

멤버관리 프로젝트

2022-01-26

들 어 가 기 에
앞 서

이 번 프 로젝트 에 대 하 여...

C#을 활용한 프로젝트중 폼을 활용하여 C
LUD 기능을 추가한 프로젝트를 구상하다
가 고객 멤버십 관리 프로그램을 구상하게
되었습니다 간단하게 사용자 로그인 및 가
입, 이후 고객 추가,수정,제거,조회 기능 및
고객별 등급에 따른 분류와 프로그램 사용
에 따른 기록(로그) 남기기 및 로그 조회기
능을 추가했습니다.

목 차

01 로그인 및 회원가입

02 멤버 등록 및
수정

03 멤버 조회

04 멤버 삭제

05 로그 조회

06 마치며....

01.

로그인 및
회원가입.

로그인 및 회원가입

간단한 로그인창과
회원가입창 구현

로그인



아이디

kwon6772

비밀번호

사용자
로그인

사용자등록

클릭시 이동

사용자 추가

아이디

ID1234

비밀번호

비밀번호
확인

확인코드

사용자 추가

로그인 및 회원가입 Code

참조 2개

```
class Manager
{
    //DB 접속 정보
    const string ORADB = "Data Source=(DESCRIPTION=(ADDRESS_LIST= " +
        "(ADDRESS=(PROTOCOL=TCP)(HOST=KB-PC)(PORT=1521)))" +
        "(CONNECT_DATA=(SERVER=DEDICATED)(SERVICE_NAME=orcl))); " +
        "User Id=test;Password=1234;";

    public static OracleConnection OraConn = new OracleConnection(ORADB);

    public static List<Login_Info> login = new List<Login_Info>();

    const string table = "login";

    참조 2개
    static void ConnectDB()
    {
        DB 접속 함수
        try
        {
            OraConn.Open();
        }
        catch (Exception ex)
        {
            System.Windows.Forms.MessageBox.Show(ex.Message);
            System.Windows.Forms.MessageBox.Show(ex.StackTrace);
        }
    }
}
```

기본 DB 접속 코드.

```
//접속
ConnectDB();
try
{
    //sql문
    string sql = $"select * from {table}";
    //새 접속
    OracleDataAdapter oda = new OracleDataAdapter();
    //오라클 생성
    oda.SelectCommand = new OracleCommand();
    //오라클 접속 참조
    oda.SelectCommand.Connection = OraConn;
    //sql 삽입
    oda.SelectCommand.CommandText = sql;

    //데이터셋 선언
    DataSet ds = new DataSet();
    //table 내 데이터를 데이터셋에 적용
    oda.Fill(ds, table);
}
```

SQL 문 및 데이터셋
선언.

```
login.Clear();
int i = 1;
//존재하는 아이디 값 확인
foreach (DataRow item in ds.Tables[0].Rows)
{
    Login_Info login = new Login_Info();
    if (ID == item["ID"].ToString())
    {
        //생성 아이디와 기존 아이디 중복 여부 체크
        System.Windows.Forms.MessageBox.Show("이미 있는 아이디 입니다.");
        OraConn.Close();
        i = 0;
        break;
    }
    if (C != "Free")
    {
        //등록 승인 코드 일치 여부 확인
        System.Windows.Forms.MessageBox.Show("승인 코드가 틀립니다.");
        OraConn.Close();
        i = 0;
        break;
    }
}
if(i == 1)
{
    //등록 구문
    sql = $"insert into {table} values ('{ID}', '{PW}')";
    System.Windows.Forms.MessageBox.Show("등록 되었습니다.");
    OracleCommand cmd = new OracleCommand();
    cmd.Connection = OraConn;
    cmd.CommandText = sql;
    cmd.ExecuteNonQuery();
}
OraConn.Close();
}
catch (Exception ex)
{
    OraConn.Close();
    System.Windows.Forms.MessageBox.Show(ex.Message);
    System.Windows.Forms.MessageBox.Show(ex.StackTrace);
}
OraConn.Close();
//접속해제
```

회원가입 시 입력된
정보에 대한 체크.

