OOPS

<https://www.interviewbit.com/c-sharp-interview-questions/>

https://www.c-sharpcorner.com/article/oops-interview-questions-c-sharp/

**Class**-buliding block of an application .Consists of fields and methods

**Objects**-blue print.Instance of class

**Class Members:**

1.Instance-accessible from the object

Ex: var person=new person();

Person.Introuduce();//Invoking from object

2.static-accessible from class

Ex: Console.WriteLine(); //console is the class and writeline is the static method

**Static Class:**

The advantage of using a static class is **that the compiler can check to make sure that no instance members are accidentally added**. The compiler will guarantee that instances of this class cannot be created. Static classes are sealed and therefore cannot be inherited. They cannot inherit from any class except Object.

**Why use static members?**

If you use the "static" keyword for the declaration of a field / property or a method, then it is called a "Static member". The static members are accessible directly, without even creating one object also.

1.to represent concepts that are singleton

2.dateTime.now//we should have one space of memory that does nt have another

3.Console.writeline();

Declaring static mem:

Public class person

{

Public static int peoplecount=0;

}

**Sealed Class:**

A sealed class, in C#, is a class that cannot be inherited by any class but can be instantiated.

The design intent of a sealed class is to indicate that the class is specialized and there is no need to extend it to provide any additional functionality through inheritance to override its behavior.

A sealed class is mostly used for security reasons by preventing unintended derivation by which the derived class may corrupt the implementation provided in the sealed class.

Partial class:

**Constructors:(** **https://www.c-sharpcorner.com/UploadFile/0c1bb2/constructors-and-its-types-in-C-Sharp/)**

What?

A method that is called when an instance of class is created

Why we need?

To initialize some of the fields in class

A constructor has the same name as the class

1.Public class customer

{

Public customer()//default connstructor

{

}

}

2.public class customer

{

Public string Name;

Public Customer(string name)//parameterized constr

{

this.Name=name;

}

}

this-this is the keyword references the current object

here in the above ex why we used this keyword?

->bcz the filed declared as uppercase Name and parameter passed is lower case after of we assign together without this keyword that looks like meaningless . That’s why using this keyword whatever in the right side that will copied to left side by using this.

**Constructor overloading-**

**https://www.c-sharpcorner.com/UploadFile/8a67c0/method-overloading-and-method-overriding-in-C-Sharp/**

Same method name but different signature like paramerts may be different and return type also different.

Why we need const overloading actually?

Ex: public class customer

{

Public customer(){……}

Public customer(string name){………}

Public customer(int id, string name){………}

}

Sometimes we may know only name then we use second one.or sometime we need id and name then we 3rd constrictor and sometime we don’t need neither.

**Methods:**

Overloading methods:

Same method name but different signature

**Access Modifiers:**

1.public 2.private 3.protected 4.Internal 5.protected internal

Why we need access modifiers?

A way to control the access to a class and/or its members

Why we need to control the access?

Bcz to create safety in our programs

1.private-accessible only from the class

Public class customer

{

Private string Name;

}

Var john=new customer();

John.name//won’t complete

2.public-accessible from eveywhere

Public class customer

{

Public string Name;

}

Var john=new customer();

John.name

3.protected-accessible only from the class and its derived class

4.Internal-Accessible only from the same assembly.

5.Protected Internal-Accessible only from the same assembly or any derived classes

**Encaspulation**: suppose in bank many secured details were ex:name,acnoumber,city,balance

We can take private as acnumber, balance // and make it public using getter/setter methods

Public as name ,branch,city

n c#, Encapsulation is **a process of binding the data members and member functions into a single unit**. ...

* Abstraction is the **method of hiding the unwanted information**. Whereas encapsulation is a method to hide the data in a single entity or unit along with a method to protect information from outside.
* Define fields as private
* Provide getter/setter method as public
* Set/writing something(proving value)
* Get/reading something(retrieving value)

Wrapping of data and showing essential features.

<https://www.c-sharpcorner.com/article/encapsulation-in-C-Sharp/>

difference between encapsulation and abstraction?

**Encapsulation** is basically to bind data members & member functions into a single unit called **Class**. Whereas **Abstraction** is basically to hide complexity of implementation & provide ease of access to the users.

**Properties: what?**

A class member that encapsulates a getter/setter for accessing a field.

Why we need?

To simply create a getter/setter with less code.

**Abstarction:**

Abstraction is an important part of object oriented programming. It means that only the required information is visible to the user and the rest of the information is hidden.

https://www.tutlane.com/tutorial/csharp/csharp-abstraction

Abstract and override and virtual

**Method hiding(new keyword):**

For hiding the base class method from derived class simply declare the derived class method with a new keyword.

By using the “new” keyword base class method is hidden ,giving preference to direved class.

Method Overriding:

**used to achieve runtime polymorphism**

**To perform overriding we need to use virtual or override keyword**

https://www.c-sharpcorner.com/UploadFile/2072a9/method-overriding-in-C-Sharp/

Modifying the implementation of an inherited method.

