

# REPORT



---

과 목 명 : 데이터베이스 프로그래밍

---

담 당 : 석 상 기 교수님

---

제 출 일 : 2019년 12월 06일

---

소 속 : 컴퓨터공학과

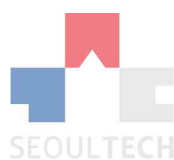
---

학 번 : 15109343

---

성 명 : 오재곤

---



서울과학기술대학교

Seoul National University of Science & Technology

## 1-1. PROC : 호기심 수 입력받아 그 수 보다 큰 회원 닉네임 출력하기.

```
#include <stdio.h>
#include <string.h>
#include <sqlca.h>

EXEC SQL INCLUDE sqlca.h;
EXEC SQL BEGIN DECLARE SECTION;
char connstr[30];
int cu;
char name[20];
EXEC SQL END DECLARE SECTION;

void SqlConnect()
{
    strcpy_s(connstr, 30, "csdb115/csdb115@gon");
    EXEC SQL CONNECT :connstr;

    if (sqlca.sqlcode != 0 && sqlca.sqlcode != -1405)
    {
        printf("\n connect error message = %s", sqlca.sqlerrm.sqlerrmc);
    } else
        printf("\n ***** DB CONNECT SUCCESS *****");
}

void main()
{
    SqlConnect();
    int curious=0;

    printf("\n호기심 수를 입력하세요:\n");
    scanf("%d",&curious);

    EXEC SQL DECLARE user_cursor CURSOR FOR SELECT O_number,O_m_name
FROM outsider;
    EXEC SQL OPEN user_cursor;
    EXEC SQL FETCH user_cursor INTO :cu, :name;

    if(sqlca.sqlcode != 0 && sqlca.sqlcode != -1405){
        EXEC SQL CLOSE user_cursor;
        return;
    }
}
```

```

while(1){
    if(cu >= curious){
        printf("호기심 수: %d\n",cu);
        printf("닉네임: %s\n",name);
    }
    EXEC SQL FETCH user_cursor INTO :cu, name;

    if(sqlca.sqlcode !=0 && sqlca.sqlcode != -1405) {
        EXEC SQL CLOSE user_cursor;
        break;
    }
}
EXEC SQL CLOSE user_cursor;
}

```

```

***** DB CONNECT SUCCESS *****
호기심 수를 입력하세요:
35
호기심 수: 35
닉네임: g6
호기심 수: 36
닉네임: g7
호기심 수: 37
닉네임: g8
호기심 수: 38
닉네임: g9
호기심 수: 39
닉네임: g10

```

1-2 모든 회원이 닉네임과 사는 곳 조회

```
#include <stdio.h>
#include <string.h>
#include <sqlca.h>
```

```
EXEC SQL BEGIN DECLARE SECTION;
char connstr[30];
int no;
char name[20];
EXEC SQL END DECLARE SECTION;
```

```
void SqlConnect()
{
    strcpy_s(connstr, 30, "csdb115/csdb115@gon");
    EXEC SQL CONNECT :connstr;
    if (sqlca.sqlcode != 0 && sqlca.sqlcode != -1405)
    {
        printf("\n connect error message = %s", sqlca.sqlerrm.sqlerrmc);
    } else
        printf("\n ***** DB CONNECT SUCCESS *****");
}
```

```
void main()
{
    EXEC SQL BEGIN DECLARE SECTION ;
    char m_nickname[10];
    char m_place[20];
    EXEC SQL END DECLARE SECTION ;

    SqlConnect();

    EXEC SQL DECLARE I_cursor CURSOR FOR SELECT m_nickname,
    m_place FROM member;
    EXEC SQL OPEN I_cursor;
    EXEC SQL FETCH I_cursor INTO : m_nickname, m_place;

    if(sqlca.sqlcode != 0 && sqlca.sqlcode != -1405) {
        EXEC SQL CLOSE I_cursor;
        return ;
    }

    while(1){
        printf("\n-----\n");
```

```

printf("회원 닉네임: %s\n", m_nickname);
printf("사는곳: %s\n", m_place);
printf("\n-----\n");

EXEC SQL FETCH I_cursor INTO : m_nickname, m_place;

if(sqlca.sqlcode != 0 && sqlca.sqlcode != -1405) {
    EXEC SQL CLOSE I_cursor;
    break;
}
}
}

```

2-1. JDBC : 남자회원의 이름을 입력하면 매칭된 여자회원의 이름을 출력하시오.

