

게시판 프로그램 만들기

chapter

4

본격적으로 게시판을 만들어 보겠습니다. 이 책에서 만들 게시판은 계층형 게시판으로, 게시물에 대해 답변할 수 있는 게시판입니다. 또한 이 게시판에 파일 업로드 기능을 추가하면 자료실로 이용할 수도 있습니다. 따라서 이 책에서 만드는 최종적인 결과물은 자료실이 될 것입니다.

01

사전 준비 사항

ASP 게시판 프로그래밍에 들어가기 전에 준비 작업이 필요합니다.

ASP를 이용하여 데이터베이스 작업을 하려면 일반적인 클라이언트 컴퓨터는 할 수 없습니다. 여기서 클라이언트 컴퓨터는 업무용으로 사용하면서 인터넷에 접속할 수 있는 환경만 갖춘 컴퓨터를 뜻합니다(예를 들면 윈도우 98이 기본 사양으로 설치된 컴퓨터).

ASP로 코딩한 내용을 웹 서버에 업로드하지 않은 상태에서, 코딩된 내용을 컴퓨터에서 바로 확인해 보려면, 최소한 웹서버 프로그램이 설치되어 있어야 합니다. 앞에서도 몇 번 얘기한 것처럼 ASP가 서버에서 실행되는 웹애플리케이션 제작 언어이기 때문입니다.

또한 게시판 프로그램은 모든 내용을 데이터베이스에 저장하고, 그 데이터베이스에서 불러와 사용합니다. 따라서 데이터베이스를 다루는 것은 필수불가결한 요소라 할 수 있습니다. 웹프로그램에서 데이터베이스를 이용하려면 두 가지를 서로 연결시켜주어야 하는데, 이러한 역할을 하는 것이 바로 ODBC입니다. ODBC를 통해 DSN이라는 연결 객체를 만들어 사용하게 됩니다.

ASP를 코딩하기 전에 점검해야 할 사항을 알아보겠습니다.

1.1 ASP 실행 환경

ASP로 웹애플리케이션을 제작하려면 ASP 실행(개발) 환경을 구성해줘야 합니다. 이 환경은 크게 둘로 구분할 수 있습니다. 즉, 개발자의 컴퓨터에 웹서버를 설치해 놓고 직접 웹프로그래밍을 할 것인지, 아니면 별도의 웹서버 컴퓨터를 사용해 개발할 것인지 결정해야 합니다.

이 두 가지 개발 방법은 개발자의 시스템 환경에 맞추면 됩니다. 예를 들어, 웹을 개발하는 프리랜서가 집에서 한 대의 컴퓨터로 웹프로그래밍을 한다면, 개발자의 컴퓨터에는 ASP를 실행할 수 있는 웹서버가 설치되어 있어야 합니다. 그래야 제작된 ASP 파일을 테스트할 수 있습니다.

다른 예로, 웹서버를 운영중인 회사에 근무하는 개발자이거나, ASP가 실행될 수 있는 호스팅, 혹은 계정을 가지고 있는 개발자라면, 자신의 컴퓨터에서 프로그래밍한 ASP 파일을 웹서버에 올려 테스트하고 개발할 수 있습니다.

어떤 개발 환경을 선택할 것인지는 독자의 상황에 맞추도록 합니다. 그러나 다음에 설명되는 것들을 위의 두 가지 환경과 관계없이 공통으로 필요한 사양합니다. 그래야 ASP를 실행해 볼 수 있습니다.

1) 운영체제

ASP를 실행하기 위한 운영체제(OS)로는 윈도우 95/98/ME/NT/2000/XP가 있지만, 서버급 운영체제인 윈도우 NT/2000/XP Server를 권장합니다. 서버급 운영체제에는 운영체제와 잘 맞는 웹서버가 포함되어 있기 때문입니다. 윈도우 95/98/ME에서 ASP를 실행하려면 별도의 웹서버를 설치해서 사용해야 합니다.

2) 웹서버

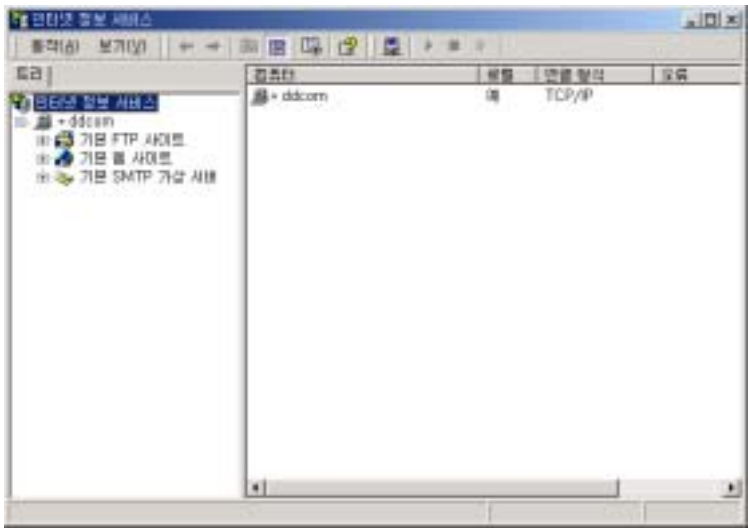
ASP 웹프로그래밍이 제대로 실행되려면 웹서버(Web Server)에서 ASP 언어를 지원해줘야 합니다. 따라서 다음과 같은 웹서버가 작동되고 있어야 합니다.

- ▶ 윈도우 95/98/ME: PWS(Personnal Web Server)
- ▶ 윈도우 NT/2000/XP: IIS(Internet Infomation Server)

윈도우 95/98/ME에서는 별도로 웹서버인 PWS를 설치해야 하며, 윈도우 NT/2000/XP에서는 시스템 설치시에 함께 설치하면 됩니다. 만일 윈도우 NT/2000/XP를 설치했는데, IIS가 보이지 않는다면, 추가로 설치하면 됩니다.

그림 4-1

IIS(Internet Information Server) 웹서버



3) 데이터베이스 연결(ODBC)

ASP 웹프로그래밍에서 가장 중요한 것은 데이터베이스 연결입니다. 윈도우 운영체제에서 데이터베이스 연결을 설정하는 것은 그다지 어려운 일은 아닙니다. 독자가 직접 ODBC(Open DataBase Connectivity)로 연결하여 사용할 수 있습니다. 다음 그림은 윈도우 2000에서 사용하는 ODBC 관리자입니다.

그림 4-2

윈도우 2000의 ODBC 관리자



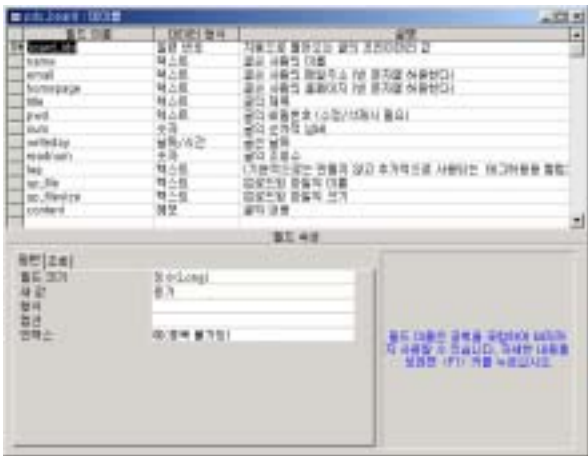
이 책에서 사용되는 데이터베이스는 마이크로소프트 액세스 파일이며, 다음 환경에서 테스트되었습니다.

OS	윈도우 2000
웹서버	IIS(Internet Infomation Server) 5.0
데이터베이스 연결	ODBC(MDAC 2.7)
데이터베이스	Microsoft Access DB
사용할 웹 프로그래밍 언어	ASP
출판할 웹서버	IIS(Internet Infomation Server) 5.0



mdb 파일

mdb 파일은 마이크로소프트 엑세스에서 만든 데이터베이스 파일로, 테이블 형식으로 데이터를 저장합니다. 테이블은 오른쪽 그림과 같은 디자인을 가지고 있으며, 각 필드마다 데이터를 입력할 수 있습니다.



4) 출판할 웹서버

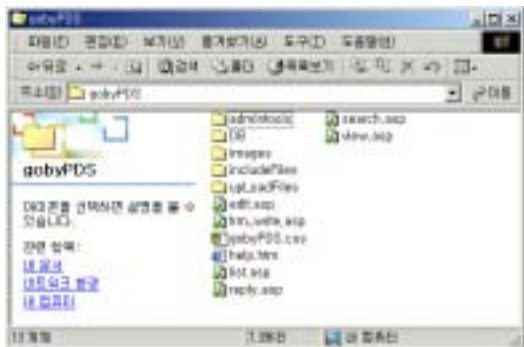
작성된 웹프로그램을 출판할 웹서버, 즉 실제로 사용자가 보게 될 웹프로그램이 들어있게 될 웹서버는 현재 사용중인 클라이언트 컴퓨터의 웹서버일 수도 있고, 별도의 장소에 있는 웹서버일 수도 있습니다.

현재 작업중인 사용자의 컴퓨터에서 ASP 프로그램이 제대로 수행된다고 하더라도 출판할 웹서버에서 웹프로그래밍 언어를 지원하지 않는다면 무용지물이 되고 맙니다. 따라서 이것을 반드시 확인해줘야 합니다. ASP를 사용했다면 윈도우 NT나 2000/XP 서버에 출판할 것을 권장합니다. 만일 리눅스나 유닉스 서버에 ASP 파일을 출판하려 한다면, 반드시 ASP 지원 여부를 확인해야 합니다.

게시판에 관련된 ASP 프로그램은 gobyPDS 폴더에 들어 있는데, gobyPDS 폴더를 열어 보면, 다음 그림처럼 게시판 레이아웃과 CSS 파일이 들어 있고, 다섯 개의 하위 폴더가 있습니다. images 폴더는 게시판에서 사용할 이미지들을 모아 놓은 폴더이고, upLoadFiles 폴더는 웹에 파일을 올릴 때 파일이 저장되는 폴더입니다. images 폴더와 upLoadFiles 폴더는 크게 신경쓰지 않아도 됩니다.

그림 4-4

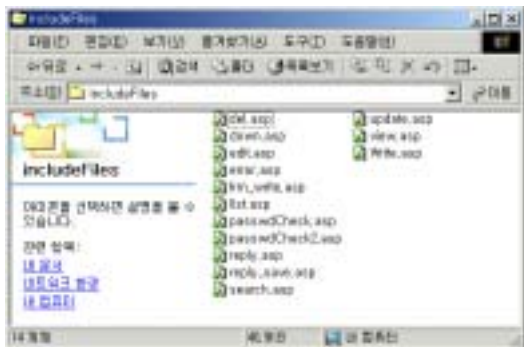
소스 파일의 구성 요소



includesFiles 폴더에는 실제 게시판 프로그램이 들어 있습니다. 여기 들어 있는 파일들은 gobyPDS에 있는 레이아웃 파일에 include 형식으로 삽입되어 사용됩니다. include 형식은 ASP 프로그래밍의 개념을 설명할 때 살펴 본 대로 객체지향 프로그래밍의 방법입니다.

그림 4-5

실제 프로그램이 들어 있는 includesFiles 폴더



DB 폴더는 게시판의 데이터가 저장되는 폴더로, 게시판의 데이터가 저장되는 goby PDSDB.mdb 파일이 들어 있습니다.

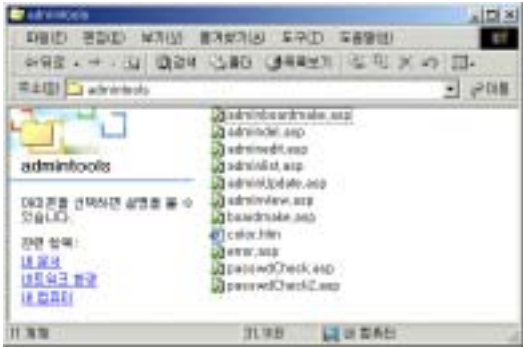
그림 4-6

데이터베이스 파일이 들어 있는 DB 폴더



그림 4-7

Admintools 폴더에는
게시판 관리자를 위한
프로그램이 들어 있다.



DB 연결 객체 만들기

gobyPDS라는 게시판 프로그램을 사용하기 위해 가장 먼저 해야 할 일은 ODBC를 설정하는 것입니다. ODBC를 설정하면 DSN이라는 데이터베이스 소스 네임이 생성되는데, 이 이름으로 ASP 파일과 데이터베이스가 연결되도록 연결 객체를 만듭니다.

데이터베이스 연결 객체를 만드는 방법은 두 가지가 있습니다. 하나는 시스템에 설정된 DSN을 이용하는 방법이고, 다른 하나는 사용자 정의 String으로 DSN 없이 데이터베이스와 바로 연결해 만드는 방법입니다.

첫 번째 방법은 사용자가 웹서버를 직접 다루고 있는 상황에서만 가능한데, ODBC 설정을 해줘야 하기 때문입니다. 즉, 호스팅이나 계정을 사용해서 ASP 웹애플리케이션을 만든다면 웹서버에 직접 액세스할 수 없기 때문에 DSN을 설정할 없습니다. 반면에 String으로 설정하면 웹서버를 직접 다룰 수 없더라도 연결 객체를 설정할 수 있습니다. 이 책에서는 두 가지 방법을 모두 사용해 보겠습니다.

ODBC 설정

ODBC는 데이터베이스에 연결하기 위한 이름 즉, DSN을 만드는 것이라고 생각하면 됩니다. DSN 없이 데이터베이스에 직접 액세스하려 한다면 이 과정은 생략해도 됩니다. 여기서는 책과 함께 제공된 소스에 들어 있는 데이터베이스 파일인 gobyPDSDB.mdb를 사용해 윈도우 2000 Professional/Server에 설정해 보겠습니다.

④ 데이터베이스 파일과 연결할 ODBC 드라이버를 추가합니다. [추가] 버튼을 클릭합니다.



⑤ ‘새 데이터 원본 만들기’ 대화상자가 나타납니다. 우리가 사용할 데이터베이스 원본 파일은 마이크로소프트 액세스 파일이므로, Microsoft Access Driver(*.mdb)를 선택하고 [마침] 버튼을 클릭합니다.



⑥ ‘ODBC Microsoft Access 설정’ 대화상자가 나면, ‘데이터 원본 이름’에 적당한 이름을 입력합니다. 이 데이터베이스 이름은 ASP 코드 내에서 연결 객체를 사용할 때 이용되며, 다른 데이터베이스와 구별하는데 사용됩니다. 그리고 ‘설명’에는 데이터베이스에 대한 간단한 설명을 입력합니다. 그런 다음, 데이터베이스 파일을 선택하기 위해 ‘데이터베이스’ 항목에서 [선택] 버튼을 클릭합니다.



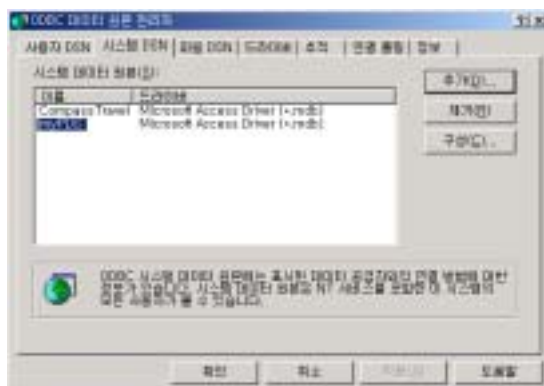
- ⑦ 데이터베이스 파일을 선택할 수 있는 대화상자가 나타나면, '웹사이트의 루트 \gobyPDS_v100(access97)\gobyPDS\DB' 폴더에 있는 gobyPDSDB.mdb 파일을 찾아서 선택하고 [확인] 버튼을 클릭합니다.



- ⑧ 다음 그림처럼 데이터베이스의 파일의 경로명과 함께 표시됩니다. ODBC 설정이 끝나면, [확인] 버튼을 클릭해 대화상자를 닫습니다.



- ⑨ '시스템 DSN' 탭에 myPDS라는 새로운 이름의 시스템 데이터 원본이 추가된 것을 알 수 있습니다. [확인] 버튼을 클릭해 대화상자를 닫습니다.

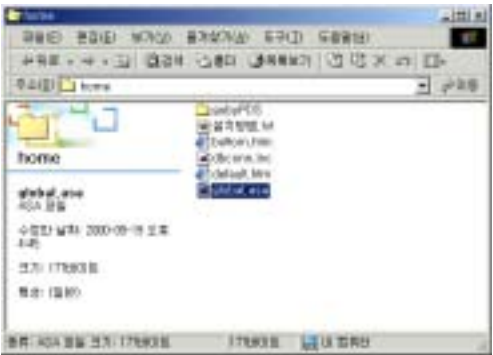


global.asa 파일 설정

게시판 소스의 루트 폴더를 보면 global.asa 파일이 들어 있습니다. 이 파일은 반드시 웹사이트의 루트에 있어야 합니다. 만일 사용자가 게시판 소스를 다른 작업 폴더에 넣었을 경우, global.asa 파일을 웹사이트의 루트로 이동시킵니다.

그림 4-8

global.asa 파일은 반드시 웹사이트 루트에 있어야 한다



만약 사용자의 웹사이트 루트에 global.asa 파일이 이미 존재한다면 다음 코드를 global.asa 파일에 추가시킵니다.

```
<!--METADATA TYPE= "typelib" NAME= "ADODB Type Library" FILE="C:\Program Files\Common Files\SYSTEM\ADO\msado15.dll" -->
```

데이터베이스 파일 살펴보기

DB 폴더에 있는 gobyPDSDB.mdb 파일은 마이크로소프트 액세스를 이용하여 만들어진 데이터베이스 파일입니다. 사용자의 시스템에 액세스 프로그램이 설치되어 있다면, 이 파일을 더블클릭합니다. 소스에 포함된 mdb 파일은 액세스 97에서 만들어진 것이기 때문에 액세스 2000 이상의 버전에서 열 때는 데이터를 변환해 주는 것이 좋습니다. 다음 그림은 액세스 2000에서 불러온 것입니다.

그림 4-9

MDB 파일을 액세스 2000에서 불러온 화면

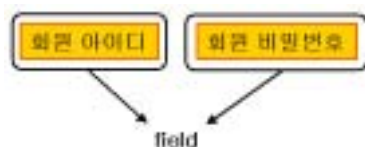


데이터베이스의 구조를 보면 gobyPDSDB.mdb에 tbl_GobyPDS와 tbl_GobyPDSAdmin의 두 가지 테이블이 들어 있는 것을 볼 수 있습니다. tbl_GobyPDS는 게시판에 사용되는 데이터가 저장되는 테이블이고, tbl_GobyPDSAdmin은 게시판 관리에 필요한 데이터와 관리자 데이터가 저장되는 테이블입니다.



데이터베이스 구조에 대해

컴퓨터 프로그래밍에서 데이터베이스 조직이나 구조를 스키마라고 합니다. 데이터베이스의 스키마는 필드(field)부터 시작되는데, 필드는 데이터베이스의 최소 단위이며 실제로 데이터가 기록되는 곳입니다.



이 필드가 모여 레코드(record)를 구성하는데, 데이터베이스에서 자료를 액세스하려면 이 레코드를 중심으로 데이터를 읽고, 저장하고, 삭제하고, 수정하게 됩니다.



또한 레코드가 모이면 하나의 테이블을 이루는데, 레코드를 액세스하기 위한 대상이 됩니다.



마지막으로 테이블이 모여 비로소 데이터베이스를 형성하게 됩니다.

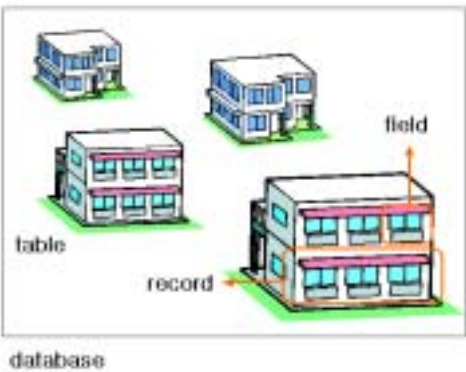
데이터베이스의 종류에 따라 테이블로만 구성된 데이터베이스가 있는가 하면, 저장 프로시저를 활용하거나 뷰를 다양하게 활용하여 스키마를 갖춘 데이터베이스도 있습니다.



역으로 생각해 보면, 데이터베이스 내에는 테이블이 있고, 테이블에는 데이터들이 저장되는 레코드가 있고, 레코드에는 실제 데이터를 담는 필드가 있는 것입니다.

간단한 예로 우리가 사는 아파트에 비교해 볼까요? 아파트는 보통 단지로 조성됩니다. 서울에 있는 잠실 5단지라고 하면 누구나 찾을 수 있습니다. 단지에 해당하는 것이 바로 데이터베이스입니다. 이 단지 내에는 여러 개의 동이 있습니다. 10층짜리 동도 있고 20층짜리 고층 동도 있으며,

타워형이나 복도형 등은 다양한 형태의 동이 있습니다. 이것을 테이블로 보면 됩니다. 여기서 테이블은 서로 다른 구조를 가질 수 있다는 것을 알 수 있습니다. 또한 하나의 동은 층별로 구성되는데, 각 층을 레코드로 생각할 수 있습니다. 그리고 각 층에는 사람들이 사는 호가 있는데, 이 호가 필드에 해당됩니다. 각 호에는 번호가 붙어 있어 구별할 수 있듯이 필드는 각각 다른 이름으로 구별되는 독립적인 것입니다.



앞에 있는 그림에서 'tbl_GobyPDS' 테이블을 선택하고, [디자인] 메뉴를 클릭하면, 다음 그림처럼 테이블을 구성하는 필드를 볼 수 있습니다.

그림 4-10
tbl_GobyPDS
테이블의 구조



마찬가지로 tbl_GobyPDSAdmin을 선택하고 [디자인] 메뉴를 클릭해 보면, 다음 그림처럼 테이블을 구성하는 필드가 나타납니다.

그림 4-11

tbl_GobyPDSAdmin
테이블의 구조

필드 이름	데이터 형식	길이
forum_id	텍스트	
name	텍스트	
bgo	텍스트	
tbc	텍스트	
tbt	텍스트	
tableWidth	텍스트	
PageSize	숫자	
SHaPage	숫자	
AdminID	텍스트	
AdminPassword	텍스트	
writedate	날짜/시간	
Header	워드	
Footer	워드	

필드 속성

필드 (조회)

필드 크기: 장수(Length) 문자

필드 형식: (예: 금액, 통화)

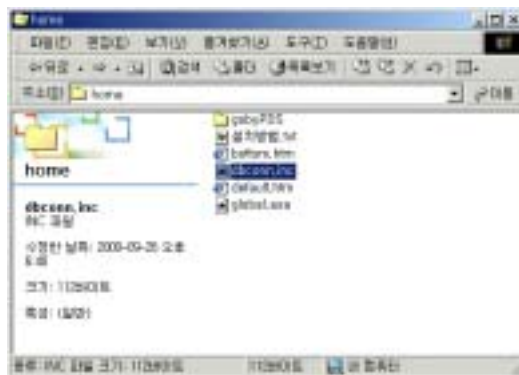
DSN으로 DB 연결 인스턴스 만들기

앞에서 데이터베이스 연결 객체를 만드는 방법에는 두 가지가 있다고 언급했습니다. 먼저 DSN을 사용해 만드는 방법부터 알아보겠습니다.

압축을 푼 작업 폴더의 루트에는 dbconn.inc 파일이 있습니다(그림 4-10 참조). 이 파일의 역할은 데이터베이스의 연결을 담당하는 연결 인스턴스(객체)를 생성하는 것입니다. 따라서 데이터베이스의 입출력이 필요한 파일에 이 dbconn.inc 파일을 include해서 사용하면, 입출력이 필요한 파일마다 객체 생성을 위해 코딩할 필요가 없습니다.

그림 4-12

데이터베이스 입출
력이 필요한 파일에
dbconn.inc를
인클루드해서
사용한다



이 파일의 내용은 다음과 같이 구성되어 있습니다.

```
1 <%
2 Dim Conn
3 set Conn=Server.CreateObject("ADODB.Connection")
4 Conn.open("DSN=gobyPDSDB") ' odbc 설정 내용
5 %>
```

2행은 Conn이라는 변수를 선언합니다.

3행은 Server 객체의 CreateObject 메소드를 사용하여 2행에서 선언한 변수 Conn을 인스턴스(연결 객체)로 만듭니다. 이 인스턴스는 데이터베이스 연결에 사용됩니다.

4행은 Conn 인스턴스의 open 메소드를 사용하여 데이터베이스를 엽니다. DSN의 값은 사용자가 시스템 ODBC 설정에서 지정한 이름을 입력합니다. 여기서는 gobyPDSDB 대신에 myPDS라고 입력하였습니다.

한 가지 의문이 생기지 않으셨나요? 분명 ASP 코드인데 파일 확장자가 asp가 아니라 inc인 이유는 무엇일까요? inc는 include의 약자입니다. 이 확장자로 된 파일은 asp 파일에 포함되는 일부분을 별도로 만들어 asp에 끼워 넣을 때 주로 사용합니다. 이렇게 되면 모든 asp 페이지에서 동일한 코드를 사용하는 작업이 있을 때 하나의 파일로 공유할 수 있습니다. 또한 예를 들어 설명한 dbconn.inc처럼 시스템 사양이 다른 컴퓨터에서도 이 파일의 내용만 수정(여기서는 DSN 이름을 수정하였음)함으로써 전체 asp 파일을 한 번에 바꿀 수 있는 장점도 있습니다.

String으로 DB 연결 인스턴스 만들기

이번에는 사용자 정의 String으로 DB 연결 객체를 생성해 보겠습니다. String 방법으로 연결 객체를 생성할 때는 웹서버에 있는 데이터베이스 파일의 위치, 즉 하드디스크상의 절대 경로를 알아야 합니다. 이것을 모르면 String 방식으로 연결 객체를 생성할 수 없습니다. 여기서 우리가 배운 것을 활용해 보겠습니다. 바로 Server.MapPath()를 활용하는 것입니다. 다음과 같은 코드로 ASP 파일을 만들어 웹서버에 업로드한 후, 웹브라우저로 접속해 봅니다.


```
<HTML>
<HEAD></HEAD>
<BODY>
<center>
<font face="돋움" size="2"> 내 웹사이트의 절대 경로는 : <%=Server.MapPath(".")%>
</font></center>
</BODY>
</HTML>
```

그러면 다음 그림처럼 자신의 웹사이트에 대한 하드디스크 상의 경로가 표시됩니다. 이 경로를 잘 적어둡니다.

