WEB TECHNOLOGY

UNIT 3

Multiple Choice Questions:

1. Web services can be discovered using _______
   a. UDDDI
   b. UDII
   c. UDDII
   d. **UDDI**
2. ____is the basis for Web services.
   a. PHP
   b. **XML**
   c. CSS
   d. CGI
3. Which of the following is considered as Web Service Platform Elements?
   a. SOAP
   b. WSDL
   c. UDDI
   d. All of these..
4. WSDL Stands for :
   a. Web Services Design Language
   b. Web Services Description Language
   c. Web Services Development Language
   d. None of these
5. UDDI Stands for .
   a. Universal Description, Development and Integration
   b. None of these
   c. Universal Description, Discovery and Integration
   d. Universal Development, Design and Integration
6. is used to convert your application into Web-Application.
   a. Java Service
   b. Struts Services
   c. Web Services
   d. Browser Action
7. Which object data is included in bookmarks and e-mailed URLs?
   a. ViewState
   b. cookies
   c. Query strings.
   d. All of the above
8. Which object in ASP.NET provides a global storage mechanism for state data that needs to be accessible to all pages in a given Web application?
   a. Session
   b. **Application**.
   c. ViewState
   d. None of the above
9. What is/are the advantages of Session State?
   a. It helps to maintain user data to all over the application and can store any kind of

object.
b. Stores every client data separately.
c. Session is secure and transparent from user.
d. All of the above.

10. Which file is used to write the code to respond to the Application_Start event?
a. Any ASP.NET web page with an .aspx extension
b. Web.config
c. Global.asax.
d. None of the above.

11. When a User‘s Session times out which event should you respond to?
a. Application_Start
b. Session_End.
c. Session_Start
d. Application_End

12. What are the client-side state management options that ASP.NET supports?
a. Application
b. Session
c. Querystring.
d. Option a and b are correct

13. What type of data can Cookies store?
a. **String**.
b. DateTime
c. System.Int32
d. None of the above.

14. What are the types of cookies?
a. Session cookies
b. Persistent cookies
c. Dummy cookies
d. Option A and B are correct.

15. From the following which is not a valid state management object?
a. Querystate.
b. Cookies
c. Application state
d. Hidden form fields

16. What ASP.NET object encapsulates the state of the client and the browser?
a. Application Object
b. Session Object.
c. Response Object
d. Request Object

17. How many types of authentication ASP.NET supports?
a. Windows Authentication
b. .NET Passport Authentication.
c. Forms Authentication.
d. All of the above.

18. In which file you should write the connection string, so you can access it in all the web page
for same application?
a. In App_Data folder.
b. In Web.config file.
c. In MasterPage file.

d. None of the above.

19. Where do we include the user lists for Form authentication?
    a. < credential>.
    b. < authorization>
    c. < Identity>
    d. < authentication>

20. Web services communicate using .
    a. Open protocols.
    b. Open-Close protocols
    c. Close protocols
    d. None of these

21. Web Services are .
    a. Application Components
    b. Application Designing Tool
    c. None of these
    d. Application IDE

22. Long form of SOAP is .
    a. Simple Object Access Protocol.
    b. None of these
    c. Secure Object Access Protocol
    d. Simplified Object Arbitary Protocol

23. Web-applications were developed in order to interact .
    a. Different platforms.
    b. Server
    c. Browsers
    d. None of these

24. Web Services is used to -
    a. Connect existing software
    b. Reuse application-components.
    c. Solve the interoperability problem
    d. All of these

25. SOAP is a .
    a. Carrier
    b. Markup
    c. Protocol.
    d. Language

Long Answer Questions:

1. What is web service? What are the benefits of Web Service.

The term "web service" describes a standardized way of integrating web-based applications using the XML, SOAP, WSDL and UDDI open standards over an Internet Protocol backbone. XML is the data format used to contain the data and provide metadata around it, SOAP is used to transfer the data, WSDL is used for describing the services available and UDDI lists what services are available.