로그인 및 회원가입 Code

```
//sql문
string sql = $"select * from {table}";
//새 접속선언
OracleDataAdapter oda = new OracleDataAdapter();
//oracle 관련문 선언
oda.SelectCommand = new OracleCommand();
//oracle 접속 선언
oda.SelectCommand.Connection = OraConn;
//sql문 사용
oda.SelectCommand.CommandText = sql;

//데이터셋 선언
DataSet ds = new DataSet();
//데이터셋을 접속한 DB의 테이블 내용 적용
oda.Fill(ds, table);

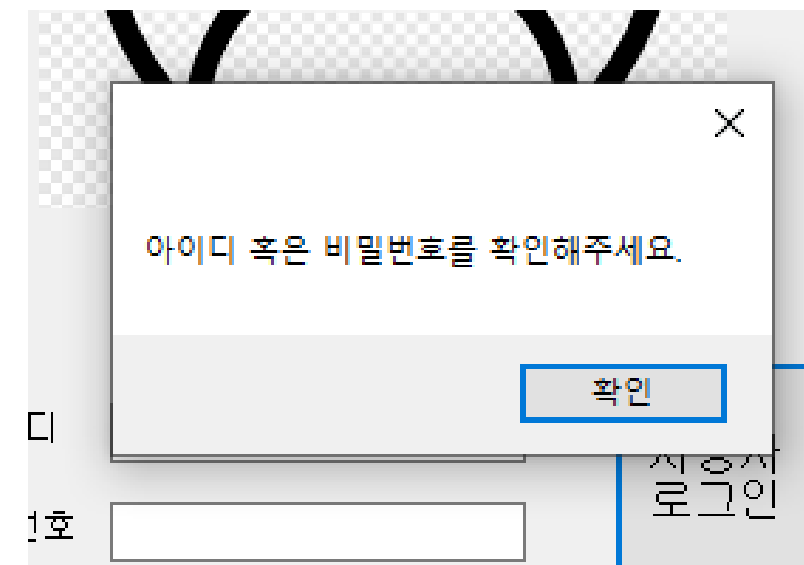
login.Clear();
// 확인용 i 선언
int i = 0;
foreach (DataRow item in ds.Tables[0].Rows)
{
    Login_Info login = new Login_Info();
    if (ID == item["ID"].ToString() && PW == item["PW"].ToString())
    {
        //정보가 일치시 i 를 1로 변경
        i = 1;
        //멤버관리창 선언

        OraConn.Close();
        return true;
    }
}
```

SQL,데이터셋 선언,테이블의 정보를 비교

```
if (i == 0)
{
    //모두 비교하여도 i가 0인 경우 안내메세지 도출
    System.Windows.Forms.MessageBox.Show("아이디 혹은 비밀번호를 확인해주세요.");
}
OraConn.Close();
return false;
}
catch (Exception)
{
    OraConn.Close();
    throw;
}
```

비교해도 조회되지 않을 경우 메세지창 출력



(대랜~)오류 메세지.

02.

멤버
링.

멤버 등록

회원 추가란에 이름, 주민번호, 전화번호를 입력
회원 등급란에 체크박스로 멤버십 등급을 정한 후
회원 추가 버튼을 누르면 멤버 리스트에 올라간걸
확인 할수 있다.

또한 멤버 등급에 따라 해당 데이터 열에 배경색이
흰색,노란색,하늘색 등으로 바뀌는걸 볼수있다.

멤버 추가 방식

회원 추가

이름

김회원

주민번호

960522-2682527

전화번호

010-1234-5678

회원 등급

☐ VVIP

☐ VIP

☒ 일반

회원 추가

회원 수정

멤버 추가 결과

추가 결과 →

	Name	Jumin	PhoneNum	Grade	AddTime
▶	2	2	2	VIP	2022-01-25 오후 12:12
	권용규	990223-1682729	010-5545-6542	일반	2022-01-24 오후 12:23
	김회원	960522-2682527	010-1234-5678	일반	2022-01-25 오후 12:19
	블리자드	850623-1682554	010-4458-1222	VVIP	2022-01-25 오후 12:10

멤버 등록 Code.

```
private void Addbtn_Click(object sender, System.EventArgs e)
{
    string grade = "";
    try
    {
        if(NameBox.Text == "")
        {
            //이름 확인
            MessageBox.Show("고객의 이름을 입력해 주세요.");
            return;
        }
        if(JuminBox.Text == "")
        {
            //주민번호 확인
            MessageBox.Show("고객의 주민번호를 입력해 주세요.");
            return;
        }
        if (PhoneBox.Text == "")
        {
            //휴대전화 미소지자를 위한 대우
            PhoneBox.Text = "휴대폰 미소유";
        }
    }
}
```

