Q: Why would you like to change the company?

1) I am looking for a more challenging career in a firm with a larger employee base such as yours.
2) Keeping in mind my career goals, the time has come for me to move onto the next rung of the ladder and make a mark for myself. This can be achieved in a company like this.
3) It is just a career move to enhance my knowledge in my own area of interest. After completion of this question only interview will go for further questions

Q: Difference between stored procedure and function

1) Procedure can return zero or n values whereas function can return one value which is mandatory.
2) Procedures can have input, output parameters for it whereas functions can have only input parameters.
3) Procedure allows select as well as DML statement in it whereas function allows only select statement in it.
4) Functions can be called from procedure whereas procedures cannot be called from function.
5) Exception can be handled by try-catch block in a procedure whereas try-catch block cannot be used in a function.
6) We can go for transaction management in procedure whereas we can't go in function.
7) Procedures cannot be utilized in a select statement whereas function can be embedded in a select statement.

Q: Difference between Abstract and Interface

Abstract Class:
-Abstract class provides a set of rules to implement next class
-Rules will be provided through abstract methods
-Abstract method does not contain any definition
-While inheriting abstract class all abstract methods must be override
-If a class contains at least one abstract method then it must be declared as an “Abstract Class”
-Abstract classes cannot be instantiated (i.e. we cannot create objects), but a reference can be created
-Reference depends on child class object’s memory
-Abstract classes are also called as “Partial abstract classes”
-Partial abstract class may contain functions with body and functions without body
-If a class contains all functions without body then it is called as “Fully Abstract Class” (Interface)

Interface:
-If a class contains all abstract methods then that class is known as “Interface”
-Interfaces support like multiple inheritance
-In interface all methods r public abstract by default
-Interfaces r implementable
-Interfaces can be instantiated, but a reference cannot be created

Q: Index types in SQL Server

Clustered Index
Only 1 allowed per table physically rearranges the data in the table to confirm to the index constraints for use on columns that are frequently searched for ranges of data for use on columns with low selectivity.

Non-Clustered Index
Up to 249 allowed per table creates a separate list of key values with pointers to the location of the data in the data pages For use on columns that are searched for single values

A clustered index is a special type of index that reorders the way records in the table are physically stored. Therefore table can have only one clustered index. The leaf nodes of a clustered index contain the data pages. A non-clustered index is a special type of index in which the logical order of the index does not match the physical stored order of the rows on disk. The leaf node of a non-clustered index does not consist of the data pages. Instead, the leaf nodes contain index rows.

Q: Included Column Index (New in SQL Server 2005)

In SQL Server 2005, the functionality of non-clustered indexes is extended by adding non-key columns to the leaf level of the non-clustered index. Non-key columns can help to create cover indexes. By including non-key columns, you can create non-clustered indexes that cover more queries. The Database Engine does not consider non-key columns when calculating the number of index key columns or index key size. Non-key columns can be included in non-clustered index to avoid exceeding the current index size limitations of a maximum of 16 key columns and a maximum index key size of 900 bytes. Another advantage is that using non-key column in index we can have index data types not allowed as index key columns generally.

In following example column Filename is varchar(400), which will increase the size of the index key bigger than it is allowed. If we still want to include in our cover index to gain performance we can do it by using the Keyword INCLUDE.

```
USE AdventureWorks
GO
CREATE INDEX IX_Document_Title
ON Production.Document (Title, Revision)
INCLUDE (FileName)
```

Non-key columns can be included only in non-clustered indexes. Columns can’t be defined in both the key column and they INCLUDE list. Column names can’t be repeated in the INCLUDE list. Non-key columns can be dropped from a table only after the non-key index is dropped first. For Included Column Index to exist there must be at least one key column defined with a maximum of 16 key columns and 1023 included columns.

Avoid adding unnecessary columns. Adding too many index columns, key or non-key as they will affect negatively on performance. Fewer index rows will fit on a page. This could create I/O increases and reduced cache efficiency. More disk space will be required to store the index. Index maintenance may increase the time that it takes to perform modifications, inserts, updates, or deletes, to the underlying table or indexed view.