A Web service exposes a number of methods to provide functions that can be used by one or more applications, regardless of the programming languages, operating systems, and hardware platforms used to develop them. The methods that provide such function are called Web methods.

Web services provide the following benefits:

- Easy accessibility: Any computer that has access to the Internet can easily access a Web service. This enables a number of applications residing on different software and hardware platforms to exchange data between each other.
- Flexibility in structure: Depending on the requirements of a business, different types of Web services can be created and used in an application. For example, you can create simple Web services that provide a fundamental functionality that can be used in multiple applications. You can also create Web service to integrate the existing applications that might have been created using different software and hardware platforms.
- Easy Integration: Using Web services, you can easily integrate business applications developed on different software and hardware platforms resulting in low setup costs. In addition, you can integrate Web applications with applications such as spreadsheets or back-end database to provide online information.

2. What is web service? What are the elements in web service.

A Web service has the following elements:

XML: XML is used to exchange data over the Internet. To enable data interchange, you require a standard data representation format that can be understood by any platform. Because XML is a plain-text format that can be understood by any type of hardware device over the Web, it is able to fulfil this requirement.

SOAP: To be able to communicate with each other, a Web service and a client application must agree upon a common protocol. SOAP is a standard communication protocol for interchanging information in a structured format in a distributed environment. The information exchanged between the client application and the Web service is called a message. Messages include the calls made by the client application to a web method and the data returned by the Web method to be client. When a client application makes a request for a Web method, a SOAP packet is created. This packet contains the name of the Web method to be invoked and the parameters to be passed to the Web method in an XML format. This information is used to invoke the Web method with the appropriate parameters.

Web Services Description Language (WSDL): To be able to use a Web service, the developers of a client application need to know the methods exposed by the Web service and the parameters to be passed to these methods. For this reason, you need a standard method to describe the methods that are called by a Web service. This information should be readily accessible to the Web service clients during the design phase. WSDL is a markup language that describes a Web service.

Universal Description, Discovery and Integration (UDDI): It provides a standard mechanism to register and discover a Web service. When a Web provider wants to make a Web service available to client applications, the provider describes the Web service by using a WSDL document. Then, the provider registers the Web service in the UDDI Directory, which contains pointers to the Web service and the WSDL document for the Web service

3. What are the controls that can be used for managing states of ASP.NET Application.

An application can have three types of states: application, session and view. Before going into detail about these three types of application states, let us have a look at the various means of storing state information:

- HiddenField control: The HiddenField control is not visible when the application is viewed in the browser; the content of the control is sent in the HTTP Form collection control along with the values of other controls to the server on page reload; this control acts as a storage area for storing any page specific information.
- Cookies: These are the text files that store data, such as user ID and preferences at the client end; when a browser again requests the same web page, the cookie is sent along with the request; the Web server then retrieves the information from the cookie.
- Query Strings: These are the information strings added at the end of a URL to mention the state of a Web application; however, the use of query string is not secure because the query string values are exposed to the internet through the URL.

4. Write a Short note on Session State.
   Each client accessing a Web application maintains distinct session with a Web server and there is also specific information associated with each of these sessions. Session-state is used to store this information. The session-state is defined in the section of the web.config file and stores the data specific to a user session in session variables. Different session variables are created for each user session. Also, the session variables can be accessed from any page of a Web application. When a user accesses a web page a session ID for the user is created. The session ID is transferred between the server and the client over the HTTP protocol using cookies.

5. Write a Short note on View State.

   The view-state stores the page-specific information when a page is posted back to the server. When a page is processed, the current state of the web page and control is hashed into a string and saved as a hidden field. Such a state of web page is called view-state and is defined as a hidden field on a web page.

   The ViewState property is used to save the view-state for each control, such as HTML controls and Web controls used in a web page. If the ViewState property is not used, then the values written in various controls, such as Textbox and Checkbox on the web page are not retained when the page is reloaded.

