

3.2.3 ADO 访库的技术实现方法

ADO(ActiveX Data Objects)是一组优化的访问数据库专用对象集,为 ASP 提供了完整的站点数据库访问解决方案。它可作用于服务器端,以提供含有数据库信息的主页内容。通过执行 SQL 命令,让用户在浏览器画面中输入、更新和删除站点服务器的数据库信息。

ADO 使用内置的 RecordSet 对象作为数据的主要接口。ADO 可使用 VBScript、JScript 语言来控制对数据库的访问,以及查询结果的输出显示。

ADO 可连接多种支持 ODBC 的数据库,如 SQL Server、Oracle、Informix 等。

在使用时,首先需在控制面板的 ODBC 中建立相应的 DSN(数据源名)。要选择系统数据源名,指定所用的驱动程序,"SQL Server",在数据来源名称中输入 DSN 名,并选定服务器和数据库。

使用 ADO 访问数据库是 ASP 编程的关键所在^[32]。

仍以人事管理为例。一页面用来显示所有员工的基本信息,包括编号、姓名、性别、出生日期、学历、籍贯、民族、住址、身份证号、所在部门等。

打开数据库,连接基本人事信息表并进行查询的编程如下:

1) 与数据库建立连接并打开数据库,其实现语法如下:

```
<%  
MM_connKDS_STRING = "dsn=dbKDSSample;uid=xx,pwd=xx;"  
%>
```

2) 设定 SQL 语句,并从表中取出相应数据:

```
<%  
strSQL="SELECT * FROM tblEmployee where blnLeave=0 "  
set rs = Server.CreateObject("ADODB.Recordset")  
rs.ActiveConnection = MM_connKDS_STRING  
rs.Source = strSQL  
rs.CursorType = 3
```

```
rs.CursorLocation =3
```

```
rs.LockType = 3
```

```
rs.Open()
```

```
rs.pageSize=50
```

```
%>
```

3) 利用 RecordSet 对象所提供的属性显示查询结果。

```
<%
```

```
if rs.bof and rs.eof then
```

```
response.write "<table width=500><tr><td width=400 align=center><font  
color=red>没有符合条件的员工! </font></td></tr></table>"
```

```
else
```

```
%>
```

```
<td >
```

```
<div align="center"><%=rs.Fields.Item("strEmpID").Value)%></div>
```

```
</td>
```

```
<td >
```

```
<div align="center"><%=rs.Fields.Item("strEmpName").Value)%></div>
```

```
</td>
```

```
<td >
```

```
<div align="center"><%=rs.Fields.Item("strEmpNameJ").Value)%></div>
```

```
</td>
```

```
<td >
```

```
<div align="center"><%=rs.Fields.Item("strGender").Value)%></div>
```

```
</td>
```

```
<td >
```

```
<div align="center"><%=rs.Fields.Item("dateBirthDate").Value)%></div>
```

```
</td>
```

```
<td >
```

```
<div align="center"><%=rs.Fields.Item("strEducation").Value)%></div>
```

```
</td>
```

```
.....
```

4) 利用 where 子句, 实现按不同字段查询:

```
if request.querystring("strQry")="strEmpID" then strSQL=strSQL& "and
strEmpID like%" + request.querystring("qry") + "%"
if request.querystring("strQry")="strEmpName" then strSQL=strSQL+ "and
strEmpName like%" + request.querystring("qry") + "%"
if request.querystring("strQry")="strEmpNameJ" then strSQL=strSQL+ "and
strEmpNameJ like%" + request.querystring("qry") + "%"
if request.querystring("strQry")="strSite" then strSql=strSql+ " and strSite
like%" + request.querystring("qry") + "%"
if request.querystring("strQry")="strDepID" then strSql=strSql+ " and strDepID
like%" + request.querystring("qry") + "%"
if request.querystring("strQry")="strDepName" then strSql=strSql+ " and
strDepName like%" + request.querystring("qry") + "%"
if request.querystring("strQry")="strPrjName" then strSql=strSql+ " and
strPrjName like%" + request.querystring("qry") + "%"
```

5) 利用 order by 子句, 实现按不同字段排序:

```
if request.queryString("order")<>"" then
    strSQL=strSQL & " order by " & request.queryString("order")
else
    strSQL=strSQL & " order by strEmpID"
end if
```

