BIYANI
Where you can trust

Biyani Girls College

I Internal Examination Sept. 2019

Class: - BCA III

Subject: - ASP .NET

MM: 40 Set: B Time: 1 ½ Hrs.

[I] Very short answer questions (Max 40 words). (5 * 2 = 10)

1. What do you mean by Constant?

   Ans:-A constant is a type of field or local variable whose value is set at compile time and can never be changed at run time. It is similar to a variable by having a name, a value, and a memory location.

2. Define web service?

   Ans:-A web service is a web-based functionality accessed using the protocols of the web to be used by the web applications. There are three aspects of web service development:

   Creating the web service

   Creating a proxy

   Consuming the web service

3. What do you mean by MultiView ?

   Ans:-MultiView and View controls allow you to divide the content of a page into different groups, displaying only one group at a time. Each View control manages one group of content and all the View controls are held together in a MultiView control.

   The MultiView control is responsible for displaying one View control at a time. The View displayed is called the active view.

4. What do you mean by Metadata?

   Ans:-Metadata is machine-readable information about a resource, or "data about data." Such information might include details on content, format, size, or other characteristics of a data source. In .NET, metadata includes type definitions, version information, external assembly references, and other standardized information.

5. Describe Operators in C#?

   Ans:-An operator is a symbol that tells the compiler to perform specific mathematical or logical manipulations. C# has rich set of built-in operators and provides the following type of operators −

   Arithmetic Operators

   Relational Operators

   Logical Operators

   Bitwise Operators

   Assignment Operators

   Misc Operators

[II] Short answer questions (Max 80 words). (2 * 5 = 10)

1. Describe Validation controls with a example of Registration Page?

Ans:-ASP.NET validation controls validate the user input data to ensure that useless, unauthenticated, or contradictory data don't get stored.

ASP.NET provides the following validation controls:

i) RequiredFieldValidator
ii) RangeValidator
iii) CompareValidator
iv) RegularExpressionValidator
v) CustomValidator
vi) ValidationSummary
vii) RequiredFieldValidator Control

i) The RequiredFieldValidator control ensures that the required field is not empty. It is generally tied to a text box to force input into the text box.

The syntax of the control is as given:

```
<asp:RequiredFieldValidator ID="rfvcandidate"
   runat="server" ControlToValidate ="ddlcandidate"
 ErrorMessage="Please choose a candidate"
  InitialValue="Please choose a candidate">
</asp:RequiredFieldValidator>
```

ii) RangeValidator Control

The RangeValidator control verifies that the input value falls within a predetermined range.

```
<asp:RangeValidator ID="rvclass" runat="server" ControlToValidate="txtclass"
   ErrorMessage="Enter your class (6 - 12)" MaximumValue="12"
   MinimumValue="6" Type="Integer">
</asp:RangeValidator>
```

iii) CompareValidator Control

The CompareValidator control compares a value in one control with a fixed value or a value in another control.

```
<asp:CompareValidator ID="CompareValidator1" runat="server"
   ErrorMessage="CompareValidator">
</asp:CompareValidator>
```

iv) RegularExpressionValidator

The RegularExpressionValidator allows validating the input text by matching against a pattern of a regular expression. The regular expression is set in the ValidationExpression property.

```
<asp:RegularExpressionValidator ID="string"  runat="server"
ErrorMessage="string"   ValidationExpression="string"
ValidationGroup="string">
   </asp:RegularExpressionValidator>
```

v) CustomValidator

The CustomValidator control allows writing application specific custom validation routines for both the client side and the server side validation.

```
<asp:CustomValidator ID="CustomValidator1" runat="server"
  ClientValidationFunction=.cvf_func. ErrorMessage="CustomValidator">
</asp:CustomValidator>
```

vi) ValidationSummary

The ValidationSummary control does not perform any validation but shows a summary of all errors in the page. The summary displays the values of the ErrorMessage property of all validation controls that failed validation.

```
<asp:ValidationSummary ID="ValidationSummary1" runat="server"
  DisplayMode = "BulletList" ShowSummary = "true" HeaderText="Errors:" />
```

2. Difference between TextBox and ListBox?

Ans:- Textbox control is most usable web server control in asp.net.

