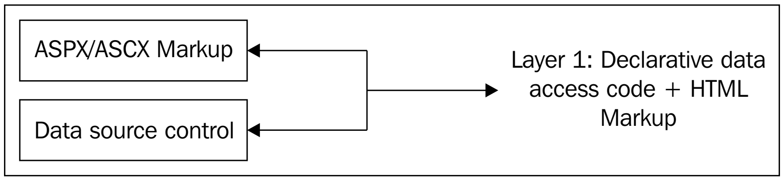
```
            Website:
            <asp:TextBox ID="txtWebsite" runat="server" Width="173px">
                                                  </asp:TextBox><br />
            Comments:
            <asp:TextBox ID="txtComments" runat=
                    "server" TextMode="MultiLine"></asp:TextBox><br />
            <br />
            <asp:Button ID="btnAddComment" runat="server" Text=
              "Add New Comment" OnClick="btnAddComment_Click" /><br />
            <br />
            <asp:AccessDataSource ID="commentsDataSource" runat=
                              "server" DataFile="~/App_Data/Guestbook.mdb"
             SelectCommand="SELECT * FROM [GuestBook]"
             InsertCommand="INSERT INTO GuestBook(FullName,
                        EmailID, Website, Comments) VALUES
              (@FullName,EmailID,Website,Comments)">
                <InsertParameters>
                    <asp:ControlParameter Name="@FullName"
                         ControlID="txtFullName" PropertyName="Text" />
                    <asp:ControlParameter Name="@EmailID"
                         ControlID="txtEmailID" PropertyName="Text" />
                    <asp:ControlParameter Name="@Website"
                         ControlID="txtWebsite" PropertyName="Text" />
                    <asp:ControlParameter Name="@Comments"
                         ControlID="txtComments" PropertyName="Text" />
                </InsertParameters>
            </asp:AccessDataSource>
           <asp:Repeater id="rptComments" runat="server" DataSourceID=
                                                 "commentsDataSource">
                           <ItemTemplate>
                                    Name:<%# Eval("FullName")%>
                                    <br>
                                    Email:<%# Eval("EmailID")%>
                                    <br>
                                    Website:&nbsp;<%# Eval("Website")%>
                                    <br>
                                    Dated: <%# Eval("EntryDate")%>
                                    <br>
                                    Comments:<%# Eval("Comments")%>
                                     </ItemTemplate>
                                  </asp:Repeater>
            </div>
        </form>
    </body>
    </html>
```

Set the mode of the **DataSourceMode** property to **DataReader** instead of **Dataset** for better performance.

The use of Data Source Controls can cut down the use of data access code and shorten the development time. In the above example, we inserted the data without writing any code. Similarly, we can perform updates, deletes, and other CRUD operations, and use data controls with other controls such as the GridView and the DetailsView.

The Data Source Controls wrap the data access code logic, and use declarative markup in the ASPX page. So using these controls means that we are strongly coupling the GUI markup with the data access code, which may be fine for small applications such as a personal website, but should not be used for any commercial medium-to-large web applications.

When using data controls, the architecture is still 1-tier, but the layers change. Instead of using data fetching, and binding code in the code-behind classes, we simply use the declarative markup of Data Source Controls. This is good for turning out applications quickly, but we lose finer control over data-access.



We have all of the code, as well as the markup, in the ASPX page. As mentioned earlier, having the data access code in the GUI violates the basic principle of code separation—the GUI should be independent of the business logic or data access code. Therefore, data controls are useful for small projects only. Here is a list of the advantages and disadvantages of using data controls in web projects:

Advantages:

- Cuts down heavily on coding and saves cost and development time.
- Declarative markup allows changes to propagate on the server without re-compiling the site.
- Ready support for data-bound controls such as GridView, DetailsView, and DataList.