



Unit I

1. How is c# better than java?

C# is a new language developed exclusively to suit the features of .NET platform. It can be used for a variety of application that are supported by .NET platform:

- Console application
- Windows application
- Developing windows controls
- Developing ASP.NET projects
- Creating web controls
- Providing web services
- Developing .NET component library

2. What are the differences between the c# and c++?

- * C# compiles straight from source code to executable code, with no object files.
- * In c++, class definition does not use a semicolon at the end
- * C# does not support #include statement
- * In c#, switch can also be used on string values.

3. What are the differences between the c# and java?

Although c# uses .net run time that is similar to java runtime the c# compiler produces an Executable code.

- C# has more primitive data types
- Unlike java, all c# data types are objects.
- Arrays are declared differently in c++

4. State some of the new features that are unique to c# language?

- It is a brand new language derived from the c/c++ family.
- It simplifies and modernizes c++
- It is the only component oriented language available today
- It is a concise, lean and modern language
- It has a lean and consistent syntax.

5. What is .net?

.net is the software framework that includes everything required for developing software for web services.

6. What is managed code?

The code that satisfies the CLR at runtime in order to execute is referred to as managed code. For example c# compiler generates IL, code(managed code).

7. What is the importance of the main method in a c# program?

C# includes a feature that enables us to define more than one class with the main method. Since main is the entry point for program execution there are now than one entry points. In facts there should be only one. This problem can be resolved by specifying which main is to be used to the compiler at the time of compilation as shown below; Csc filename.cs/main: classname

8. What is a local variable?

Variable declared inside methods are called local variables. Local variable can also be declared inside program blocks that are defined between an opening brace {and the closing brace}. The scope of a local variable starts after immediately after its identifier in the declaration and extends up to the end of the block containing the declaration.

9. What is initialization? Why is it necessary?

A variable must be given a value after it has been declared but before it is used in an expression. A simple method of giving value to a variable is through the assignment statement as follow,

```
Variable name=value;
```

10. Why do we require type conversion operation?

There is a need to convert a data of one type to another before it is used in arithmetic operations or to store a value of one type into a variable of another type .for example consider the code below :

```
byte b1 = 50;
byte b2 = 60;
byte b3 = b1 + b2;
```

11. What do you mean by implicit conversion?

It can always be performed without any loss of data. For numeric types, this implies that the destination type can be fully representing the range of the sources type. For example, a short can be converted implicitly to an int, because the shott range is a subset of the int range.

Therefore,

```
Short b= 75;
```

```
Int a=b;
```

```
//implicit conversion.
```

Are valid statement.

12. What do you mean by explicit conversion?

There are many conversions that cannot be implicitly made between types. If we attempt such conversions, the complier will give an error message.

13. What is value parameter?

Value parameters are used for passing parameters into methods by value. The parameters are declared with no modifier is passed by value and is called a value parameters.

14. What are parameters arrays?

In c#, we can define methods that can handle variable numbers of arguments using what are known as parameters arrays. Parameters arrays are declared using the keyword params.

Example:

```
Void function (params int[]x)
{
.....
.....
}
```

15. What is a variable size array?

C# also allows you to create a special type of two-dimensional array called a jagged array. A jagged array is an array of array in which the length of each array can differ. Thus a jagged array can be used to create a table in which the lengths of the rows are not the same.

16. What is an immutable string? How can we modify such a string?

String objects are immutable, meaning that we cannot modify the characters contained in them.

We are assigning the modified value to a new string variable. For instance,
`s1 = s1.Insert(3, "r");`

Is illegal .we are trying to modify the immutable string object s1.

17. What are the differences between classes and structs?

category

classes

structs

Data type

Inheritance

Destructors

assignment

Reference type and

therefore stored on the

heap

Support inheritance

Supported

Assignment copies the

reference

Value type and therefore

stored on the stack.

Behave like simple data

types.

Do not support

inheritance

Not supported

Assignment copies the

value

18. What is the purpose of a constructor in a structure?

The constructor is a method which is used to set values of data members at the time of declaration. The constructor method has the same name as struct and declared as public. The constructor is invoked as follows:

`Number n1=new number(100);`

This statement creates a struct object n1 and assigns the value 100 to its only data members number.

19. Why do we use methods as members of a struct?

The methods may be designed to perform certain operation on the data stored in struct objects. note that a struct is not permitted to declare a destructor.

20. Distinguish between ref and out parameters

A reference parameter does not create a new storage location. A parameters declared with the ref modifier is a reference parameters. Example:

`Void modify(ref int x)`

The output parameter are used to pass result backs to the calling method. This is achieved by declaring parameters with an out keyword. An output parameter does not create a new location.