6) 由于查询结果数量很多, 所以要进行分页显示:

```
<%
if request.queryString("move")="next" then
    if session("currentPage")<>rs.PageCount then
        session("currentPage")=session("currentPage")+1
    else
        session("currentPage")=rs.PageCount
    end if
end if
if request.queryString("move")="prev" then
```

```

        session("currentPage")=session("currentPage")-1
    end if
    if request.QueryString("move")="first" then
        session("currentPage")=1
    end if
    if request.QueryString("move")="last" then
        session("currentPage")=rs.PageCount
    end if
    if isnumeric(request.QueryString("move")) then
        session("currentPage")=cint(request.QueryString("move"))
    end if
    if session("currentPage")="" or session("currentPage")=0 then
        session("currentPage")=1
    end if
    'response.write session("currentPage")
    if (not rs.bof) and (not rs.eof) then
        rs.AbsolutePage=session("currentPage")
    end if
    %>
    <table border="0" width=700 align="center">
        <tr>
            <td width=60 align="center">
                <% If (session("currentPage")> 1) Then %>
                <a
href="<%=filename%>.asp?move=first&order=<%=request.QueryString("Order"
)%>&strQry=<%=request.QueryString("strQry")%>&qry=<%=request.QueryString("qry")%>">首</a>

                <% End If ' end %>
            </td>
            <td width=40 align="center">
                <% If (session("currentPage")> 1) Then %>

```

```

        <a
href="<%=filename%>.asp?move=prev&order=<%=request.querystring("Order"
)%>&strQry=<%=request.querystring("strQry")%>&qry=<%=request.querystring("qry")%>"> ↑ </a>

        <% End If' end %>
    </td>
    <% for i=1 to rs.pagecount %>
    <td
width=4><a
href="<%=filename%>.asp?move=<%=i%>&order=<%=request.querystring("O
rder")%>&strQry=<%=request.querystring("strQry")%>&qry=<%=request.querystring("qry")%>">
        <% if session("currentPage")=i then
            response.write "<font color=red>" & i &"</font>"
        else
            response.write i
        end if %></a>
        <% if i=50 then response.write "</td><tr><td>"%>
    </td>
    <% next %>
    <td width=40 align="center">
        <% If (session("currentPage")< rs.PageCount) Then %>
        <a
href="<%=filename%>.asp?move=next&order=<%=request.querystring("Order"
)%>&strQry=<%=request.querystring("strQry")%>&qry=<%=request.querystring("qry")%>"> ↓ </a>
        <% End If' end %>
    </td>
    <td width=60 align="center">
        <% If (session("currentPage")< rs.PageCount) Then %>
        <a
href="<%=filename%>.asp?move=last&order=<%=request.querystring("Order")

```

```

%>&strQry=<%=request.querystring("strQry")%>&qry=<%=request.querystring("qry")%>">末</a>

    <% End If ' end %>

</td>

</tr>

</table>

<%
    i=i+1
    rs.MoveNext()
Wend
%>

```

3.2.4 ASP 与组件对象的使用

在利用 ASP 技术进行动态 Web 开发的时候,应遵循这样一个原则:即 ASP 的服务器端脚本必须与微软倡导的组件对象(Component Object Model)配合使用才能开发出具有实用价值的信息产品。ASP 脚本相当于一种粘合剂,把一个个具有特定功能的组件对象粘合在一起,以形成最终的软件产品。这一软件制作的工艺思想类似于硬件工厂生产板卡的过程,那一个个的组件对象就相当于集成电路的芯片,而 ASP 脚本只是焊接芯片的焊接剂,由此可见组件对象起着核心作用。

组件对象模型 COM 是微软提出的一种基于二进制的 Windows 软件标准,它是由 OLE 技术逐渐发展而来的。在使用不同语言工具写成的组件对象之间,依据 COM 的标准可以进行交互。COM 是技术概念和标准,其商业概念的称谓则使用 ActiveX。

上文提到的 ADO 就是系统提供的用于访问后台数据库的组件。此外,ASP 本身还提供了五个内置的对象。这五个内置对象可被 ASP 脚本直接使用,它们