Definition: Method Overriding is **a technique that allows the invoking of functions from another class (base class) in the derived class**.

In C#, for overriding the base class method in a derived class, you have to declare a base class method as virtual and derived class method asoverride shown below:

Virtual Ketword:

The Virtual keyword is used for generating a virtual path for its derived classes on implementing method overriding. The Virtual keyword is used within a set with an override keyword. It is used as:

1. // Base Class
2. **class** A
3. {
4. **public** **virtual** **void** show()
5. {
6. Console.WriteLine("Hello: Base Class!");
7. Console.ReadLine();
8. }
9. }

**Override Keyword**  
The Override keyword is used in the derived class of the base class in order to override the base class method. The Override keyword is used with the virtual keyword, as in:

1. // Base Class
2. **class** A
3. {
4. **public** **virtual** **void** show()
5. {
6. Console.WriteLine("Hello: Base Class!");
7. Console.ReadLine();
8. }
9. }
11. // Derived Class
12. **class** B : A
13. {
14. **public** **override** **void** show()
15. {
16. Console.WriteLine("Hello: Derived Class!");
17. Console.ReadLine();
18. }
19. }

Abstract:it does not have a body it is just a declaration

* Does not include implementation
* If a member is declared as abstract the containing class needs to be declared as abstract too.
* In derived class must implement all abstract method in the base abstract class
* Suppose if we declared two abstract method then we need to override it two times
* Cannot be instaiated

Where do we use abstract class in real time?

When we have the requirement of a **class that contains some common properties or methods** with some common properties whose implementation is different for different classes, in that situation, it's better to use Abstract Class then Interface

Public abstract class shape

{

Public abstract void draw();

}

Public class circle:shape

{

Public override void draw()

{

//implememtaion for circle

}

}

**Interface: how to call write check some of the examples**

**https://www.geeksforgeeks.org/c-sharp-interface/**

What? It is used to avhieve the multiple inheritance

A language construct that is similar to a class(in terms of syntax) but its fundamentally different.

Ex:

Public interface ITaxCalculator

{

Int calculate();

}

Here instead of class declared as interface , all interfaces starts with I

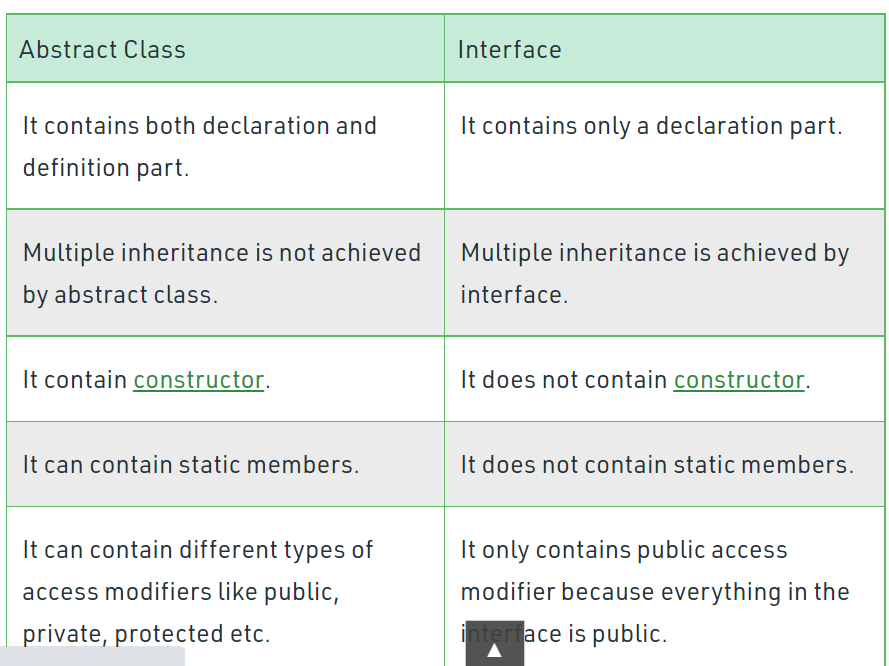
Interfaces does not have implementation and does not have access modifiers

Then why we need?

To build loosely couple applications

Ex: suppose in a hotel if any chef feels sick so that they will replace –nothing but loosely couples

But if they real that chef to do the baking and everething suppose if he absent for any day that is highly dependedt-represent titly coupled



**Inheritance:**

A kind of relationship between two classes that allows one to inherit code from other

Benefits is:

Code is re-use and code does not repeat

**Boxing and Unboxing**

Types in C#:

1.value types 2.refrence types

1.value type

* All stored on the stack
* Ex:all primitive data types:byte,int ,float…..
* Struct type

2.Reference type

* All stroed in the heap
* Ex:any classes (object,array,string…)

Boxing->the procees of converting a value type instance to an object reference.