Q: Another example to test:

Create following Index on Database AdventureWorks in SQL SERVER 2005

```
USE AdventureWorks
GO
CREATE NONCLUSTERED INDEX
IX_Address_PostalCode
ON Person.Address (PostalCode)
INCLUDE (AddressLine1, AddressLine2, City,
StateProvinceID)
GO
```

Test the performance of following query before and after creating Index. The performance improvement is significant.

```
SELECT AddressLine1, AddressLine2, City,
StateProvinceID, PostalCode
FROM Person.Address
WHERE PostalCode BETWEEN '98000'
AND '99999';
GO
```

Q: What are differences between Array list and Hash table?

Ans:
1) Hash table store data as name, value pair. While in array only value is store.
2) To access value from hash table, you need to pass name. While in array, to access value, you need to pass index number.
3) you can store different type of data in hash table, say int, string etc. while in array you can store only similar type of data.

Q: What are differences between system.stringbuilder and system.string?

The main difference is system.string is immutable and system.stringbuilder is a mutable. Append keyword is used in string builder but not in system.string.
Immutable means once we created we cannot modified. Suppose if we want give new value to old value simply it will discarded the old value and it will create new instance in memory to hold the new value.

Q: What are the differences between Application object and session object?
Ans: The session object is used to maintain the session of each user. If one user enter in to the application then they get session id if he leaves from the application then the session id is deleted. If they again enter in to the application they get different session id.
But for application object the id is maintained for whole application.

Q: What are the different types of indexes?
Ans: Two types of indexes are there one is clustered index and non-clustered index

Q: How many types of memories are there in .net?
Ans: Two types of memories are there in .net stack memory and heap memory

Q: Is it possible to set the session out time manually?
Ans: Yes we can set the session out time manually in web.config.

Q: What are differences between function and stored procedure?
Ans:
1) Function returns only one value but procedure returns one or more than one value.
2) Function can be utilized in select statements but that is not possible in procedure.
3) Procedure can have an input and output parameters but function has only input parameters only.
4) Exceptions can be handled by try catch block in procedures but that is not possible in function.

Q: What are the differences between Abstract and interface?
Ans: 1) Abstract cannot be instantiated but we can inherit. Interface it cannot be inherit it can be instantiate
2) Interface contain only declarations no definitions. Abstract contain declarations and definitions.
3) The class which contains only abstract methods is interface class. A class which contains abstract method is called abstract class
4) Public is default access specifier for interface we don't have a chance to declare other specifiers. In abstract we have chance to declare with any access specifier

Q: Can you Explain Page lifecycle in .net?
Ans:
-Page_Init: Page Initialization
-LoadViewState: View State Loading
-LoadPostData: Postback Data Processing
-Page_Load: Page Loading
-RaisePostDataChangedEvent: PostBack Change Notification
-RaisePostBackEvent: PostBack Event Handling
-Page_PreRender: Page Pre Rendering Phase
-SaveViewState: View State Saving
-Page_Render: Page Rendering
-Page_Unload: Page Unloading

Q: Can you Explain .NET architecture in .net?

Ans:
1) The .Net framework class library
2) CLR which is Common Language Runtime
3) Languages and
4) Web Technology (ASP.NET)

Q: What is the difference between primary key and unique key with not null?
Ans: There is no difference between primary key and unique key with not null.

Q: What is boxing and unboxing concepts in .net?
Ans: Boxing is a process of converting value type into reference type
Unboxing is a process of converting reference type to value type.

Q: What are the differences between value type and reference type?
Ans: Value type contain variable and reference type are not containing value directly in its memory.
Memory is allocated in managed heap in reference type and in value type memory allocated in stack. Reference type ex-class value type-struct, enumeration

Q: Is it possible to host the website from desktop?
Ans: Yes

Q: Why we go for page rendering in Asp.Net Page life cycle?
Ans: Browser understands an only html control that’s why in page rendering we will convert the aspx controls into html controls.