Unit-2

1. What is a constructor?

A constructor initializes an object when it is created. It has the same name as its class and is syntactically similar to a method. However, constructor have no explicit return type. The general form of constructor if shown here:

```
Access class-name()
{
//constructor code
}
```

2. What is a static constructor?

A static constructor is called before any object of the class is created. This is useful to do any housekeeping work that needs to be done once. it is usually used to assign initial values to static data members.

3. What are the restrictions of static methods?

That there is no access modifier on static constructors. It cannot take any. A class can have only one static constructor.

4. What is property?

Another type of class members is the property. A property combines a field with the methods that access it. if you want to create a field that is available to users of an objects, but you want to maintain control over the operations allowed on that field.

5. What is read-only property?

There are situations where we would like to decide the value of a constant member at run-time. We may also like to have different constant values for different objects of the class. To overcome these shortcomings, c# provides another modifier known as read only

6. What is an indexer?

Array indexing is performed using the []operator. An indexer allows an object to be indexed like an array. The main use of indexes is to support the creation of specified arrays that are subjects to one or more constraints.

7. What are the two form of inheritance?

The two form of inheritance classical form, containment form

8. What is hiding a method?

We can use the modifier new to tell the compiler that the derived class method “hides” the base class method.

9. What is polymorphism?

Polymorphism means 'one name, many form'. Polymorphism can be achieved in two ways. c# supports both of them

*Operation polymorphism

*Inclusion polymorphism

10. What is early binding?

The compiler is able to select and bind the appropriate method to the object for a particular call at compiler time itself. This process is called early binding, or static binding, it is also known as compiler time polymorphism.

11. What is late binding?

The decision on exactly which method to call is delayed until runtime and, therefore, it is also known as runtime polymorphism. Since the method is linking with a particular class much later after compilation, this process is termed late binding.

12. What is operation overloading?

This means, C# has the ability to provide the operator with a special meaning for data type. This mechanism of giving such special meaning to an operator is known as operator overloading.

13. List the operators that can be overloaded?

Category

Operations

Binary arithmetic

Unary arithmetic

Binary bitwise

Unary bitwise

+, *, /, %, -

+, -, ++, --

&, |, ^, <<, >>

!, ~, true, false

14. List the operators that cannot be overloaded?

Category

Operations

Conditional operators

Compound assignment

&&, ||

+=, -=, *=, /=, %=

15. What is a delegate?

The dictionary meaning of delegate is "a person acting for another person". In C#, it really means method acting for another method. Creating and using delegates involves four steps.

They include:

* Delegate declaration

* Delegate methods definition

* Delegate instantiation

* Delegate invocation

16. What delegate method?

Delegate methods are any functions (defined in a class) whose signature matches the declare signature matches the delegate signature exactly.

17. What is multicast delegate?

It is possible for certain delegate to hold and invoke multiple methods. Such delegate are called multicast delegates. Multicast delegates, also known as combinable delegates, must satisfy the following conditions:

The return types of the delegate must be void.

None of the parameter of the delegate type can be declared as output parameter, using out keyword.

18. What is an event?

An event is a delegate type class member that is used by the objects or class to provide a notification to other objects that event has occurred. The client object can act on an event by adding an event handler to the event

19. What is an interface?

An interface can contain one or more methods, properties, indexers and events but none of them are implemented in the interface itself. It is the responsibility of the class that implements to define the code for implementation of these members.

20. What is the purpose of using the finally block?

C# supports another statements known as a finally statements that can be used to handle an exception that is not caught by any of the previous catch statements. A finally blocks can be used to handle any exception generated within a try block.

Unit-III**1. List out the different types of applications that can be created on .net?**

.Net offers closely related but distinguishable suites of tools for building windows or web applications. Both are based on the premises that many application have user interfaces centered on interacting with the user through form and controls, such as buttons, list boxes, text, and so forth

2. What are the advantage in using a dataset?

Dataset is a subset of the entire database cached on your machines without a continuous connection to the database-disconnected architecture which reduce burden on the database server which may help your application scale well.

3. Define relational database?

A database is a repository of data. A relational database organizes your data into tables. Consider the north wind database provided with Microsoft SQL server and Microsoft access.

4. What are the advantages of ADO.NET?

The significant advantage to disconnecting your data architecture from your database. The biggest advantage is that your application, whether running on the web or on a local machine, will create a reduced burden on the database server which may help your application to scale well. A disconnected architecture is resource frugal.

5. What is declarative reference integrity?

Relational database use declarative reference integrity to establish constraints on the relationship among the various tables.

This helps avoid two types of mistakes.

First, you can't enter a record with an invalid customer ID

Secondly, you can't delete a customer record if that customerID is used in any order. The integrity of your data and its relationship is thus protected.