Ex:

Int num=10;

Object obj=num;//here object is the base of all classes in .net framework and assign a vlue to it.

What happened is?

This box stored in CLR and stored in heap instead of stack

Stack heap

Num=10 num=10//when boxing happens CLR creates object in heap&then create reference in stack

Obj reference

Unboxing-oppsite to boxing

Object obj=10;//at this time boxing happens

Int number=(int)obj;

Why we need boxing and unboxing ?

With Boxing and unboxing one can link between value-types and reference-types by allowing any value of a value-type to be converted to and from type object

Or

The opposite operation is Unboxing which is the process of converting back the reference type into the value type. This process verifies that the receiving data type is equivalent to the boxed type as;

1. **int** c = (**int**)b; // unboxing

**Generics:**

Object is the parent type for evry class in dot net we can easily pass any types to this method but there is a problem here that is performance

If we use this class for storing value types evry time we insert value type in the list it has to be boxed and every time when we need to access that value that has to be unboxed results in performance penalty . If we take reference type if we book is an object and vice versa this casting cost performance pentaly.

Here came Generics

We can create a class one time and resuse it. no penalty compared to reference type or value types.

Ex: class DataStore<T>

{

public T Data { get; set; }

}

he DataStore is a generic class. T is called type parameter, which can be used as a type of fields, properties, method parameters, return types, and delegates in the DataStore class

**Delegates**

**I**t is an object that knows how to call a method (or group of methods)

A reference or pinter to a function

Why we need a delegate?

We can use function or method only?

->for designing extensible and flexible applications(eg:framework)

Ex:Imagine ur responsible for designing framework that is for processing photos

Suppose if we used three filters like Brightness,Contrast size

Public void process(string path)

{

Var photo=new photo();

Var filter=new filter();

Filter.ApplyBrightness();

Filter.ApplyContrast();

Filter.Resize();

}

Suppose if we release this framework and another developer wants to use this framework and want to apply new filter that we are not defined. Then the problem we need to create a filter then compile and reploying code.

Imagine every time for adding a new filter need to complie and redeploycode and this is waste of time.

So by delegates we can make this framework extensible such that a developer can create a own filter without relying and the same problem is solved by interface also using some polymorphism.

Using delegate

Public delegate void photofilter(photo photo)

{

Public void process(string path, photofilter filter)//passing delegate here

{

Var photo=new photo();

Filter (photo);//this code does not know what filter has been applied and it is responsibility of client whoever using they can define what they want.Some developer need only resize or some need clor so here no need of again and again compile and redeploying the code here framework is extensible.

}

//

**Lambda Expressions**

An anonymous method

* No access modifier
* No name
* No return statement

Lambda expressions are anonymous functions that contain expressions or sequence of operators. All lambda expressions use the lambda operator =>, that can be read as “goes to” or “becomes”. The left side of the lambda operator specifies the input parameters and the right side holds an expression or a code block that works with the entry parameters. Usually lambda expressions are used as predicates or instead of delegates (a type that references a method).

Expression Lambdas

*Parameter => expression  
Parameter-list => expression  
Count => count + 2;  
Sum => sum + 2;  
n => n % 2 == 0*

**Events:**

* A mechanism for communicating between objects
* Helps in building loosely coupled application->that is components or classes are not titly coupled to gether. It is easy to extend without breaking.
* Helps extending applications.

Public class vedioencoder

{

Public void encode(Video video)

{

//encoding logic…

…………..

\_mailservice.Send(new Mail());//when the encoding is finished we using Mail service to send an email to the person owns vedio

}

}

But in the future we want text message to send also then what we will do

Public class vedioencoder

{

Public void encode(Video video)

{

//encoding logic…

…………..

\_mailservice.Send(new Mail());

\_mailservice.Send(new Text());//by adding this extra line the it has to be recompile and redeploy once again

}

}

Then we need to minimal imapcat compared to this by using events

Video encoded

VedioEnoder

MailService

Video Encoder

Publisher subscriber

Event sender event reciever

Message Service

We have video encoder class has publisher and sender , mail service class has subscriber and receiver

In this video encoder does not know anything abt mail servive. In the future if we want to extend our application and capability to send the text message then we can simply create new class ‘message service’.

Basically video encoder does not need to recompile and redeployed again

**LINQ-Langaugage Integrated query**

Gives the capability to query objects in memory like collections(linq to objects)

Databases(linq to entities)

XML(linq to xml)

Ex:

IList<string> stringList = new List<string>() {

"C# Tutorials",

"VB.NET Tutorials",

"Learn C++",

"MVC Tutorials" ,

"Java"

};

// LINQ Query Syntax

var result = from s in stringList

where s.Contains("Tutorials")

select s;

Assembly

<https://www.c-sharpcorner.com/UploadFile/78607b/what-is-assembly/#:~:text=by%20the%20CLR.-,An%20Assembly%20is%20a%20basic%20building%20block%20of%20.,a%20logical%20unit%20of%20functionality>.