입력된 정보의 이상 체크

```
else
{
    //라디오 체크의 대한 분류
    if (VVIP.Checked == true)
    {
        grade = "VVIP";
    }
    if (VIP.Checked == true)
    {
        grade = "VIP";
    }
    if (None.Checked == true)
    {
        grade = "일반";
    }
    //함수로 쿼리문 작성
    Manager.executeQuery("insert", NameBox.Text, JuminBox.Text,
        PhoneBox.Text, grade);
    Manager.selectQuery();
    //로그에 표시될 내용
    Log_Info($"{NameBox.Text} 님 등록(등급 : {grade})");
    //안내메세지창 띄우기
    MessageBox.Show($"{NameBox.Text} 님 등록 완료.");
    //등록후 텍스트 박스초기화.
    NameBox.Text = "";
    JuminBox.Text = "";
    PhoneBox.Text = "";
    //그리드뷰 새로고침
    refresh();
}
```

이상이 없을시 등급 체크 후 추가

그리드뷰 새로고침

참조 4개

```
void refresh()
{
    MemberView.DataSource = null;
    try
    {
        if(Manager.members.Count > 0)
        {
            MemberView.DataSource = Manager.members;
        }
    }
    catch (System.Exception ex)
    {
        MessageBox.Show(ex.Message);
        MessageBox.Show(ex.StackTrace);
    }
}
```

멤버 등록 Code.

```
참조 3개
public static void executeQuery(string menu, string name, string jumin=null, string phonenum=null, string grade=null)
{
    //DB접속
    ConnectDB();
    //스트링형 쿼리문 공백으로 선언
    string query = "";

    try
    {
        OracleCommand cmd = new OracleCommand();
        cmd.Connection = OraConn;
        //입력된 정보를 토대로 Query 함수에 정보 전달
        query = Query(menu, name, jumin, phonenum, grade);
        //작성
        cmd.CommandText = query;
        //실행
        cmd.ExecuteNonQuery();
    }
    catch (Exception ex)
    {
        OraConn.Close();
        System.Windows.Forms.MessageBox.Show(ex.Message);
        System.Windows.Forms.MessageBox.Show(ex.StackTrace);
        throw new Exception(query + " - " + ex.Message + "오류위치" + Environment.NewLine + ex.StackTrace);
    }

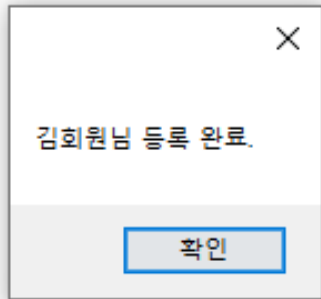
    OraConn.Close();
}
```

우선 executequery 함수로 와서 위에 입력된 정보를 Query 함수로 보낸다.

```
참조 1개
static string Query(string menu, string name,string jumin, string phonenum, string grade)
{
    string query = "";
    switch (menu)
    {
        case "update":
            query = $"update {Table} set name = '{name}', Phonenum = '{phonenum}'," +
            $"grade = '{grade}', Datatime = sysdate where jumin = '{jumin}'";
            break;
        case "insert":
            query = $"insert into {Table} values ('{name}', '{jumin}', '{phonenum}', '{grade}', sysdate)";
            break;
        case "delete":
            query = $"delete from {Table} where name = '{name}' and jumin like '{jumin}'";
            break;
        default:
            break;
    }

    return query;
}
```

함수로 전달된 내용에 따라 switch문을 통해 나눈 케이스에 맞게끔 sql 문이 작성되게끔 한다



추가가 되었을 경우
화면에 보이게 됨

멤버 수정

그리드뷰에 저장된 정보를 클릭시 하단 텍스트 박스에 자동으로 해당 정보가 들어가도록 설계했다

이후 원하는 정보를 수정한뒤 수정버튼을 클릭시 다음과 같이 수정되서 적용되는걸 볼수있다.