그림 4-13

웹사이트의 물리적인
경로가 표시된 화면



String 방식으로 연결 객체를 만들려면 다음과 같은 코드를 dbconn.inc 파일에 코딩합니다. 소스 코드에서 ‘mdb 파일의 절대 경로’ 부분은 웹사이트의 절대 경로를 사용합니다. 즉, d:\ddcom_family\studiomx\design이 웹사이트의 루트라면, 이 폴더 하위의 gobyPDS\DB\gobyPDSDB.mdb 파일이 있는 것입니다. 따라서 전체 mdb 파일의 경로는 d:\ddcom_family\studiomx\design\gobyPDS\DB\gobyPDSDB.mdb가 됩니다.

```
<%
Dim Conn
set Conn=Server.CreateObject("ADODB.Connection")
Conn.open "Driver={Microsoft Access Driver (*.mdb)};DBQ=mdb 파일의 절대 경로"
%>
```

이 책에서는 외부 호스팅이나 계정에 파일을 업로드한 후, ASP 파일을 테스트할 것입니다. 따라서 연결 객체를 String 방식으로 사용하도록 하겠습니다.

03

리스트 목록 화면 프로그래밍

데이터베이스의 데이터를 화면으로 표시해주는 리스트 목록 화면을 만들어 보겠습니다.

우리가 만들 리스트 목록 화면은 다음과 같이 구성됩니다.

레이아웃 파일	gobyPDS\list.asp
프로그램 파일	gobyPDS\includeFiles\list.asp
실행	http://도메인, 또는 계정 URL/gobyPDS/list.asp?forum_id=pds

프로그램을 실행해보면 다음 그림과 같습니다. list.asp 파일에 접속할 때 매개변수 forum_id를 사용하는데, forum_id에 저장되는 pds는 게시판 이름에 해당합니다. 이 이름으로 하나의 게시판 프로그램을 통해 여러 개의 게시판을 만들 수 있습니다. 참고로 게시판 이름, 즉 forum_id는 관리자 프로그램에서 생성하고, pds는 미리 만들어 놓은 forum_id 값입니다. 현재 pds 게시판에는 데이터베이스에 저장된 데이터가 없기 때문에 게시물이 없습니다.

그림 4-14

리스트 목록 화면
프로그램의 결과



먼저 gobyPDS\list.asp 파일부터 살펴보겠습니다. list.asp 소스 코드의 내용은 [예제 4-1]과 같습니다.

예제 4-1

gobyPDS/list.asp

```
1 <%@ Language=VBScript %>
2 <html>
3 <head>
4 <title>자료실 리스트</title>
5 <meta http-equiv="Content-Type" content="text/html; charset=euc-kr">
5 <!--#include file=../dbconn.inc -->
6 <LINK rel="stylesheet" type="text/css" href="gobyPDS.css">
7 <script language=javascript>
8 function searchOk() {
9 if (searchForm.searchKey.value == "" ) {
10 alert("검색어를 입력하세요");
11 searchForm.searchKey.focus();
12 return false;
13 }
14 searchForm.submit();
15 }
16 function newwindow() {
17 var str,width,height;
18 URL="help.htm";
19 width=450;
20 height=300;
21 str="''toolbar=no, location=no, directories=no, status=no, menubar=no,
    scrollbars=no, resizable=no, copyhistory=no,";
22 str=str+"width="+width;
23 str=str+",height="+height+"''";
24 window.open(URL,'remote',str);
25 }
26 </script>
27 </head>
28 <body bgcolor="#FFFFFF">
29 <table border="0" cellspacing="0" cellpadding="0" align=center>
30 <tr align="left" valign="top">
31 <td >
32 <!--#include file=includeFiles/list.asp-->
33 </td>
34 </tr>
35 </table>
36 </body>
37 </html>
```

1행은 VBScript 코드로 제작된 ASP 파일이라는 것을 알려주는 코드입니다. 이 코드가 없어도 ASP 실행에는 문제가 되지 않습니다.

2~37행은 HTML 코드입니다. 5행의 `<!--#include file=../dbconn.inc -->`는 상위 폴더(..)의 dbconn.inc 파일을 현재 위치에 삽입하라는 코드입니다. 즉 다음의 코드를 직접 삽입하는 것과 같은 역할을 합니다.

```
<%
Dim Conn
set Conn=Server.CreateObject("ADODB.Connection")
Conn.open "Driver={Microsoft Access Driver (*.mdb)};DBQ=mdb 파일의 절대 경로"
%>
```

6행은 외부 CSS 스타일을 연결하는 것입니다. 외부 CSS 스타일 파일을 연결한다는 것은 웹사이트의 전체적인 레이아웃과 아이덴티티를 위해 별도의 파일로 만들어 사용한다는 뜻입니다. 이렇게 하면 사이트의 레이아웃이 변경되거나 스타일을 바꿀 때, 파일을 하나 하나 열어 수정하는 대신 이 파일만 수정하면 간단히 처리할 수 있습니다.

7~26행은 자바스크립트 코드로, 두 개의 함수가 정의되어 있습니다. searchOk 함수는 데이터 검색을 위한 것이고, newwindow는 새로운 윈도우를 열어 도움말을 표시하기 위한 함수입니다.

28~36행은 실제 웹브라우저 화면에 표시되는 `<body>` 태그 부분입니다. 크게 어려운 내용은 없지만, 32행에 있는 `<!--#include file=includeFiles/list.asp-->` 코드는 includeFiles 폴더에 있는 list.asp 파일을 현재 위치에 삽입한다는 것을 의미합니다. 따라서 gobyPDS 폴더의 list.asp 파일은 데이터베이스 연결과 자바스크립트 실행 외에는 아무런 기능이 없습니다. 리스트 화면은 includeFiles라는 하위 폴더의 list.asp 파일에서 설정합니다.

그렇다면 includeFiles 폴더에 있는 list.asp의 내용을 살펴볼 필요가 있습니다. [예제 4-2]는 includeFiles/list.asp 파일의 소스입니다.

예제 4-2

includeFiles/list.asp

```
1 <%
2 forum_id=Request.QueryString("forum_id")
3 strSQL="SELECT * FROM tbl_GobyPDSAdmin WHERE forum_id='"&forum_id&"'"
```

```
4 set rs=conn.execute(strSQL)
5 if rs.EOF then Response.Redirect "includefiles/error.asp?code=notCreate
  Board"
6 name=rs("name")
7 bgc=rs("bgc")
8 thc=rs("thc")
9 thcf=rs("thcf")
10 tableWidth=rs("tableWidth")
11 ps=rs("pagesize")
12 conternp=rs("skippage")
13 header=rs("header")
14 footer=rs("footer")
15 set rs=nothing
16 Response.Write header
17 page=Request.QueryString("page")
18 if page <> "" Then
19     curPage = page
20     if curPage < 1 Then
21         curPage = 1
22     end if
23 else
24     curPage=1
25 end if
26 set rs=server.CreateObject("ADODB.RecordSet")
27 strSQL="SELECT pk_id, forum_id, group_id, group_order, msglevel,
  subject, writer, email, readno, downno, writedate, filename, content"
28 strSQL=strSQL&" FROM tbl_GobyPDS where forum_id='"&forum_id&"' order by
  group_id desc, group_order asc"
29 rs.CursorLocation=adUseClient
30 rs.open strSQL,conn,adOpenForwardOnly,adLockReadOnly
31 str=str&"<table width='"&tableWidth&" border=0 cellspacing=0 cellpadding=4>"
32 str=str&"<tr align=left valign=top>"
33 str=str&"<td height=22><img src=images/jn_7.gif width=12 height=12>"
34 str=str&"&nbsp;<b><font color=#000000 class=l> "&name&"</font></b>"
  </td>"
35 str=str&"<td height=22 align=right>[<a href='admintools/adminlist.asp'>
  admin</a>][<a href='frm_write.asp?Forum_id='"&Forum_id&"'>글 올리기 </a>]
  </td>"
36 str=str&"</tr>"
37 str=str&"</table>"
```

[illegible]

[illegible]


```
105 str=str&rs("writer")
106 else
107 str=str"<a href='mailto:'&rs("email")&"'>&rs("writer")&"</a>"
108 end if
109 str=str"</td>"
110 str=str"<td align=center>&writedate&"</td>"
111 str=str"<td align=center>&rs("readno")&"</td>"
112 str=str"<td align=center>&rs("downno")&"</td>"
113 str=str"</tr>"
114 i=i+1
115 rs.MoveNext
116 loop
117 if Request.QueryString("startpage")="" then
118     startpage=1
119 else
120     startpage=Request.QueryString("startpage")
121 end if
122 str=str"</table>"
123 str=str"<table width='&tablewidth&' border=0 cellpadding=0
        cellspacing=0>"
124 str=str"<tr align=left valign=top bgcolor=#CCCCCC>"
125 str=str"<td width='&tablewidth&' height=1 colspan=2><img src=
        images/jn_0.gif width=1 height=1></td>"
126 str=str"</tr>"
127 str=str"<td height=30 align=left><font color=dimgray>&r_count& 개의
        글이 있습니다.</font></td>"
128 str=str"<td align=right>"
129 if Cint(totpage) > Cint(conternp) then
130     if startpage=1 then
131         str=str"< <font color=dimgray>이전 '&conternp&'개 </font> ] "
132     else
133         str=str"<[<a href='list.asp?forum_id='&forum_id&'&page='&cint
        (startpage)-conternp&'&startpage='&cint(startpage)-conternp&'>
        이전 '&conternp&' 개 </a>] "
134     end if
135     For a=startpage to startpage+conternp-1
136         if a>totpage then
137             exit for
138         else
139             if a=Cint(curpage) then
```

```
140         str=str"<b> "&a&" </b>"
141         else
142             str=str" <a href='list.asp?forum_id="&forum_id&"
                &page="&a&"&startpage="&startpage&"'>"&a&"</a> "
143             End if
144         end if
145     Next
146     if ((startpage\conternp)=(totpage\conternp)) then
147         str=str"&[ <font color=dimgray>다음 "&conternp&" 개</font> ]"
148     else
149         str=str"&[<a href='list.asp?forum_id="&forum_id&"&page="&a&"
                &startpage="&a&"'> 다음 "&conternp&" 개 </a>] "
150     end if
151 else
152 str=str"&[ <font color=#808000>이전 "&conternp&" 개 </font> ] "
153 For a=startpage to totpage
154     if a=Cint(curpage) then
155         str=str"<b> "&a&" </b>"
156     else
157         str=str" <a href='list.asp?forum_id="&forum_id&"&page="
                "&a&"'>"&a&"</a> "
158     end if
159 next
160     str=str"&[ <font color=#808000> 다음 "&conternp&" 개 </font> ] "
161 end if
162 str=str"</td>"
163 str=str"</td>"
164 str=str"</tr>"
165 str=str"<tr>"
166 str=str"<td align=center colspan=2>"
167 str=str"<form action=search.asp Method=get name=search>"
168 str=str"<font color=dimgray><b>제목</b></font><input type=checkbox
        name=searchType 174 value=1 checked>"
169 str=str"<font color=dimgray><b>글쓴이</b></font> <input type=checkbox
        name=searchType      value=2>"
170 str=str"<font color=dimgray><b>내용</b></font> <input type=checkbox
        name=searchType      value=4>"
171 str=str"<input type=text name=searchkey size=20 maxlength=80 style=
        'border:1 solid #808080'>"
172 str=str"&nbsp;<input type=submit value=검색 style='border:1 solid
        black; height:19px'>"
```

```

173 str=str+"<input type=hidden name=forum_id value="&forum_id">"
174 str=str+"<a href='javascript:newwindow()'>확장검색</a>]"
175 str=str+"</form>"
176 str=str+"</td>"
177 str=str+"</tr>"
178 str=str+"</table>"
179 end if 'rs.eof end
180 set rs=nothing
181 set conn=nothing
182 Response.Write str
183 Response.Write footer
184 %>

```

리스트 목록 화면은 많은 기능을 포함하고 있기 때문에 소스가 매우 긴 편입니다. 자, 그럼 소스를 세부적으로 나눠서 하나씩 설명해 보겠습니다.

```
2 forum_id=Request.QueryString("forum_id")
```

2행의 `forum_id=Request.QueryString("forum_id")` 코드는 `get` 방식으로 전달된 값을 `forum_id`라는 변수에 저장합니다. 이전 예에서 `list.asp` 파일에 접속할 때 `forum_id`라는 매개 변수를 사용한 것을 기억하고 있을 것입니다. 이 매개변수에 저장된 값은 `pds`였죠. 따라서 2행은 그 값을 받아 다시 동일한 변수인 `forum_id` 변수에 저장하는 것입니다. 정리하면, 좌항의 `forum_id`는 새로 설정되는 변수이고, 우항의 `forum_id`는 `get` 방식으로 전달된 매개 변수입니다. 코드가 같다고 해서 같은 값으로 생각하는 일이 없길 바랍니다.

```
3 strSQL="SELECT * FROM tbl_GobyPDSAdmin WHERE forum_id='"&forum_id&'"
4 set rs=conn.execute(strSQL)
```

3행은 데이터베이스의 `tbl_GobyPDSAdmin` 테이블에서 모든 필드의 값을 가져오는 SQL 구문을 설정합니다. 여기서 `where`는 조건식으로, `forum_id` 변수 값이 테이블의 `forum_id` 필드 값과 일치하는 데이터만 가져옵니다. 이 `select` 문을 `strSQL` 변수에 저장합니다.



SQL 문법 : Select

Select 구문은 데이터베이스로부터 데이터를 가져올 경우 사용되는 구문입니다. 기본 문법 형식은 다음과 같습니다.

```
Select (* 또는 필드명) from 테이블 명
[where 필드명=문자열]
[order by 필드명 (desc or asc)]
```

4행은 rs라는 인스턴스를 생성합니다. rs 인스턴스는 conn 인스턴스에 execute 메소드를 사용하여 strSQL를 실행합니다. 따라서 rs는 tbl_GobyPDSAdmin 테이블에서 forum_id 값이 일치하는 단 하나의 레코드만 불러오게 됩니다. 나중에 보면 알겠지만, forum_id의 값은 동일한 값이 없는 유일한 값입니다. 게시판의 이름이기 때문이죠.

```
5 if rs.EOF then Response.Redirect "includefiles/error.asp?code=notCreateBoard"
```

5행은 if 조건문으로, rs.EOF는 전체 레코드의 마지막까지 읽었는데도 일치하는 값이 없을 경우를 의미합니다. 다시 말해, 매개 변수 forum_id의 값으로 만들어진 게시판이 없다는 것입니다. 이렇게 되면 includefiles/error.asp?code=notCreateBoard로 하이퍼링크됩니다. 여기서 EOF는 End Of File의 약자로, 데이터가 존재하지 않아서 마지막 레코드셋을 읽어 들였다면 값이 true가 되고, 마지막이 아닌 경우에는 false가 됩니다. 다음 그림은 forum_id의 값이 tbl_GobyPDSAdmin 테이블에 없을 때 이동하는 페이지입니다.

그림 4-15

forum_id 값이 테이블에 없을 때 이동하는 페이지



```

6  name=rs("name")
7  bgc=rs("bgc")
8  thc=rs("thc")
9  thcf=rs("thcf")
10 tableWidth=rs("tableWidth")
11 ps=rs("pagesize")
12 conternp=rs("skippage")
13 header=rs("header")
14 footer=rs("footer")
15 set rs=nothing

```

6~14행은 레코드 필드값을 해당 변수에 저장합니다. 혼동을 피하기 위해 레코드 필드의 이름과 동일한 변수명을 사용했습니다. 즉, 데이터베이스로부터 필요한 필드값을 불러와서 각 필드명과 동일한 이름의 변수에 이 값을 저장합니다. 데이터베이스에서 필드값을 불러오는 방법은 앞에서 생성한 rs 인스턴스를 이용하는데, rs("필드명")의 형태로 사용합니다. 여기서 rs 인스턴스는 데이터베이스로부터 데이터를 불러오는 쿼리 구문(Select)과 동일합니다.

15행에서는 지정된 rs 인스턴스가 더 이상 필요없으므로 소멸시킵니다.

```

16 Response.Write header

```

16행은 Response.Write로 header 변수의 값을 출력합니다. header는 13행에 있는 데이터베이스의 rs 인스턴스의 header 필드 값을 의미합니다. 이 값은 관리자 모드에서 지정하는 값으로, 게시물 리스트 화면의 머리글에 해당합니다.

```

17 page=Request.QueryString("page")
18 if page <> "" Then
19   curPage = page
20   if curPage < 1 Then
21     curPage = 1
22   end if
23 else
24   curPage=1
25 end if

```

17행은 get 방식으로 전송된 데이터 중 page의 값을 page라는 변수에 저장합니다. 이 값은 게시판의 페이지 번호를 의미합니다.

18번, 19행에서는 page 값이 존재(page > "")할 경우, curPage 변수에 17행에서 설정한 page 값을 다시 저장합니다. 만약 페이지 값이 없다면 23행의 else 구문으로 이동해서 24행의 curPage 값을 1로 설정한 다음, 25행에서 if 문을 종료합니다.

20행은 if 조건문으로, curPage 값이 1보다 작으면 curPage의 값을 1로 설정하고, 22행에서 if 조건문을 종료한 후에 25행으로 이동합니다.

정리하면 17~25행은 페이지 값을 지정하는 코드로, 페이지 값이 1보다 작거나 없으면 1로 설정하고, 1보다 크면 그 수를 page 값에 저장합니다.

리스트 화면에 접속하려면 페이지 번호를 다음과 같이 넣어줘야 합니다.

```
http://www.studiomx.co.kr/design/gobyPDS/list.asp?forum_id=pds&page=1
```

페이지 번호가 존재할 경우, 해당 페이지부터 게시판의 리스트를 볼 수 있습니다. 페이지 번호를 매개변수로 지정하지 않으면 항상 1페이지를 화면에 불러오게 됩니다.

```
26 set rs=server.CreateObject("ADODB.RecordSet")
27 strSQL="SELECT pk_id, forum_id, group_id, group_order, msglevel,
    subject, writer, email, readno, downno, writedate, filename, content"
28 strSQL=strSQL&" FROM tbl_GobyPDS where forum_id='"&forum_id&"' order by
    group_id desc, group_order asc"
29 rs.CursorLocation=adUseClient
30 rs.open strSQL,conn,adOpenForwardOnly,adLockReadOnly
```

26행은 rs라는 레코드셋 인스턴스를 만듭니다. 이 라인을 보면 15행에서 왜 rs를 소멸시켰는지 알 수 있을 것입니다. 동일한 이름이기 때문입니다.

27행과 28행은 하나로 합칠 수 있는데, 예제에서는 구문을 간결하기 위해 분리한 것입니다. SELECT pk_id, forum_id, group_id, group_order, msglevel, subject, writer, email, readno, downno, writedate, filename, content FROM tbl_GobyPDS where forum_id='"&forum_id&"' order by group_id desc, group_order asc" 문은 strSQL 변수로 대치됩니다. select 다음에 필드명을 콤마(,)로 구별해 나열한 것은 필요한 필드값만 데이터베이스에서 불러오기 위함입니다.



레코드와 레코드셋

데이터베이스에서 데이터를 실제로 저장하는 장소는 테이블, 좀 더 자세히는 테이블 안의 필드에 데이터가 입력되는 것입니다. 테이블에 입력된 데이터는 행을 기준으로 구분할 수 있는데, 데이터를 저장하고 있는 테이블의 한 행을 레코드라고 합니다. 오른쪽 그림에서 세로열을 컬럼이라고 하는데, 컬럼은 필드 속성을 가지고 있습니다. 이렇게 필드들이 모여 하나의 가로행을 만들게 되는데, 이것을 레코드라고 합니다. 레코

드셋을 말 그대로 풀이해 보면, 레코드들의 집합입니다. 따라서 데이터들의 집합을 의미합니다

Product	Description	Unit	Weight	Volume	Price	Quantity	Amount
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12	12	12	12
Orange	24	12	12	12	12	12	12
Apple	24	12	12	12			



레코드셋의 기본 활용

웹에서는 레코드셋과의 연동을 위해 ADO 컴포넌트를 이용합니다. ADO 컴포넌트는 데이터베이스 연결 객체와 레코드셋 객체를 사용하여 데이터베이스를 연결하고 데이터를 불러옵니다. 연결 객체는 데이터베이스와의 연결 및 실행을 담당하고 레코드셋 객체는 데이터베이스에 쿼리 문을 전송해 필요한 데이터들을 가져오게 합니다. 연결 객체는 ODBC(Open DataBase Connection)를 사용해서 연결을 시도하게 됩니다. 데이터베이스와의 연결 후, 레코드셋 객체는 필요한 데이터를 레코드셋으로부터 불러와서 웹에서 사용할 수 있게 합니다. 레코드셋을 읽어오는 방식은 여러 가지가 있지만, 이 책에서는 String 방식으로 연결 객체를 생성한 후, 레코드셋 객체를 만듭니다.

다음 코드는 ODBC를 통해 DSN을 만들고, 그 DSN으로 연결 객체를 생성하고 있습니다. 그리고 Select 구문은 데이터베이스로부터 데이터를 가져올 때 사용되는 구문입니다.

```
<%
Set DBcon = Server.CreateObject("ADODB.Connection")
DBcon.open("DSN=odbc등록이름을 기입;UID=DB 사용자 계정 기입;PWD=사용자 계정 패스 워드 기입
")

Set Rs=Server.CreateObject("ADODB.RecordSet")
Rs.open "Select * from 테이블명", DBcon
%>
```

첫 번째 Set 이하 코드는 연결 객체를 사용하여 데이터베이스와의 연결을 설정하고, 두 번째 Set 이하 코드는 레코드셋 객체를 사용하여 필요한 데이터를 불러오는 것입니다.

28행의 order by는 정렬순서를 지정합니다. group_id 필드값은 내림차순으로, group_order 필드의 값은 오름차순으로 정렬하라는 의미입니다(desc: 내림차순, asc: 오름차순).

29행의 `rs.CursorLocation=adUseClient`는 레코드셋의 커서 위치를 지정합니다. 여기서 `adUseClient`는 데이터 ADO가 커서를 클라이언트에서 관리하도록 해줍니다. 이것은 레코드셋을 열기 전에 수행되어야 합니다. 서버측에서 커서를 조작하려면 `adUseServer` 프라퍼티를 사용하는데, 기본값으로 이 프라퍼티가 설정되어 있습니다.

30행에서 레코드셋을 엽니다. 여기서 `adOpenForwardOnly`는 커서의 타입인데, 레코드셋을 앞으로 이동시키는 커서 타입 상수입니다. `adLockReadOnly`는 락 타입으로 레코드 값을 읽기전용으로 불러옵니다.

```
31 str=str+"<table width="&tablewidth&" border=0 cellpadding=4>"
32 str=str+"<tr align=left valign=top>"
33 str=str+"<td height=22><img src=images/jn_7.gif width=12 height=12>"
34 str=str+"&nbsp;<b><font color=#000000 class=l> "&name&"</font></b>"
   str=str+"</td>"
35 str=str+"<td height=22 align=right>[<a href='admintools/adminlist.asp'"
   str=str+"admin</a>] [<a href='frm_write.asp?Forum_id="&Forum_id&"'>글 올리기</a>]"
   str=str+"</td>"
36 str=str+"</tr>"
37 str=str+"</table>"
38 str=str+"<table width="&tablewidth&" border=0 cellpadding=0"
   str=str+"cellpadding=4>"
39 str=str+"<tr bgcolor="&thc&">"
40 str=str+"<td width=30 align=center height=20> <font color="&thcf&"
   str=str+"size=2>번호</font></td>"
41 str=str+"<td width=30 align=center height=20> <font color="&thcf&"
   str=str+"size=2> &nbsp;</font></td>"
42 str=str+"<td align=center height=20> <font color="&thcf&" size=2>"
   str=str+"제목</font></td>"
43 str=str+"<td width=80 align=center height=20> <font color="&thcf&"
   str=str+"size=2> 작성자</font></td>"
44 str=str+"<td width=80 align=center height=20> <font color="&thcf&"
   str=str+"size=2> 등록일</font></td>"
45 str=str+"<td width=30 align=center height=20> <font color="&thcf&"
   str=str+"size=2> 조회</font></td>"
46 str=str+"<td width=30 align=center height=20> <font color="&thcf&"
   str=str+"size=2> 다운</font></td>"
47 str=str+"</tr>"
```

31~47행은 HTML을 ASP 내에서 기술합니다. 47번에 최종 str 변수는 전체 내용을 모두 포함하는데, 188행의 Response.Write str 명령으로 출력됩니다.

31번과 38행의 tableWidth는 10행에서 설정된 변수로, 관리자 프로그램에서 설정된 테이블의 크기가 대입됩니다. 34행의 name은 6행에서 설정된 변수로, 관리자 프로그램에서 테이블의 제목이 대입됩니다.

39행의 thc는 8행에서 설정된 변수로, 관리자 프로그램에서 설정된 셀의 색상이 대입됩니다. 40~46행의 thcf는 9행에서 설정된 변수로, 관리자 프로그램에서 설정된 폰트의 색상이 대입됩니다.

이 부분의 코드가 실행되면 다음 그림과 같은 화면이 만들어집니다.



```

48 if rs.EOF then
49 str=str&"<tr><td colspan=5 align=center>아직 등록된 자료가 없습니다</td>
   </tr>"
50 else

```

48행은 if 조건문입니다. rs.EOF의 값이 참이면 49행을 실행하고, 거짓이면 else 이하 코드를 실행합니다. 이 코드의 끝(end if)은 179행입니다. rs.EOF는 26~30번 코드에서 실행된 SQL 구문으로, 데이터베이스의 레코드를 불러온 결과, 조건에 맞는 레코드가 하나도 없어 레코드셋을 생성할 수 없는 경우를 의미합니다.

48~50, 179행이 없어도 프로그램을 실행하는데 문제가 없습니다. 다만 데이터베이스에 저장된 데이터가 없으면, 다음과 같은 에러 메시지가 나타납니다.

그림 4-16

데이터베이스에 저장된
데이터가 없을 때
나타나는 에러 메시지



```
51 r_count=rs.RecordCount
52 rs.PageSize=ps
53 RS.AbsolutePage = curPage
54 totPage = r_count \ rs.PageSize
55 if not totPage*rs.PageSize = r_count then totPage=totPage+1
56 i=0
```

51~56행은 페이지와 관련된 변수의 값을 지정합니다. 51행은 rs 객체의 RecordCount 속성을 사용하여 총 레코드 개수를 r_count 변수에 저장합니다. 이 값은 읽어들이 전체 레코드의 수가 됩니다.