TextBox control is a rectangular box which is used to take user to input. In simple word the TextBox is a place where user can input some text on asp.net web form. To use TextBox on page we can write code or just drag and drop from toolbox.

```
<asp:TextBox ID=”TextBox1″ runat=”server”></asp:TextBox>
```

The ListBox control is similar to the DropDownList control with two important differences. First, the ListBox control requires more screen real estate because it always displays a certain number of list items. Furthermore, unlike the DropDownList control, the ListBox control enables a user to select multiple items.

```
<asp:ListBox
  id="lstProducts"
  DataSourceID="SqlDataSource1"
  DataTextField="TITLE"
  DataValueField="ID"
  Rows="4"
  Runat="server" />
```

[III] Long answer questions (Max 150 words) (2 * 10 = 20)

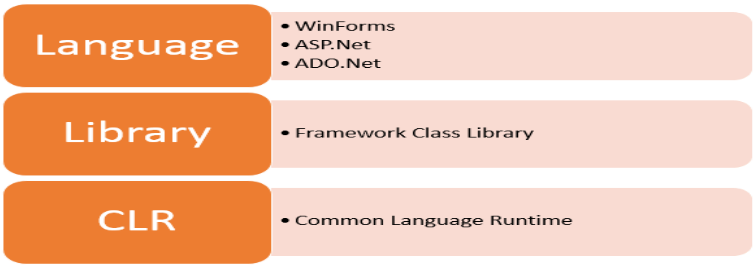
1. What do you mean by .Net? Describe .Net architecture in detail?

Microsoft .NET architecture is the programming model for the .NET platform. The .NET Framework provides a managed execution environment, simplified development and deployment and integration with a wide variety of programming languages.

The .NET Framework has two key parts:

The **.NET Framework class library** is a comprehensive, object-oriented collection of reusable types that you can use to develop applications. The .NET Framework class library includes ADO.NET, ASP.NET, and Windows Forms.

The common language runtime (CLR) is the core runtime engine for executing applications in the .NET Framework. You can think of the CLR as a safe area - a "sandbox" - inside of which your .NET code runs. Code that runs in the CLR is called managed code. It is fully protected from the outside environment and highly optimized within, taking advantage of the services that the CLR provides such as security, performance, deployment facilities, and memory management, including garbage collection.



Components :

CLR (Common Language Runtime) : It is a run-time environment which executes the code written in any .NET programming language. .Net framework provides the support for many languages like C#, F#, C++, Cobra, Jscript.Net, VB.Net, Oxygene etc.

FCL (Framework Class Library) : A large number of class libraries are present in this framework which is known as FCL.

Types of Applications : Mainly the applications which are built in .Net framework is divided into the following three categories :

WinForms : Form – Based applications are considered under this category. In simple terms, we can say client based applications which read and writes the file system comes under this category.

ASP .NET : Web-Based applications come under this category. ASP.Net is a framework for web and it provides the awesome integration of HTML, CSS and JavaScript which makes it useful to develop the web applications, websites and web services. Web services were added in .Net Framework 2.0 and considered as a part of ASP.NET web applications.

ADO .NET : It includes the application which are develop ed to communicate with the database like MS SQL Server, Oracle etc. comes. It mainly consists of classes that can be used to connect, retrieve, insert and delete data.

Managed Languages and Common Language Specification

.NET supports managed and unmanaged programming languages. Applications created from managed languages, such as C# and VB.NET, execute under the management of a common runtime, called the common language runtime.

There are several differences between a compiled managed application and an unmanaged program.

Managed applications compile to Microsoft Intermediate Language (MSIL) and metadata. MSIL is a low-level language that all managed languages compile to instead of native binary. Using just-in-time compilation, at code execution, MSIL is converted into binary optimized both to the environment and the hardware. Since all managed languages ultimately become MSIL, there is a high degree of language interoperability in .NET.