Q: Write a sample query for self join?
Ans: Select e1.ename, e2.empid from emp e1, emp e2 where e1.empid=e2.mgrid;

Q: Can we change the index of primary key on table?
Ans: No

Q: How to change the name of the table or stored procedure in sql?
Ans: sp_rename oldtablename newtablename
For changing the column name
Sp_rename ‘tablename.[Oldcolumnname]’,’newcolumnname’,’Column’
Ex:sp_rename ‘tblemp.first’,’namechange’,’Column’

Q: How to find out which index is defined on table?
Ans: sp_helpindex tablename

Q: Can you write the program to find the length of string without using library function?
Ans:
```
for (int i=0; str[i]!=”\n”; i++)
{
Count++;
}
```

Q: What is the difference between scope_identity() and current_identity()?
Ans: Scope_identity and current _identity both are similar and it will return the last identity value generated in the table.
Scope_Identity will return the identity value in table that is currently in scope

Q: What are difference between GET and POST Methods?
Ans:
GET Method ():
1) Data is appended to the URL.
2) Data is not secret.
3) It is a single call system
4) Maximum data that can be sent is 256.
5) Data transmission is faster
6) This is the default method for many browsers

POST Method ():
1) Data is not appended to the URL.
2) Data is Secret
3) It is a two call system.
4) There is no Limit on the amount of data. That is characters any amount of data can be sent.

5) Data transmission is comparatively slow.
6) No default and should be explicitly specified.

Q: What are difference between truncate and delete?
Ans: 1) Delete keeps the lock over each row where Truncate keeps the lock on table not on all the row.
2) Counter of the Identity column is reset in Truncate where it is not reset in Delete.
3) Trigger is not fired in Truncate where as trigger is fired in Delete.
4) In TRUNCATE we cannot rollback.
5) In DELETE we can rollback

Q: What is the difference Grid View and between Data Grid (Windows)?
Ans:
1) GridView Control Enables you to add sorting, paging and editing capabilities without writing any code.
2) GridView Control Automatically Supports paging by setting the ‘PagerSetting’ Property.The Page Setting Property supports four Modles
a. Numeric(by default)
b. Next Previous
c. NumericFirstLast
d. Next PreviousLast

3) It is Used in asp.net
4) GridView Supports RowUpdating and RowUpdated Events.
5) GidView is Capable of Pre-Operations and Post-Operations.
6) GridView Has EditTemplates for this control
7) It has AutoFormat

DataGrid(Windows)
1)DataGid Control raises single Event for operations
2)DataGird Supports the SortCommand Events that occur when a column is Soted.
3)DataGrid Supports UpdataCommand Event that occurs when the UpdateButton is clicked for an item in the grid.
4)DataGrid is used in Windows GUI Application.
5)It doesnot have EditTemplates for this control
6)It doesnot have AutoFormat

Q: If I write System.exit (0); at the end of the try block, will the finally block still execute?
Ans: No in this case the finally block will not execute because when you say system.exit(0),the control immediately goes out of the program, and thus finally never executes.

Q: What are the different levels of State management in ASP.NET?
Ans:
State management is the process by which you maintain state and page information over multiple requests for the same or different pages.
There are 2 types State Management:
1. Client – Side State Management
This stores information on the client's computer by embedding the information into a Web page, a uniform resource locator (url), or a cookie. The techniques available to store the state information at the client end are listed down below:
a. View State – Asp.Net uses View State to track the values in the Controls. You can add custom values to the view state. It is used by the Asp.net page framework to automatically save the values of the page and of each control just prior to rendering to the page. When the page is posted, one of the first tasks performed by page processing is to restore view state.
b. Control State – If you create a custom control that requires view state to work properly, you should use control state to ensure other developers don’t break your control by disabling view state.
c. Hidden fields – Like view state, hidden fields store data in an HTML form without displaying it in the user's browser. The data is available only when the form is processed.
d. Cookies – Cookies store a value in the user's browser that the browser sends with every page request to the same server. Cookies are the

best way to store state data that must be available for multiple Web pages on a web site.
e. Query Strings - Query strings store values in the URL that are visible to the user. Use query strings when you want a user to be able to e-mail or instant message state data with a URL.

2. Server – Side State Management
a. Application State - Application State information is available to all pages, regardless of which user requests a page.
b. Session State – Session State information is available to all pages opened by a user during a single visit.
Both application state and session state information is lost when the application restarts. To persist user data between application restarts, you can store it using profile properties.

Q: Abstract Class:
Ans: Abstract class is a class which can’t be instantiate. Class should have “Abstract” key word with the name. In any one of the method of class having abstract method with in it, then it should be define as abstract class. The class which derived the abstract class should have definition of the abstract method. These classes which derived the abstract class and implement the abstract methods call concrete class.
Abstract class may have the definition of function or may not. Below is the simple example of an abstract class

```
public abstract alass AbstractStudent
    {
        String Roll
        {
            get;
            set;
        }

        String FirstName
        {
            get;
            set;
        }

        String LastName
        {
            get;
            set;
        }

        Public String GetStudentDetails()
             {

                  // Implementation of Method
             }

        public String SaveStudentDetails ()
            {
                  // Implementation of Method
            }

        public abstract String CalculateWage();

    }
```

So, the class having one abstract method so we need to mention the class as "abstract" .