52행은 11행에서 지정한 ps 변수에 저장된 값을 rs.PageSize에 대입합니다. 이 값은 관리자 프로그램에서 설정한 값으로, 한 화면에 표시될 데이터의 개수를 의미합니다.

53행의 RS.AbsolutePage에는 17~25행에서 지정한 curPage 값을 대입합니다. 이 값은 현재 페이지를 표시합니다.

54번 totPage는 전체 레코드를 PageSize로 나눈 값으로, 게시판의 전체 페이지 수를 의미합니다. 만약 전체 레코드(r_count) 수가 20이고 페이지(rs.PageSize)를 20으로 설정한다면, 전체 페이지(totPage)는 1이 될 것입니다. 그러나 전체 레코드(r_count) 수가 25라면, totpage의 값은 1.25가 됩니다. 이 때 소수점 이하의 값은 버리게 되는데, 이로 인해 5개의 게시물은 표시될 자리가 없어집니다. 따라서 55번째 라인에 역으로 계산한 값이 r_count 값과 같지 않으면 1을 더해 페이지를 하나 더 만들게 됩니다.



연산 기호 / 와 \

보통 나누기를 할 때는 /를 사용합니다. 10/4를 계산하면 결과 값이 2.5가 됩니다. 그러나 10\4를 계산하면, 결과 값이 2가 됩니다. 즉, 결과 값에서 정수 부분만 반환하는 것입니다.

56번째 라인은 i라는 변수에 0을 설정합니다. 이 i 값은 표시될 게시물의 크기(PageSize)와 비교해서 한 화면에 지정된 개수의 게시물을 출력하도록 해줍니다. 즉, PageSize의 값이 20이면, 한 화면에 20개의 게시물만 표시되게 합니다.

```

57 do while not rs.EOF and rs.PageSize > i
....
116 loop

```

57~116행은 do while 반복문으로, 지정된 조건이 충족하는 동안만 반복합니다. 여기서 지정된 조건은 레코드셋이 존재하고(not rs.EOF), PageSize의 값이 I보다 클 때까지입니다.



연산 기호 and와 or

and와 or는 두개의 조건을 결합할 때 사용합니다. 먼저 and는 좌항과 우항의 값 모두 만족을 해야 참(true)가되면 둘 중 하나라도 만족을 하지 않으면 거짓(false)가 됩니다. 그러나 or는 좌항과 우항의 값 중 하나만이라도 만족하면 참(true)가 됩니다.

```

58 check_fileexe = rs("filename")
59 if not isNull(check_fileexe) then
60 pos = instrrev(check_fileexe, ".")+1
61 check_fileexe = mid(check_fileexe, pos)
62 end if
63 Select Case Ucase(check_fileexe) '확장자에 따른 이미지 선택
64 Case "HWP", "COM", "EXE", "TXT", "WAV", "XLS", "DOC", "ZIP", "RAR", "GIF",
   "BMP", "MP3", "PDF", "PPT"
65     fileicon=check_fileexe & ".gif"
66     Case "RA", "RAM"
67         fileicon = "ra.gif"
68     Case "HTM", "HTML"
69         fileicon= "htm.gif"
70     Case "JPG", "JPEG"
71         fileicon="jpg.gif"
72     Case "MPG", "MPEG", "AVI"
73         fileicon="mpg.gif"
74     Case else
75         fileicon="etc.gif"
76 End Select

```

58~76행은 업로드된 파일의 확장자에 다른 아이콘을 표시해주는 기능입니다.

78~80행은 for 반복문입니다. 이 반복문의 역할은 답변 글의 들여쓰기 기능을 삽입하기 위한 것입니다. 즉, 답변의 단계(msglevel)에 따라 들여쓰는 정도가 설정됩니다. cnt의 값이 0부터 시작해서 Cint(rs("msglevel"))의 값에 도달할 때까지 for ~ next 사이의 구문이 반복적으로 실행됩니다. 이 값은 데이터베이스에서 가져오며, 79행의 blank 값이 로 대체됩니다. 만약 msglevel의 값이 3이라면 가 여섯 개 출력됩니다. 여기서 출력의 의미는 출력된 만큼 빈 공간을 만들어 내는 것입니다. 참고로 는 빈 문자열을 출력하는 HTML 특수 문자 코드입니다.

```

81 content=rs("content")
82 If Len(content) > 150 Then content=Mid(content,1,150)&"..."
83 content = replace(content, "\"", "&quot;")
84 subject=rs("subject")
85 If Len(subject) > 20 Then subject=Mid(subject,1,20)&"..."
86 subject = replace(subject, "\"", "&quot;")

```

81행은 레코드셋의 content 값을 content라는 변수에 저장합니다.

82행은 Len이란 함수를 사용해서 문자의 길이를 얻어낸 후, 이 값이 150보다 크면 content의 내용을 150으로 줄인 후, 나머지를 ...으로 표시하게 합니다.

83행은 content 내의 "를 HTML에서 표시하기 위해 "로 대체합니다. 이렇게 문자를 바꾸는 이유는 ASP 코드 내에서 문자열을 표시할 때 사용하는 "와 구별이 되지 않기 때문에 HTML에서 사용하는 특수 문자 코드인 "로 바꿔주는 것입니다.



replace 함수

주어진 문자열(a)에서 그 문자열 안에 있는 어떤 특정한 문자열(b)을 다른 문자열(c)로 바꾸는 역할을 합니다. 기본 형식은 다음과 같습니다.

Replace(문자열(a), 문자열안의 바꾸고자 하는 문자열(b), 바꿀 문자열(c))

84행 역시 레코드셋의 subject의 값을 subject라는 변수에 저장합니다.

85행은 문자의 길이가 20보다 크면 나머지 부분을 ...로 표시합니다.

86행은 83행과 동일한 것으로, subject 내의 "를 HTML에서 표시하기 위해 "로 대체합니다.

```

87 writedate = rs("writedate")
88 writedate = month(writedate)&"/"&day(writedate)&" "&hour(writedate)&":
    "& Minute(writedate)

```

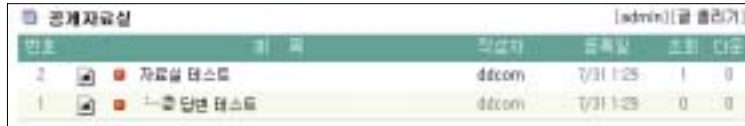
87번과 88번은 날짜에 관한 부분입니다. 87행에서는 레코드셋의 writedate 값을 writedate라는 변수에 저장합니다. month, day, hour는 각각 월, 일, 시간을 추출해주는 함수입니다. 88행은 입력된 날짜를 각각 분리해서 다시 조합합니다.

```
89 if (i mod 2) = 0 then str=str+"<tr bgcolor=#ffffff>" else str=str+"<tr
    bgcolor = #F4F9EB>"
```

89행은 셀의 색을 지정하는 부분인데, i의 값을 2로 나누어 나머지가 0이면 #ffffff, 그렇지 않으면 #F4F9EB로 셀의 색상을 바꿉니다. 즉, 홀수와 짝수를 구별하는 것입니다.

그림 4-17

셀 단위로 색상을
다르게 해준 화면



번호	제 목	작성자	등록일	조회	답수
2	자료실 테스트	datcom	7/31 1:25	1	0
1	하루 단편 테스트	datcom	7/31 1:25	0	0



mod

mod는 주어진 식의 나머지를 구하는 연산 기호입니다. 예를 들어, 10 mod 6의 결과는 4입니다.

```
90 str=str+"<td align=center >"&(r_count-i-(curPage-1)*rs.PageSize)&"</td>"
```

90행부터는 다시 str 값이 추가됩니다. 90행은 게시물의 번호인데, (r_count-i-(curPage-1)*rs.PageSize)가 바로 번호를 만들어 내는 식입니다. 예를 들어, 전체 레코드 수(r_count)가 20이고 한 페이지에 표시될 게시물의 수가(PageSize) 20이라면, 현재 페이지(curPage)는 1이 됩니다. 여기서 PageSize 값이 20일 경우, i는 앞에서 설정한 조건에 따라 0부터 19까지의 값을 차례대로 갖게 됩니다. 따라서 (20-i-(1-1)*20)라는 식에 의해 게시물의 번호는 20,19,18,... 순으로 정렬됩니다.

만약 총 레코드 수가 50이고 한 페이지에 표시될 게시물의 수가 20인, 세 번째 페이지의 세 번째 게시물의 번호는 어떻게 될까요? (50-i-(3-1)*20)이라는 식에 의해 처리되어 10,9,8,... 의 순서로 게시물의 번호가 정렬됩니다.

```
91 str=str+"<td align=center ><img src=images/"& fileicon &"></td>"
```

91행의 파일 아이콘이 표시되는 부분입니다. images 폴더에 있는 이미지 파일을 삽입하는데, fileicon 변수에는 58-76 라인에서 지정된 파일명이 대입됩니다.


```
92 str=str("<td align=left>"
93 if now() < dateadd("d",1,rs("writdate")) then
94 str=str("<IMG SRC='images/jn_11.gif'>"
95 else
96 str=str("<IMG SRC='images/jn_11b.gif'>"
97 End If
```

92행부터 97행은 새로 등록된 게시물에 24시간 동안 뽀짝이는 아이콘을 삽입하는 역할을 합니다.

93행은 현재 시간 값과 비교하여 새로운 게시물에 대한 이미지를 삽입합니다. dateadd 함수는 지정된 시간을 추가하는 함수로, 날짜에 1을 더해 그 값이 현재 시간보다 크면 94행을 실행해 새로운 게시물로 간주하게 됩니다. 즉, 하루 24시간 동안 나타나도록 하는 것입니다. 물론 그렇지 않다면 96행을 실행할 것입니다.

```
98 str=str&blank
99 if Cint(rs("msglevel"))>0 then
100 str=str("<img SRC='images/re.gif' border=0 WIDTH=26 HEIGHT=12>"
101 end if
102 str=str("&nbsp;<a href='view.asp?no="&rs("pk_id")&"&page="&curPage&"
&forum_id="&forum_id&" title="& content &">"& subject &"</a>"
```

98~101행은 레코드셋의 msglevel 값이 0보다 크면, images 폴더의 re.gif 이미지를 삽입합니다. 추후에 글쓰기 화면인 write.asp 파일을 확인해 보면, 일반적인 글을 썼을 때는 msglevel의 값이 0으로 기록되고, 답변 글을 기록하는 reply_save.asp 파일에서는 답변 글의 단계에 따라 1부터 수가 증가되면서 기록됩니다.

102행에서는 제목(subject)을 표시하고, 하이퍼링크를 설정해 게시물을 자세히 볼 수 있습니다. 하이퍼링크를 설정할 때는 get 방식으로 pk_id, curPage, forum_id 값을 view.asp 파일에 함께 전송합니다. pk_id는 데이터베이스에 저장된 레코드의 일련번호이고, curPage는 현재 보고 있는 게시판의 페이지 번호이며, forum_id는 게시판 이름입니다.

```

103 str=str"<td align=center>"
104 if rs("email")="" then
105 str=str&rs("writer")
106 else
107 str=str"<a href='mailto:'&rs("email")&'>'&rs("writer")&'</a>"
108 end if
109 str=str"</td>"

```

104~108행은 레코드셋에 이메일이 없으면 작성자만 표시하고, 이메일이 있으면 작성자를 표시하면서 이메일 주소로 하이퍼링크시킵니다.

```

110 str=str"<td align=center>"&rs("readdate")&"</td>"
111 str=str"<td align=center>"&rs("readno")&"</td>"
112 str=str"<td align=center>"&rs("downno")&"</td>"
113 str=str"</tr>"

```

110~112행은 글을 쓴 날짜와 읽은 횟수, 다운 횟수를 표시합니다.

```

114 i=i+1
115 rs.MoveNext
116 loop

```

114행은 i의 값을 1 증가시키고, 115행은 다음 레코드셋으로 이동하며, 116번은 48행으로 되돌아가 EOF의 값이 true가 될 때까지 반복합니다.

```

117 if Request.QueryString("startpage")="" then
118 startpage=1
119 else
120 startpage=Request.QueryString("startpage")
121 end if
122 str=str"</table>"

```

117~121행은 get 방식으로 전송된 startPage의 값이 있으면 이 값을 startPage 변수에 저장하고, 값이 없으면 startPage의 값을 1로 설정합니다. 122행은 38행에서 시작한 <table> 태그를 닫습니다.

```

123 str=str+"<table width=\"%tablewidth%\" border=0 cellpadding=0
      cellpadding=0>"
124 str=str+"<tr align=left valign=top bgcolor=#CCCCCC>"
125 str=str+"<td width=\"%tablewidth%\" height=1 colspan=2><img src=
      images/jn_0.gif width=1 height=1></td>"
126 str=str+"</tr>"
127 str=str+"<td height=30 align=left><font color=dimgray>\"&r_count&\" 개의
      글이 있습니다.</font></td>"

```

123~127행은 새로운 테이블을 하나 그린 다음, 첫 번째 셀에 전체 글의 개수를 표시합니다. 전체 글의 수는 레코드셋의 수(r_count)와 같습니다.

2 개의 글이 있습니다.	[이전 10 개] [다음 10 개]
---------------	-------------------------

```

128 str=str+"<td align=right>"
129 if Cint(totpage) > Cint(conternp) then
...
151 else
...
161 end if

```

128~161행은 스킵 페이지(skippage)에 관한 코드인데, if 조건문으로 되어 있습니다. 먼저 totpage의 값이 conternp의 값보다 크면 129~154행을 실행하고, 같거나 작으면 151행부터 실행합니다. 이 부분은 세부적으로 나누어 더 자세히 살펴보도록 하겠습니다.

```

130 if startpage=1 then
131 str=str&"[ <font color=dimgray>이전 "&conternp&"개 </font> ] "
132 else
133 str=str&"[<a href='list.asp?forum_id="&forum_id&"&page="&cint
      (startpage)-conternp&"&startpage="&cint(startpage)-conternp&"'> 이
      전"&conternp&" 개 </a>] "
134 end if

```

129행에서 totpage가 관리자 프로그램에서 지정된 스킵 페이지(conternp)보다 크면, 현재 시작 페이지(startpage)의 값에 따라 ‘이전 .. 개’라는 텍스트에 하이퍼링크가 설정됩니다. 이 때 페이지 값이 1이면, 131행에서 ‘이전 .. 개’라는 텍스트에 하이퍼링크를 설정하지 않으며, 그 외의 값이면 133행에서 ‘이전 .. 개’라는 텍스트에 하이퍼링크를 설정합니다. 만약 totpage가 관리자 프로그램에서 지정된 스킵 페이지(conternp)와 같거나 작으면 151행부터 실행됩니다.

```

135 For a=startpage to startpage+conternp-1
136     if a>totpage then
137         exit for
138     else
139         if a=Cint(curpage) then
140             str=str&"<b> "&a&" </b>"
141         else
142             str=str&" <a href='list.asp?forum_id="&forum_id&"
              &page="&a&"&startpage="&startpage&"'>"&a&"</a> "
143         End if
144     end if
145 Next

```

135~145행은 페이지 번호를 표시해 주는 코드인데, for ... next 반복문으로 처리합니다. 예를 들어 스킵 페이지가 5이고 현재 페이지가 6이라면, for 구문은 6부터 10까지의 수를 표시합니다. 만일 이 값이 전체 페이지(totpage) 값보다 크면, 137행에서 for 반복문을 빠져나와 146행으로 이동합니다.

139행의 if 조건문에서 a의 값이 현재 페이지이면 볼드체로 표시하고, 그렇지 않으면 페이지로 이동할 수 있도록 142행에서 하이퍼링크를 설정합니다.

```

146 if ((startpage\conternp)=(totpage\conternp)) then
147 str=str&"[ <font color=dimggray>다음 "&conternp&" 개</font> ]"
148 else
149 str=str&"<a href='list.asp?forum_id="&forum_id&"&page="&a&"
      &startpage="&a&"'> 다음 "&conternp&" 개 </a>] "
150 end if

```

146행은 마지막 페이지 번호까지 웹 문서에 표시되는 경우입니다. 현재 시작 페이지를 스킵 페이지로 나누었을 때의 값(몫)과 전체 페이지 수를 스킵 페이지로 나누었을 때의 값이 같으면 마지막 페이지가 표시되기 때문에, 147행에서 ‘다음 ...개’에 하이퍼링크를 설정하지 않습니다.

149행은 앞의 두 식의 값이 서로 다를 경우, 설정된 페이지 개수만큼 다음 페이지로 이동시킵니다.

```

151 else
152 str=str&"[ <font color=#808000>이전 "&conternp&" 개 </font> ] "
153 For a=startpage to totpage
154 if a=Cint(curpage) then
155     str=str&"<b> "&a&" </b>"
156 else
157     str=str&" <a href='list.asp?forum_id="&forum_id&"&page=
      "&a&"'>"&a&"</a> "
158 end if
159 next
160 str=str&"[ <font color=#808000> 다음 "&conternp&" 개 </font> ] "

```

129행의 조건이 틀릴 경우, 예를 들어 스킵페이지를 5로 설정했는데 전체 페이지가 5가 되지 않으면, ‘이전’과 ‘다음’에 하이퍼링크를 설정하지 않고 페이지만 표시합니다.

```

162 str=str&"</td>"
163 str=str&"</td>"
164 str=str&"</tr>"

```

str 값에 셀 태그를 추가합니다.

```
165 str=str("<tr">
166 str=str("<td align=center colspan=2>"
167 str=str("<form action=search.asp Method=get name=search>"
168 str=str("<font color=dimgray><b>제목</b></font><input type=checkbox
      name=searchType 174 value=1 checked>"
169 str=str("<font color=dimgray><b>글쓴이</b></font> <input type=checkbox
      name=searchType      value=2>"
170 str=str("<font color=dimgray><b>내용</b></font> <input type=checkbox
      name=searchType      value=4>"
171 str=str("<input type=text name=searchkey size=20 maxlength=80 style=
      'border:1 solid #808080'>"
172 str=str("&nbsp;<input type=submit value=검색 style='border:1 solid
      black; height:19px'>"
173 str=str("<input type=hidden name=forum_id value='&forum_id'>"
174 str=str("<[a href='javascript:newwindow()'>확장검색</a>]"
175 str=str("</form>"
176 str=str("</td>"
177 str=str("</tr>"
178 str=str("</table>"
```

165~178번은 검색을 위한 폼입니다. 174행에서 '확장검색'이라는 텍스트에 자바스크립트로 하이퍼링크가 설정되어 있습니다. 현재 보고있는 includFiles/list.asp가 포함하는 파일인 list.asp 레이아웃 파일의 16행에 있는 newwindow() 파일을 호출하게 됩니다.

제목 ☒	글쓴이 ☐	내용 ☐		검색 [확장검색]
--	------------------------------	-----------------------------	----------------------	-----------

```
179 end if 'rs.eof end
```

5행에서 시작한 if문을 종료합니다.

```
180 set rs=nothing
181 set conn=nothing
```

rs, conn 인스턴스를 소멸시킵니다. 소멸시키는 이유는 이들이 생성될 때 사용한 메모리를 다시 시스템에 반환시키기 위해서입니다.

```
182 Response.Write str
```

드디어 str에 저장된 HTML 코드를 출력합니다.

```
183 Response.Write footer  
184 %>
```

183행은 관리자 프로그램에서 지정한 footer 값을 출력합니다. 마지막으로 184행에서 asp 코드를 닫습니다.

04

세부 정보 보기 화면 프로그래밍

세부 정보를 보여주는 view.asp의 소스를 분석해 어떻게 프로그램되어 있는지 알아봅니다.

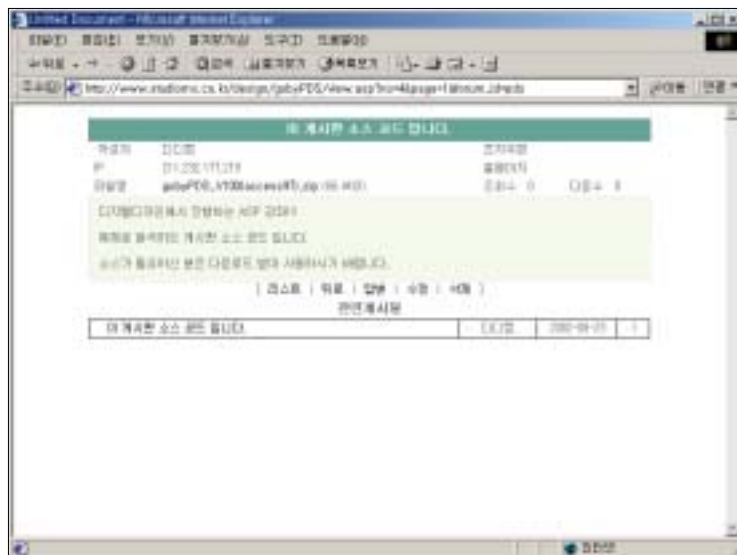
세부 정보 보기 화면(view.asp)은 다음과 같이 구성되어 있습니다.

레이아웃 파일	gobyPDS\view.asp
프로그램 파일	gobyPDS\includeFiles\view.asp
실행	http://도메인, 또는 계정 URL/gobyPDS/view.asp?no=숫자&page=숫자&forum_id=pds

이 프로그램을 실행하면 다음 그림과 같습니다. view.asp 파일에 접근하기 위해서는 froum_id, no, page 세 개의 매개변수가 사용됩니다. 우선 froum_id 매개변수는 게시판판을 구별해 주고, no 매개변수는 등록된 글을 구별해 줍니다. 예를 들어 no 매개변수 값이 4라고 가정하면, forum_id 값이 pds인 게시판에서 no 값이 4인 게시물을 출력하는 것입니다. 여기서 page 매개변수의 게시물 출력과 관련된 것이 아닙니다. 이 변수는 view.asp 파일에서 list.asp 파일로 돌아갈 때 사용하는 것으로, list.asp 파일에서 목록을 선택한 위치로 다시 돌아갈 수 있게 해주는 값입니다.

그림 4-18

세부 정보보기 화면



보기 화면은 리스트 화면에서 게시물의 제목을 클릭했을 때 게시물의 내용을 볼 수 있는 화면을 말합니다. [예제 4-3]은 gobyPDS 폴더에 있는 view.asp의 소스입니다.

예제 4-3

gobyPDS\view.asp

```

1  <%@ Language=VBScript %>
2  <html>
3  <head>
4  <title>Untitled Document</title>
5  <meta http-equiv="Content-Type" content="text/html; charset=euc-kr">
6  <!--#include file=../dbconn.inc -->
7  <LINK rel="stylesheet" type="text/css" href="gobyPDS.css">
8  </head>
9  <%
10 forum_id=Request.QueryString("forum_id")
11 %>
12 <body bgcolor="#FFFFFF">
13   <table border="0" cellspacing="0" cellpadding="0" align=center>
14     <tr align="left" valign="top">
15       <td>
16         <!--#include file=includeFiles/view.asp-->
17       </td>
18     </tr>
19   </table>
20 </body>
21 </html>

```

이 소스를 보면 앞에서 살펴본 gobyPDS\list.asp 파일과 유사한 것을 알 수 있습니다. 이 파일 역시 페이지 레이아웃 역할을 하는 파일입니다.

6행에서 dbconn.inc 파일이 삽입되었고, 10행에서는 forum_id 변수가 선언되었습니다. 이 라인은 Request 객체를 사용하여 list.asp 파일에서 전송되는 forum_id 매개변수 값을 다시 forum_id 변수에 저장하고 있습니다. 이제 forum_id 값이 게시판을 구별해주는 값이라는 것을 확실히 인식했으리라 생각합니다.

16행에는 includeFiles 폴더의 view.asp 파일이 삽입되었습니다. 그렇다면 핵심적인 코드는 includeFiles 폴더의 view.asp 파일 내에 있다고 할 수 있습니다. 이 코드를 자세히 살펴 보도록 하겠습니다.