6. Define Data Adapter?

The dataset is an abstraction of relational data base.ADO.NET uses a data Adapter as a bridge between the dataset and data source, which is the underlying database.dataAdapter provides the Fill () method to retrieve data from the database and populate the dataset.

7. Define Data Reader?

The data reader provides connected, forward-only, read-only access to a Collection of tables. By executing either a SQL statement or stored procedures.datareader is a lightweight object.

8. What are the advantages of web applications?

- They can be accessed from any browser that can connect to the server.
- Update can be made at the server

- You can achieve better performance by building a desktop application.

9. What is the step to create windows application?

First open visual studio and file ----->new ----> project.

In project window, create a new c# windows application and name it sharpwindowsform.

10. Differentiate between data reader and dataset?

The actually uses a data reader to populate itself. A data reader is lean, mean access

Methods that return results as soon as they are available, rather than for the whole of the query to be populated into a dataset.

11. How to implement the copy button event.

Now that you can check the files and pick the target directory, you are ready to handle the copy button-click event. The very first thing you need to do is to get a list of which files were selected .

12. What is ADO.NET?

ADO.NET looks very similar to ADO, its predecessor. The differences are that ADO.NET is native to .NET and that it is primarily a disconnected data architecture. Data is retrieved from a database and cached on your local machine.

13. Define normalization.

Normalization not only makes your use of the database more efficient, but also it reduces the likelihood of data corruption. If you kept the customer's name in both the customer table and the order table, you would run the risk that a change in one table might not be reflected in the other table. By keeping only the customerID in order, you are free to change the address in customers, and the change is automatically reflected for each order.

14. Define SQL.

The most popular language for querying and manipulating database is SQL, usually pronounced "Sequel". SQL is a declarative language, as opposed to a procedural language. The heart of SQL is query. Query is statement that return a set of record from the database.

15. Define ADO.NET object model.

The ADO.NET object model is rich, but at its heart it is a fairly straightforward set of classes. The most important of these is dataset. The dataset represent a subset of the entire database, cached on your machine without a continuous connection to the database.

16. Define DBCommand.

The DBCommand object allows you to send a command to the database. Often these objects are implicitly create when you create a dataadapter, but you can explicitly access these object.

17. Define DBConnection.

The DBConnection objects represent a connection to data source. This connection can be shared among different command object.

18. Define Data tables and data columns.

The data table has a number of public properties, including the columns connection, which returns the data columns connection object. Each data column object represents a column in a table.

19. Define rows.

Data table's rows collection returns a set of rows for that table. Use this collection to examine the results of queries against the database, iterating through the rows to examine each record in turn.

20. What are the advantages of ADO.NET model?

Highly efficient, but to be effective the dataset must be a robust subset of the database, capturing not just a few rows from a single table, but also a set of tables with all metadata necessary to represent the relationship and constraints of the original database.

Unit IV

1. What are the uses of server side controls?

.NET wants you to use server side controls that can often generate unnecessary trips back to the server by default. Plus, .NET also wants to make the determination as to how it will act with various browsers. While not impossible, it is much more cumbersome to create custom workarounds for special situations. Not to mention the fact that new browsers are released in between .NET service pack updates.

2. List out the server side state management options supported by ASP.NET.

Application state

Session state

Profile properties

Database support

3. Differentiate the postback events with nonpostback events?

Post back events are those that cause the form to be posted back to the server immediately. In contrast, many events are considered nonpostback in that the form isn't posted back to the server immediately. Instead, these events are cached by the control until the next time a post back event occurs.

4. What is WSDL?

A lot of work is being done for you automatically. HTML pages describing your web services and its methods are generated and this page includes links to pages in which the methods can be tested.

5. List down the events in life cycle of a web page?

Initialize, load view state, process postback data, load, handle postback events, prerender, save state, render, dispose

6. Define the prerender event in life cycle of a web page?

This is your last chance to modify the output prior to rendering using the OnPreRender () method.

7. Define Client-side support?

You make use of a web service by writing client code that acts as though it were communicating directly with a local object, but in reality communicates with the host server through a proxy.

8. Define server-side support?

While creating a web service requires no special programming on your part, you need only write the implementing code, add the [web method] attribute and let the server do the rest.

9. What is SOAP?

SOAP is a lightweight, message-based protocol built on XML, HTTP and SMTP. Two other protocols are desirable, but not required. For a client to use a SOAP-enabled web service.

10. Define web form life cycle?

The life cycle begins with request for the page, which cause the server to load it. When the request is complete, the page is unloading.

11. What are web services?

.Net web services help you write components whose methods can be invoked across the internet using any .Net programming language.

12. How to create the proxy?

To create the proxy, enter wsdl at the visual studio command-line prompt, followed by the path to the wsdl contract. For example

Wsdll<http://localhost:19975/calculatorws/service.asmx?wsdl>

13. What are the two types of server side controls?

* The first is server-side HTML controls. These are HTML controls that you tag with the attribute runat=server.