\*\*\*\*\* DB CONNECT SUCCESS \*\*\*\*\*

회원 닉네임: g1  
사는곳: konkuk1

회원 닉네임: g2  
사는곳: konkuk2

회원 닉네임: g3  
사는곳: konkuk2

회원 닉네임: g4  
사는곳: konkuk4

회원 닉네임: g5  
사는곳: konkuk5

회원 닉네임: g6  
사는곳: donkuk1

회원 닉네임: g7  
사는곳: donkuk2

회원 닉네임: g8  
사는곳: donkuk3

회원 닉네임: g9  
사는곳: donkuk4

회원 닉네임: g10  
사는곳: donkuk5

## 2-1. JDBC : 남자회원의 이름을 입력하면 매칭된 여자회원의 이름을 출력하시오.

```
package db1;
import java.sql.*;
import java.util.Scanner;

public class ex01 {

    public static void main(String[] args) {
        Connection conn;
        PreparedStatement pstmt;
        ResultSet rs;

        try {
            Scanner scan = new Scanner(System.in);
            //드라이버를 로딩한다.
            Class.forName("oracle.jdbc.driver.OracleDriver");
            //연결을 설정한다.
            conn = DriverManager.getConnection(
                "jdbc:oracle:thin:@117.17.198.43:1531:ORCL",
                "CSDB115", "CSDB115");
            pstmt = conn.prepareStatement("SELECT I_SANE, I_M_NAME
from insider where I_SANE=?");

            System.out.println("남성의 닉네임을 입력하세요.");
            String input = scan.nextLine();
            System.out.println("");

            //인자값을 설정한다.
            pstmt.setString(1, input);
            //sql문을 실행한다.
            rs= pstmt.executeQuery();
            while(rs.next())
            {
                //데이터를 받아온다.
                String I_name = rs.getString("I_SANE");
                String Y_name = rs.getString("I_M_NAME");

                //결과를 출력한다.
                System.out.println("남자 닉네임 : "+ I_name);
                System.out.println("여자 닉네임 : "+ Y_name);
            }

            //Result Set을 닫는다.
            rs.close();
            //Statement를 닫는다.
            pstmt.close();
            //연결을닫는다.
            conn.close();
        }catch(Exception e) {e.printStackTrace();}
    }
}
```

```
C:\Users\tjfbw\workspace\db1\src>javac ex01.java
```

```
C:\Users\tjfbw\workspace\db1\src>cd ..
```

```
C:\Users\tjfbw\workspace\db1\src>java db1.ex01
```

남성의 닉네임을 입력하세요:

g1

남자 닉네임 : g1

여자 닉네임 : g6



2-2 JDBC: 회원의 취미가 'sing'인 사람을 'soccer'로 변경.

```
package db2;
import java.sql.*;

public class db2 {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Connection conn;
        PreparedStatement ps;
        try {
            Class.forName("oracle.jdbc.driver.OracleDriver");
            conn = DriverManager.getConnection(
                "jdbc:oracle:thin:@117.17.198.43:1531:ORCL",
                "CSDB115", "CSDB115");
            ps = conn.prepareStatement("update member set m_hobby=?
where m_hobby=?");

            ps.setString(1, "soccer");
            ps.setString(2, "sing");

            ps.executeUpdate();

            System.out.println("업데이트가 완료되었습니다.");

            ps.close();
            conn.close();
        } catch (Exception e) {e.printStackTrace();}

    }
}
```

\*변경 전

```
SQL> select m_nickname, m_hobby from member where m_hobby='sing';
```

M_NICKNAME	M_HOBBY
g10	sing

\*변경 후

```
C:\Users\jfbw\workspace\db2\src>javac db2.java
```

```
C:\Users\jfbw\workspace\db2\src>cd ..
```

```
C:\Users\jfbw\workspace\db2\src>java db2.db2  
업데이트가 완료되었습니다.
```

```
SQL> select m_nickname, m_hobby from member where m_hobby='sing';
```

선택된 레코드가 없습니다.

```
SQL> select m_nickname, m_hobby from member where m_hobby='soccer';
```

M_NICKNAME	M_HOBBY
g10	soccer