예제 4-4

includeFiles/view.asp

[예제 4-4]는 includeFiles/view.asp 파일의 소스입니다.

```

1 <%
2 no=Request.QueryString("no")
3 curpage=Request.QueryString("page")
4 forum_id=Request.QueryString("forum_id")
5 strSQL="SELECT thc, thcf, tableWidth FROM tbl_GobyPDSAdmin WHERE
   forum_id='"&forum_id&'"
6 set rs=conn.execute(strSQL)
7 if rs.EOF then Response.Redirect "includefiles/error.asp?code=notCreate
   Board"
8 tableWidth=rs("tableWidth")
9 thc=rs("thc")
10 thcf=rs("thcf")
11 set rs=nothing
12 strSQL="SELECT * FROM tbl_GobyPDS WHERE pk_id = "&no
13 set rs=conn.execute(strSQL)
14 group_id=rs("group_id")
15 content=replace(rs("content"),vbCrLf,"<br>")
16 if isNull(rs("filesize")) then
17 filesize = 0
18 else
19 filesize = Round((CInt(rs("filesize"))/1024)*10)/10&"KB"
20 end if
21 %>
22 <table border="0" width="<%=tableWidth%>" cellpadding="2">
23 <tr bgcolor=#<%=thc%>>
24 <td colspan=6 align="middle" height=25><b><font size="2" color="#<%=
   thcf%>"> <%= rs("subject")%></font> </b></td>
25 </tr>
26 <tr height="20">
27 <td> &nbsp; 작성자 </td>
28 <td> &nbsp; <%=rs("writer")%> </td>
29 <td> &nbsp; 전자우편 </td>
30 <td> &nbsp; <a href="mailto:<%=rs("email")%>"><%=rs("email")%></a> </td>
31 </tr><tr height="20">
32 <td> &nbsp; IP </td>
33 <td> &nbsp; <%=rs("ip")%> </td>
34 <td> &nbsp; 홈페이지 </td>
35 <td> &nbsp; <a href="<%=rs("homepage")%>" target="_blank"><%=rs
   ("homepage")%> </a> </td>

```

```

37 </tr><tr height="20">
38<td> &nbsp;&nbsp;&nbsp;&파일명 </td>
39<td> &nbsp;&nbsp;&<a href="includeFiles/down.asp?pk_id=<%=no%>&filename=
      <%=rs("filename")%>&forum_id=<%=forum_id%>"><%=rs("filename")%>
      </a>( <%=filesize%>) </td>
40<td> &nbsp;&nbsp;&<조회수 &nbsp;&nbsp;&&<%=rs("readno")%> </td>
41<td> &nbsp;&nbsp;&<다운수 &nbsp;&nbsp;&&<%=rs("downno")%> </td>
42 </tr><tr>
43<td colspan="4" bgcolor=#F4F9EB style="BORDER-BOTTOM: 1px dashed;
   BORDER-LEFT: 1px dashed; BORDER-RIGHT: 1px dashed; BORDER-TOP: 1px
   dashed; PADDING-LEFT: 10px; PADDING-RIGHT: 10px; PADDING-TOP: 10px;
   PADDING-BOTTOM: 10px">
44<%= content%></td></tr>
45 </table>
46 <table border=0 width=<%=tableWidth%>>
47 <tr>
48<td align="center">
49 [&nbsp;&nbsp;&<a href="list.asp?forum_id=<%=forum_id%>&page=<%=curPage%>">리
   스트 </a>
50 &nbsp;&nbsp;&|&nbsp;&nbsp;&<a href="javascript:history.back()"> 뒤로 </a>
51 &nbsp;&nbsp;&|&nbsp;&nbsp;&<a href="reply.asp?no=<%= no%>&amp;forum_id=<%=forum_id %>
   &amp;page=<%=curPage%>"> 답변 </a>
52 &nbsp;&nbsp;&|&nbsp;&nbsp;&<a href="includefiles/passwdCheck.asp?no=<%=no%>
   &amp;forum_id=<%=forum_id%>&amp;job=edit&amp;page=<%=curPage%>"> 수정
   </a>
53 &nbsp;&nbsp;&|&nbsp;&nbsp;&<a href="includefiles/passwdCheck.asp?no=<%=no%> &amp;
   forum_id=<%=forum_id%>&amp;job=del&amp;page=<%=curPage%>"> 삭제 </a>
   &nbsp;&nbsp;&;]
54 </td></tr>
55 </table>
56 </form>
57 <%
58 set rs=nothing
59 strSQL="UPDATE tbl_GobyPDS SET readno = readno + 1 WHERE pk_id = "&no
60 conn.execute(strSQL)
61 strSQL="SELECT * FROM tbl_GobyPDS WHERE forum_id = '"& forum_id &'""
62 strSQL=strSQL&" and group_id = "& group_id &" order by group_id desc,
   group_order asc"
63 set re=conn.execute(strSQL)
64 if not re.eof then

```

```
65 %>
66 <table border=0 width=<%=tableWidth%>>
67 <tr>
68 <td align="center"><font size="2"><b>관계시물</b></font> </td>
69 </tr>
70 </table>
71 <table border="1" width="<%=tableWidth%" cellspacing="0" cellpadding="
    "2" bordercolor="#ffffff" bordercolorlight="#000000">
72 <tr>
73 <%
74 do while not re.EOF
75 blank=""
76 for cnt=0 to Cint(re("msglevel"))
77     blank=blank&"&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&";
78 next
79 subject=re("subject")
80 If Len(subject) > 25 Then subject=Mid(subject,1,25)&"..."
81 subject = replace(subject, "\"", "&quot;");
82 %>
83 <td align="left">
84 &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~
85 <a href="view.asp?no=<%=re("pk_id")%>&amp;page=<%=curPage%>
    &amp;forum_id=<%=forum_id%>">
86 <% if Cint(re("msglevel"))>0 then %>
87 <img SRC="images/re.gif" border="0" WIDTH="26" HEIGHT="12">
88 <% end if %>
89 <%=subject%> </a>
90 <td align="center" width="80">
91 <% if re("email")="" then %>
92 <%=re("writer")%>
93 <% else %>
94 <a href="mailto:<%=re("email")%>"> <%=re("writer")%> </a>
95 <% end if %>
96 </td>
97 <td align="center" width="80"> <%=left(re("writedate"),10)%> </td>
98 <td align="center" width="30"> <%=re("readno")%> </td>
99 </tr>
100 <%
101 re.MoveNext
102 loop
```

```

103 end if
104 set re=nothing
105 %>
107 </table>

```

소스를 자세히 분석해 보겠습니다.

```

1 <%
2 no=Request.QueryString("no")
3 curpage=Request.QueryString("page")
4 forum_id=Request.QueryString("forum_id")

```

2~4행은 list.asp로부터 get 방식으로 전송받은 데이터 값을 해당 변수에 저장합니다. [예제 3-2] list.asp의 102행을 보면 제목(subject)을 클릭할 때, no, page, forum_id의 값이 다음과 같이 get 방식으로 하이퍼링크되는 것을 알 수 있습니다.

```

102 str=str+"&nbsp;<a href='view.asp?no="&rs("pk_id")"&page="&curPage"&
    forum_id="&forum_id&"' title="& content "&">"& subject "&"</a>"

```

```

5 strSQL="SELECT thc, thcf, tableWidth FROM tbl_GobyPDSAdmin WHERE
    forum_id='"&forum_id&"'"
6 set rs=conn.execute(strSQL)
7 if rs.EOF then Response.Redirect "includefiles/error.asp?code=notCreate
    Board"
8 tableWidth=rs("tableWidth")
9 thc=rs("thc")
10 thcf=rs("thcf")
11 set rs=nothing

```

5~11행은 데이터베이스의 tbl_GobyPDSAdmin 테이블로부터 thc, thcf, tableWidth 필드의 값을 가져오는데, 데이터를 가져오는 조건은 WHERE forum_id="&forum_id&"가 됩니다. 이 조건문은 4행의 forum_id 변수에 저장된 값과 데이터베이스의 forum_id 필드의 값이 같은 경우의 레코드만 가져오게 합니다.

6행은 rs라는 인스턴스를 생성합니다. 이 인스턴스(객체)는 conn 인스턴스에 execute 메소드를 사용하여 strSQL을 실행합니다. 여기서 strSQL을 실행한다는 것은 5행을 실행하는 것입니다.

7행은 if 조건문으로, rs.EOF는 전체 레코드의 마지막까지 읽었을 때 일치하는 값이 없는 경우를 의미합니다. 다시 말해, 매개 변수 forum_id의 값으로 만들어진 게시판이 없다는 뜻입니다. 게시판이 존재하지 않으면 includefiles/error.asp?code=notCreateBoard로 하이퍼링크됩니다. 조건식으로 사용된 EOF는 End Of File의 약자로, 데이터가 존재하지 않아서 마지막 레코드셋을 읽어 들였다면 true가 되고, 마지막이 아니면 false가 됩니다. 다음 그림은 forum_id의 값이 tbl_GobyPDSAdmin 테이블에 없을 때 이동되는 페이지입니다.

그림 4-19

forum_id 값이 테이블에 없을 때 이동하는 페이지



8~10행은 rs 인스턴스를 사용해 레코드의 필드값을 불러와서 각 변수(tableWidth, thc, thcf)에 저장합니다.

11행은 생성한 rs 인스턴스를 소멸시킵니다.

```

12 strSQL="SELECT * FROM tbl_GobyPDS WHERE pk_id = "&no
13 set rs=conn.execute(strSQL)
14 group_id=rs("group_id")
15 content=replace(rs("content"),vbCrLf,"<br>")
16 if isNull(rs("filesize")) then
17 filesize = 0
18 else
19 filesize = Round((Clng(rs("filesize"))/1024)*10)/10&"KB"
20 end if
21 %>

```

12행은 5번과 마찬가지로 데이터베이스로부터 필요한 데이터를 불러오기 위한 실행 구문입니다. 자세히 설명하자면, pk_id 값이 list.asp 파일에서 넘어온 값(no)과 일치하는 데이터의 레코드를 tbl_GobyPDS 테이블에서 불러오는 코드입니다. 13행은 6행처럼 rs 인스턴스를 생성합니다.

14번과 15번은 레코드의 필드값을 변수에 저장합니다. 단, 15번의 경우 replace 함수를 사용하여 content 필드값에 있는 vbCrLf를
 태그로 바꾸어 저장하게 됩니다. 여기서 vbCrLf는 키보드의 엔터 값으로 생각하면 됩니다. 즉, 글을 입력할 때 줄 바꿈을 하기 위해 엔터 키를 누르면 vbCrLf로 인식하게 되고, 실제로 줄이 바뀌지도록 HTML 태그인
로 바꾸어 주는 것입니다.

16~ 20행은 파일의 사이즈를 지정합니다. 먼저 16번은 if 조건문으로, isNull 함수를 사용해 파일 사이즈의 크기가 Null이면, 즉 파일 사이즈의 크기가 존재하지 않으면 17번을 실행하라는 의미입니다.



CLng 함수와 Round 함수

<CLng() 함수>

CLng 함수는 주어진 식을 long 형식으로 바꾸어 줍니다. 즉, 표시할 수 있는 숫자의 범위를 확대하는 것입니다. 기본 형식은 다음과 같습니다.

CLng(식)

<Round() 함수>

Round 함수는 주어진 수를 반올림 합니다. 기본 형식은 다음과 같습니다.

Round(수)

19행은 파일 사이즈가 존재할 경우 ClnG 함수에 의해 rs("filesize")를 long 형식으로 변환한 후, 1024로 나누고 다시 10을 곱합니다. 그런 다음 Round 함수를 사용해 반올림하고, 다시 10으로 나누어 filesize 변수에 저장합니다. 19행의 식은 byte로 저장된 파일 사이즈를 KByte로 바꿔 소수점 첫째자리까지 표시하기 위한 것입니다.

```

22 <table border="0" width="<%=tableWidth%>" cellspacing="0" cellpadding="2">
23 <tr bgcolor="#<%=thc%>>
24 <td colspan=6 align="middle" height=25><b><font size="2" color="#<%=
   thcf%>"> <%= rs("subject")%></font> </b></td>
26 </tr>
27 <tr height="20">
28 <td >&nbsp; 작성자 </td>
29 <td >&nbsp; <%=rs("writer")%> </td>
30 <td > &nbsp; 전자우편 </td>
31 <td> &nbsp; <a href="mailto:<%=rs("email")%>"><%=rs("email")%></a> </td>
32 </tr><tr height="20">
33 <td > &nbsp; IP </td>
34 <td > &nbsp; <%=rs("ip")%> </td>
35 <td > &nbsp; 홈페이지 </td>
36 <td > &nbsp; <a href="<%=rs("homepage")%>" target="_blank"><%=rs
   ("homepage")%> </a> </td>
37 </tr><tr height="20">
38 <td> &nbsp; 파일명 </td>
39 <td> &nbsp; <a href="includeFiles/down.asp?pk_id=<%=no%>&filename=
   <%=rs("filename")%>&forum_id=<%=forum_id%>"><%=rs("filename")%>
   </a>( <%=filesize%>) </td>
40 <td> &nbsp; 조회수 &nbsp; &nbsp; <%=rs("readno")%> </td>
41 <td> &nbsp; 다운수 &nbsp; &nbsp; <%=rs("downno")%> </td>
42 </tr><tr>
43 <td colspan="4" bgcolor=#F4F9EB style="BORDER-BOTTOM: 1px dashed;
   BORDER-LEFT: 1px dashed; BORDER-RIGHT: 1px dashed; BORDER-TOP: 1px
   dashed; PADDING-LEFT: 10px; PADDING-RIGHT: 10px; PADDING-TOP: 10px;
   PADDING-BOTTOM: 10px">
44 <%= content%></td></tr>
45 </table>
46 <table border=0 width=<%=tableWidth%>>
47 <tr>
48 <td align="center">

```

```

49 [ &nbsp;   <a href="list.asp?forum_id=<%=forum_id%>&page=<%=curPage%>"> 리
스트 </a>
50 &nbsp;   |&nbsp;   <a href="javascript:history.back()"> 뒤로 </a>
51 &nbsp;   |&nbsp;   <a href="reply.asp?no=<%= no%>&forum_id=<%=forum_id%>
>&page=<%=curPage%>"> 답변 </a>
52 &nbsp;   |&nbsp;   <a href="includefiles/passwdCheck.asp?no=<%=no%>
&forum_id=<%=forum_id%>&job=edit&page=<%=curPage%>"> 수정
</a>
53 &nbsp;   |&nbsp;   <a href="includefiles/passwdCheck.asp?no=<%=no%> &
forum_id=<%=forum_id%>&job=del&page=<%=curPage%>"> 삭제 </a>
&nbsp;   ]&nbsp;   
54 </td></tr>
55 </table>
56 </form>

```

이 부분은 일반적인 HTML 문서입니다. 여기서 굵게 표시된 코드는 ASP 코드가 삽입된 부분입니다. 즉, 변수에 저장된 값이 대입된다고 생각하면 됩니다. 그리고 43행은 CSS를 사용해 <td> 태그에 속성을 설정한 것입니다.

```
57 <%  
58 set rs=nothing  
59 strSQL="UPDATE tbl_GobyPDS SET readno = readno + 1 WHERE pk_id = "&no  
60 conn.execute(strSQL)  
61 strSQL="SELECT * FROM tbl_GobyPDS WHERE forum_id = '"& forum_id &'"'  
62 strSQL=strSQL+"& and group_id = "& group_id &" order by group_id  
desc,group_order asc"  
63 set re=conn.execute(strSQL)  
64 if not re.eof then  
65 %>
```

57행부터 ASP 코드가 다시 시작됩니다. 58행은 rs 인스턴스를 소멸시킵니다.

59행은 readno 값을 1 증가시키고 새로운 값을 업데이트시키는 구문입니다. 즉, tbl_GobyPDS 테이블의 필드인 pk_id 값과 리스트 파일에서 전송된 no 변수값이 일치하는 레코드의 readno 값을 1 증가시키는 구문입니다.



SQL 문법 : Update

Update는 이미 존재하는 데이터를 수정합니다. 즉, Insert 구문에 의해 데이터베이스에 이미 입력된 데이터를 다시 업데이트하는 것입니다. 기본 형식은 다음과 같습니다.

```
Update 테이블명 set 컬럼명1=데이터1, 컬럼명2=데이터2, ...
[Where 조건문]
```

기본 형식을 보면 데이터1, 데이터2,... 들이 각각의 컬럼명(필드명)1, 컬럼명2,...에 다시 입력되는 것을 알 수 있습니다.

60행은 strSQL(59행)을 실행하여 실제 데이터베이스의 값을 업데이트합니다.

61행부터는 게시물의 내용과 관련된 부분이 아닙니다. 여기서부터는 게시물의 내용 밑에 출력되는 답변 목록을 표시합니다. 즉, 누군가 어떤 게시물에 답변을 달면, 게시물 보기 화면에서 게시물 내용과 답변 리스트가 함께 출력되도록 구성된 것입니다. 다음 그림의 하단 리스트 부분이 앞으로 진행될 소스가 출력되는 부분입니다.

그림 4-20

게시물 내용과 답변
리스트가 함께
출력된 화면



61행과 62행은 현재 게시물의 forum_id와 group_id 값을 데이터베이스의 forum_id와 group_id 필드값과 비교해서 동일한 값을 가지는 레코드를 불러옵니다.

63행은 strSQL을 실행하여 re 인스턴스를 생성합니다. 66행은 if 조건문으로, re.eof가

Loop While 조건

76번은 for~next 반복문으로, 0부터 Cint(re("msglevel")) 값까지 반복합니다. 여기서 Cint() 함수는 괄호 안의 값을 정수로 변환하는 역할을 합니다. 이 값은 답변 리스트를 출력할 때 들여쓰기로 활용됩니다. cnt는 단순한 변수로, 마지막에 rs("msglevel") 값이 저장됩니다.



For ~ Next 반복문

For ~ Next 문은 입력한 시작점(=시작)에서 끝나는 점까지 For~Next 사이의 실행문을 반복 실행시킵니다. 한 번 실행될 때마다 자동으로 시작점이 1씩 증가됩니다. 이 증가값을 조정하려면 step을 사용하여 step 뒤에 증가값을 설정하면 됩니다. 예를 들어, step 2라고 설정하면 반복문이 한 번 실행될 때마다 시작점이 2씩 증가됩니다. step 값이 생략되면 기본값으로 1씩 증가됩니다. 기본 형식은 다음과 같습니다.

```
For I = 시작 To 끝 [Step 증가값]
    실행문
Next
```

```
79 subject=re("subject")
80 If Len(subject) > 25 Then subject=Mid(subject,1,25)&"..."
81 subject = replace(subject, "\"", "&quot;")
82 %>
```

80행은 If 조건문으로, Len 함수를 사용하여 subject 필드의 길이값을 구한 다음, 이 길이값을 비교하여 그 값이 25글자 이상이면, 25글자만 출력하고 나머지는 ...로 대체합니다. 여기서 Mid(subject, 1, 25) 함수는 subject 필드값을 첫 번째 자리부터 25번째 자리까지만 표시되게 하는 역할을 합니다.

81행은 replace 함수를 사용하여 subject의 문자 중에 "가 있으면 HTML에서 제대로 표시되도록 "로 대체한 다음, subject 변수에 다시 저장합니다.

```

83 <td align="left">
84 &nbsp;  <%=blank%>
85 <a href="view.asp?no=<%=re("pk_id")%>&amp;page=<%=curPage%>
   &amp;forum_id=<%=forum_id%>">
86 <% if Cint(re("msglevel"))>0 then %>
87 <img SRC="images/re.gif" border="0" WIDTH="26" HEIGHT="12">
88 <% end if %>
89 <%=subject%> </a>
90 <td align="center" width="80">
91 <% if re("email")="" then %>
92 <%=re("writer")%>
93 <% else %>
94 <a href="mailto:<%=re("email")%>"> <%=re("writer")%> </a>
95 <% end if %>
96 </td>
97 <td align="center" width="80"> <%=left(re("writedate"),10)%> </td>
98 <td align="center" width="30"> <%=re("readno")%> </td>
99 </tr>

```

83~99행은 목록에 출력되는 내용을 설정합니다.

84행은 blank 변수 값을 웹페이지에 표시하는 것으로, 75행 또는 77번에서 설정된 값이 표시됩니다. 즉, 들여쓰기 공백이 표시되는 것입니다.

85행은 출력되는 제목(subject) 값에 링크를 설정합니다. 제목을 클릭했을 때 view.asp 파일로 다시 돌아가 게시물이 출력되도록 no, page, forum_id 매개변수 값을 함께 설정하고 있습니다.

86~88행은 답변 목록 이미지를 표시하기 위한 코드입니다. 86행에 있는 Cint(re("msglevel"))>0은 msglevel 필드값이 0보다 클 때를 뜻하는 것으로, 게시물에 답변을 달 때 답변글의 msglevel 값은 항상 0보다 크게 됩니다. 즉, 87행에서 답변 글임을 표시하는 이미지를 제목 앞에 표시하게 됩니다.

89행은 링크가 설정된 제목이 웹페이지에 표시되게 합니다. 91~95행은 작성자를 표시하는 부분으로, email의 존재 유무에 따라 작성자가 표시되는 방법을 달리하고 있습니다. 91번은 자료실에 글을 등록할 때 이메일을 함께 등록하지 않아서 re("email") 값이 없을 경우이며, 92행을 실행시켜 작성자의 이름만 표시되게 합니다.

93행은 if 조건문을 만족하지 않을 때, 즉 re("email") 값이 존재하는 경우를 뜻하는 것으로, 94행을 실행시킵니다. 94행은 작성자의 이름에 링크를 설정해서 표시하는 것으로, 작성자 이름을 클릭했을 때 자동으로 메일 프로그램이 실행되게 설정되어 있습니다. 97행은 작성날짜를 표시하는 것으로 left 함수를 사용하여 좌측 첫 번째 자리부터 시작해서 10번째 자리까지만 출력되게 하고 있습니다.



left 함수

left 함수는 문자열을 지정한 수만큼 왼쪽부터 시작하여 반환합니다. 그 기본형식은 다음과 같습니다.

left(문자열, 지정한 수)

98행은 조회수를 표시하기 위한 코드입니다.

```
100 <%
101 re.MoveNext
102 loop
103 end if
104 set re=nothing
105 %>
106 </table>
```

101행은 현재 사용된 데이터베이스의 레코드에서 다음 레코드로 이동시키는 구문입니다. 예를 들어, 영희와 순이가 하나씩 글을 쓰게 되면 두 개의 글, 즉 레코드가 존재하게 됩니다. 첫 번째 반복문에서 영희가 쓴 글이 표시되고 난 후, 101행의 구문이 존재하지 않으면 반복문에 의해 계속해서 영희의 글이 표시됩니다. 또한 반복문이 종료되지 않아서 무한 루프에 빠지게 됩니다. 따라서 이 구문은 매우 중요하다고 할 수 있습니다. 반복 구문이 종료되려면 마지막 레코드가 실행되어야 하는데, 순이가 작성한 마지막 레코드가 실행되지 않았으므로 반복문이 계속해서 실행되어 무한 루프에 빠지는 것입니다. 즉, 101행은 반복문이 한 번 실행될 때마다 다음 레코드로 이동시켜 모든 레코드를 실행시킬 수 있도록 해주는 것입니다.

102행은 do 반복문의 실행 영역을 표시하는 것으로, 반복문이 끝나지 않았을 경우 74행으로 리턴합니다.

103행은 64행의 조건문이 끝났음을 표시합니다. 마지막으로 104행은 re 인스턴스를 소멸시킵니다.

다음 그림은 위 코드가 실행된 결과입니다.

이 게시판 소스 코드입니다.	디디컬	2002-08-23	11
이 게시판 소스 코드입니다.	메니아	2002-08-23	0
얼씨대 해야줘....^^	화이팅	2002-08-23	0

05

입력 양식과 데이터베이스 저장 프로그래밍

입력 양식을 보여주는 frm_write.asp 파일과 데이터베이스를 저장하는 write.asp 파일의 소스를 분석합니다.

입력 양식과 데이터베이스 저장 프로그램은 다음과 같이 구성되어 있습니다.

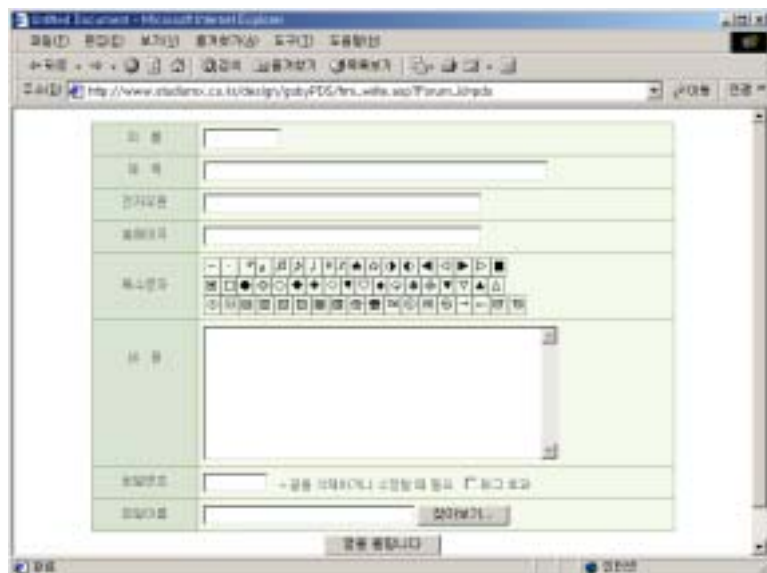
레이아웃 파일	gobyPDS\frm_write.asp, gobyPDS\includeFiles\frm_write.asp,
데이터베이스 저장 프로그램 파일	gobyPDS\includeFiles\write.asp
실행	http://도메인, 혹은 계정 URL/gobyPDS/frm_write.asp?Forum_id=pds

입력양식 만들기: frm_write.asp

이 프로그램을 실행하기 위해서는 우선 하나의 매개변수가 필요합니다. 어떤 게시판에서 글을 작성하는지 알 수 있어야 그 게시판에 작성된 글을 등록할 수 있기 때문입니다. 따라서 게시판을 구별해 줄 수 있는 매개변수인 forum_id를 사용하여 접속합니다. 다음 그림은 frm_write.asp 파일이 실행된 그림입니다.

그림 4-21

내용 입력 양식



이 화면에서 [글을 올립니다] 버튼을 클릭하면 includeFiles 폴더의 write.asp 파일이 실행되어 입력된 데이터를 데이터베이스에 저장합니다.

입력 양식이란 데이터베이스에 데이터를 입력하기 위한 양식으로, 리스트 화면에서 이미지, 텍스트, 버튼으로 링크되는 경우가 많습니다. 리스트 화면의 오른쪽 상단의 [글 올리기] 텍스트를 클릭하면 gobyPDS 폴더의 frm_write.asp 파일로 링크됩니다.

그림 4-22

[글 올리기] 버튼이
삽입된 리스트 화면



[예제 4-5]는 gobyPDS\frm_write.asp의 소스 코드입니다.

예제 4-5

gobyPDS\frm_write.asp

```
1  <%@ Language=VBScript %>
2  <html>
3  <head>
4  <title>Untitled Document</title>
5  <meta http-equiv="Content-Type" content="text/html; charset=euc-kr">
6  <!--#include file=../dbconn.inc -->
7  <LINK rel="stylesheet" type="text/css" href="gobyPDS.css">
8  <script language="Javascript">
9  function send(form) {
10 if (form.writer.value=="")
11   {alert("\n이름을 입력해 주세요.");document.form.writer.focus();return;}
12 if (form.subject.value=="")
13   {alert("\n글제목을 입력해 주세요.");document.form.subject.focus();return;}
14 if (form.content.value=="")
15   {alert("\n내용을 입력해 주세요.");document.form.content.focus();return;}
16 if (form.passwd.value=="")
17   {alert("\n패스워드를 입력해 주세요.");document.form.passwd.focus();return;}
17 form.submit();
18 }
19 function check_email()
20 {
```



```
21 str=document.form.email.value
22 if (str != "") {
23     if(str.indexOf('@') == -1) {
24         alert("올바른 e-mail형식이 아닙니다.\r\n      다시입력하세요.")
25         document.form.email.focus()
26 document.form.email.select()
27 }
28 }
29 return;
30 }
32 </script>
33 </head>
34 <body bgcolor="#FFFFFF">
35     <table border="0" cellspacing="0" cellpadding="0" align=center>
36         <tr align="left" valign="top">
37             <td>
38                 <!--#include file=includeFiles/frm_write.asp-->
39             </td>
40         </tr>
41     </table>
42 </body>
43 </html>
```

이 파일은 보기 화면(view.asp)과 구조가 비슷합니다. 6행에서 dbconn.inc 파일이 삽입되고 있으며, 38행에는 includeFiles 폴더의 frm_write.asp 파일을 삽입하고 있습니다.