Name	Jumin
2	2
권용규	990223-1682729
김회원	960522-2682527
블리자드	850623-1682554

수정 후
변화

Name	Jumin	F
2	2	2
권용규	990223-1682729	0
김회원	960522-2682527	0
블리자드	850623-1682554	0

추가

이름

김회원

주민번호

960522-2682527

전화번호

010-1234-5678

회원 등급

☐ VVIP

☐ VIP

☒ 일반

회원 추가

회원 수정

수정 전
정보

수정 후
정보

회원 추가

이름

김회원

주민번호

960522-2682527

전화번호

010-1234-5678

회원 등급

☐ VVIP

☒ VIP

☐ 일반

회원 추가

회원 수정

멤버 수정 Code.

```
private void Updatebtn_Click(object sender, System.EventArgs e)
{
    //회원 추가와 코드가 거의 유사하다
    string grade = "";
    try
    {
        if (NameBox.Text == "")
        {
            MessageBox.Show("고객의 이름을 입력해주세요.");
            return;
        }
        if (JuminBox.Text == "")
        {
            MessageBox.Show("고객의 전화번호를 입력해주세요.");
            return;
        }
        else
        {
            if (VVIP.Checked == true)
            {
                grade = "VVIP";
            }
            else if (VIP.Checked == true)
            {
                grade = "VIP";
            }
            else if (None.Checked == true)
            {
                grade = "일반";
            }
            Manager.executeQuery("update", NameBox.Text, JuminBox.Text,
            PhoneBox.Text, grade);
            Manager.selectQuery();
            Log_Info($"{NameBox.Text} 님 정보 수정 완료");
            NameBox.Text = "";
            JuminBox.Text = "";
            PhoneBox.Text = "";
            refresh();
        }
    }
    catch (System.Exception ex)
```

함수 내용은 거의 동일하다.

이부분이 작동된다.

이부분만 다름.

```
참조 1개
static string Query(string menu, string name, string jumin, string phonenum, string grade)
{
    string query = "";
    switch (menu)
    {
        case "update":
            query = $"update {Table} set name = '{name}', Phonenum = '{phonenum}', " +
            $"grade = '{grade}', Datetime = sysdate where jumin = '{jumin}'";
            break;
        case "insert":
            query = $"insert into {Table} values ('{name}', '{jumin}', '{phonenum}', '{grade}', sysdate)";
            break;
        case "delete":
            query = $"delete from {Table} where name = '{name}' and jumin like '{jumin}'";
            break;
        default:
            break;
    }
    return query;
}
```

03.

멤버
조회.

멤버 조회

그리드뷰에 저장된 정보중 자신이 찾는 정보 중
전화번호를 통하여 조회할수있도록 설계하였다.
전화번호는 DB에서 프라이머리 키로 지정했기에
전화번호는 겹칠수가 없다.

```
public static void find(string f)
{
    ConnectDB();

    string sql = $"select * from {Table} where Phonenum = '{f}'";
    int i = 0;
    OracleDataAdapter oda = new OracleDataAdapter();
    oda.SelectCommand = new OracleCommand();
    oda.SelectCommand.Connection = OraConn;
    oda.SelectCommand.CommandText = sql;

    DataSet ds = new DataSet();
    oda.Fill(ds, Table);
    foreach (DataRow item in ds.Tables[0].Rows)
    {
        if (f.Equals(item["Phonenum"]))
        {
            System.Windows.Forms.MessageBox.Show($"성함 : {item["Name"]}주민번호 : {item["jumin"]} +
            $"전화번호 : {item["Phonenum"]}멤버쉽 등급 : {item["grade"]}");
            i = 1;
            break;
        }
    }
    if (i == 0)
    {
        System.Windows.Forms.MessageBox.Show("등록된 정보가 없습니다.");
    }
    OraConn.Close();
}
```

멤버 조회 코드

조회 정보 입력

조회 결과

조회 결과(실패)

04.

멤버
삭제.

멤버 삭제

그리드뷰에 저장된 정보중 자신이 삭제하고자 하는
정보에 대하여 이름,주민번호 로 찾아내어 삭제하
게끔 설계했다.