   The hidden field of a view-state is placed at the top of each page. The source code to maintain a view-state contains an input tag with three attribute, type, name and value. The type attribute is said to hidden. The name attribute contains the name of the controls, such as Web and HTML, for which the ViewState property is being used. The value attributes specifies a string value, which is not in human readable format. The string value is a combination of control values that are used in a web page. Therefore when the web page is reloaded, the server reads the values in the value attribute and restores the values of the controls when the page is sent back to the client

6. What is the use SOAP in ASP.NET Web Service.

To be able to communicate with each other, a Web service and a client application must agree upon a common protocol. SOAP is a standard communication protocol for interchanging information in a structured format in a distributed environment. The information exchanged between the client application and the Web service is called a message. Messages include the calls made by a client application to a Web method and the data returned by the Web method to the client. When a client application makes a request for a Web method, a SOAP packet is created. This packet contains the name of the Web method to be invoked and the parameters to be passed to the Web method in an XML format. This information is used to invoke the Web method with the appropriate parameters. When the SOAP packet arrives at the Web server on which the Web service resides, the Web method name and its parameters are extracted from the SOAP packet and the appropriate Web method is invoked.

7. Write a short note on WSDL.

To be able to use a Web service, the developers of a client application need to know the methods exposed by the Web service and the parameters to be passed to these methods. Therefore, you need a standard method to describe the methods that are exposed by a Web service. This information should be readily accessible to the Web service clients during the design phase. This is achieved by using an XML vocabulary called Web Services Description Language(WSDL). WSDL is a markup language that describes a Web service. A WSDL document contains the following information about a Web service:

- The Web service available for a given Web site
- The purpose for which these service can be used
- The types of parameters that need to be passed to a Web service
- The types of value that is returned by a Web service
- The format used to access these Web services
- The URL at which a We service can be accessed

8. Write a short note on UDDI.

UDDI provides a standard mechanism to register and discover a Web service. When a Web service provider wants to make a Web service available to client applications, the provider describes the Web service by using a WSDL document. Then, the provider registers the Web service in the UDDI Directory. The UDDI Directory contains pointers to the web service and the WSDL document for the Web service. Client application can then discover the Web service by using the UDDI Directory.

9. Explain different phases of development of application

There are four phases in the development of an application:

- Designing
- Implementing
- Testing
- Deployment

After the designing, implementing and testing phases are over; the application is ready for production. Deployment can be viewed as the process of installing the application from the development computer to the production computer. In case of ASP.NET ,the production computer refers to the server on which the ASP.NET application will be hosted

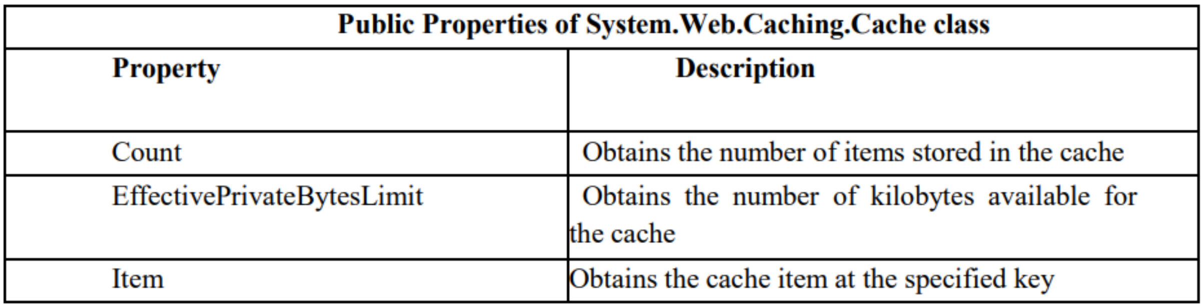
10. What are the types of deployment of Projects.