세부적으로 하나씩 분석해 보도록 하겠습니다.

```
9  function send(form) {
10  if (form.writer.value=="")
11      {alert("\n이름을 입력해 주세요.");document.form.writer.focus();return;}
12  if (form.subject.value=="")
13      {alert("\n글제목을 입력해 주세요.");document.form.subject.focus();return;}
14  if (form.content.value=="")
15      {alert("\n내용을 입력해 주세요.");document.form.content.focus();return;}
16  if (form.passwd.value=="")
17      {alert("\n패스워드를 입력해 주세요.");document.form.passwd.focus();return;}
18  form.submit();
```

```

18 }
19 function check_email()
20 {
21   str=document.form.email.value
22   if (str != "") {
23     if(str.indexOf('@') == -1) {
24       alert("올바른 e-mail형식이 아닙니다.\r\n      다시입력하세요.")
25       document.form.email.focus()
26       document.form.email.select()
27     }
28   }
29   return;
30 }

```

9~30행은 자바스크립트 소스입니다. 이 소스는 두 개의 자바스크립트 함수로 구성되어 있습니다. 첫 번째, send() 함수는 9~18행인데, includeFiles\frm_write.asp 파일에서 폼 버튼을 클릭했을 때 이름, 제목, 내용, 패스워드에 입력된 데이터를 검사한 다음, 입력된 데이터가 없으면 데이터를 입력하라는 메시지가 표시되게 합니다.

19~30행이 두 번째 함수인 check_email 함수입니다. 이 함수의 역할은 includeFiles\frm_write.asp 파일에서 입력하는 email의 값이 올바른 이메일인지 아닌지 검사하는 것입니다. 올바른 이메일이 아니면 올바르게 못한 이메일이라는 메시지가 표시됩니다.

다음으로 입력양식 레이아웃을 표시하는 includeFiles\frm_write.asp 파일의 소스를 알아보겠습니다.

예제 4-6

includeFiles\
frm_write.asp

```

1  <%
2  forum_id=Request.QueryString("forum_id")
3  strSQL="SELECT tableWidth FROM tbl_GobyPDSAdmin WHERE
      forum_id='"&forum_id&'"
4  set rs=conn.execute(strSQL)
5  if rs.EOF then Response.Redirect "includefiles/error.asp?code=notCreateBoard"
6  tableWidth=rs("tableWidth")
7  set rs=nothing
8  if not isEmpty(session("member")) then
9    writer=session("member")(0)

```

[illegible]

[illegible]

```
78 <td colspan=2 align=middle><input type="button" value="글을 올립니다"
    onclick="send(this.form)">
79 </td>
80 </tr>
81 </table>
82 </form>
```

소스를 세부적으로 하나씩 분석해 보겠습니다.

```
1 <%
2 forum_id=Request.QueryString("forum_id")
3 strSQL="SELECT tableWidth FROM tbl_GobyPDSAdmin WHERE
    forum_id='"&forum_id&'"
4 set rs=conn.execute(strSQL)
5 if rs.EOF then Response.Redirect "includefiles/error.asp?code=notCreateBoard"
6 tableWidth=rs("tableWidth")
7 set rs=nothing
```

2행은 리스트 파일로부터 게시판을 구별해 주는 변수값인 forum_id의 값을 Request 인스턴스(객체)를 사용하여 forum_id 변수에 저장합니다.

3행은 2행에서 저장된 forum_id 값과 tbl_gobyPDSAdmin 테이블의 forum_id 필드값을 비교하여 동일한 값을 갖는 레코드의 tableWidth 필드값을 불러옵니다.

4행은 rs 인스턴스를 생성하는 구문으로, conn 인스턴스의 execute 메소드를 사용하여 3행을 실행시키게 됩니다.

5행은 3행에서 실행된 구문에 의해 불러지는 데이터가 존재하지 않을 경우 에러를 보여주는 페이지로 이동하게 합니다.

6행은 동일한 테이블 레이아웃을 유지하기 위해 tbl_GobyPDSAdmin 테이블로부터 tableWidth 값을 가져옵니다. 마지막으로 7행은 rs 인스턴스(객체)를 소멸시킵니다.

```
8  if not isEmpty(session("member")) then
9  writer=session("member")(0)
10  passwd=session("member")(2)
11  email=session("member")(3)
12  homepage=session("member")(4)
13  readonly = "readonly"
14  end if
15  %>
```

8~12행은 세션값과 관련된 부분입니다. 이 세션 값은 게시판과 함께 운영되는 회원관리 프로그램에서 로그인을 해야 생성됩니다. 세션 값은 회원 정보가 저장되는 테이블에서 일부 정보를 불러와 폼 입력 양식 일부에 자동으로 입력되게 함으로써 유저에게 편리함을 주기 위해 필요한 것입니다. 현재 자료실 소스에는 회원관리 프로그램이 포함되어 있지 않기 때문에 이 소스에 따른 효과를 실제로 확인할 수는 없습니다.

Session 변수

Session은 클라이언트(웹브라우저)가 사이트에 접속하면 자동으로 발생합니다. 세션은 사이트에 접속하는 각 클라이언트(웹브라우저)마다 독립적으로 발생합니다. 즉, 클라이언트가 사이트에 접속하여 발생하는 세션은 고유의 값을 갖게 되는 것입니다. 이 세션은 클라이언트(웹브라우저)가 사이트를 떠나거나 웹브라우저를 종료하면 자동으로 사라집니다. 파일과 파일 사이에서 동일한 변수값을 사용하기 위해 변수값을 전송해야 하는 일반 변수(forum_id 등)와 달리, Session 변수는 한 번 선언하면 클라이언트가 사이트에 접속해 활동하고 있는 한, 어떤 파일에서든지 변수값 전달없이 바로 사용할 수 있습니다. 예를 들어, 세션변수 session ("member")=10이라고 선언하면 다른 파일에서 session("member") 변수를 사용했을 때 자동으로 10으로 설정됩니다.

```
16 <form action="includeFiles/write.asp" method="post" name="form" encType
    = "multipart/form-data">

.....,

82 </form>
```

16~82행은 폼 태그로 구성되어 있습니다. 즉, 폼 양식이라는 뜻입니다. 해당 폼 필드에 입력된 데이터는 includeFiles 폴더에 있는 write.asp 파일로 전송됩니다. 이 때 전송 방법은 post입니다.

예제 4-7

includefiles\write.asp

데이터베이스 저장: write.asp

[예제 4-7]은 데이터베이스를 저장하는 write.asp 파일의 코드입니다. 이 코드가 웹에서 제대로 실행되려면 사이트갤럭시업로드(SiteGalaxyUpload) 컴퍼넌트가 웹서버로 사용되는 시스템에 설치되어 있어야 합니다.

```

1  <%@ Language=VBScript %>
2  <!--#include file="../../dbconn.inc"-->
3  <%
4  server.ScriptTimeout = 5000
5  Function CheckWord(CheckValue)
6  CheckValue = replace(CheckValue, "&" , "&amp;")
7  CheckValue = replace(CheckValue, "<" , "&lt;")
8  CheckValue = replace(CheckValue, ">" , "&gt;")
9  CheckValue = replace(CheckValue, "'", "'")
10 CheckWord = CheckValue
11 End Function
12 Function CheckHTML(CheckValue)
13 CheckValue = replace(CheckValue, "&" , "&amp;")
14 CheckValue = replace(CheckValue, "'", "'")
15 CheckHTML = CheckValue
16 End Function
17 Set UploadForm = Server.CreateObject("SiteGalaxyUpload.Form")
18 forum_id=trim(UploadForm("forum_id"))
19 passwd=trim(UploadForm("passwd"))
20 writer=trim(UploadForm("writer"))
21 email=trim(UploadForm("email"))
22 homepage=trim(UploadForm("homepage"))
23 upfile=UploadForm("upfile")
24 ip=trim(UploadForm("ip"))
25 tag=UploadForm("tag")
26 if tag then
27 content=CheckHTML(UploadForm("content"))
28 subject=CheckHTML(UploadForm("subject"))
29 tag=1
30 else
31 content=CheckWord(UploadForm("content"))
32 subject=CheckWord(UploadForm("subject"))
33     tag=0

```



```
34 end if
35 strSQL="SELECT MAX(group_id) FROM tbl_gobyPDS WHERE forum_id =
    '&forum_id&' "
36 set MaxGroup_ID = conn.execute(strSQL)
37 if isNull(MaxGroup_ID(0)) then Group_id = 0 else Group_id =
    Cint(MaxGroup_ID(0))+1
38 set MaxGroup_ID=nothing
39 group_order=0
40 Msglevel=0
41 readno=0
42 if Len(upfile) > 0 then
43 set fc = CreateObject("Scripting.FileSystemObject")
44 folder="uploadfiles\"
45 storedir=server.mappath("../")&folder
46 attach_file=UploadForm("upfile").filepath
47 filesize=UploadForm("upfile").size
48 filename = mid(attach_file, instrrev(attach_file,"")+1) '
49 strname=mid(filename, 1, instrrev(filename,".")-1)
50 strext=mid(filename,instrrev(filename,".")+1)
51 strfilename=storedir & filename
52 fexist = true
53 count=0
54 do while fexist
55
56     if(fc.fileexists(strfilename)) then
57         count=count+1
58         filename=strname&"-"&count&"."&strext
59         strfilename=storedir & filename
60     else
61         fexist=false
62     end if
63 loop
64 UploadForm("upfile").saveas strfilename
65 set fc=nothing
66 else
67 filesize=0
68 filename=""
69 end if
70 if content ="" then content=" "
71 strSQL="INSERT INTO tbl_GobyPDS(forum_id, group_id, group_order,
    msglevel, subject, "
```

```

72 strSQL=strSQL&"writer, email, readno,downno, passwd, homepage, filename,
   filesize, ip, tag, content ) VALUES ('"
73 strSQL=strSQL& forum_id&"', "
74 strSQL=strSQL&group_id&" , "
75 strSQL=strSQL&group_order    &" , "
76 strSQL=strSQL&msglevel&" ,'"
77 strSQL=strSQL&subject&"', '"
78 strSQL=strSQL&writer&"', '"
79 strSQL=strSQL&email&"', "
80 strSQL=strSQL&"0,0,'"
81 strSQL=strSQL&passwd&"', '"
82 strSQL=strSQL&homepage&"', '"
83 strSQL=strSQL&filename&"', '"
84 strSQL=strSQL&filesize&"', '"
85 strSQL=strSQL&ip&"', "
86 strSQL=strSQL&tag&" ,'"
87 strSQL=strSQL&content&"')"
88 conn.execute(strSQL)
89 set conn=nothing
90 Response.Redirect "../list.asp?forum_id="&forum_id
91 %>

```

소스를 하나씩 자세히 분석해 보겠습니다.

```

1  <%@ Language=VBScript %>
2  <!--#include file="../../../dbconn.inc"-->
3  <%
4  server.ScriptTimeout = 5000

```

1행은 비주얼베이직 스크립트로 작성된 문서임을 표시합니다. 2행은 dbconn.inc 파일을 삽입하는 코드입니다.

4행은 server 객체의 ScriptTimeout 속성을 사용하여 프로그램 수행시 무한 루프에 들어갔을 때 빠져나올 시간을 설정합니다. 단위는 초입니다.

```
5 Function CheckWord(CheckValue)
6 CheckValue = replace(CheckValue, "&" , "&amp;")
7 CheckValue = replace(CheckValue, "<" , "&lt;")
8 CheckValue = replace(CheckValue, ">" , "&gt;")
9 CheckValue = replace(CheckValue, "'" , "&quot;")
10 CheckWord = CheckValue
11 End Function
```

5~11행은 VBscript의 Function 프로시저를 사용한 구문으로, 프로시저 안에서 문장을 수행한 후 최종 결과 값을 리턴하는 경우에 사용합니다. 즉, 6~9행까지 실행한 다음, 10행에서 최종값(리턴값)을 CheckWord에 저장합니다. 여기서 중요한 것은 마지막 반환 값(CheckValue)을 자신의 함수명(CheckWord)에 저장한다는 점입니다.

5행은 함수명으로 CheckWord를 선언하는 것으로, 뒤에서 설명하는 31, 32행에서 이 함수를 호출하게 됩니다.

6~9행은 replace 함수를 사용해 CheckValue 값에 들어있는 &, <, >, ' 을 &, <, >, " 로 바꾸어 CheckValue에 다시 저장하는 것입니다. 즉, <, >을 특수문자로 바꿔 주므로써 HTML 태그가 사용되지 못하도록 하는 것입니다. 예를 들어, CheckValue 값에 텍스트를 강조하기 위한 태그가 사용되었다면, 이 태그는 로 바뀌어 태그의 역할을 할 수 없게 됩니다. 웹브라우저에는 일반 텍스트처럼 라고 표시됩니다. ' 을 " 으로 전환하는 이유는 DB에 저장할 때 ' 이 포함된 문자열을 그냥 저장하면, 에러가 발생하기 때문입니다. DB에서는 문자를 인식하는 방법으로 ' 를 사용하여 '문자열' 로 표시합니다. 이 때 문자열에 ' 이 포함되어 있으면 ' 가 두 개 표시되므로 SQL 구문이 제대로 인식하지 못해 에러가 발생합니다. 다음 그림은 9행이 생략된 경우, DB에 폼 입력 내용을 저장할 때 에러가 출력되는 화면입니다.

그림 4-23

includefiles\write.asp에서 9행이 생략되었을 때 나타나는 에리 메시지



10행은 함수명에 CheckValue 값을 저장하는 것으로, 이 CheckValue 값이 리턴값이 됩니다. Function 프로시저에서는 마지막 리턴값을 함수명에 저장합니다.

11행에서 Function 프로시저가 종료됩니다.



Function 프로시저

Function 프로시저는 호출한 프로시저로부터 인수값을 받아 Function 프로시저안의 실행구문에 의해 실행한 다음 그 결과값을 다시 호출한 프로시저에 반환합니다. 그 기본형식은 다음과 같습니다.

Function 함수명(인수 리스트)

[실행구문]

함수명=리턴값

End Function

```

12 Function CheckHTML(CheckValue)
13 CheckValue = replace(CheckValue, "&" , "&amp;")
14 CheckValue = replace(CheckValue, "'", "'")
15 CheckHTML = CheckValue
16 End Function
    
```

12행부터 16행 역시 function 프로시저가 사용되는 구문입니다. 13, 14행에서 실행 구문을 실행한 다음, 그 결과값을 15행에서 함수명에 저장합니다. 그런 다음, 저장된 결과값을 호출한 곳으로 리턴하게 됩니다.

12행은 함수명으로 CheckHTML을 선언합니다. 이 프로시저는 27행과 28행에서 호출하게 됩니다.

13~14행은 replace 함수를 사용하여 & , ' 을 각각 & , " 로 변환한 다음 CheckValue에 저장합니다.

15행은 함수명에 리턴값(CheckValue)을 저장합니다.

16행에서 프로시저가 종료됩니다.

```

17 Set UploadForm = Server.CreateObject("SiteGalaxyUpload.Form")
18 forum_id=trim(UploadForm("forum_id"))
19 passwd=trim(UploadForm("passwd"))
20 writer=trim(UploadForm("writer"))
21 email=trim(UploadForm("email"))
22 homepage=trim(UploadForm("homepage"))
23 upfile=UploadForm("upfile")
24 ip=trim(UploadForm("ip"))
25 tag=UploadForm("tag")

```

여기서부터는 파일을 업로드하기 위한 업로드 컴포넌트가 필요한 부분입니다. 예제에서는 사이트갤럭시 컴퍼넌트를 사용하고 있는데, 이 컴퍼넌트는 내장된 컴포넌트가 아닙니다. 즉, 필요에 따라 설치하는 것이므로, 컴포넌트를 사용하려면 컴포넌트에서 정의된 객체의 속성이나 메소드를 사용해야 합니다.

간단히 요약하자면, 일반적으로 규정된 객체의 메소드나 속성을 사용하는 것이 아니라 업로드 컴포넌트에서 정의된 객체의 메소드와 속성을 사용해야 한다는 것입니다. 따라서 컴포넌트마다 사용하는 메소드나 속성이 달라지게 됩니다. 이 메소드나 속성을 사용하려면 컴포넌트를 제공하는 사이트에서 필요한 매뉴얼을 보고 독자가 직접 학습하도록 합니다. 예제에서는 사이트갤럭시 컴포넌트를 사용하여 업로드하는 방법에 대해서만 설명합니다.

17행은 사이트갤럭시 컴포넌트를 사용하여 UploadForm 인스턴스를 생성합니다. 일반적으로 폼 양식 파일에서 텍스트만 전송받는 것이라면 업로드 컴포넌트를 사용하지 않고, 간단히 request 인스턴스(객체)를 사용하여 전송받을 수 있습니다. 여기서 업로드 컴포넌트를 사용해 UploadForm 인스턴스를 생성하는 이유는 텍스트만이 아닌 실제 파일 전체를 전송받기 위해서입니다. 즉, request 인스턴스는 단순히 텍스트만 전송받을 수 있기 때문에 실제 파일을 업로드하기 위한 업로드 컴퍼넌트가 필요한 것이고, 이 컴퍼넌트의 사용방법에 따라 파일을 전송받는 인스턴스(UploadForm)를 따로 생성하는 것입니다. 사이트갤럭시 컴퍼넌트를 사용하여 이러한 업로드 인스턴스를 생성하기 위한 소스는 다음과 같습니다.

```
Set 인스턴스명(UploadForm) =Server.CreateObject("SiteGalaxyUpload.Form")
```

18~25행은 get 방식으로 전송된 데이터를 지정된 변수에 저장합니다. 그런데

request.querystring을 사용하지 않고 Uploadform 인스턴스로 데이터를 받고 있습니다. 앞에서 설명했으니 이유를 알고 있으리라 생각합니다(마찬가지로 텍스트만이 아니라 파일을 업로드하기 위해서 사이트갤럭시 컴포넌트를 사용하였기 때문입니다. 따라서 이 컴퍼넌트의 사용방법에 따라 새롭게 생성한 UploadForm 인스턴스를 이용하여 데이터를 받고 있는 것입니다).

또한 trim 함수를 사용해 전송받는 문자열의 앞·뒤 공백을 제거한 다음 변수에 저장하고 있는데, 그 이유는 글쓰기 파일(frm_write.asp)에서 글을 입력할 때 유저의 실수로 입력하는 글의 앞이나 뒤에 공백이 삽입될 경우가 있기 때문입니다. 이러한 공백은 예상치 못한 오류를 발생시킬 수 있는데, 예를 들어 비밀번호를 '1234'로 입력하고 자신도 모르게 스페이스 키를 누르면 비밀번호(1234) 뒤에 공백이 삽입됩니다. 이 상태에서 데이터베이스에 글을 저장하면 비밀번호는'1234공백'이 됩니다. 즉, 데이터베이스는 공백도 하나의 문자로 인식하여 비밀번호를 '1234공백'으로 저장하므로, 등록된 글을 삭제하기 위해 비밀번호를 입력할 때 틀린 비밀번호라고 표시됩니다. 따라서 문자열을 데이터베이스에 저장하기 전에, trim 함수를 사용하여 전송받은 문자열의 앞·뒤 공백을 제거하는 것입니다.



Trim 함수

Trim 함수는 문자열의 앞·뒤 공백을 제거한 다음 그 결과값을 반환합니다. 기본형식은 다음과 같습니다.

Trim(문자열)

```
26 if tag then
27 content=CheckHTML(UploadForm("content"))
28 subject=CheckHTML(UploadForm("subject"))
29 tag=1
30 else
31 content=CheckWord(UploadForm("content"))
32 subject=CheckWord(UploadForm("subject"))
33 tag=0
34 end if
```

26~34행은 글쓰기 파일(frm_write.asp)에서 태그효과란의 체크 여부에 따른 조건문을

표시하고 있습니다. 글쓰기 파일에서 '태그 효과' 항목에 체크하면 27~29행이 실행되고, 그렇지 않으면, 31~33행까지 실행됩니다.

26행은 조건문으로, tag 값이 존재할 경우 그 이하 구문을 실행시키는 소스입니다.

27행은 CheckHTML 함수를 호출하여 리턴되는 결과값을 content 변수에 저장합니다. 좀 더 구체적으로 살펴보면, 12번의 CheckHTML(CheckValue) 함수를 호출하는데, 여기서 호출이란 CheckHTML(CheckValue) 함수 구문을 실행시킨다는 뜻입니다. 27행에서 12번의 CheckHTML(CheckValue) 함수를 호출하는데 있어 CheckValue 인수값으로 UploadForm("content") 값을 전달하게 됩니다. 그러면 전달받은 UploadForm("content")값은 CheckValue 값이 되고 13번~15행까지 실행된 다음, 그 결과값을 다시 27번의 CheckHTML로 전송합니다. 이 값이 다시 content 변수에 저장되는 것입니다.

28행은 CheckHTML 함수를 호출하여 리턴되는 결과값을 subject 변수에 저장하는 것으로, 그 과정은 27번과 동일합니다.

29행은 tag 변수의 값을 1로 설정합니다. 이 값은 등록되는 글이 HTML 효과를 주는 글임을 구별해 주는 값이 됩니다.

31행은 tag 값이 존재하지 않을 경우, 즉 HTML 효과를 주지 않을 때 실행되는 구문으로, CheckWord 함수를 호출하여 리턴되는 결과값을 content 변수에 저장합니다. 이 과정은 27번과 동일한데 호출하는 함수가 CheckWord이므로 5행의 구문이 실행됩니다.

32행은 CheckWord 함수를 호출하여 리턴되는 결과값을 subject 변수에 저장하는 것으로, 그 과정은 31행과 동일합니다.

32행은 tag 변수의 값을 0으로 설정합니다. 이 값은 등록되는 글이 HTML 효과를 주지 않는 글임을 구별해 주는 값이 됩니다.

```

35 strSQL="SELECT MAX(group_id) FROM tbl_gobyPDS WHERE forum_id =
    '&forum_id&'
36 set MaxGroup_ID = conn.execute(strSQL)
37 if isNull(MaxGroup_ID(0)) then Group_id = 0 else Group_id =
    Cint(MaxGroup_ID(0))+1
38 set MaxGroup_ID=nothing
    
```

35행은 tbl_gobyPDS 테이블에서 group_id 값 중 최대값을 불러오는 select 구문입니다. 36행은 실제 쿼리문을 실행시켜 값을 가져온 후, 그 값을 MaxGroup_ID 인스턴스에 저장 합니다.

37행은 MaxGroup_ID 값이 없으면 그 값을 0으로 설정하고, 값이 있으면 그 값에 1을 더하게 됩니다. MaxGroup_ID 값이 없다는 것은 아직 등록된 게시물이 없다는 것을 의미합니다.

38행에서 MaxGroup_ID 객체를 소멸시킵니다.

```
39 group_order=0
40 Msglevel=0
41 readno=0
```

39~41행은 group_order, Msglevel, readno 변수 값을 0으로 설정합니다. 즉, 처음 등록하는 글은 group_order, Msglevel, readno 값이 0으로 초기화되는 것입니다. group_order 값은 같은 그룹 내에서 표시되는 목록의 순서를 나타내며, Maglevel 값은 표시되는 목록의 들여쓰는 공백 크기를 설정합니다. readno 값은 조회수를 의미합니다.

```
42 if Len(upfile) > 0 then
43 set fc = CreateObject("Scripting.FileSystemObject")
44 folder="\uploadfiles\"
45 storedir=server.mappath("../")&folder
46 attach_file=UploadForm("upfile").filepath
47 filesize=UploadForm("upfile").size
48 filename = mid(attach_file, instrrev(attach_file,"\")+1) '
49 strname=mid(filename, 1, instrrev(filename,".")-1)
50 strext=mid(filename,instrrev(filename,".")+1)
51 strfilename=storedir & filename
52 fexist = true
53 count=0
54 do while fexist
55
56     if(fc.fileexists(strfilename)) then
57         count=count+1
```



```

58         filename=strname&"-"&count&". "&strex
59         strfilename=storedir & filename
60     else
61         fexist=false
62     end if
63 loop
64 UploadForm("upfile").saveas strfilename
65 set fc=nothing
66 else
67 filesize=0
68 filename=""
69 end if
70 if content ="" then content=" "

```

42~69행은 파일을 업로드하는 코드입니다.

42행은 if 조건문으로, 업로드된 파일의 길이를 검사하여 0보다 크면 then 이하의 코드를 수행합니다. 즉, 문자의 수가 0보다 크다는 것은 업로드된 파일이 존재한다는 것을 의미합니다. 이 한 줄에 의해 파일 첨부 여부를 간단히 알 수 있는데, 만약 첨부된 파일이 없으면 67행부터 실행됩니다.

43행은 내장된 FSO 컴퍼넌트를 사용하여 fc라는 인스턴스를 생성합니다. 56행에서 이 인스턴스를 사용하여 업로드하는 파일과 동일한 이름의 파일이 있는지 검사하게 됩니다.



FileSystemObject 컴퍼넌트(FSO)

FileSystemObject 컴퍼넌트는 컴퓨터 파일 시스템을 제어할 수 있게 해줍니다. 즉, 파일과 폴더의 복사, 삭제, 이동, 존재 유·무 확인 등에 사용됩니다.

44행은 파일이 저장될 폴더의 경로를 변수에 저장합니다.

45행은 server 객체의 mappath 메소드를 사용하여 파일이 저장될 실제 경로를 추출한 후, folder 변수 값을 더합니다. 이 경로가 서버에 파일이 저장되는 절대 경로입니다.

46행은 업로드될 파일의 경로를 attach_file 변수에 저장합니다.

47행은 파일의 용량을 filesize 변수에 저장합니다.

48행은 instrrev 함수를 사용하여 경로명 뒤에서 첫 번째 \ 까지 찾고, 이에 해당하는 수를 반환한 후, 1을 더합니다. 이 값은 다시 mid 함수에 대입되어 파일 이름만 추출하게 됩니다.



InStrRev() 함수

InStrRev() 함수는 문자열의 끝에서부터 차례대로 검색하여 발견된 곳의 자리값을 반환합니다. 기본 형식은 다음과 같습니다.

InStrRev(검색 문자열, 찾을 문자열)

49행은 파일 이름에서 파일명만 분리해내는 과정으로, strname 변수에는 파일명만 저장됩니다.

50행은 파일 이름에서 확장자만 분리한 다음, strext 변수에 분리된 확장자를 저장합니다. 파일명과 확장자 분리 작업을 하는 이유는 나중에 설명됩니다.

51행은 저장 경로(storedir)와 파일명을 합칩니다.

52행과 53행은 do while ... loop에서 사용될 변수를 지정합니다.

54행은 fexist 변수가 참일 경우 63행까지의 구문을 반복 실행시킵니다.

56행은 fileexists 메소드를 사용하여 파일을 업로드할 때 서버에 동일한 파일이 이미 존재하면 then 이하 구문을 실행시킵니다. 57행에서 count 값에 1을 더하고 58행에서 파일명에 - count를 붙여 파일 이름을 만들어 냅니다. 이 파일 이름을 58행에서 전체 경로로 저장합니다. 예를 들어, ddcom.jpg 파일이 서버에 존재한다면, 동일한 파일을 업로드할 때는 ddcom-1.jpg 파일명으로 바뀌어 업로드됩니다.