Extension method:

If I wanted to add any functionality to int ,string then we want extension method without touching the class solution in extension.

Singleton Pattern:

Singleton design pattern in C# is one of the most popular design patterns. In this pattern, a class has only one instance in the program that provides a global point of access to it.

Exception Handling: <https://www.javatpoint.com/exception-handling-in-java>

https://www.c-sharpcorner.com/article/exception-handling-in-C-Sharp/

The **Exception Handling** is one of the powerful mechanism to handle the runtime errors

 ClassNotFoundException, IOException, SQLException, RemoteException, etc.

Suppose there are 10 statements in a Java program and an exception occurs at statement 5; the rest of the code will not be executed, i.e., statements 6 to 10 will not be executed. However, when we perform exception handling, the rest of the statements will be executed. That is why we use exception handling in [Java](https://www.javatpoint.com/java-tutorial).

Why we need?

Exception handling **ensures that the flow of the program doesn't break when an exception occurs**.

Try: The "try" keyword is used to specify a block where we should place an exception code. It means we can't use try block alone. The try block must be followed by either catch or finally

Catch: The "catch" block is used to handle the exception. It must be preceded by try block which means we can't use catch block alone. It can be followed by finally block later.

Finally: The "finally" block is used to execute the necessary code of the program. It is executed whether an exception is handled or not.

For example, if you open a file, it must be closed whether an exception is raised or not.

AZ-900(https://www.edureka.co/blog/interview-questions/azure-interview-questions/)

* IaaS: cloud-based services, pay-as-you-go for services such as storage, networking, and virtualization.

 Azure VM, Amazon EC2

* PaaS: hardware and software tools available over the internet.

For example: Web Apps, Mobile Apps in Azure

* SaaS: software that’s available via a third-party over the internet.

Google Apps, Salesforce,

SQL

SELECT name, salary

FROM #Employee e1

 WHERE N-1 = (SELECT COUNT(DISTINCT salary) FROM #Employee e2

 WHERE e2.salary > e1.salary)SELECT name, salary  
  
Read more: <https://javarevisited.blogspot.com/2016/01/4-ways-to-find-nth-highest-salary-in.html#ixzz7FHKXi82y>

Or

SELECT TOP 1 max(salary)

FROM

 ( SELECT DISTINCT TOP N max( salary)

FROM #Employee

ORDER BY salary DESC

) AS temp

ORDER BY salary

select \* from employee

where salary=(select Max(salary)

from employee);

N=1,2,3 if u need find top 3 salary then N=3

By default ORDER BY clause print rows in ascending order, since we need the highest salary at the top, we have used ORDER BY DESC, which will display salaries in descending order. Again DISTINCT is used to remove duplicates. The outer query will then pick the topmost salary, which would be your Nth highest salary.  
  
Read more: <https://javarevisited.blogspot.com/2016/01/4-ways-to-find-nth-highest-salary-in.html#ixzz7FHLV7voO>

String and stringbuilder

StringBuilder is used to represent a **mutable string** of characters. Mutable means the string which can be changed. So String objects are immutable but StringBuilder is the mutable string type. It will not create a new modified instance of the current string object but do the modifications in the existing string object.

<https://www.c-sharpcorner.com/blogs/difference-between-string-and-stringbuilder-in-c-sharp1>

difference between readonly and constant

<https://www.geeksforgeeks.org/difference-between-readonly-and-const-keyword-in-c-sharp/>

in mvc

global.aspx--https://stackoverflow.com/questions/2340572/what-is-the-purpose-of-global-asax-in-asp-net

routing---https://www.tutorialsteacher.com/mvc/routing-in-mvc

ado.net objects

web api

get

put

waterfall

agile

scrum

SQL

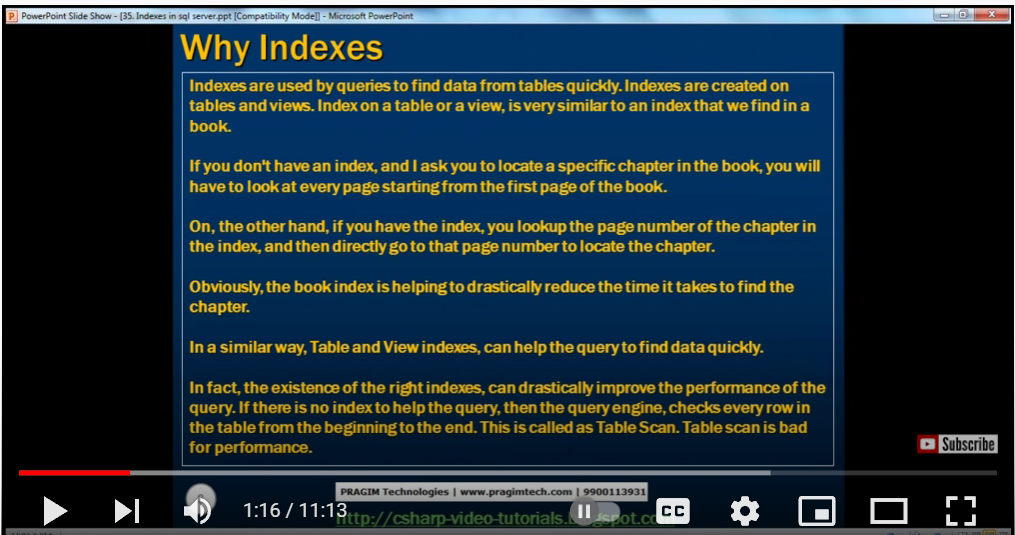
Index: <https://www.sqlshack.com/what-is-the-difference-between-clustered-and-non-clustered-indexes-in-sql-server/>