The different types of deployment projects available in visual studio.NET are as follows:

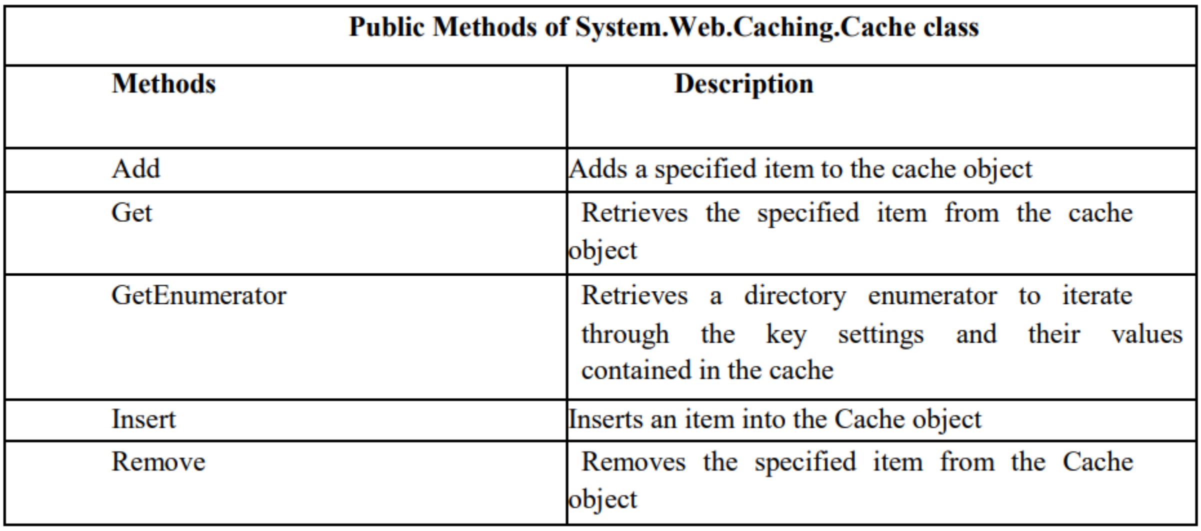
- Setup project: Help to create a MSI file that can be used for deploying an application. The MSI file created contains the application, installation instructions, information about the registry entries to be made, and dependencies. When this MSI file gets executed, all the components are copied to the Program Files directory on the destination computer. Setup projects are used to deploy the windows form application.
- Web Setup project: Help to create a MSI file. It is very similar to the Setup project, except that when the MSI file gets executed, all the components are copied to the virtual root directory in IIS and the web application is registered on the Web server.
- Merge module project: Helps to unite all files, registry entries, resources, and the setup logic required to deploy the package as one unit. Merge module allows multiple applications to share the setup code. It identifies all the dependencies for a component and makes sure that the correct versions of the components are installed. A merge module project enables you to solve the problem related to versioning. When a new version of any component needs to be deployed, a new merge module project can be created. A new merge module project contains the dependencies for the new version of the component.
- CAB project: Helps to generate CAB files to bundle the ActiveX controls. These CAB files can then be used to download components form the Web server to a Web browser. If it is required that the components should run on the client side, a CAB project should be created. A CAB project does not have any editors. The Solution Explorer can be used to add application components to a CAB file. After an application component is added, you can set properties, such as the level of compression and the location of dependent file on the Web, for each component by using the Properties window. This type of deployment project creates a file with the extension, .msm. the .msm file cannot be used independently. For this reason, it must be merged with the project that creates the MSI file

11. Write a short note on Caching.

Caching is a technique to store frequently accessed Web pages (or data), temporarily in memory that are faster to access than accessing it from the original source. For example, when you access a Web page for the first time, it takes a while to load. Subsequent visits to that Web page will be a lot faster because the Web page has been cached on your computer. You may cache a whole Web page, parts of a Web page, or application data according to your requirements. However, you may not want certain items to be cached independently. For example, you are running a news website and you want the content on your website to be refreshed at regular intervals. You may want the content on your website to refresh every half an hour, ten minutes, or even every minute. System.Web.Caching.Cache class is used to implement cache for a Web application in ASP.NET. This class can not be inherited. Some important public properties and public methods of the System.Web.Caching.Cache class and given in the following table



Public Properties of System.Web.Caching.Cache class	
Property	**Description**
Count	Obtains the number of items stored in the cache
EffectivePrivateBytesLimit	Obtains the number of kilobytes available for the cache
Item	Obtains the cache item at the specified key