64행은 업로드 파일을 저장하는 코드입니다.

66~69행은 게시물에 첨부된 파일이 없을 때 수행되는 코드로, filesize와 filename의 값을 각각 0과 null로 지정합니다.

70행은 26행 이하에서 선언한 content 변수 값이 null일 경우, content 변수값으로 빈 여백을 설정합니다.

```

71 strSQL="INSERT INTO tbl_GobyPDS(forum_id, group_id, group_order,
    msglevel, subject, "
72 strSQL=strSQL&"writer, email, readno,downno, passwd, homepage, filename,
    filesize, ip, tag, content ) VALUES ("
73 strSQL=strSQL&forum_id&"', "
74 strSQL=strSQL&group_id&" , "
75 strSQL=strSQL&group_order&" , "
76 strSQL=strSQL&msglevel&" ,'"
77 strSQL=strSQL&subject&"', '"
78 strSQL=strSQL&writer&"', '"
79 strSQL=strSQL&email&"', "
80 strSQL=strSQL&"0,0, '"
81 strSQL=strSQL&passwd&"', '"
82 strSQL=strSQL&homepage&"', '"
83 strSQL=strSQL&filename&"', '"
84 strSQL=strSQL&filesize&"', '"
85 strSQL=strSQL&ip&"', "
86 strSQL=strSQL&tag&" , '"
87 strSQL=strSQL&content&"')"
88 conn.execute(strSQL)
89 set conn=nothing
90 Response.Redirect "../list.asp?forum_id="&forum_id
91 %>

```

71~91행은 데이터를 데이터베이스에 입력하는 코드입니다. 코드의 내용일 길면 예제처럼 분리해서 코딩합니다. 즉, strSQL 변수에 계속해서 분리된 내용이 추가되는 것입니다. 이 라인에서는 ' (홀 따옴표)의 사용을 주의해서 살펴봐야 합니다. 저장 프로그램을 코딩할 때 ' (홀 따옴표)의 사용을 실수하는 경우가 많으니 제대로 이해하고 있어야 합니다. Insert 구문에서 저장되는 데이터가 문자일 경우, ' (홀 따옴표)로 데이터를 돌려주게 되고, 숫자이면 ' (홀 따옴표)가 필요 없습니다.

72행은 데이터베이스에 데이터를 저장하는 Insert 구문으로, insert into 테이블명(필드명 1, 필드명2,) values (값1, 값2, ...)의 형식을 가집니다.



SQL 문법 : Insert

Insert 구문은 데이터베이스에 데이터를 저장하기 위한 구문입니다. 기본 형식은 다음과 같습니다.

Insert into 테이블명([필드명1, 필드명2,....]) **Values** (데이터1, 데이터2,.....)

Insert 구문에서는 저장되는 데이터의 값이 문자일 경우와 숫자일 경우를 구분해줘야 합니다. 저장되는 데이터의 값이 문자이면, '(홀 따옴표)로 둘러줘야 하고, 숫자면 '(홀 따옴표)가 필요 없습니다.

89행은 conn 인스턴스를 소멸하고, 90행에서 list.asp 파일로 이동합니다.

06

데이터베이스 수정 프로그래밍

데이터베이스에 입력된 내용을 수정하는 수정 양식을 만들어 봅니다.

내용을 수정하려면 게시물을 하나 선택하고, [수정] 메뉴를 클릭하면 됩니다.

그림 4-24

게시물을 선택하고
[수정] 메뉴를 클릭



그러면 다음 그림처럼 수정 권한 여부를 판별하는 비밀번호 입력란이 표시됩니다. 게시물을 수정할 수 있는 권한은 관리자와 글을 쓴 게시자에게만 있습니다.

그림 4-25

수정 권한을 주는
비밀번호 입력 화면



여기서 [확인] 버튼을 클릭하면, 다음 그림처럼 입력양식 폼이 표시되는데, 데이터베이스의 필드 값을 그대로 불러옵니다.

그림 4-26

내용을 수정할 수 있는
입력양식 폼



게시물의 내용을 수정하고 [글을 수정합니다] 버튼을 클릭하면 includesFiles 폴더의 update.asp 파일로 하이퍼링크됩니다.

수정양식에서 사용되는 파일은 모두 여섯 개로, 데이터베이스 연결을 위한 dbconn.inc 파일, 비밀번호 확인을 위한 passwdCheck.asp, passwdCheck2.asp 파일, 수정양식인 edit.asp 파일과 이 파일 내에 포함되어 있는 includeFiles 폴더의 edit.asp 파일, 그리고 수정된 내용을 데이터베이스에 업데이트해주는 update.asp 파일입니다.

순서대로 소스를 분석해 보겠습니다.

암호 입력 폼 : passwdCheck.asp

파일 경로

admintools/passwd
Check.asp

```

1  <%@ Language=VBScript %>
2  <HTML>
3  <HEAD>
4  </HEAD>
5  <BODY onload="document.form.passwd.focus();">
6  <P>&nbsp;&nbsp;&nbsp;</P>
7  <P>&nbsp;&nbsp;&nbsp;</P>
8  <P>&nbsp;&nbsp;&nbsp;</P>
9  <P>&nbsp;&nbsp;&nbsp;</P>
10 <form method=post action="passwdCheck2.asp" name=form>
11 <input type=hidden name=forum_id value="<%=Request.QueryString
    ("forum_id")%>">
12 <input type=hidden name=page value="<%=Request.QueryString("page")%>">
13 <input type=hidden name=no value="<%=Request.QueryString("no")%>">
14 <input type=hidden name=job value="<%=Request.QueryString("job")%>">
15 <TABLE border=1 borderColor=black cellPadding=5 cellSpacing=0
    width="250" align=center>
16 <TR>
17 <TD align=middle><STRONG><FONT face=굴림 size=2 color=dimgray>
    비밀번호</FONT></STRONG></TD>
18 <TD align=middle><INPUT id=password1 name=passwd type=password size=8
    maxlength=8></TD></TR>
19 </TABLE>
20 <p align=center><INPUT type="submit" value="확인" id=submit1 name=submit1>
    </p></form>
21 </BODY>
22 </HTML>

```

passwdCheck.asp 파일은 위의 그림처럼 비밀번호를 입력받는 폼입니다. 입력받은 비밀번호는 10행의 action 값에 의해 passwdCheck2.asp 파일로 post 방식으로 전송합니다. get 방식으로 전송하면 비밀번호가 URL에 전송되기 때문에 post 방식을 이용하는 것입니다.

11~14행은 숨김 필드를 사용하여 view.asp 파일에서 넘어온 정보를 입력되는 비밀번호와 함께 전송하기 위한 코드입니다. 참고로 view.asp 파일에서 넘어오는 정보는 다음과 같습니다. forum_id는 게시판을 구별하기 위한 정보이고, page는 게시판에서 게시물의 위치(페이지)를 나타내는 정보이며, no는 데이터베이스에서 게시물을 구별해 주기 위한 정보입니다. 마지막으로 job은 수정인지 삭제인지 구별하기 위한 정보입니다. job이 필요한 이유는 수정이나 삭제의 비밀번호 확인이 모두 하나의 파일(passwdCheck2.asp)에서 실행되기 때문에 이 수정과 삭제를 구별해 주기 위한 매개변수(job)가 필요한 것입니다.

암호 인증 프로그램 : passwdCheck2.asp

파일 경로

admintools/passwd
Check2.asp

```

1  <%@ Language=VBScript %>
2  <!--#include file="../../../dbconn.inc"-->
3  <%
4  forum_id=Request.Form("forum_id")
5  page=Request.Form("page")
6  pk_id=Request.form("no")
7  passwd=Request.Form("passwd")
8  job=Request.Form("job")
9  strSQL="SELECT passwd FROM tbl_GobyPDS WHERE pk_id = "&pk_id
10 set rs=conn.execute(strSQL)
11 passwd2=rs(0)
12 set rs=nothing
13 strSQL="SELECT adminpasswd FROM tbl_GobyPDSAdmin WHERE
   forum_id='"&forum_id&"'"
14 set rs=conn.execute(strSQL)
15 passwd3=rs(0)

```

```

16 set rs=nothing
17 set conn=nothing
18 if trim(passwd)=trim(passwd2) or trim(passwd)=trim(passwd3) then
19     if job="edit" then
20         Response.Redirect "../edit.asp?forum_id=" & forum_id & "&no=" & pk_id & "&
            page=" & page
21     else
22         Response.Redirect "del.asp?forum_id=" & forum_id & "&no=" & pk_id & "&
            &page=" & page
23     end if
24 else
25     Response.Redirect "error.asp?code=passwd"
26 end if
%>

```

2행은 데이터베이스 연결을 위해 dbconn.inc 파일을 삽입합니다. 4~8행은 post 방식으로 전달된 폼 필드의 값을 변수에 저장합니다.

9행은 pk_id 변수에 저장된 폼 필드 값과 tbl_GobyPDS 테이블의 pk_id 값이 일치하는 데이터에서 passwd 필드의 값만 불러오는 구문입니다.

10행은 실제 passwd 값을 불러 오도록 9행의 select 구문을 실행시킵니다.

11행은 tbl_GobyPDS 테이블에서 불러온 passwd 값을 passwd2 변수에 저장합니다. 여기서 rs(0)은 배열을 의미하는 것으로, 10행이 실행되어 데이터베이스로부터 불러오는 첫 번째 데이터를 말합니다. 즉, rs(0)은 rs("passwd")를 의미합니다. 예를 들어, 불러오는 데이터가 여러 개 있다면 rs(0), rs(1), ... 과 같은 방식으로 데이터를 호출할 수 있습니다.

12행은 rs 인스턴스를 소멸시킵니다. 13행은 forum_id 변수에 저장된 폼 필드 값과 tbl_GobyPDSAdmin 테이블의 forum_id 값이 일치하는 데이터에서 adminpasswd 필드의 값만 불러오는 구문입니다. 이 구문이 추가된 이유는 관리자 패스워드를 불러오기 위함입니다. 관리자는 등록된 모든 글의 수정이나 삭제가 가능해야 합니다.

14행은 실제 adminpasswd 값을 불러 오도록 13행의 select 구문을 실행시킵니다.

15행은 11행과 동일한 방식입니다. rs(0) 배열을 사용하여 adminpasswd의 값을 passwd3 변수에 저장합니다.

18행은 입력된 비밀번호가 글을 등록할 때 지정했던 비밀번호(데이터베이스의 비밀번호)와 동일한지 검사하며, 또한 관리자 비밀번호인지도 검사합니다. 비밀번호가 일치하거나 관리자 비밀번호이면 19행으로 이동하고, 그렇지 않으면 24행으로 이동합니다. 여기서 trim() 함수를 사용한 이유는 혹시 모를 에러를 방지하기 위함입니다. 앞에서도 설명된 것처럼, 올바른 비밀번호를 입력했는데 실수로 여백을 추가했을 경우, 이 함수를 사용하여 비밀번호 좌·우 여백을 삭제하므로 에러가 발생하지 않습니다.

19행은 전송된 폼 필드의 값 중에서 job의 값이 edit이면 then 이하 구문을 실행시킵니다. 20행은 job의 값이 edit일 경우, edit.asp 파일로 이동시킵니다.

21행은 job의 값이 edit가 아닐 경우에 실행되는 조건문으로, 여기서는 job의 값이 del일 경우를 의미합니다. 이 값은 삭제 프로그램을 설명할 때 다시 한 번 언급됩니다.

25행은 비밀번호가 틀렸을 경우, error.asp로 하이퍼링크시킵니다.

수정 양식

먼저 edit.asp 파일부터 알아보겠습니다.

edit.asp

```

1  <%@ Language=VBScript %>
2  <html>
3  <head>
4  <title>Untitled Document</title>
5  <meta http-equiv="Content-Type" content="text/html; charset=euc-kr">
6  <!--#include file=../dbconn.inc -->
7  <LINK rel="stylesheet" type="text/css" href="gobyPDS.css">
8  <script language="Javascript">
9  function send(form) {
10 if (form.writer.value=="") {alert("\n이름을 입력해 주세요.");
    document.form.writer.focus();return;}
11 if (form.subject.value=="") {alert("\n글제목을 입력해 주세요.");
    document.form.subject.focus();return;}
12 if (form.content.value=="") {alert("\n내용을 입력해 주세요.");
    document.form.content.focus();return;}

```

파일 경로

gobyPDS/edit.asp

```

13  if (form.passwd.value=="") {alert("\n패스워드를 입력해 주세요.");
    document.form.passwd.focus();return;}
14  form.submit();
15  }
16  </script>
17  </head>
18  <body bgcolor="white" link="#000080" onLoad="document.form.writer.
    focus()">
19    <table width="720" border="0" cellspacing="0" cellpadding="0" align=
      center>
20      <tr align="left" valign="top">
21        <td>
22          <!--#include file=includeFiles/edit.asp-->
23        </td>
24      </tr>
25    </table>
26  </body>
27  </html>

```

edit.asp 파일은 frm_write.asp 파일과 유사한 코드입니다. 이 부분에 대한 이해가 부족하면 frm_write.asp 파일을 참조하기 바랍니다.

6행은 데이터베이스 연결을 위하여 연결 파일인 dbconn.inc 파일을 현재의 문서에 삽입합니다.

7행은 문서에 스타일 시트를 적용하기 위해 외부 스타일 시트인 gobyPDS.css 파일로의 링크를 설정하고 있습니다. 즉, 미리 작성된 스타일 시트를 연결해서 현재 문서에 적용하는 것입니다.

8~16행은 자바스크립트 소스로, 수정 폼 필드에 내용이 입력되지 않을 경우 내용을 입력하라는 메시지가 표시되도록 합니다. 좀 더 구체적으로 말하면 includeFiles/edit.asp 파일에 있는 폼 필드에 유저가 내용을 입력하지 않아 폼 필드 값이 존재하지 않는 경우에 메시지가 출력되도록 합니다.

22행은 실제로 폼 양식이 들어있는 파일인 includeFiles/edit.asp를 현재 문서에 삽입합니다.

계속해서 실제 폼 양식이 들어있는 파일에 대해 알아보겠습니다.

파일 경로

includeFiles\edit.asp

includeFiles\edit.asp

[illegible]


```

63      <td bgColor=#F4F9EB><input type="password" size="8" maxLength="8"
        name="passwd" style=" border:1 solid #808080" value="<%=rs("passwd")
        %>" readonly>
64      <% if rs("tag") then%>
65      <INPUT name=tag type=checkbox value=1 checked>태그 효과</FONT></td>
66      <% else %>
67      <INPUT name=tag type=checkbox value=1>태그 효과</FONT></td>
68      <% end if %>
69  </tr>
70  <tr align="left" valign="middle" bgcolor=#F4F9EB>
71      <td align="middle" bgcolor=#D8E3D2> 파일이름 </td>
72      <td bgColor=#F4F9EB><INPUT type="file" name="upfile" size=30>
        </td>
73</tr>
74 </table>
75<tr align="left" valign="middle">
76      <td align=center colspan=2 ><br>
77      <input type="button" value="글을 수정합니다" onclick="send(this.form)">
78      </td>
79      </tr>
80      </td>
81      </tr>
92 </table>
93 </form>
94 <% set conn=nothing %>

```

includeFiles\edit.asp 파일 역시 includeFiles\frm_write.asp와 코드가 비슷하니 참조하기 바랍니다. 세부적으로 설명해 보겠습니다.

2~4행은 이전 페이지에서 넘어오는 정보를 Request 객체를 사용하여 전송받는 것입니다. 5행은 데이터베이스의 tbl_GobyPDSAdmin 테이블에서 tablewidth 값을 불러오는 것으로, 테이블의 가로 크기를 의미합니다. 이 값을 불러오는 이유는 수정 파일의 테이블 크기를 다른 파일과 동일하게 함으로써 전체적인 레이아웃을 맞춰주기 위함입니다. 여기서 불러오는 tablewidth 값은 21행에서 테이블의 width 속성값으로 사용됩니다.

6행은 5행을 실제로 실행시키는 구문으로, 이 구문이 곧 rs 인스턴스가 됩니다. 이렇게 생성된 인스턴스는 데이터베이스에서 데이터를 불러올 수 있게 됩니다.

7행은 6행에서 생성한 rs 인스턴스를 사용하여 데이터를 불러올 때 데이터가 존재하지 않으면 includefiles/error.asp로 이동시키는 구문입니다. 여기서 rs.EOF는 데이터가 존재하지 않는 경우라는 것을 뜻합니다.

10행은 tbl_GobyPDS 테이블로부터 모든 데이터(*)를 불러오는 구문으로, pk_id="&no 조건이 주어졌기 때문에 이 조건에 맞는 레코드의 모든 데이터를 불러오게 됩니다. 여기서 pk_id는 데이터베이스에서 등록된 글을 구별하는 필드값이고, no는 사용자가 수정하기 위해 선택한 글의 pk_id 값이 됩니다. 따라서 사용자가 선택한 글의 데이터만 불러오게 됩니다. 11행은 10행을 실행시키는 구문으로, rs 인스턴스가 됩니다.

41~55행은 자바스크립트 소스로, 기호 문자를 표시하는 역할을 합니다. 특수문자를 직접 입력하는 대신 반복문과 배열을 사용하여 기호 문자가 표시될 수 있도록 버튼으로 만들어서 쉽게 넣을 수 있게 합니다.



수정 프로그래밍 : includeFiles\update.asp

파일 경로
includeFiles\
update.asp

```
1 <%@ Language=VBScript %>
2 <!--#include file="../../../dbconn.inc"-->
3 <%
4 Function CheckWord(CheckValue)
5 CheckValue = replace(CheckValue, "&" , "&amp;")
6 CheckValue = replace(CheckValue, "<" , "&lt;")
7 CheckValue = replace(CheckValue, ">" , "&gt;")
8 CheckValue = replace(CheckValue, "'", "'")
9 CheckWord = CheckValue
10 End Function
11 Function CheckHTML(CheckValue)
12 CheckValue = replace(CheckValue, "&" , "&amp;")
13 CheckValue = replace(CheckValue, "'", "'")
14 CheckHTML =CheckValue
15 End Function
16 Set UploadForm = Server.CreateObject("SiteGalaxyUpload.Form")
17 pk_id=trim(UploadForm("pk_id"))
```

```

18 delfilename=trim(UploadForm("delfilename"))
19 prefilesiz=trim(UploadForm("prefilesiz"))
20 forum_id=trim(UploadForm("forum_id"))
21 writer=trim(UploadForm("writer"))
22 email=trim(UploadForm("email"))
23 homepage=trim(UploadForm("homepage"))
24 upfile=UploadForm("upfile")
25 tag=UploadForm("tag")
26 if tag then
27 content=CheckHTML(UploadForm("content"))
28 subject=CheckHTML(UploadForm("subject"))
29 tag=1
30 else
31     content=CheckWord(UploadForm("content"))
32 subject=CheckWord(UploadForm("subject"))
33 tag=0
34 end if
35 if Len(upfile) > 0 then
36 set fc = CreateObject("Scripting.FileSystemObject")
37 folder="\upLoadFiles\"
38 storedir=server.mappath("../")&folder
39 if Len(delfilename) > 0 then
40     delfile=storedir&delfilename
41     fc.DeleteFile(delfile)
42 end if
43 attach_file=UploadForm("upfile").filepath
44 filesize=UploadForm("upfile").size
45 filename = mid(attach_file, instrrev(attach_file,"")+1)
46 strname=mid(filename, 1, instrrev(filename,".")-1)
47 strext=mid(filename,instrrev(filename,".")+1)
48 strfilename=storedir & filename
49 fexist = true
50 count=0
51 do while fexist
52     if(fc.fileexists(strfilename)) then
53         count=count+1
54         filename=strname&"-"&count&"."&strext
55         strfilename=storedir & filename
56     else
57         fexist=false

```

```

58     end if
59 loop
60     UploadForm("upfile").saveas strfilename
61     set fc=nothing
62 else
63     filename=delfilename
64     filesize=prefilesize
65 end if
66 strSQL="UPDATE tbl_gobyPDS SET subject = '& subject &',"
67 strSQL=strSQL&"email = '& email &',"
68 strSQL=strSQL&"homepage = '& homepage &',"
69 strSQL=strSQL&"filename = '& filename &',"
70 strSQL=strSQL&"filesize = '& filesize &',"
71 strSQL=strSQL&"tag = '& tag &',"
72 strSQL=strSQL&"content = ' "& content &' WHERE pk_id = "&pk_id
73 conn.execute(strSQL)
74 set conn=nothing
75 Response.Redirect "../list.asp?forum_id="&forum_id&"&page="&page
76 %>

```

includeFiles\update.asp 파일도 includeFiles\write.asp와 코드가 거의 유사하니 참조하기 바랍니다. 코드를 자세히 살펴보겠습니다.

4~15행은 Function 프로시저를 사용한 구문입니다. function 구문은 includeFiles\write.asp에서 사용된 구문과 동일합니다. 우선 4행의 첫 번째 함수인 CheckWord 함수가 호출되면, 5~8행을 실행하고, 9행에서 그 결과값을 함수명에 저장합니다. 그러면 이 함수를 호출한 곳으로 결과값이 리턴됩니다.

11행의 두 번째 함수 역시 마찬가지입니다. CheckHTML 함수가 호출되면 12행부터 13라인까지 실행한 다음, 14행에서 그 결과값을 함수명에 저장하게 되고, 함수를 호출한 곳으로 결과값을 리턴합니다.

16행 이하는 업로드 컴포넌트를 이용해 UploadForm 인스턴스를 생성하고, Request 인스턴스 대신 UploadForm 인스턴스를 사용해 정보를 전송받는 소스입니다. 여기서 사용된 업로드 컴포넌트는 앞에서 설명된 사이트갤럭시 컴포넌트입니다.

16행은 사이트갤럭시 컴퍼넌트를 사용하여 UploadForm 인스턴스를 생성합니다.

17~25행은 생성된 UploadForm 인스턴스를 활용하여 정보를 전송받고 있습니다. 전송 받은 정보는 trim 함수를 사용하여 좌·우 여백을 제거한 다음 변수에 저장합니다.

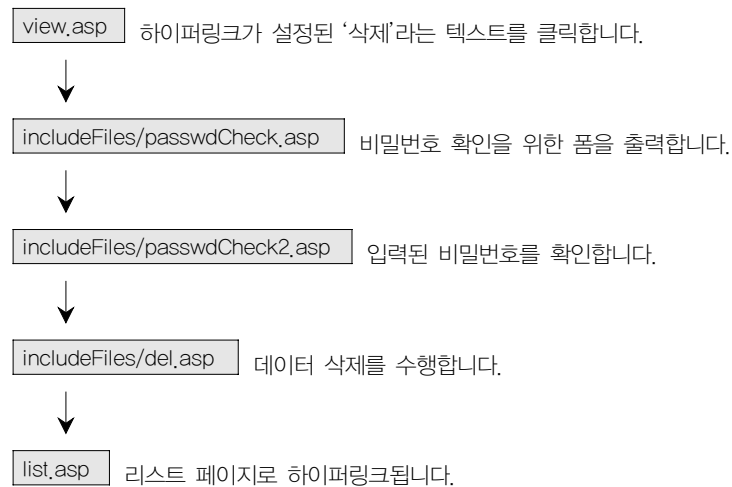
70행에서는 update 구문이 사용되고 있습니다. update 구문은 update 테이블명 set 필드이름1=필드값1, 필드이름2=필드값2, ... 의 형식을 가집니다.

06

데이터 삭제

게시판에서 데이터를 삭제하는 프로그래밍을 합니다.

이 게시판에서 데이터의 삭제는 다음과 같은 과정을 통해 이루어집니다.



include/view.asp 파일에서 '삭제' 텍스트에 하이퍼링크된 내용은 다음과 같습니다.

```
<a href="includefiles/passwdCheck.asp?no=<%=no%>&forum_id=<%=forum_id%>&job=del&page=<%=curPage%>"> 삭제 </a>&nbsp;]
```

하이퍼링크는 데이터 값을 전송하기 위해 get 방식으로 링크되었습니다. 전송되는 데이터 값에는 no(게시물의 고유 ID 값), forum_id(게시판 ID 값), job(편집이나 삭제를 구별해주는 값), page(삭제 후 되돌아올 페이지 값)이 정의되어 있습니다.

이 값들은 include/passwdCheck.asp 파일로 전달됩니다.

파일 경로

include/passwd
Check.asp

암호 입력 양식 : include/passwdCheck.asp

```

1  <%@ Language=VBScript %>
2  <HTML>
3  <HEAD>
4  </HEAD>
5  <BODY onload="document.form.passwd.focus();">
5  <P>&nbsp;</P>
6  <P>&nbsp;</P>
7  <P>&nbsp;</P>
8  <P>&nbsp;</P>
9  <form method=post action="passwdCheck2.asp" name=form>
10 <input type=hidden name=forum_id value="<%=Request.QueryString
    ("forum_id")%>">
11 <input type=hidden name=page value="<%=Request.QueryString("page")
    %>">
12 <input type=hidden name=no value="<%=Request.QueryString("no")%>">
13 <input type=hidden name=job value="<%=Request.QueryString("job")%>">
14 <TABLE border=1 borderColor=black cellPadding=5 cellSpacing=0
15   width="250" align=center>
16   <TR>
17     <TD align=middle><STRONG><FONT face=굴림
        size=2 color=dimgray>비밀번호</FONT></STRONG></TD>
18     <TD align=middle><INPUT id=password1 name=passwd type=password size=
        8 maxlength=8></TD></TR>
19   </TABLE>
20   <p align=center><INPUT type="submit" value="확인" id=submit1 name=
        submit1>
21   </p>
22   </form>
23   </BODY>
24   </HTML>

```

이 소스의 결과는 다음 그림과 같습니다.

이 문서는 일반 HTML 문서입니다. 9행에서 <form> 태그가 시작되고 있으며, 액션은 passwdCheck2.asp로 되어 있습니다. 입력받은 비밀번호를 passwdCheck2.asp 파일에 post 방식으로 전송합니다.

10~13행은 view.asp 파일에서 전송된 데이터 값들을 hidden 필드의 값으로 각각 지정합니다. 이 값 또한 비밀번호와 함께 passwdCheck2.asp 파일에 post 방식으로 전송됩니다.

전송된 값이 어떻게 표시되는지에 대한 것은 다음의 passwdCheck2.asp를 통해 알아보겠습니다.