삭제 후
변화

	Name	Jumin	PhoneNum	Grade	AddTime
	2	2	2	VIP	2022-01-25 오후 12:12
	권용규	990223-1682729	010-5545-6542	일반	2022-01-24 오후 12:23
	블리자드	850623-1682554	010-4458-1222	VVIP	2022-01-25 오후 12:10

회원 제거

이름

김회원

주민번호

960522-2682527

제거

×

김회원고객님 해지 처리 완료

확인

삭제할 정보 입력

삭제 시 안내메세지

멤버 삭제 Code.

```
참조 1개
private void Delbtn_Click(object sender, System.EventArgs e)
{
    try
    {
        if(Del_NameBox.Text == "")
        {
            MessageBox.Show("이름을 입력해 주세요.");
            return;
        }
        if (Del_JuminBox.Text == "")
        {
            MessageBox.Show("주민번호를 입력해 주세요.");
            return;
        }
        else
        {
            Manager.executeQuery("delete", Del_NameBox.Text, Del_JuminBox.Text);
            Manager.selectQuery();
            refresh();
            Log_Info($"{Del_NameBox.Text} 고객님의 해지 처리 완료");
            MessageBox.Show($"{Del_NameBox.Text} 고객님의 해지 처리 완료");
            Del_JuminBox.Text = "";
            Del_NameBox.Text = "";
        }
    }
    catch (System.Exception)
    {
        throw;
    }
}
```

입력 정보의 부재 검사

이부분이 작동된다.

```
참조 1개
static string Query(string menu, string name, string jumin, string phonenum, string grade)
{
    string query = "";
    switch (menu)
    {
        case "update":
            query = $"update {Table} set name = '{name}', Phonenum = '{phonenum}', " +
                $"grade = '{grade}', Datetime = sysdate where jumin = '{jumin}'";
            break;
        case "insert":
            query = $"insert into {Table} values ('{name}', '{jumin}', '{phonenum}', '{grade}', sysdate)";
            break;
        case "delete":
            query = $"delete from {Table} where name = '{name}' and jumin like '{jumin}'";
            break;
        default:
            break;
    }
    return query;
}
```

05.

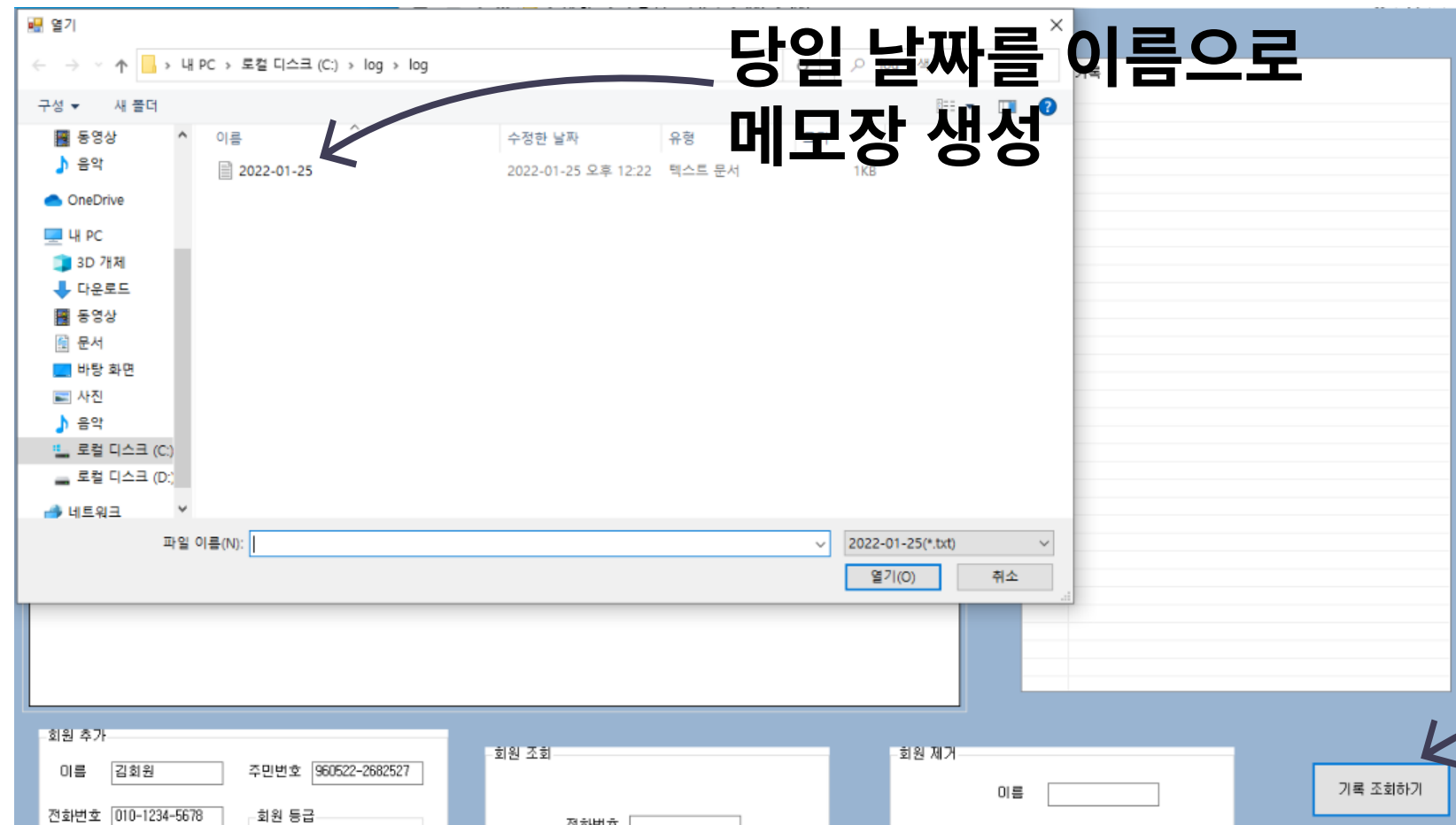
로그
조회.