Public Methods of System.Web.Caching.Cache class	
Methods	**Description**
Add	Adds a specified item to the cache object
Get	Retrieves the specified item from the cache object
GetEnumerator	Retrieves a directory enumerator to iterate through the key settings and their values contained in the cache
Insert	Inserts an item into the Cache object
Remove	Removes the specified item from the Cache object

Caching Techniques ASP.NET 2.0 provides three types of caching techniques:

- Output Caching (also called Page Level Caching)
- Partial-Page Output Caching
- Data Caching

12. Write a note on output caching (Page Level Caching).

Output caching or Page Level caching is the simplest technique of caching in ASP.NET. In output caching, if a Web page is requested by the client machine for the first time, it is accessed from the Web Server and sent to the client machine; also a copy of the Web page is stored in the cache. Now if the same Web page is requested further, the Web page is not accessed from the server instead the copy of the Web page which is stored on the cache is sent to the client machine. If a copy of the Web page is not found in the cache, the Web page is accessed from the server and sent to the client machine; also a copy of the Web page is stored in the cache for later use.

You can easily implement Output caching by just adding the @ OutputCache directive in the Web page as follows:

<%@ OutputCache Duration =“10” VaryByParam=”none” %>

You can also define as to where you want to cache the data either on the client-side, or the server-side or the proxy server by including a location attributes:

<%@ OutputCache Duration="90" Location ="Any | Client |" -

+"Downstream | Server None" VaryByParam="none" %>

13. Write a note on Partial Page output caching (Fragment Caching).
    It is not always that you require caching the complete page. Sometimes you need to implement caching on only some of the contents of a Web page. Other contents of the Web page will be refreshed each time the page is requested. This is called partial caching or fragment caching. The steps to apply the partial caching are as follows:
    To cache the contents of User control, either use @ OutputCache directive or make it possible programmatically by using the PartialCachingAttribute class. The following code will direct the caching engine to caching the control in the Output cache for 500 seconds:
    <%@ OutputCache Duration="500" VaryByParam="None" %>
    If you prefer to use code-behind technique, you can use the following lines of code in the metadata of your class declaration:
    <PartialCaching(500)>
    Now, when someone requests for the Web page where this User control is used, the caching engine will store the User control for 500 seconds in its Output cache and will serve it from the cache for the subsequent requests. The rest of the contents of the page will be retrieved from the server on every request.

14. Write a note on Data caching.
    The Data Caching is one of the most powerful features in the whole caching scenario in .NET. So far, we have seen declarative ways of enabling caching in our ASP.NET application, where Data caching is all about doing it programmatically. Data caching is very useful when you need to cache your own custom data into cache. For example, we need to cache a dataset which can be used later somewhere in your application. For this, we need Data caching.
    In ASP.NET, caching is implemented using two mechanisms-application caching and page out caching. Application caching allows you to cache data that you generate, such as a DataSet or a custom report. This page out caching saves the output of page processing and reuses the output. It supports the following basic types of dependency models:
    - File Based Dependency
    - Key Based Dependency
    - SQL Based Dependency
    - Aggregate based Dependency
    - Custom Dependency

15. What is Thread? Explain different thread states.
    A thread is defined as the execution path of a program. Each thread defines a unique flow of control. If your application involves complicated and time consuming operations like database access or some intense I/O operations, then it is often helpful to set different execution paths or threads, with each thread performing a particular job. Threads are lightweight processes.
    Thread States:

A thread can have one of the following states:

- Ready or Run-able state—A thread in this state has all resources it needs, excluding the processor. It awaits its turn in the ready queue (a queue of ready-to-use threads) to be allocated the processor.
- Running state—A thread in this state has all resources it needs, inclusive of the processor.Hence, a thread in this state is said to be in the running or executing state. Note that in a single processor, you can at any point in time have only one thread in the running state.
- Wait state—A thread in this state waits for the IO to be complete. Once the IO activity for the thread is complete, the thread is scheduled again to the ready queue.