* the alternative to marking HTML controls as server-side controls is to use ASP.NET server controls, also called ASP controls or web controls.

14. How to add the server-side controls to web form?

* by writing HTML into the HTML page

* by dragging controls from the toolbox to the design page,

*by programmatically adding them at runtime.

15. Define Load View state.

The view state property of the control is populated. The view state information comes from hidden variable on the control, used to persist the state across round trip to server.

16. Define Render.

This is where the output to be sent back to the client browser is generated. You can overwrite it using render method.

17. Define Dispose.

This is the last phase of the life cycle. It gives you an opportunity to do any final cleanup and release reference to any expensive recourse, such as data base connection.

18. Define process post back data?

During the phase, the data sent to the server in the posting is processed. if any of this data results in a requirement to update the viewstate, that update is performed via the loadpostdata()method.

19. Define save state.

Near the beginning of the life cycle, the persisted view state was loaded from the hidden variable, now it is saved back it is saved back to the hidden variable, persisting as a string object that will complete the round trip to the client.

20. Define state?

A web application's state is the current value of all the controls and variables for the current user in the current session.

Unit V

1. What are assemblies?

An assembly is a collection of files that appear to be a single DLL or executable (exe). Assemblies are .NET units of reuse, versioning, security, and deployment.

2. What is the difference between single call and singleton?

With a well-known singleton object, all messages for the object, from all clients, are dispatched to a single object running on the server.

With a well-known single-call object, each new message from a client is handled by a new object.

3. Define metadata?

Assemblies are the .NET unit of reuse, versioning, security, and deployment.

In addition to the objects code for the application, assemblies contain resources.

4. What is PE files.

On disk, assemblies are portable executable files, PE files are not new. The format of a .NET PE file is exactly the same as a normal Windows PE file. PE files are implemented as DLLs or EXEs. They consist of one or more modules.

5. Define multimodule assemblies

A multimodule assembly consists of multiple files. The assembly manifest in this case can reside in a standalone file, or it can be embedded in one of the modules. When the assembly is referred to at runtime, it loads the file containing the manifest and then loads the required modules as needed.

6. Define shared assemblies

If you want to share your assembly, it must meet certain stringent requirements.

Your assembly must have a strong name. Strong names are globally unique.

To share your assembly, place it in the global assembly cache. This is an area of the file system set aside by the CLR to hold shared assemblies.

7. Define private assemblies

Assemblies come in two flavors:

Private and shared. Private assemblies are intended to be used by only one application.

Shared assemblies are intended to be shared among many applications.

8. Define attributes

Attributes are a mechanism for adding metadata, such as compiler instructions and other data about your data, methods, and classes to the program itself. Attributes are inserted into the metadata and are visible through ILDasm and other metadata-reading tools.

9. Define reflection

Reflection is the process by which a program can read its own metadata or metadata from another program. A program is said to reflect on itself or on another program, extracting metadata from the reflected assembly and using that metadata either to inform the user or to modify the program's behavior.

10. Define :Marshaling

The process of moving an object to be remoted is called marshaling.

11. what are the uses of server side controls?

.NET want you to use server side controls that can often generate unnecessary trips back to the server by default.plus .NET also wants to make the determination as to how the control will act with various browsers.while not impossible,it makes it much more cumbersome to create custom workarounds for special situation.

12. List out the server side state management option supported by asp.net?

Application state

Session state

Profile properties

Database support

13. Define DLL hell?

DLL hell is where loading one program breaks another program.

Windows file protection (WTP) protects system DLLs from being updated or

Deleted by unauthorized agents.

14. Define Joining threads?

When you tell a thread to stop processing and wait until a second thread

Complete its work, you are said to be joining the first thread to the second.

15. Define Starting threads?

The simplest way to create a thread is to create a new instance of the

Thread class. The thread constructor takes a single argument.

16. Define Remoting?

When an object is marshaled, either by value or by proxy, across a process

Or machine boundary, it is said to be remoted.

17. Define application (app) domain?

App domain can be independently started and halted. They are secure, lightweight and versatile. An app domain can provide fault tolerance. If you start an object in a second app domain and it crashes.

18. What are the methods and properties of the app domain class.

CurrentDomain

createDomain()

GetCurrentThreadID()

Unload ()

FriendlyName

GetData ()

Load ()

setData()

14

19. Define Member Filter?

A delegate that filter the list of members in the member info array of object. You use a type.filtername filter, which is a field of the type class that filter on the name.

20. List out the four task of reflections?

Viewing metadata

Performing type discovery

Late binding to method and properties

Creating types at runtime (reflection emit)