암호 인증 프로그램 : include/passwdCheck2.asp

파일 경로

include/passwd
Check2.asp

```

1  <%@ Language=VBScript %>
2  <!--#include file="../../../dbconn.inc"-->
3  <%
4  forum_id=Request.Form("forum_id")
5  page=Request.Form("page")
6  pk_id=Request.form("no")
7  passwd=Request.Form("passwd")
8  job=Request.Form("job")
9  strSQL="SELECT passwd FROM tbl_GobyPDS WHERE pk_id = "&pk_id
10 set rs=conn.execute(strSQL)
11 passwd2=rs(0)
12 set rs=nothing
13 strSQL="SELECT adminpasswd FROM tbl_GobyPDSAdmin WHERE
    forum_id="&forum_id&""
14 set rs=conn.execute(strSQL)
15 passwd3=rs(0)
16 set rs=nothing
17 set conn=nothing
18 if trim(passwd)=trim(passwd2) or trim(passwd)=trim(passwd3) then
19     if job="edit" then
20         Response.Redirect "../edit.asp?forum_id="&forum_id&"&no="&pk_id&"
            &page="&page
21     else
22         Response.Redirect "del.asp?forum_id="&forum_id&"&no="&pk_id&"&page
            ="&page

```

```

23     end if
24     else
25         Response.Redirect "error.asp?code=passwd"
26     end if
27 %>

```

4~8행은 post 방식으로 전송된 데이터 값들을 모두 해당 변수에 저장합니다. 9행은 tbl_GobyPDS 테이블에서 pk_id 값이 일치하는 passwd 값을 가져오는 select 구문입니다. 이 라인에서 글을 올린 게시자의 비밀번호를 가져옵니다. 11행은 passwd2에 passwd 값을 저장하고 12행에서 rs 인스턴스 값을 소멸시킵니다.

같은 방법으로 13~16행은 게시판 관리자의 passwd 값을 가져오는 select 구문입니다. 이 값은 15행에서 passwd3 변수에 저장됩니다.

18행은 if 조건문으로, 게시판 관리자의 비밀번호와 게시자의 비밀번호 중에 하나만 일치하면 19행의 if 조건문을 수행합니다. 만약 일치하지 않으면 25행의 error.asp 파일로 하이퍼링크됩니다.

19행의 조건문에서 job의 값이 edit일 때와 del일 때를 구분해서 각각 다른 파일로 하이퍼링크시킵니다. 여기서는 job의 값이 del이기 때문에 del.asp 파일로 하이퍼링크됩니다. 데이터 값은 get 방식으로 전송됩니다.

다음으로 include/del.asp 파일을 알아보도록 하겠습니다.

삭제 프로그램 : include/del.asp

```

1  <%@ Language=VBScript %>
2  <!--#include file="../../../dbconn.inc"-->
3  <%
4  forum_id=Request.QueryString("forum_id")
5  pk_id=Request.QueryString("no")
6  page=Request.QueryString("page")
7  strSQL="DELETE from tbl_GobyPDS WHERE pk_id = "&pk_id
8  conn.execute(strSQL)
9  set conn=nothing
10 Response.Redirect "../list.asp?forum_id="&forum_id&"&page="&page
11 %>

```

파일 경로

include/del.asp

include/del.asp 파일은 아주 간단한 asp 코드입니다.

일단 전송된 데이터 값은 변수에 저장되고, 7행에서는 delete 구문으로 pkid 값이 일치되는 레코드를 삭제합니다.

10행은 데이터 값을 삭제한 후, list.asp 파일로 하이퍼링크시키는 구문입니다.

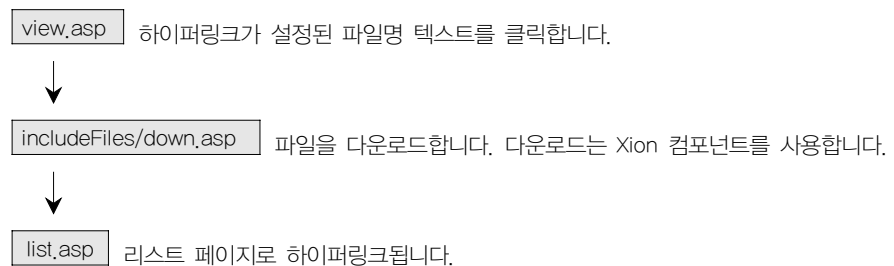
데이터 삭제에 관한 설명은 이것으로 마치겠습니다.

07

데이터 다운로드

게시판에서 데이터를 다운로드하게 해주는 프로그래밍을 합니다.

파일 다운로드 는 다음과 같은 과정을 통해 이루어집니다.



파일의 다운로드 는 asp 코드에서 Xion 업/다운로드 컴포넌트를 사용하여 이루어집니다. 먼저 view.asp에서 파일명(하이퍼링크)을 클릭함으로써 includeFiles/down.asp가 수행 됩니다.

```
....  
<a href="includeFiles/down.asp?pk_id=<%=no%>&filename=<%=rs("filename")%>  
&forum_id=<%=forum_id%>"><%=rs("filename")%> </a><%=filesize%>  
....
```

하이퍼링크를 클릭했을 때 pk_id, filename, forum_id 값이 get 방식으로 includeFiles/down.asp에 전송됩니다.

[예제 4-8]은 includeFiles/down.asp 파일의 소스 코드입니다.

예제 4-8

includeFiles/down.asp

```
1 <%@ Language=VBScript %>  
2 <!--#include file="../../dbconn.inc"-->  
3 <%  
4 no=Request.QueryString("pk_id")  
5 filename=Request.QueryString("filename")
```

```
6 forum_id=Request.QueryString("forum_id")
7 folder="../upLoadFiles/"
8 strSQL="UPDATE tbl_GobyPDS SET downno = downno + 1 WHERE pk_id = "&no
9 conn.execute(strSQL)
10 set conn=nothing
11 dim xdn
12 set xdn = Server.CreateObject("Xionsoft.XionFileDownload")
13 xdn.DownFromFile folder&filename
14 set xdn = nothing
%>
```

매우 간단한 코드지만, Xion 업/다운 컴포넌트를 모르면 사용할 수 없는 코드입니다.

먼저 view.asp 파일에서 전송된 데이터 값을 해당 변수에 저장하고, 7행에서는 folder 변수를 지정합니다.

8행은 다운로드 횟수를 데이터베이스에 갱신하기 위한 update 구문입니다.

11~13행은 파일을 다운로드 해주는 Xion 컴포넌트 구문입니다.

이것으로 다운로드에 관한 설명을 마치겠습니다.

08

답변 쓰기 프로그래밍

게시판의 등록된 글에 답변을 달았을 때 리스트가 계층적으로 표시되는 것을 계층형 게시판이라고 합니다. 계층형 게시판의 핵심은 단계적인 답변 등록이 이루어지도록 만드는 것입니다. 다소 어려운 감이 있지만, 개념을 잘 이해하면 학습에는 큰 무리가 없을 것입니다.

답변형 게시판의 전체적인 구성은 생각보다 간단합니다. 예를 들어 ‘ASP 특강’이라는 글을 작성하면 그 글에 고유 정수값을 부여하고, 글의 순서를 결정할 또 다른 정수값을 부여합니다(고유 정수값을 ID라고 하고, 또 다른 정수값을 order라고 가정하겠습니다).

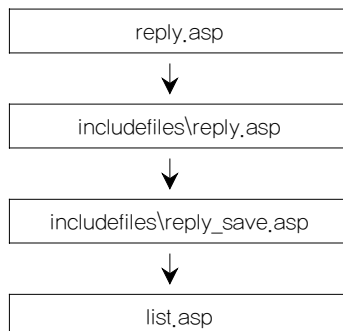
‘ASP 특강’이라는 글에 대한 답변글을 작성하면, 처음 글과 동일한 ID 값을 부여받게 되고, order 값은 1 증가됩니다. 이런 식으로 order 값에 의해 같은 ID 값을 가지는 글들이 정렬됩니다. 그런 다음, ID 값이 같은 글들을 하나의 글로 취급하여 게시판에 역순으로 정렬합니다. 즉, 원래의 글과 답변글을 하나의 글로 묶어서 게시판에 등록하는 것입니다. 이렇게 등록된 묶음은 그 안에서 order 값에 의해 정렬됩니다. 다음 그림은 답변이 첨부된 계층형 게시판입니다.

그림 4-27

계층형 게시판 화면

게시판 목록		[admin][글 등록]			
번호	제목	작성자	등록일	조회	답변
1	ASP 특강	C	3/17/18:00	0	0
2	1-강 EEE	E	3/17/18:00	1	0
3	1-강 EEE	D	3/17/18:00	2	0
4	1-강 EEE	E	3/17/18:00	0	0
5	1-강 FFF	F	3/17/18:00	0	0
6	AAA	A	3/17/18:00	0	0

답변 양식에서 파일의 진행 방향을 간단히 살펴보면 다음과 같습니다.



순서대로 소스를 분석해 보도록 하겠습니다.

답변 입력 양식

reply.asp

파일 경로

gobyPDS/reply.asp

```
1  <%@ Language=VBScript %>
2  <html>
3  <head>
4  <title>Untitled Document</title>
5  <meta http-equiv="Content-Type" content="text/html; charset=euc-kr">
6  <!--#include file=../dbconn.inc -->
7  <LINK rel="stylesheet" type="text/css" href="gobyPDS.css">
8  <script language="Javascript">
9  function send(form) {
10 if (form.writer.value=="")
11 {alert("\n이름을 입력해 주세요.");document.form.writer.focus();return;}
12 if (form.subject.value=="")
13 {alert("\n글제목을 입력해 주세요.");document.form.subject.focus();return;}
14 if (form.content.value=="")
15 {alert("\n내용을 입력해 주세요.");document.form.content.focus();return;}
16 if (form.passwd.value=="")
17 {alert("\n패스워드를 입력해 주세요.");document.form.passwd.focus();return;}
18 form.submit(); }
19 </script>
20 </head>
21 <body link="#000080" onLoad="document.form.writer.focus()">
22 <table border="0" cellspacing="0" cellpadding="0" align=center>
23 <tr align="left" valign="top">
24 <td>
25 <!--#include file=includeFiles/reply.asp-->
26 </td>
27 </tr>
28 </table>
29 </body>
30 </html>
```

이 소스는 frm_write.asp 파일과 비슷합니다. 9~18행은 자바 스크립트를 사용하여 폼

에서 정보가 입력되지 않았을 때 출력되는 부분을 설정합니다. 또한 include를 사용하여 주요 소스가 들어 있는 includefiles\reply.asp 파일을 포함하고 있습니다.

includefiles/reply.asp

파일 경로

includefiles/reply.asp

```
1  <%
2  no=Request.QueryString("no")
3  page=Request.QueryString("page")
4  forum_id=Request.QueryString("forum_id")
5  strSQL="SELECT thc, thcf, tableWidth FROM tbl_GobyPDSAdmin WHERE
6  forum_id='"&forum_id&'"
7  set rs=conn.execute(strSQL)
8  if rs.EOF then Response.Redirect "includefiles/error.asp?code=notCreateBoard"
9  tableWidth=rs("tableWidth")
10 thc=rs("thc")
11 thcf=rs("thcf")
12 set rs=nothing
13 strSQL="SELECT * FROM tbl_GobyPDS WHERE pk_id="&no
14 set rs=conn.execute(strSQL)
15 content=replace(rs("content"),vbCrLf,"<br>")
16 if not isEmpty(session("member")) then
17 writer=session("member")(0)
18 passwd=session("member")(2)
19 email=session("member")(3)
20 homepage=session("member")(4)
21 readonly = "readonly"
22 end if
23 %>
24 <table border="0" width="<%=tableWidth%>" cellspacing="0" cellpadding
   ="2">
25 <tr bgcolor=#<%=thc%>>
26 <td colspan=6 align="middle" height=25><b><font size="2" color="#<%=
   thcf%>">
27 <%= rs("subject")%> </b></td>
28 </tr>
29 <tr height="20">
30 <td>&nbsp;작성자 </td>
31 <td>&nbsp;<%=rs("writer")%> </td>
32 <td> &nbsp;전자우편 </td>
```

```
33     <td> &nbsp;<a href="mailto:<%=rs("email")%>"><%=rs("email")%></a>
34     </td></tr><tr height="20">
35     <td> &nbsp;<IP> </td>
36     <td> &nbsp;<%=rs("ip")%> </td>
37     <td> &nbsp;<홈페이지> </td>
38     <td> &nbsp;<a href="<%=rs("homepage")%>" target="_blank">
39     <%=rs("homepage")%></a> </td>
40     </tr><tr height="20">
41     <td> &nbsp;<파일명> </td>
42     <td> &nbsp;<a href="includeFiles/down.asp?pk_id=<%=no%>&
43     filename=<%=rs("filename")%>&forum_id=<%=forum_id%>"><%=rs("filename")%></a>
44     (<%=Round((Cln(rs("filesize"))/1024)*10)/10&"KB"> </td>
45     <td> &nbsp;<조회수> &nbsp;<%=rs("readno")%> </td>
46     <td> &nbsp;<다운수> &nbsp;<%=rs("downno")%> </td>
47     </tr><tr align="left" valign="middle" bgcolor="#F4F9EB">
48     <td colspan="6" bgcolor="#F4F9EB" style="BORDER-BOTTOM: 1px dashed;
49     BORDER-LEFT: 1px dashed; BORDER-RIGHT: 1px dashed; BORDER-TOP: 1px
50     dashed;
51     PADDING-LEFT: 30px; PADDING-RIGHT: 30px; PADDING-TOP: 10px;
52     PADDING-BOTTOM: 10px">
53     <%= content%></td></tr>
54     </table>
55     <form action="includeFiles/reply_save.asp" method="post" name="form"
56     encType = "multipart/form-data">
57     <input type="hidden" name="forum_id" value="<%=forum_id%>">
58     <input type="hidden" name="page" value="<%=page%>">
59     <input type="hidden" name="group_id" value="<%=rs("group_id")%>">
60     <input type="hidden" name="group_order" value="<%=rs("group_order")%>">
61     <input type="hidden" name="msglevel" value="<%=rs("msglevel")%>">
62     <input type="hidden" name="name" value="<%=rs("writer")%>">
63     <input type="hidden" name="email2" value="<%=rs("email")%>">
64     <input type="hidden" name="ip" value="<%=Request("REMOTE_ADDR")%>">
65     <table width="<%=tableWidth%>" border="0" cellspacing="0" cellpadding
66     ="0">
67     <tr align="left" valign="top" bgcolor="#84A76F">
68     <td width="<%=tableWidth%>">
69     <table width="<%=tableWidth%>" border="0" cellspacing="1" cellpadding
70     ="5">
```

[illegible]

이 코드는 includesfiles 안에 있는 view.asp와 frm_write.asp를 합쳐놓은 것과 매우 유사하니 참고하기 바랍니다.

1~53행은 includesfiles\view.asp와 유사하며, 54행부터는 includesfiles\frm_write.asp와 거의 동일합니다. 여기서는 서로 다른 부분만 확인하도록 하겠습니다.

16~22행은 세션값을 검사한 다음, 변수에 세션값을 저장합니다. 이 세션값은 게시판과 함께 운영되는 회원관리 프로그램에서 로그인해야 생성됩니다.

56~62행은 데이터베이스에서 정보를 불러와 전달하기 위한 코드입니다. 입력 타임을 hidden으로 설정해서 보이지는 않지만, 폼 태그의 정보를 전달할 때 다른 정보와 마찬가지로 전달됩니다.

63행은 접속한 웹브라우저의 IP 주소를 불러옵니다.

이 외의 다른 행의 내용은 includesfiles 안에 있는 view.asp, frm_write.asp 파일과 일치합니다.

데이터베이스에 답변 저장: includefiles/reply_save.asp

파일 경로

includefiles/
reply_save.asp

```

1  <%@ Language=VBScript %>
2  <!--#include file="../../../dbconn.inc"-->
3  <%
4  Function CheckWord(CheckValue)
5  CheckValue = replace(CheckValue, "&" , "&amp;")
6  CheckValue = replace(CheckValue, "<" , "&lt;")
7  CheckValue = replace(CheckValue, ">" , "&gt;")
8  CheckValue = replace(CheckValue, "'", "'&#039;")
9  CheckWord = CheckValue
10 End Function
11 Function CheckHTML(CheckValue)
12 CheckValue = replace(CheckValue, "&" , "&amp;")
13 CheckValue = replace(CheckValue, "'", "'&#039;")
14 CheckHTML = CheckValue
15 End Function
16 Set UploadForm = Server.CreateObject("SiteGalaxyUpload.Form")
17 forum_id=UploadForm("forum_id")
18 page=UploadForm("page")
19 group_id=UploadForm("group_id")
20 group_order=UploadForm("group_order")
21 msglevel=UploadForm("msglevel")

```

```
22 dim getorder
23 getorder=0
24 set rs=server.CreateObject("ADODB.RecordSet")
25 strSQL="select group_id,group_order,msglevel from tbl_GobyPDS"
26 strSQL=strSQL&" where forum_id='"&forum_id&"' and group_id= "&Cint
   (group_id)&" order by 27 group_order"
28 rs.CursorLocation=adUseClient
29 rs.CursorType=adOpenDynamic
30 rs.LockType=adLockBatchOptimistic
31 rs.Open strSQL,conn
32 if not(rs.EOF and rs.BOF) then
33     do while not rs.EOF
34         if (Cint(rs("msglevel")) <= msglevel) and (Cint(rs("group_order"
           )) >(group_order) then
35             group_order=rs("group_order")
36             getorder=true
37             exit do
38         end if
39         rs.MoveNext
40 loop
41 if getorder then
42     do while not rs.EOF
43         rs("group_order")=Cint(rs("group_order"))+1
44         rs.MoveNext
45     loop
46     rs.UpdateBatch
47 else
48     rs.MoveLast
49     group_order=Cint(rs("group_order"))+1
50 end if
51 end if
52 rs.Close
53 set rs=nothing
54 msglevel=msglevel+1
55 upfile=UploadForm("upfile")
56 writer=trim(UploadForm("writer"))
57 email=trim(UploadForm("email"))
58 readno=0
59 downno=0
60 passwd=trim(UploadForm("passwd"))
```



```
61 homepage=trim(UploadForm("homepage"))
62 ip=trim(UploadForm("ip"))
63 tag=UploadForm("tag")
64 if tag then
65 content=CheckHTML(UploadForm("content"))
66 subject=CheckHTML(UploadForm("subject"))
67 tag=1
68 else
69 content=CheckWord(UploadForm("content"))
70 subject=CheckWord(UploadForm("subject"))
71 tag=0
72 end if
73 if Len(upfile) > 0 then
74 set fc = CreateObject("Scripting.FileSystemObject")
75 folder="\uploadFiles\"
76 storedir=server.mappath("../")&folder
77 attach_file=UploadForm("upfile").filepath '파일이 저장될 경로를 지정
78 filesize=UploadForm("upfile").size
79 filename = mid(attach_file, instrrev(attach_file,"\")+1)
80 strname=mid(filename, 1, instrrev(filename,".")-1)
81 strext=mid(filename,instrrev(filename,".")+1)
82 strfilename=storedir & filename
83 fexist = true
84 count=0
85 do while fexist
86     if(fc.fileexists(strfilename)) then
87         count=count+1
88         filename=strname&-"&count&". "&strext
89         strfilename=storedir & filename
90     else
91         fexist=false
92     end if
93 loop
94 'Response.Write strfilename
95 UploadForm("upfile").saveas strfilename '컴포넌트에 의해 실지 파일을 업로드
96 set fc=nothing
97 else
98 filesize=0
99 filename=""
100 end if
101 if content = "" then content=" "
```

```
102 strSQL="INSERT INTO tbl_GobyPDS(forum_id, group_id, group_order,
    msglevel, subject, "
103 strSQL=strSQL&"writer, email, readno,downno, passwd, homepage,
    filename, filesize, ip, tag,
104 content ) VALUES ('"
105 strSQL=strSQL& forum_id      &"', "
106 strSQL=strSQL& group_id      &" , "
107 strSQL=strSQL& group_order    &" , "
108 strSQL=strSQL& msglevel      &" ,'"
109 strSQL=strSQL& subject       &"', '"
110 strSQL=strSQL& writer        &"', '"
111 strSQL=strSQL& email         &"', "
112 strSQL=strSQL&              "0,0,'"
113 strSQL=strSQL& passwd        &"', '"
114 strSQL=strSQL& homepage      &"', '"
115 strSQL=strSQL& filename      &"', '"
116 strSQL=strSQL& filesize      &"', '"
117 strSQL=strSQL& ip            &"', "
118 strSQL=strSQL& tag           &" ,'"
119 strSQL=strSQL& content       &"')"
120 conn.execute(strSQL)
121 set conn=nothing
122 Response.Redirect "../list.asp?forum_id="&forum_id&"&page="&page
123 %>
```

이 소스가 답변의 핵심적인 부분이니 확실히 이해하도록 합니다.

```
4 Function CheckWord(CheckValue)
5 CheckValue = replace(CheckValue, "&" , "&amp;")
6 CheckValue = replace(CheckValue, "<" , "&lt;")
7 CheckValue = replace(CheckValue, ">" , "&gt;")
8 CheckValue = replace(CheckValue, "'" , "''")
9 CheckWord = CheckValue
10 End Function
11 Function CheckHTML(CheckValue)
12 CheckValue = replace(CheckValue, "&" , "&amp;")
13 CheckValue = replace(CheckValue, "'" , "''")
14 CheckHTML = CheckValue
15 End Function
```

4~10행은 html을 지원하지 않기 위해서 사용된 코드입니다. 먼저 4행은 CheckWord()라는 함수를 만듭니다.

5~8행은 replace() 함수를 사용하여 지정된 문자가 CheckValue 안에 있을 때 그 문자를 새롭게 지정된 문자로 바꿔줍니다.

9행은 CheckWord 함수 이름을 변수로 하여 CheckValue 값을 저장해 반환합니다. VB 스크립트에서는 해당 함수명에 반환값을 지정하면, 그 값이 바로 반환값이 됩니다. 10행에서 Function 프로시저를 종료합니다.

11~15행은 html을 지원하기 위해 사용된 코드이며, 12~15행은 5~10행의 형식과 같습니다.

```

16 Set UploadForm = Server.CreateObject("SiteGalaxyUpload.Form")
17 forum_id=UploadForm("forum_id")
18 page=UploadForm("page")
19 group_id=UploadForm("group_id")
20 group_order=UploadForm("group_order")
21 msglevel=UploadForm("msglevel")

```

16행은 사이트갤럭시 컴포넌트를 사용하여 UploadForm 객체를 생성합니다. 17~21행은 Uploadform 객체를 사용하여 불러온 정보값을 변수에 저장합니다.

```

22 dim getorder
23 getorder=0
24 set rs=server.CreateObject("ADODB.RecordSet")
25 strSQL="select group_id,group_order,msglevel from tbl_GobyPDS"
26 strSQL=strSQL+"&" where forum_id='"&forum_id&"' and group_id="&Cint
   (group_id)&" order by 27 group_order"
28 rs.CursorLocation=adUseClient
29 rs.CursorType=adOpenDynamic
30 rs.LockType=adLockBatchOptimistic
31 rs.Open strSQL,conn

```

22행과 23행은 새로운 변수를 선언한 다음 초기값을 0으로 설정합니다. 24행은 rs 객체를 생성하여 레코드 셋을 연결합니다. 25~27행은 데이터를 불러오기 위한 쿼리문입니다.

28행은 커서의 위치를 설정합니다. adUseClient에서는 데이터 ADO가 커서를 조작합니다. 즉, 레코드셋 전체 내용이 클라이언트에게 복사되며, 로컬 클라이언트 커서 서비스에 의해 제어됩니다.

29행은 커서 타입을 설정합니다. adOpenDynamic은 동적 커서로, 레코드셋을 앞뒤로 이동할 수 있습니다.

30행은 락(lock) 타입을 설정합니다. adLockBatchOptimistic은 UpdateBatch 메소드가 호출될 경우에만 락을 걸어주는 것으로, 여러 개의 레코드를 한꺼번에 수정 가능토록 합니다.

31행은 쿼리문을 실행하여 실제 데이터를 불러옵니다.

```

32  if not(rs.EOF and rs.BOF) then
33      do while not rs.EOF
34          if (Cint(rs("msglevel")) <= msglevel) and (Cint(rs("group_order"))
              >(group_order) then
35              group_order=rs("group_order")
36              getorder=true
37              exit do
38          end if
39          rs.MoveNext
40      loop
41      if getorder then
42          do while not rs.EOF
43              rs("group_order")=Cint(rs("group_order"))+1
44              rs.MoveNext
45          loop
46          rs.UpdateBatch
47      else
48          rs.MoveLast
49          group_order=Cint(rs("group_order"))+1
50      end if
51  end if
52  rs.Close
53  set rs=nothing
54  msglevel=msglevel+1

```

32행은 데이터가 존재할 경우 51행까지의 구문을 실행시킵니다. 33행은 레코드 셋의 마지막을 읽을 때까지 반복문을 실행시킵니다.

34~38행은 (Cint(rs("msglevel")) <= msglevel)와 (Cint(rs("group_order")) > (group_order) 조건을 동시에 만족할 경우, group_order라는 변수에 조건을 만족시킨 rs("group_order") 값을 저장하고, getorder 변수값을 true로 설정한 다음, 37행의 exit do 구문에 의해 41행으로 이동합니다.

34행은 코드의 수정이 필요합니다. 현재 코드로는 조건문이 제대로 실행되지 않으므로 다음과 같이 수정해야 합니다.

```
if (Cint(rs("msglevel")) <= Cint(msglevel)) and (Cint(rs("group_order")) >
Cint(group_order)) then
```

39행은 레코드셋의 다음줄로 이동시킵니다. 40행은 33행의 구문과 함께 사용되는 반복 문입니다. 즉 33~40행 사이의 구문들이 반복 실행됩니다.

41행이 참인 경우 42~46행의 구문을 실행시킵니다. 42행은 33행과 동일한 의미로, 레 코드셋의 마지막이 아니면 계속적으로 반복 구문을 실행시킵니다.

43행은 34행의 조건을 만족시킨 레코드셋의 group_order 값부터 순차적으로 1씩 증가 시킵니다. 44행은 레코드셋의 다음 줄로 이동시킵니다. 45행은 42행의 구문과 함께 사 용되는 반복 구문입니다.

46행은 UpdateBatch 메소드에 의해 43행의 구문을 실행시킵니다. UpdateBatch 메소 드는 보류중인 데이터의 변경 내용을 현재의 레코드에 일괄적으로 업데이트합니다.