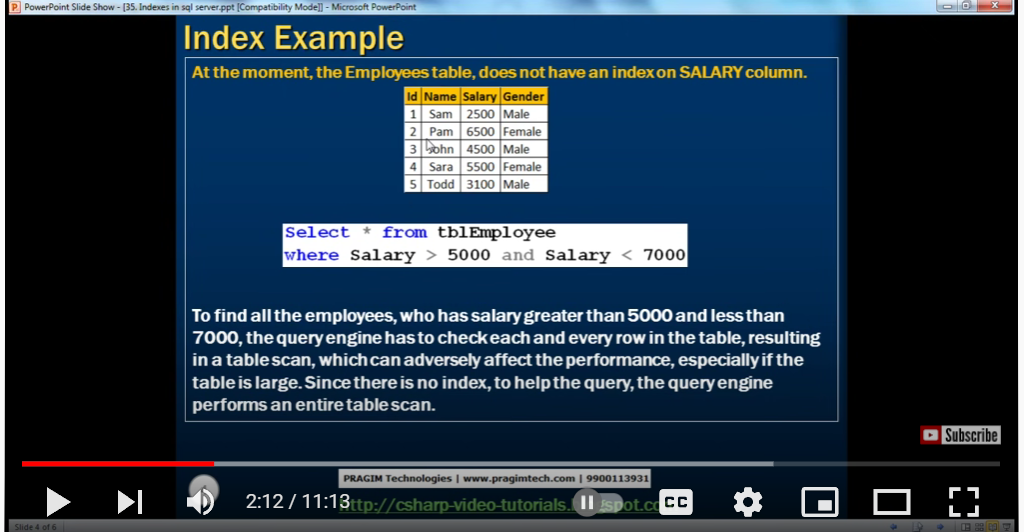
CREATE INDEX index\_name  
ON table\_name (column1, column2, ...);

Non clusteres

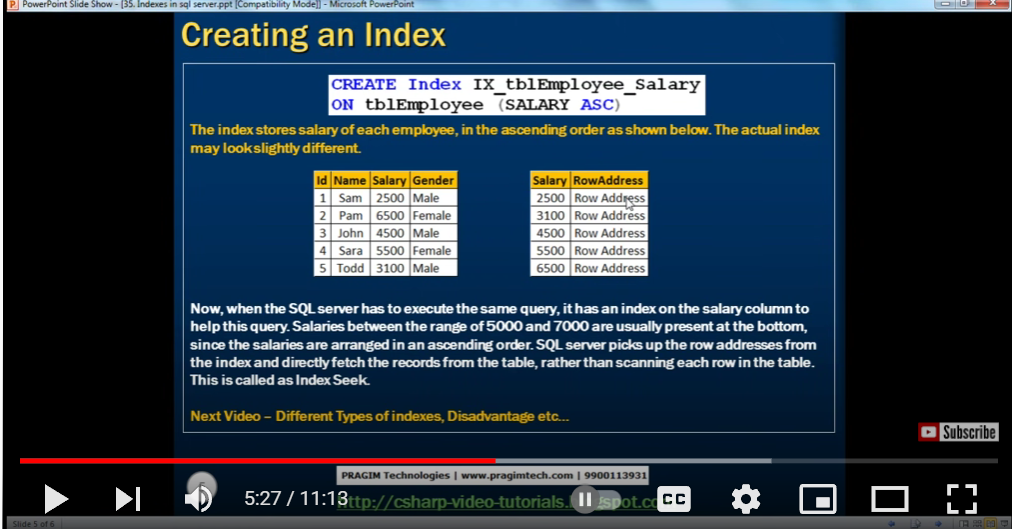
CREATE NONCLUSTERED INDEX IX\_tblStudent\_Name

ON student(name ASC)