Q: Difference between Abstract Class and Interface?
Ans:
Abstract class is a class which can’t be instantiated and which can have methods with definition as well as declaration also. This can be inherited.
As for Example:

```
public abstract class AbstractStudent
    {
        String Roll
        {
            get;
            set;
        }

        String FirstName
        {
           get;
           set;
        }
        String LastName
        {
           get;
            set;
        }

        Public String GetStudentDetails()
            {
                 // Implementation of Method
             }

        public String SaveStudentDetails ()
            {
                  // Implementation of Method
            }

        public abstract String CalculateWage();

    }
```

Interface can only contain the methods declaration and can be implemented in the class.

As for Example:

```
Public interface IStudnet
    {
        String Roll
        {
           get;
            set;
        }

        String FirstName
        {
            get;
            set;
        }

        String LastName
        {
            get;
            set;
        }

        String GetStudentDetails();
        String SaveStudentDetails ();
    }
```

Below are the few main difference between Abstract Class and Interface

a. In abstract class method can have definition as well as declaration also. But Interface should have only definition.

b. All the Methods are Public as default and don’t have any access Modifier Controls in interface, whereas for abstract class we can have access modifier for methods.

c. Abstract class can have constructor or destructor, whereas interface not.

d. Abstract class can’t be part of multiple inheritance and we can implement multiple interface.

Q: What do you mean by String objects are immutable?

Ans: String objects are immutable as its state cannot be modified once created. Every time when we perform any operation like add, copy, replace, and case conversion or when we pass a string object as a parameter to a method a new object will be created.

Example:

```
String str = "ABC";
str.Replace("A","X");
```

Here Replace() method will not change data that "str" contains, instead a new string object is created to hold data "XBC" and the reference to this object is returned by Replace() method. So in order to point str to this object we need to write below line.

```
str = str.Replace("A","X");
```

Now the new object is assigned to the variable str. earlier object that was assigned to str will take care by garbage collector as this one is no longer in used.

Q: What is dll hell problem in .NET and how it will solve?

Ans: dll hell, is kind of conflict that occurred previously, due to the lack of version supportability of dll for (within) an application

.NET Framework provides operating system with a global assembly cache. This cache is a repository for all the .net components that are shared globally on a particular machine. When a .net component installed onto the machine, the global assembly cache looks at its version, its public key and its language information and creates a strong name for the component. The component is then registered in the repository and indexed by its strong name, so there is no confusion between the different versions of same component, or DLL

Q: What is a Partial class?
Ans: Instead of defining an entire class, you can split the definition into multiple classes by using partial class keyword. When the application compiled, c# compiler will group all the partial classes together and treat them as a single class. There are a couple of good reasons to use partial classes. Programmers can work on different parts of classes without needing to share same physical file
Ex:

```
Public partial class employee
{
Public void somefunction()
{
}
}
Public partial class employee
{
Public void function ()
{
}
}
```

Q: What is difference between constants, read-only and, static?
Ans: Constants: The value can’t be changed
Read-only: The value will be initialized only once from the constructor of the class.
Static: Value can be initialized once.

Q: What is the cross page post backing?
Ans: Asp.Net 2.0 fixed this with built-in features that allowed us to easily send information from one page to another.

Button control has property PostBackUrl that can be set to URL of any page in our ASP.Net WebSite where we want to transfer all form values to.
Along with that Asp.Net 2.0 Page class has a property PreviousPage that allows us to get reference to the Page object that initiated the postback (in other words to get the actual reference to the Page object of the aspx page on which user clicked the Submit button on a HTML form).

So for example lets create two sample pages in our Web Application:
SourcePage.aspx
DestinationPage.aspx
In SoucePage in Html form we will put two TextBox controls (one for First Name and one for Last Name) and one Button component and set its PostBackUrl property to "~/DestinationPage.aspx".

