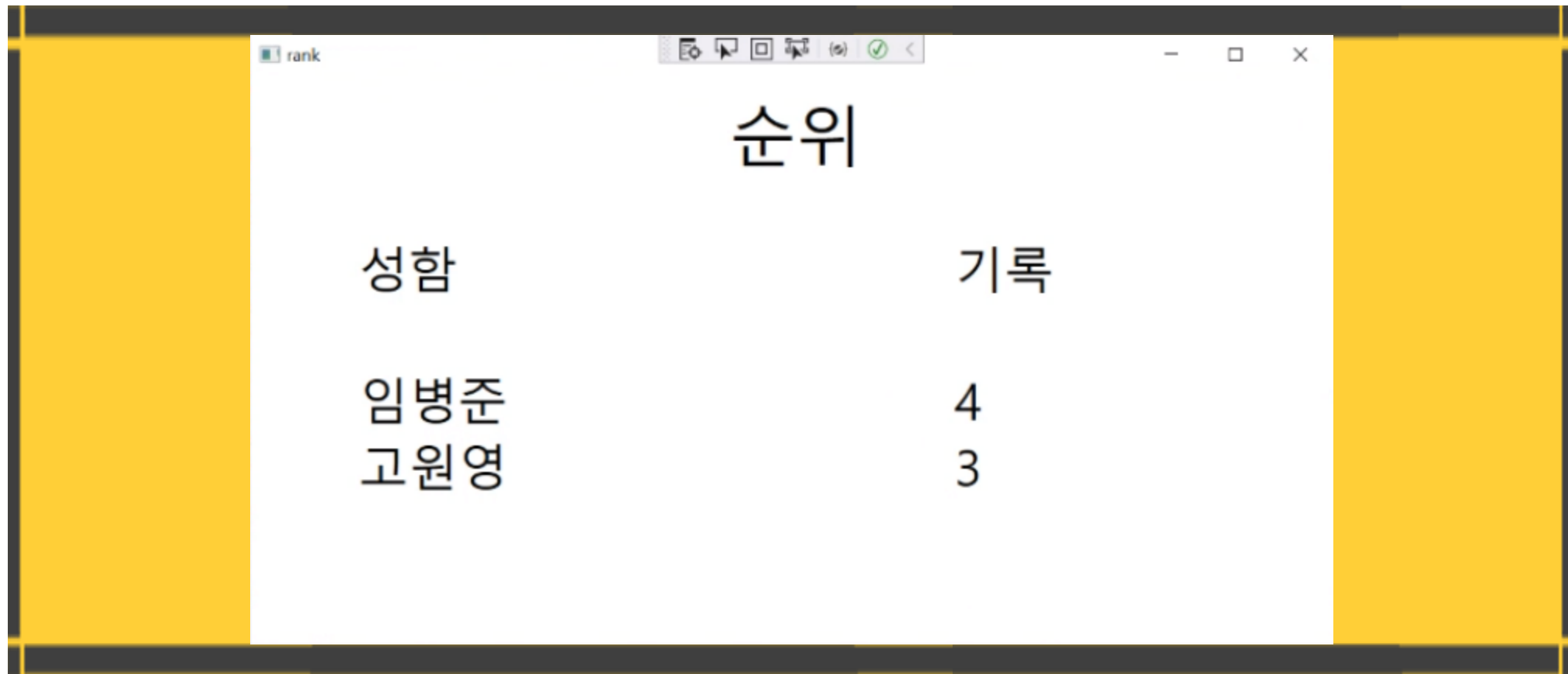
```

```

    else if (updown == 5) //청기백기내려
    {
        color.Text = "청기내리고 백기내려";
        if (check4) ss.SpeakAsync("청기내리고 백기내려");
        if (p[11].Y > p[0].Y && p[7].Y > p[0].Y)
        {
            return 1;
        }
    }
    else if (updown == 6) //청기올리고 백기내려
    {
        color.Text = "청기올리고 백기내려";
        if (check4) ss.SpeakAsync("청기올리고 백기내려");
        if (p[11].Y < p[2].Y && p[7].Y > p[0].Y)
        {
            return 1;
        }
    }
    else if (updown == 7) //청기내리고 백기올려
    {
        color.Text = "청기내리고 백기올려";
        if (check4) ss.SpeakAsync("청기내리고 백기올려");
        if (p[11].Y > p[0].Y && p[7].Y < p[2].Y)
        {
            return 1;
        }
    }
    check4 = false;
    return -1;
}

```


Chapter 04. 랭킹



The screenshot shows a web browser window with the title 'rank'. The page content is a table with two columns: '성함' (Name) and '기록' (Record). The table lists two names: '임병준' (Lim Byung-jun) with a record of 4, and '고원영' (Go Won-young) with a record of 3. The browser window has a standard toolbar with icons for back, forward, home, search, and other functions. The page is framed by yellow vertical bars on the left and right sides.

성함	기록
임병준	4
고원영	3

소스 코드

```
public partial class oldRank : Window
{
    참조 2개
    public oldRank()
    {
        InitializeComponent();
        MySqlDataReader ranklist = reSQL("select name, record from base where record > 0 order by record desc");
        string Totext = "성함ㅅㅅㅅㅅ기록ㅅㅅㅅㅅ";
        while (ranklist.Read())
        {
            Totext += ranklist.GetString(0);
            Totext += "ㅅㅅㅅㅅㅅㅅ";
            Totext += ranklist.GetInt32(1) + "ㅅㅅㅅㅅ";
        }
        rankbox.Text = Totext;
    }
    참조 1개
    public MySqlDataReader reSQL(string sql)
    {
        string mysqldb = "Server = 14.37.24.166; Port = 3306; Database = kinect; Uid = root; Pwd = rootpw";
        MySqlConnection conn = new MySqlConnection(mysqldb);
        MySqlCommand cmd = new MySqlCommand(sql, conn);
        conn.Open();
        MySqlDataReader table = cmd.ExecuteReader();

        return table;
    }

    conn.Close();
    conn.Dispose();
}

int32 rec = reSQL("select record from base where id = '" + id + "'");
if (count > rec)
{
    usingSQL("update base set record = " + count + " where id = '" + id + "'");
    System.Windows.MessageBox.Show("신기록 달성");
}
```

발표 들어주셔서 감사합니다!

Thank you!



GAME over!!