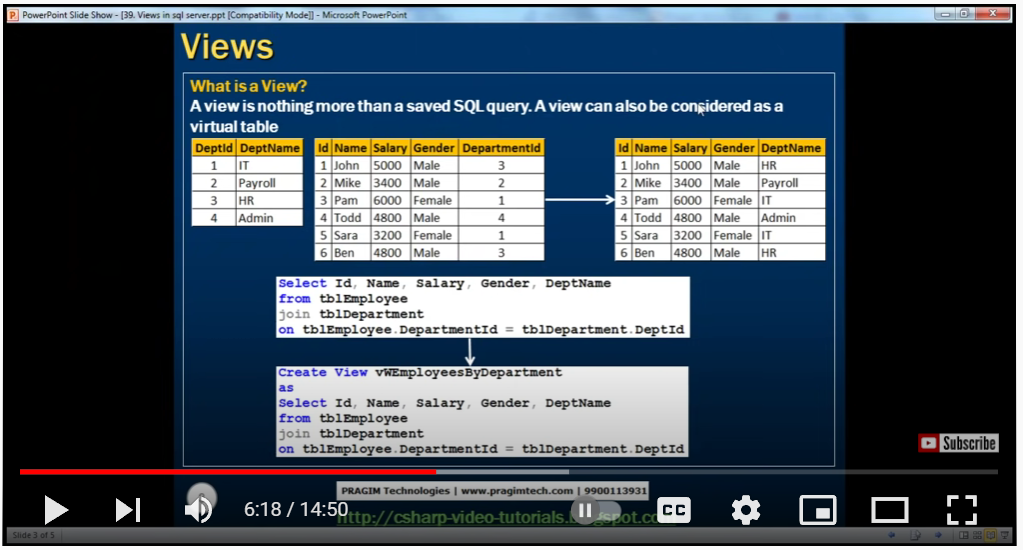


Suppose if u want to fetch the salary from 5k to 7k so we should scan the whole table suppose if data is large it results in time consumimg, so here if index there we can so that easily by taking that we can fetch easily.

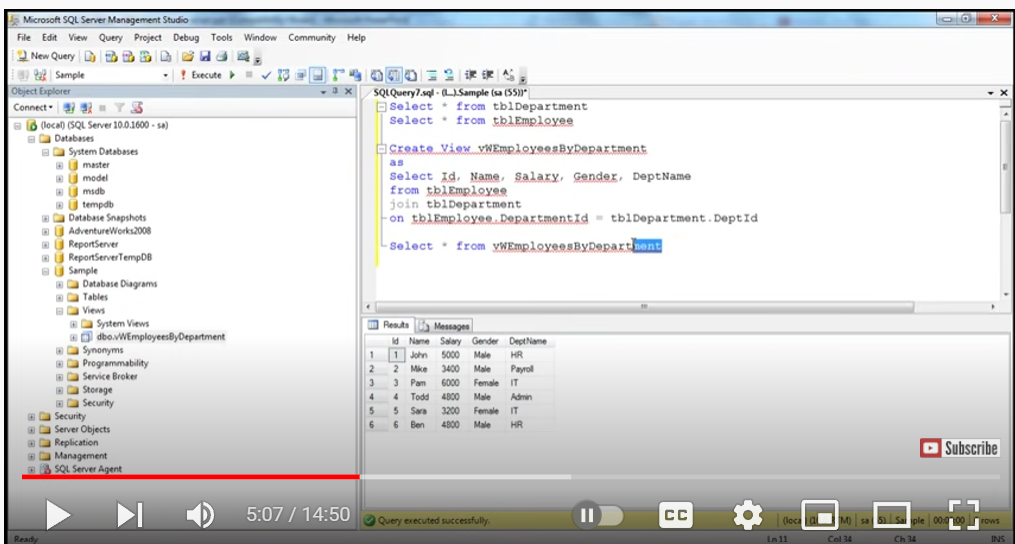


**View: it is used to view the selected data whenever required we stored in the view folder.**

**View is nothing but a saved select query**

****

**We have two tables and we join two tables using ‘joins’ and then we can use ‘view’**

****

**We can check view in particulau database we have view folder we can check there**

**Select \* from vtablenamecolumnname**

**It gives the data of two tables.**

**Advnatges:**

1.You can add/remove fields easily in a view without modifying your underlying schema

2.Views can model complex joins easily.

**Temp**

Trigger:

Like what is trigger means suppose I am the management officer in a company if any new employ appointed I need to send an wecome email but if many employes there I cannot do that manually it should send automatically when any new employ enters there trigger comes.

Syntax:

Create trigger triggername

Before|after(like trigger should act beforeor after on going event)

Insert|update|delete

On tablename

Foreachrow | foreachcol

Tigger\_body

Ex: we want to calculate the marks of the student as soon as we get details into the database.