SourcePage.aspx:

```
    <form id="form1" runat="server">
        <div>
            First Name:&nbsp;<asp:TextBox ID="FirstName" runat="server"></asp:TextBox><br />
            Last Name:&nbsp;<asp:TextBox ID="LastName" runat="server"></asp:TextBox><br /><br />
            <asp:Button ID="Button1" runat="server" Text="Submit To Destination Page" PostBackUrl="~/CrossPagePostbacks/DestinationPage.aspx" />
        </div>
    </form>
```

When our user clicks the Submit button, all the values from the HTML Form on SourcePage.aspx will be transfered to the DestinationPage.aspx and we will also be able to get reference to the SourcePage.aspx in our DestinationPage with the PreviousPage property like this:

So in our DestinationPage.aspx.cs code-behind we can easily access two TextBox controls on SourcePage.aspx and show them in two label controls like this:

```
    protected void Page_Load(object sender, 
EventArgs e)
    {
        // first check if we had a cross page 
postback
        if ( (PreviousPage != null) && 
(PreviousPage.IsCrossPagePostBack) )
        {
            Page previousPage = PreviousPage;
            TextBox firstName = 
(TextBox)previousPage.FindControl("FirstName"
);
            TextBox lastName = 
(TextBox)previousPage.FindControl("LastName"
);
            // we can now use the values from 
TextBoxes and display them in two Label 
controls..
            labelFirstName.Text = firstName.Text;
            labelLastName.Text = lastName.Text;
         }
    }
```

You probably noticed that we first checked if PreviousPage property of current page (DestinationPage.aspx) is NOT NULL, this is done to avoid running our code in case that user opens our DestinationPage.aspx directly, without running a cross page postback.
Also here we checked the another PreviousPage property called IsCrossPagePostBack to see if we really had a CrossPagePostback.
(If Server.Transfer is used to redirect to this page, IsCrossPagePostBack property will be set to FALSE.
TIP: We can be completely sure that we have a real CrossPagePostback ONLY IF:
Page.PreviousPage is NOT NULL,
PreviousPage.IsCrossPagePostback is true
This important to check to avoid errors in code.
Now this is very useful and i'm sure you are eager to use this in your next project. But wait, we are not over yet!
Finding the controls on PreviousPage with FindControl method and type-casting them from object to their real type is a little messy.
It feels like there must be a better solution for this!
And here it is: We can use the <%@ PreviousPageType %> directive in the header of our DestinationPage.aspx like this

```
    <%@ PreviousPageType 
VirtualPath="~/SourcePage.aspx" %>
```

To declare our previous page type, and then we can access Public properties of the PreviousPage without typecasting.
Now all we need to do is to create some public properties on our SourcePage.aspx.cs to expose data/Controls we want to the destionation page:

```
    public partial class SourcePage : 
System.Web.UI.Page
    {
        public string FormFirstName
        {
            get { return FirstName.Text; }
        }

        public string FormLastName
        {
            get { return LastName.Text; }
        }
    }
```

And then we can change the Page_Load code in our DestinationPage.aspx to much cleaner code like this:

```
    protected void Page_Load(object sender, 
EventArgs e)
    {
        // first check if we had a cross page 
postback
        if ( (PreviousPage != null) && 
(PreviousPage.IsCrossPagePostBack) )
```

```
        {
            SourcePage prevPage = PreviousPage;

            // we can now use the values from textboxes and display them in two Label controls..
            labelFirstName.Text = prevPage.FormFirstName;
            labelLastName.Text = prevPage.FormLastName;
        }
    }
```

SourcePage type used in the code is offcourse name of the partial class defined is SourcePage.aspx.cs that inherits System.Web.UI.Page that is automatically created for us when we created new WebForm in VisualStudio.
This code is much cleaner and easier to follow, there is no ugly typecasting, just simple property values to use to retrieve the data from previous page.

Q: When should you use Abstract Class vs Interface while programming?
Ans: When we want that sub class must implement all the methods of base class. In such a situation we will implement the interface. In the other hand when we want only some method of base class in our sub class then use base class as abstract class.

Q: What is the difference between application exception and system exception?
Ans: The difference between application exception and system exception is that system exceptions are thrown by CLR and application exceptions are thrown by applications.

Q: What is the difference between authorization and authentication?
Ans: Authorization is a process of allowing or denying resources to particular user or record
Declaration of authorization is

```
<authorization>
<allow users=”Suresh, Sanjay”/>
<deny users=”Ramana, Rakesh”>
</authorization>
```

Sometimes authorization allows the unauthorized persons at that time we will use
<deny users=”?”/>

Authentication means
Authentication is a process where we identify the credentials of user i.e. username, password and create an identity to mention user as an authenticated.