47~49행은 34행의 조건문을 만족하는 데이터가 없어서 getorder 변수가 true 값을 가 질 수 없게 됨에 따라 41번 조건문을 만족하지 못했을 때 실행됩니다. 48행은 레코드셋 의 마지막으로 이동시키고, 49행은 레코드셋의 마지막 group_order 값에 1을 더해 group_order 변수에 저장합니다.

50행은 41행의 조건문을 종결하고, 51행은 32행의 조건문을 종결합니다. 52행은 DB와 의 연결을 종료하고, 53행은 rs 객체를 소멸시킵니다. 54행은 msglevel 변수값에 1을 증가시킵니다.

55행부터 마지막까지는 Includefiles/write.asp 파일과 거의 동일하므로 별도로 설명하 지 않겠습니다.

기본적으로 추가해줘야 하는 검색 기능을 프로그래밍합니다.

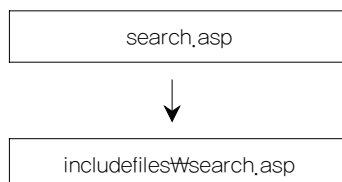
게시판 작성시 검색 기능의 추가는 기본적인 사항이라고 봐도 과언이 아닙니다. 그만큼 사용자 편의를 위해 게시판에 검색기능 추가가 빈번합니다. 이 절에서는 이러한 검색기능 추가에 대해 알아보도록 하겠습니다.

그림 4-28

검색 기능이 추가된 화면



검색 기능에서 파일의 진행 방향을 간단히 살펴보면 다음과 같습니다.



순서대로 소스를 분석해 보겠습니다.

답변 입력 양식

search.asp

파일 경로

gobyPDS/search.asp

```
1  <%@ Language=VBScript %>
2  <html>
3  <head>
4  <title>Untitled Document</title>
5  <meta http-equiv="Content-Type" content="text/html; charset=euc-kr">
6  <!--#include file=../dbconn.inc -->
7  <LINK rel="stylesheet" type="text/css" href="gobyPDS.css">
8  <script language=javascript>
9  function searchOk() {
10   if (searchForm.searchKey.value == "" ) {
11alert("검색어를 입력하세요");
12searchForm.searchKey.focus();
13return false;
14   }
15   searchForm.submit();
16   }
17   function newwindow() {
18   var str,width,height;
19   URL="help.htm";
20   width=450;
21   height=300;
22   str="'toolbar=no, location=no, directories=no, status=no, menubar=no,
      scrollbars=no,
23   resizable=no, copyhistory=no,'"
24   str=str+"width="+width;
25   str=str+", height="+height+"'"
26   window.open(URL, 'remote', str);
27   }
28  </script>
29  </head>
30  <body bgcolor="#FFFFFF">
31    <table border="0" cellspacing="0" cellpadding="0" align=center>
32      <tr align="left" valign="top">
33    <td>
34      <!--#include file=includeFiles/search.asp-->
35      </td>
```

```
36    </tr>
37  </table>
38 </body>
39 </html>
```

이 소스는 HTML 코딩 부분으로, list.asp 파일과 거의 동일합니다. 차이점이 있다면 include 파일로 search.asp 파일이 포함되어 있는 정도입니다. 9~14행은 자바 스크립트를 사용하여 폼에서 정보가 입력되지 않았을 때 출력되는 부분을 설정하고, 17~27행은 확장 검색에 대한 도움말 페이지를 출력하는 부분입니다.

includefiles/search.asp

파일 경로

includefiles/search.asp

```
1  <%
2  forum_id=Request.QueryString("forum_id")
3  strSQL="SELECT * FROM tbl_GobyPDSAdmin WHERE forum_id='"&forum_id&"'"
4  set rs=conn.execute(strSQL)
5  name=rs("name")
6  bgc=rs("bgc")
7  thc=rs("thc")
8  thcf=rs("thcf")
9  tableWidth=rs("tablewidth")
10 ps=rs("pagesize")
11 conternp=rs("skippage")
12 header=rs("header")
13 footer=rs("footer")
14 set rs=nothing
15 page=request("page")
16 if page <> "" Then
17 curPage = page
18   if curPage < 1 Then
19     curPage = 1
20   end if
21 else
22 curPage=1
23 end if
24 TempSearchkey = replace(Trim(Request.QueryString("searchkey")), "&",
   "&")
```



```
25 TempSearchkey = replace(TempSearchkey,"'", "&quot;")
26 GetSearchKey = split(TempSearchKey," ")
27 for i=0 to UBound(GetSearchKey)
28 DSPSearchkey = DSPSearchKey&GetSearchKey(i)&" "
29 if ((i+1) mod 2) = 0 then
30 if not (trim(GetSearchKey(i)) = "or" or trim(GetSearchKey(i)) = "and")
    then
31 Response.Redirect "includefiles/error.asp?code=invaildSearchKey"
32 end if
33 next
34 temp = split(Request.QueryString("searchType"),",")
35 for i=0 to UBound(temp)
36 if temp(i)<>"" then
37 searchType=searchType+Cint(temp(i))
38 end if
39 next
40 strSQL="select * from tbl_GobyPDS where forum_id='"&forum_id&"' and ("
41 select case searchType
42 case 1
43 SearchKey = "subject like"
44 for i=0 to UBound(GetSearchKey)
45 if ((i+1) mod 2) = 0 then
46 Searchkey = SearchKey&" "&GetSearchKey(i)&" subject like"
47 else
48 Searchkey = SearchKey & " '%"& GetSearchKey(i) &"%'"
49 end if
50 next
51 case 2
52 SearchKey = "writer like"
53 for i=0 to UBound(GetSearchKey)
54 if ((i+1) mod 2) = 0 then
55 Searchkey = SearchKey&" "&GetSearchKey(i)&" writer like"
56 else
57 Searchkey = SearchKey & " '%"& GetSearchKey(i) &"%'"
58 end if
59 next
60 case 4
61 SearchKey = "content like"
62 for i=0 to UBound(GetSearchKey)
63 if ((i+1) mod 2) = 0 then
```

```
64     Searchkey = SearchKey & " "&GetSearchKey(i)&" content like"
65     else
66         Searchkey = SearchKey & " '%"& GetSearchKey(i) &"%' "
67     end if
68 next
69 case 3
70 SearchKey = "subject like"
71 for i=0 to UBound(GetSearchKey)
72     if ((i+1) mod 2) = 0 then
73         Searchkey = SearchKey&" "&GetSearchKey(i)&" subject
           like"
74     else
75         Searchkey = SearchKey & " '%"& GetSearchKey(i) &"%' or
76 writer like '%"& GetSearchKey(i) &"%' "
77     end if
78 next
79
80 case 5
81 SearchKey = "subject like"
82 for i=0 to UBound(GetSearchKey)
83     if ((i+1) mod 2) = 0 then
84         Searchkey = SearchKey&" "&GetSearchKey(i)&" subject like"
85     else
86         Searchkey = SearchKey & " '%"& GetSearchKey(i) &"%' or
87 content like '%"& GetSearchKey(i) &"%' "
88     end if
89 next
90 case 6
91 SearchKey = "writer like"
92 for i=0 to UBound(GetSearchKey)
93     if ((i+1) mod 2) = 0 then
94         Searchkey = SearchKey&" "&GetSearchKey(i)&" writer like"
95     else
96         Searchkey = SearchKey & " '%"& GetSearchKey(i) &"%' or
97 content like '%"& GetSearchKey(i) &"%' "
98     end if
99 next
100 case else
101     SearchKey = "subject like"
102     for i=0 to UBound(GetSearchKey)
```

[illegible]

```
134 </font> </td>"
135 str=str("<td width=30 align=center height=20> <font color=#"& thcf &"
      size=2> 조회
136 </font></td>"
137 str=str("<td width=30 align=center height=20> <font color=#"& thcf &"
      size=2> 다운
138 </font></td>"
139 str=str("</tr>"
140 if rs.EOF then
141 str=str("<tr><td colspan=5 align=center>검색 결과가 없습니다</td></tr>"
142 else
143 r_count=rs.RecordCount
144 rs.PageSize=ps
145 RS.AbsolutePage = curPage
146 totPage = r_count \ rs.PageSize
147 if not totPage*rs.PageSize = r_count then totPage=totPage+1
148 i=0
149 do while not rs.EOF and rs.PageSize > i
150 check_fileexe = rs("filename")
151 if not isNull(check_fileexe) then
152 pos = instrrev(check_fileexe, ".")+1
153 check_fileexe = mid(check_fileexe, pos)
154 end if
155 Select Case Ucase(check_fileexe) '확장자에 따른 이미지 선택
156     Case "HWP", "COM", "EXE", "TXT", "WAV", "XLS", "DOC", "ZIP",
        "RAR", 1 157 "GIF", "BMP", "MP3", "PDF", "PPT"
158         fileicon=check_fileexe & ".gif"
159     Case "RA", "RAM"
160         fileicon = "ra.gif"
161     Case "HTM", "HTML"
162         fileicon= "htm.gif"
163     Case "JPG", "JPEG"
164         fileicon="jpg.gif"
165     Case "MPG", "MPEG", "AVI"
166         fileicon="mpg.gif"
167     Case else
168         fileicon="etc.gif"
169 End Select
170 blank=""
171 for cnt=0 to Cint(rs("msglevel"))
```



```
208 loop
209 if Request.QueryString("startpage")="" then
210     startpage=1
211 else
212     startpage=Request.QueryString("startpage")
213 end if
214 str=str&"</table>"
215 end if 'rs.eof end
216 set rs=nothing
217 set conn=nothing
218 Response.Write str
219 %>
220 <table width=<%= tableWidth %> border=0 cellpadding=0>
221 <tr align=left valign=top bgcolor=#CCCCCC>
222 <td width=<%= tableWidth %> height=1 colspan=2><img src=images/
jn_0.gif
223 width=1 height=1></td>
224 </tr>
225 <td height=30 align=left><font color=dimgray><%=r_count%> 개의 글이
226 있습니다.</font></td>
227 <td align=right><a href="list.asp?forum_id=<%=forum_id%>"> 리스트
</a>]
228 <% if Cint(totpage) > Cint(conternp) then
229     if startpage=1 then %>
230     [ <font color=dimgray>이전 <%=conternp%> 개 </font> ]
231 <%     else %>
232     [<a href="search.asp?forum_id=<%=forum_id%>&page=
233 <%=cint(startpage)-conternp%>&startpage=<%=cint(startpage )-
conternp%> &searchKey=
234 <%=TempsearchKey%>&searchType=<%=searchType%>"> 이전 <%=conternp%> 개
</a>]
235 <%     end if
236 For a=startpage to startpage+conternp-1
237     if a>totpage then
238         exit for
239     else
240         if a=Cint(curpage) then %>
241         <b> <%=a%> </b>"
242         <%     else %>
243         <a href="search.asp?forum_id=<%=forum_id%>&page=<%=a%>
&startpage=
```

```
244 <%=startpage%>&searchKey=<%=TempsearchKey%>&searchType= <%=search
    Type%>">
245 <%=a%></a>
246     <% End if
247     end if
248 Next
249 if ((startpage\conternp)=(totpage\conternp)) then %>
250 [ <font color=dimgray>다음 <%=conternp%> 개</font> ]
251 <% else %>
252 [                                     <                                     a
href="search.asp?forum_id=<%=forum_id%>&page=<%=a%>&startpage=
253 <%=a%>&searchKey=<%=TempsearchKey%>&searchType=<%=searchType%>">
254 다음 <%=conternp%> 개 </a>]
255 <% end if
256 else %>
257 [ <font color=#808000>이전 <%=conternp%> 개 </font> ]
258 <% For a=startpage to totpage
259 if a=Cint(curpage) then %>
260     <b> <%=a%> </b>
261 <% else %>
262     <a href="search.asp?forum_id=<%=forum_id%>&page=<%=a%>&
        searchKey=
263 <%=TempsearchKey%>&searchType=<%=searchType%>"><%=a%></a>
264 <% end if
265 next %>
266 [ <font color=#808000> 다음 <%=conternp%> 개 </font> ]
267 <% end if %>
268 </td>
269 </tr>
270 <tr>
271     <td align=center colspan=2>
272     <form action=search.asp Method=get name=search>
273     <font color=dimgray><b>제목</b></font><input type=checkbox name=
        searchType
274 value=1 checked>
275     <font color=dimgray><b>글쓴이</b></font> <input type=checkbox name=
        searchType
276 value=2>
277 <font color=dimgray><b>내용</b></font><input type=checkbox name=
        searchType value=4>
```

```
278 <input type=text name=searchkey size=20 maxlength=80 style='border:
    1 solid #808080'>
279 &nbsp;<input type=submit value=검색 style='border:1 solid black;
    height:19px'>
280 [<a href='javascript:newwindow()'>확장검색</a>]
281 <input type=hidden name=forum_id value=<%=forum_id%>>
282 </form>
283 </td>
284 </tr>
285 </table>
286 <% Response.Write footer %>
```

이 소스는 검색 기능을 추가하기 위해 삽입된 코드를 제외하면, includefiles/list.asp와 유사합니다. 여기서는 주로 검색 기능에 관련된 코드를 다루도록 하겠습니다. 나머지 부분은 includefiles/list.asp를 참고하기 바랍니다.

```
1 <%
2 forum_id=Request.QueryString("forum_id")
3 strSQL="SELECT * FROM tbl_GobyPDSAdmin WHERE forum_id='"&forum_id&'"
4 set rs=conn.execute(strSQL)
5 name=rs("name")
6 bgc=rs("bgc")
7 thc=rs("thc")
8 thcf=rs("thcf")
9 tableWidth=rs("tableWidth")
10 ps=rs("pagesize")
11 conternp=rs("skippage")
12 header=rs("header")
13 footer=rs("footer")
14 set rs=nothing
15 page=request("page")
16 if page <> "" Then
17 curPage = page
18     if curPage < 1 Then
19         curPage = 1
20     end if
21 else
22 curPage=1
23 end if
```

2~13행은 DB와의 연동을 통해 데이터를 불러와서 변수에 저장합니다.

14행은 rs객체를 소멸시키고, 15~23행은 페이지값을 지정하는 코드입니다.

```

24 TempSearchkey = replace(Trim(Request.QueryString("searchkey")), "&",
    "&amp;")
25 TempSearchkey = replace(TempSearchkey, "'", "&quot;")
26 GetSearchKey = split(TempSearchKey, " ")
    
```

24행은 우선 searchkey 값을 불러온 다음, trim 함수를 사용해 양 끝의 여백을 삭제합니다. 그런 후, replace 함수를 사용해 Request.QueryString("searchkey") 내의 &을 &으로 변환시켜 TempSearchkey 변수에 저장합니다.

25행은 replace 함수를 사용하여 TempSearchkey 내의 '을 "로 변환시켜 다시 저장합니다.

26행은 split 함수를 사용하여 TempSearchkey 내의 문자열을 공백을 기준으로 나눈 다음 배열화시켜 GetDearchKey 값에 저장합니다.



split 함수

<구문> split(문자열[, 문자열을 분리하기위한 기호])

split 함수는 분리 기호에 의해 분리된 문자열을 각각의 배열 값으로 저장합니다. 분리기호가 생략되면 공백이 분리 기호의 기준이 됩니다.

```

27 for i=0 to UBound(GetSearchKey)
28 DSPSearchkey = DSPSearchKey&GetSearchKey(i)&" "
29 if ((i+1) mod 2) = 0 then
30 if not (trim(GetSearchKey(i)) = "or" or trim(GetSearchKey(i)) =
    "and") then
31 Response.Redirect "includefiles/error.asp?code=invaildSearchKey"
32 end if
33 next
    
```

27~33행은 잘못된 검색 방법에 대한 에러 문구를 내보내는 코드입니다.

27행은 for 구문을 이용한 반복문인데, UBound 함수를 사용해 for 구문의 반복회수를 결정합니다.



UBound 함수

〈구문〉 UBound(배열 이름[, 차원])

UBound 함수는 지정한 배열 차원에 사용할 수 있는 가장 큰 첨자를 반환합니다. 차원값이 생략되면 1차원이 기본값으로 설정됩니다. 다음은 그 사용 예입니다.

```
Dim array(10,4)
UBound(array, 1) --> array 배열이 가지는 1차원 값 중 가장 큰 값은 10이므로
10을 반환합니다.
UBound(array, 2) --> array 배열이 가지는 2차원 값 중 가장 큰 값은 4이므로
4를 반환합니다.
```

28행은 GetSearchKey(i) 배열을 DSPSearchkey 변수에 저장하고, 29행은 i 값을 2로 나누는 나머지가 0인지 1인지 구별하여 0이면 30번 구문을 실행시킵니다.

30~31행의 구문은 공백으로 구분된 문자열 중에서 두 번째, 네 번째, ... 문자열이 and 나 or가 아닐 때 여러 구문을 실행시킵니다.

32행은 29행의 조건문을 종결하고, 33행은 for 구문과 함께 사용되는 반복구문입니다.

```
34 temp = split(Request.QueryString("searchType"),",")
35 for i=0 to UBound(temp)
36     if temp(i)<>"" then
37         searchType=searchType+Cint(temp(i))
38     end if
39 next
40 strSQL="select * from tbl_GobyPDS where forum_id='"&forum_id&"' and ("
```

34행은 searchType을 불러온 다음 split 함수를 사용하여 쉼표(,)를 기준으로 분리된 문자열을 배열화합니다. 여기서 searchType은 폼의 체크박스 name 값입니다. 폼에서 넘어온 체크박스의 name이 모두 동일할 경우, 각 체크박스의 value 값을 호출하면 쉼표(,)에 의해 각각 구분되어 순차적으로 표시됩니다.

35~39행은 반복문에 의해 searchType 값을 더해주고 있습니다. 더해주는 이유는 매우 간단합니다. 이 코드 상에서 불러오는 체크박스의 value 값은 1, 2, 4입니다. 사용자가 세 개의 체크박스에 체크할 수 있는 경우의 수는 최대 7가지입니다. 즉, 각 경우값을 살펴보면 1, 2, 4, 3, 5, 6, 7의 값이 나오게 되는 것입니다. 따라서 체크된 체크박스의 value 값을 더해주므로써 어떤 체크박스가 체크되었는지 알 수 있는 것입니다.

40행은 쿼리 구문입니다.

```

41  select case searchType
42  case 1
43  SearchKey = "subject like"
44  for i=0 to UBound(GetSearchKey)
45  if ((i+1) mod 2) = 0 then
46      Searchkey = SearchKey & " "&GetSearchKey(i)&" subject
        like"
47  else
48      Searchkey = SearchKey & " '%"& GetSearchKey(i) &"'"
49  end if
50  next

```

41~50행은 나머지가 0 또는 1일 경우에 따라 40행의 where 조건문 뒤에 연결되는 구문을 설정합니다.

41행은 37행의 구문에서 나온 searchType 값을 조건문으로 비교합니다. 42행은 searchType 값이 1인 경우 43행 이하를 실행시킵니다.

43행은 SearchKey 변수에 "subject like" 값을 저장합니다. 그 이유는 체크박스에서 체크한 부분이 제목(subject)일 경우에만 searchType 값이 1이 되기 때문입니다.

44행은 반복문입니다. 45행은 나머지가 0일 경우 그 이하 구문을 실행시키는 조건문입니다.

46행은 GetSearchKey(i) 값으로 and나 or 값이 입력된 다음 Searchkey 변수에 저장됩니다. 나머지가 0일 경우 and나 or만 들어가는 이유는 배열이 GetSearchKey(1), GetSearchKey(3), 일 때만 나머지가 0이 되기 때문입니다. 다시 말해, 27행 구문 이하에서 배열이 GetSearchKey(1), GetSearchKey(3),일 경우 and나 or 값이 아니

면 에러 파일로 이동하기 때문입니다. 따라서 GetSearchKey(1), GetSearchKey(3),배열이 존재한다면 그 값은 and나 or가 되는 것입니다.

47행은 나머지가 1일 경우 그 이하 구문을 실행시키고, 48행은 GetSearchKey(i) 값으로 유저가 검색을 위해 입력한 내용이 들어갑니다.

49행은 45번 조건문을 종료합니다. 50행은 for 구문과 함께 쓰이는 반복문입니다.



where 컬럼명 like '%" & GetSearchKey(i) &"'

일반적인 의미는 지정된 컬럼의 레코드값과 GetSearchKey(i) 값이 같은 데이터만 불러오라는 구문입니다. 하지만 여기서 정작 중요한 것은 %인데, like와 함께 사용되어 검색의 범위를 증대시켜 줍니다. 쿼리 구문에서 배열이나 변수 양 옆에 쓰이는 %가 의미하는 것은 GetSearchKey(i) 값이 포함된 문자열을 모두 불러오라는 의미입니다. %가 앞에만 붙게 되면 GetSearchKey(i) 값이 포함된 문자열중 마지막에 위치하는 문자열만 불러오게 됩니다. 즉, GetSearchKey(i)를 중심으로 %가 붙은 방향은 어떠한 문자가 와도 상관없다는 뜻입니다. 반대로 %가 뒤쪽에만 붙게 되면 GetSearchKey(i) 값이 맨 처음에 위치하는 문자열만 불러오게 됩니다.

```
51 case 2
52 SearchKey = "writer like"
53 for i=0 to UBound(GetSearchKey)
54     if ((i+1) mod 2) = 0 then
55         Searchkey = SearchKey & " "&GetSearchKey(i)&" writer
           like"
56     else
57         Searchkey = SearchKey & " '%" & GetSearchKey(i) &"'"
58     end if
59 next
60 case 4
61 SearchKey = "content like"
62 for i=0 to UBound(GetSearchKey)
63     if ((i+1) mod 2) = 0 then
64         Searchkey = SearchKey & " "&GetSearchKey(i)&" content
           like"
65     else
66         Searchkey = SearchKey & " '%" & GetSearchKey(i) &"'"
67     end if
68 next
```

51~68행은 41~50행의 내용과 유사합니다.

51~59행은 searchType 값이 2인 경우로, 체크박스에서 '글쓴이' 부분만 체크한 것입니다. 따라서 subject like 대신에 writer like가 사용된 것입니다.

61~68행은 searchType 값이 4인 경우로, 체크박스에서 '내용' 부분만 체크한 것입니다. 따라서 subject like나 writer like 대신에 content like를 사용하는 것입니다.

```
69 case 3
70 SearchKey = "subject like"
71 for i=0 to UBound(GetSearchKey)
72     if ((i+1) mod 2) = 0 then
73         Searchkey = SearchKey & " "&GetSearchKey(i)&" subject like"
74     else
75         Searchkey = SearchKey & " "&GetSearchKey(i) &"%" or
76         writer like '%"& GetSearchKey(i) &"%"
77     end if
78 next
79
80 case 5
81 SearchKey = "subject like"
82 for i=0 to UBound(GetSearchKey)
83     if ((i+1) mod 2) = 0 then
84         Searchkey = SearchKey & " "&GetSearchKey(i)&" subject like"
85     else
86         Searchkey = SearchKey & " "&GetSearchKey(i) &"%" or
87         content like '%"& GetSearchKey(i) &"%"
88     end if
89 next
90 case 6
91 SearchKey = "writer like"
92 for i=0 to UBound(GetSearchKey)
93     if ((i+1) mod 2) = 0 then
94         Searchkey = SearchKey & " "&GetSearchKey(i)&" writer like"
95     else
96         Searchkey = SearchKey & " "&GetSearchKey(i) &"%" or
97         content like '%"& GetSearchKey(i) &"%"
98     end if
```

```

99 next
100 case else
101 SearchKey = "subject like"
102 for i=0 to UBound(GetSearchKey)
103     if ((i+1) mod 2) = 0 then
104         Searchkey = SearchKey & " "&GetSearchKey(i)&" subject like"
105     else
106         Searchkey = SearchKey & " '%"& GetSearchKey(i) &"%' or
107     writer like '%"& GetSearchKey(i) &"%' or content like '%"& GetSearch
        Key(i)&"%'
108     end if
109 next
110 end select

```

69~110행도 41~50행의 내용과 유사합니다. 다만 조건의 차이로 인해서 일부분이 추가되었을 뿐입니다.

69~78행은 searchType 값이 3인 경우로, 체크박스에서 ‘제목’과 ‘글쓴이’ 두 개를 체크한 것입니다. 따라서 이미 존재하는 "subject like" 외에 76행에 있는 "writer like"가 첨부된 것입니다.

80~89행은 searchType 값이 5인 경우로, 체크박스에서 ‘제목’과 ‘내용’ 두 개를 체크한 것입니다. 따라서 이미 존재하는 subject like 외에 87행에 있는 content like가 첨부된 것입니다.

90~99행은 searchType 값이 6인 경우로, 체크박스에서 ‘글쓴이’와 ‘내용’ 두 개를 체크한 것입니다. 따라서 91행에서 searchKey 변수에 writer like를 저장합니다. 그런 다음 94행에서 writer like를 입력합니다. 결국 96행에 writer like가 존재하게 되므로 97행 구문에 content like만 첨부된 것입니다.

100~109행은 searchType 값이 7인 경우로, 체크박스에서 ‘제목’, ‘글쓴이’, ‘내용’이 모두 체크된 것입니다. 따라서 이미 존재하는 subject like 외에 107행에 있는 writer like와 content like가 첨부된 것입니다.

110행은 41행에서 시작한 조건문을 종료합니다.

```
111 strSQL=strSQL&SearchKey&"") order by group_id desc, group_order "  
112 set rs=server.CreateObject("ADODB.RecordSet")  
113 rs.CursorLocation=adUseClient  
114 rs.open strSQL,conn,adOpenForwardOnly,adLockReadOnly
```

111행은 40행에서 시작한 쿼리문을 실제로 완성시킵니다. 112행은 rs객체를 생성합니다. 113행은 커서의 위치를 클라이언트로 설정하고, 114행은 실제로 DB와의 연결을 실행합니다.

115~139행은 HTML 구문입니다.

```
140 if rs.EOF then  
141 str=str+"<tr><td colspan=5 align=center>검색 결과가 없습니다</td></tr>"  
142 else
```

140~142행은 조건에 맞는 데이터가 존재하지 않을 때 출력되는 코드입니다.

143행부터 마지막까지는 includefiles/list.asp 파일과 동일합니다. 약간의 차이가 있다면 하이퍼링크 설정시, get 방식으로 정보를 전달할 때 searchKey 값과 searchType 값이 추가로 전달된다는 것입니다.

```
227 <td align=right>[<a href="list.asp?forum_id=<%=forum_id%>">리스트 </a>]
```

227행은 리스트 항목이 추가된 것입니다.