Here we are adding 100 marks for each column

Create trigger sampletigger

Before Insert

On student

Foreachrow

Set new.marks=new.marks+6;

Before insert : are used to update or validate record values before they saved to database

After insert :they used to access the fileds

https://www.c-sharpcorner.com/UploadFile/63f5c2/triggers-in-sql-server/

Noramlization https://www.guru99.com/database-normalization.html

DDL,DML https://www.geeksforgeeks.org/sql-ddl-dql-dml-dcl-tcl-commands/

Clauses <https://www.javatpoint.com/dbms-sql-clauses>

Group by,where,having

Joins <https://www.w3schools.com/sql/sql_join.asp>

Outer join: In an outer join, unmatched rows in one or both tables can be returned.

Set https://www.w3schools.com/sql/sql\_ref\_set.asp

Union <https://www.w3schools.com/sql/sql_ref_union.asp>

Like: used in a where clause to search for a specified pattern like ‘%a%’

It gives all the details related to a

Aliases: are used using ‘as’ keyword

SELECT CustomerID AS ID, CustomerName AS Customer  
FROM Customers;

Stored procedure depth <https://www.geeksforgeeks.org/what-is-stored-procedures-in-sql/>

What Is stored proc?

Instead of writing the query again and again we can store once and we can call the stored proc whenever needed

Suppose you want name,gender from table emp

Select name,gender from emp

But everytime writing this we can just call the stored proc name

Create procedure spgetemp

As

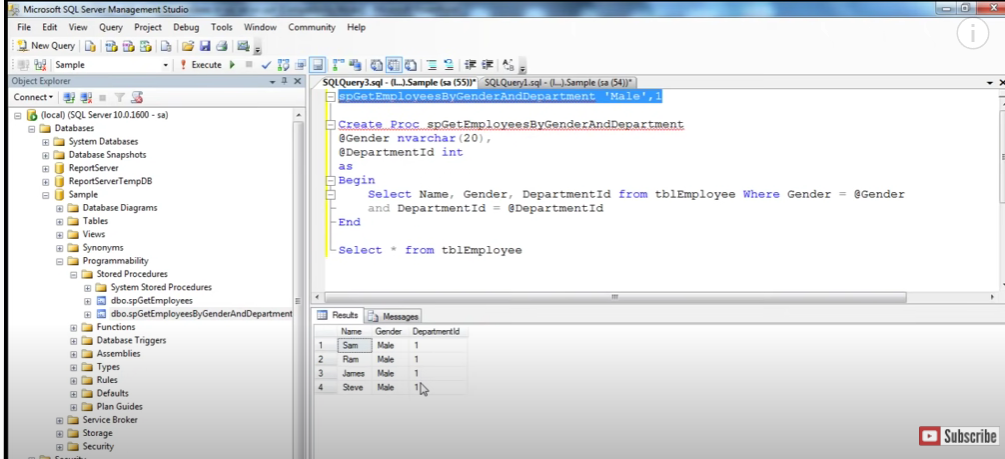
Begin

Select name,gender from emp

End

If u just call **spgetemp or exec spgetemp**

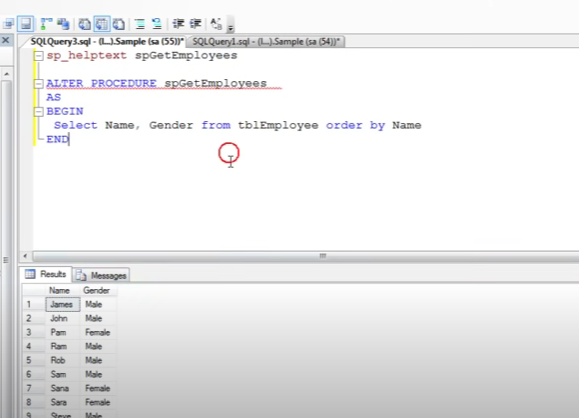
Then we can able to see the name and gender which is already stored in stored proc



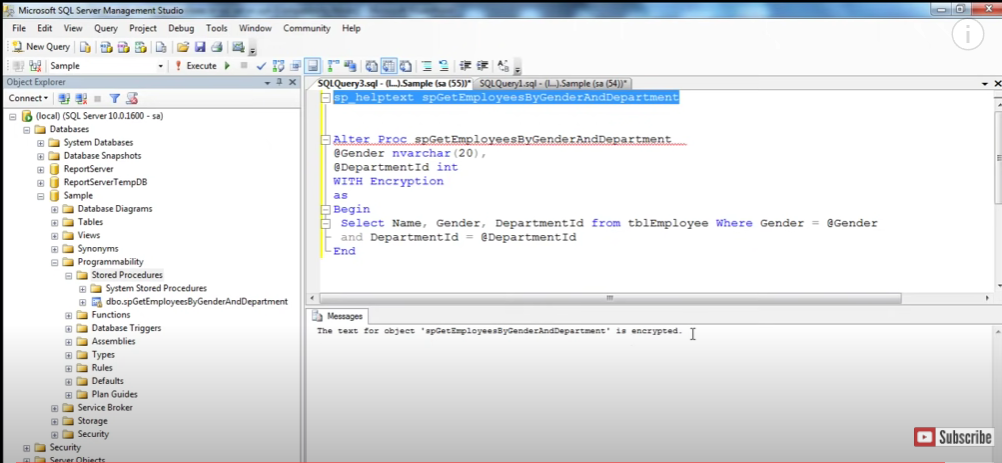
Here we have table tblempoyee from that we need gender as male and dep as 1 so using stored done here.