Q: What is the use of n-tier architecture and 3-tier architecture?
Ans: Basically 3-Tier architecture contains 3 layers

1. Application Layer or Presentation Layer
2. Business Access Layer(BAL) or Business Logic Layer(BLL)
3. Data Access Layer(DAL)

Q: How to get the version of the assembly?
Ans: lbltxt.text=Assembly. GetExecutingAssembly().GetName().Version.ToString();

Q: What is the location of Global Assembly Cache on the system?
Ans: c:\Windows\assembly

Q: What is the serialization?
Ans: Serialization is a process of converting object into a stream of bites.

Q: What is synchronization?
Ans: The mechanism needed to block one thread access to the data. If the data is being accessed by another thread.
Synchronization can be accessed by using system.monitor class
A monitor class methods are enter, exit, pulse for this lock statement is also used
Suppose if we need to synchronize some data at that time we need to place that data in this block

```
Lock
{
}
```

Whatever the data has been placed into the lock block that data has been blocked

Q: What are the thread priority levels?
Ans: Thread priority levels are five types

- 0 - Zero level
- 1 - Below Normal
- 2 - Normal
- 3 - Above Normal
- 4 - Highest

By Default priority level is 2

Q: What is the difference between .tostring(), Convert.tostring()?
Ans: The basic difference between them is "Convert" function handles NULLS while ".ToString()" does not it will throw a NULL reference exception error. So as a good coding practice using "convert" is always safe.

Q: What is Collation?
Ans: Collation refers to a set of rules that determine how the data is sorted and compared.

Q: What is the difference between Primary key and unique key?
Ans: Primary key does not allow the null values but unique key allows one null value.
Primary key will create clustered index on column but unique key will create non-clustered index by default.

Q: How many web.config files are there in 1 project?
Ans: There might be multiple web.config files for a single project depending on the hierarchy of folders inside the root folder of the project, so for each folder we can use one web.config file

Q: What is the difference between throw and throw ex?
Ans: throw ex resets the stack trace (so your errors would appear to originate from HandleException)
throw doesn't - the original offender would be preserved.

Q: What is the difference between view state and hidden field?
Ans: viewstate is secured hidden field is insecure
Viewstate will store large amount of data but hidden filed will store small amount of data.

Q: What is the difference between binary serialization and xml serialization?
Ans: The reason a binary formatter is able to deserialize directly to an interface type is because when an object is originally serialized to a binary stream metadata containing type and assembly information is stuck in with the object data. This means that when the binary formatter deserializes the object it knows its type, builds the correct object and you can then cast that to an interface type that object implements.
The XML serializer on the otherhand just serializes to a schema and only serializes the public fields and values of the object and no type information other than that (e.g. interfaces the type implements).
Here is a good post, .NET Serialization, comparing the BinaryFormatter, SoapFormatter, and XmlSerializer. I recommend you look at the following table which in addition to the previously mentioned serializers includes the DataContractSerializer, NetDataContractSerializer and protobuf-net.

Q: What is the Difference between read only and constant variables?
Ans: Read only can assign the values at runtime only.
Constant will assign the values at compile time only.
We cannot modify the both variable values.

Q: What is static keyword in .Net?
Ans: Static is same as constant variable but we can change the value of static variable and we can access the variables without creating any instances

Q: What is the use of business logic layer in 3-tier architecture in .net?
Ans: Though a web site could talk to the data access layer directly, it usually goes through another layer called the business layer. The business layer is vital in that it validates the input conditions before calling a method from the data layer. This ensures the data input is correct before proceeding, and can often ensure that the outputs are correct as well. This validation of input is called business rules, meaning the rules that the business layer uses to make "judgments" about the data.
However, business rules don't only apply to data validation; these rules apply to any calculations or any other action that takes place in the business layer. Normally, it's best to put as much logic as possible in the business layer, which makes this logic reusable across applications.
One of the best reasons for reusing logic is that applications that start off small usually grow in functionality. For instance, a company begins to develop a web site, and as they realize their business needs, they later decide to add a smart client application and windows service to supplement the web site. The business layer helps move logic to a central layer for "maximum reusability."