Metadata is data that describes data. In a managed application, also called an assembly, metadata formally defines the types employed by the program.

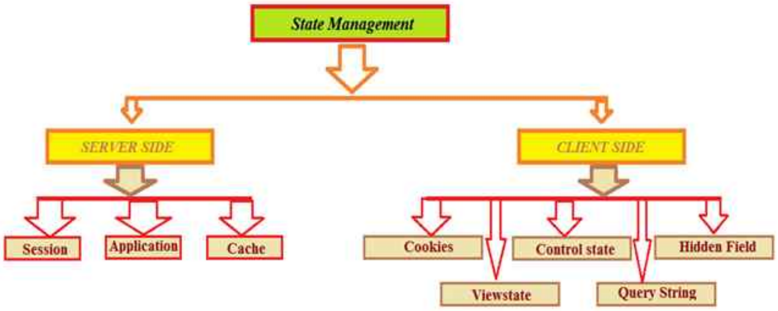
Wave a fond goodbye to the Registry. Managed applications are sweeping away the Registry, Interface Definition Language (IDL) files, and type libraries with a single concept called metadata. Metadata and the related manifest describe the overall assembly and the specific types of an assembly.

Managed applications have limited exposure to the unmanaged environment. This might be frustrating to many programmers, particularly experienced C gurus. However, .NET has considerable flexibility. For those determined to use unmanaged code, there are interoperability services.

2. What Do you mean by state Management? Describe Client side state management?

State management is a preserve state control and object in an application because ASP.NET web applications are stateless. A new instance of the Web page class is created each time the page is posted to the server. If a user enters information into a web application, that information would be lost in the round trip from the browser (MSDN).

In a single line, State management maintains and stores the information of any user till the end of the user session.


State Management

SERVER SIDE

CLIENT SIDE

Session

Application

Cache

Cookies

Control state

Hidden Field

Viewstate

Query String

Client Side statemanagement

1) Cookie

Cookie is a small and an important part of ASP.NET. In this store user information, session and application. It can be created constant and temporary and they work with browser request. Cookies are store on client side. The server can read cookies and abstract data.

Two types of cookies are available-

i) Persistence

This type of cookie works with Date and time.

```
 Response.Cookies["CookieName"].Value = "Test Cookies";
//set expire time
Response.Cookies["CookieName"].Expires = DateTime.Today.AddHours(1);
```

ii) Non-Persistence

This is a temporary cookie. It is created with access application and discards the close application.

```
 Response.Cookies["CookieName"].Value = "Test Cookies";
```

2) Control state

Control state technique is developed to maintain data work properly in order. We can use view state but suppose view state is disabled by the user, the control will not work as expected. For expected results of the control we have to use Control State. In application, the Viewstate is by default true. Sometimes we need to use custom control to manage application properly.

```
if (!IsPostBack)
{
  lblmsg1.Text = "Welcome to C# corner";
  lblmsg2.Text = "Welcome to C# corner community";
}
```

When two messages are displayed on a Postback event, then control which one is displayed by using customized control state.

3) Hidden Field

Hidden fields are used to store value to client side. Hidden field is not displayed on the browser, but it works on a request.

```
if (HiddenField1.Value != null)
{
  int val = Convert.ToInt32(HiddenField1.Value) + 1;
  HiddenField1.Value = val.ToString();
  Label1.Text = val.ToString();
```

4) Viewstate

Viewstate is a very useful client side property. It is used for page level state management. Viewstate stores any type of data and used for sending and receiving information, Viewstate is easy to apply and does not need access to any server resources. In a Viewstate, do not store big data, only store small values. Viewstate enables and disables on page level control. It also supports Encryption and Decryption and data/value is stored in hashed format. So we are not storing important data such as password, account information, etc. When more data is stored in this, then the page becomes heavy.

5) Query String

Query string stores the value in URL.

```
Response.Redirect("ShowStringValue.aspx?Username=" + txtUsername.Text);
```