If you want to see the stored proc

Sp\_helptext stored procedure



Alter



Here we alterad and encrypted

If someone tries to open this stored proc it is encrypted it will show.

**Primary key will not accept NULL values whereas Unique key can accept NULL values**. A table can have only one primary key whereas there can be multiple unique key on a table

Difference between drop,delete and truncate

Delete-if u want to delete all the rows of the table as per some condition

DELETE from WHERE

Here we can use the “ROLLBACK” command to restore the tuple

Drop: If we want to drop the table:

Drop table

Here we cannot use the “ROLLBACK” command to restore the tuple

Truncate: By using this command the existence of all the rows of the table is lost. It is comparatively faster than the delete command as it deletes all the rows fastly.

Truncate

Here we cannot use the “ROLLBACK” command to restore the tuple

Linq querires in c#

Action result: https://www.tutorialsteacher.com/mvc/action-method-in-mvc

An ActionResult is a return type of a controller method, also called an action method, and serves as the base class for \*Result classes. Action methods return models to views, file streams, redirect to other controllers.

Functions: http://www-db.deis.unibo.it/courses/TW/DOCS/w3schools/sql/sql\_functions.asp.html

|  |  |
| --- | --- |
| **Functions** | **Procedures** |
| You cannot manage transactions inside a function. | You can manage transactions inside a procedure. |
| You cannot call stored procedures from a function | You can call a function from a stored procedure. |

Sql jobs: https://www.c-sharpcorner.com/UploadFile/raj1979/create-and-schedule-a-job-in-sql-server-2008/

what is mvc helper?

An HTML Helper is just a method that returns a HTML string.

For example, you can use HTML Helpers to render standard HTML tags like HTML <input>, <button> and <img> tags etc.

Data Annotations

Required.

Range,

RegularExpression ,

Compare.

StringLength.

Data type.

DB context:

You can think of DbContext as the database connection and a set of tables, and DbSet as a representation of the tables themselves. The DbContext allows you to link your model properties (presumably using the Entity Framework) to your database with a connection string.

1.The primary class that is responsible for interacting with data as objects DbContext.

* If we are using the EF Designer, Entity Framework will generate the context.
* But if you are using the Code First approach, you will typically write the context yourself.
* The DbContext APIs is not released as part of the .NET Framework, Entity Framework team distributes EntityFramework.dll through NuGet to be more flexible and frequent with releasing new features to Code First and the DbContext APIs.

Scaff folding:

**code generation framework for ASP.NET Web applications**.

Scaffold templates are used to generate code for basic CRUD operations within your ASP.NET MVC applications against your database with the help Entity Framework

<https://www.tutorialspoint.com/asp.net_mvc/asp.net_mvc_scaffolding.htm>

why we need?

You add scaffolding to your project when you want to quickly add code that interacts with data models. **Using scaffolding can reduce the amount of time to develop standard data operations in your project**.

Mvc caching?

Caching is used to improve the performance in ASP.NET MVC. Caching is a technique which stores something in memory that is being used frequently to provide better performance. In ASP.NET MVC, OutputCache attribute is used for applying Caching. OutputCheching will store the output of a Controller in memory and if any other request comes for the same, it will return it from cache result.

Suppose you want to see the data output so kept that as thread sleep to 3 sec .

First time when u load the page it take 3 sec but if you go on click on evry time it will take 3 sec.

So in this case we can take [outputcache] which stores output when u open first time it will take 3 sec but after clicking on it it will not load again.

When we make a first time request atleast it will take 3 sec to retrieve the data and then caches the data then it gives to view and view send back to the client . so next time when we make a request to a same action method since data is already cached it return directly from the cache it will not take 3 sec.

[outputcache(Duration=10)]-so this controller action method processed atmost for 10 sec.

Dependency Injection in c#:

https://www.c-sharpcorner.com/UploadFile/85ed7a/dependency-injection-in-C-Sharp/

What is the purpose of console application?

A console application **facilitates the reading and writing of characters from a console** - either individually or as an entire line. It is the simplest form of a C# program and is typically invoked from the Windows command prompt.

What is difference between console application and window application?

A Windows form application is an application that has a graphical user interface(GUI) like the Visual C# IDE. A console program on the other hand is **a text application**. There are not fancy controls like buttons or textboxes in a console application and they are run from the command prompt.

Difference between Windows Application and Web Application

Windows application **can only be accessed** from a system in which it is installed. A web application can be accessed from any system through the internet. You need an Internet Information Services (IIS) server to run the web application.