로그 조회

프로그램의 실행 및 종료, 멤버의 추가, 제거, 수정 등의
 대한 내용이 시간대 별로 담겨있는 기록을 조회
 할수 있는 기능이다.

메모장의 입력된 활동
내역을 내용을 보여준
다.

No.	기록
1	2022-01-25 12:16:46 : 프로그램 시작
2	2022-01-25 12:19:02 : 김회원 님 등록(등급 : 일반)
3	2022-01-25 12:20:24 : 김회원 님 정보 수정 완료
4	2022-01-25 12:22:32 : 김회원 고객님의 해지 처리 완료



버튼 클릭시
선택화면 이
나온다.

로그 조회 Code.

```
참조 1개
public void Log_Start()
{
    //로그 파일 경로 지정
    string local = @"C:\log";

    //파일 이름 지정
    string title = DateTime.Now.ToString("yyyy-MM-dd");

    OpenFileDialog ofd = new OpenFileDialog();
    //메모장 형식으로 생성
    ofd.Filter = title + "(*.txt) | *.txt";
    //경로
    ofd.InitialDirectory = local + "\log";
    //파일 유무 검사
    if(ofd.ShowDialog() == DialogResult.OK)
    {
        listView1.Items.Clear();

        string filename = "";
        filename = ofd.FileName;
        //없을시 메시지 호출
        if (!System.IO.File.Exists(filename))
        {
            MessageBox.Show("해당 파일이 없습니다");
            return;
        }
        else
        {
            //선택된경우 내용 표시.
            string[] value = System.IO.File.ReadAllLines(filename);
            for(int i = 0; i < value.Length; i++)
            {
                ListViewItem lvi = new ListViewItem();
                lvi.Text = (i + 1).ToString();
                lvi.SubItems.Add(value[i]);

                listView1.Items.Add(lvi);
            }
        }
    }
}
```

로그 기록을 리스트
에 표시 및 파일 생성

```
public bool Log_Info(string M)
{
    try
    {
        //문자열 2개 선언
        string checkfolder = "";
        string filename = "";

        //경로 선언
        string local = @"C:\log";

        checkfolder = local + "\log";

        //파일이 존재하지 않을경우 생성
        if (!System.IO.Directory.Exists(checkfolder))
        {
            System.IO.Directory.CreateDirectory(checkfolder);
        }
        //파일 이름 지정
        filename = checkfolder + "\log" + DateTime.Now.ToString("yyyy-MM-dd") + ".txt";
        //파일에 내용 작성
        System.IO.StreamWriter FileWriter = new System.IO.StreamWriter(filename, true);
        FileWriter.Write(DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss") + " : " + M + "\r\n");

        FileWriter.Flush();
        //작성 종료
        FileWriter.Close();

    }
    catch (System.Exception)
    {
        return false;
    }
    return true;
}
```

로그 기록 파일에 내
용 저장 하는 함수

06. THANK YOU

질문 있으신 분은 손을 들고 질문해 주세요.