All the queries are compiled everytime when we run them

They are sent to Sql Server separately

If you want to send them together to server , use batch

By using go Keyword

use practice

Select \* from student

Select \* from student where rollno =2

Select \* from student where rollno > 2

Go

It creates a batch , all the queries will be sent together to server

But , can we run this block again

Stored Procedure : It’s a block of Sql statements. Once we create it, we

can call it many times . Resuablity

create procedure GetA

AS

Begin

Select \* from student

Select \* from student where rollno =2

Select \* from student where rollno > 2

end

How do we execute it

Exec GetA

Stored Procedure :

Once you create and run it, it is compiled for the first time, after that it is stored in

a pre-compiled form.

Syntax of stored procedure

Create procedure <procedurename> (parameters)

AS

Begin

Statements

End

How to call / execute it

Exec <procedurename>

create procedure GetStudents

As

Begin

Select \* from student

End

exec GetStudents

create procedure InsertStudent

As

Begin

Insert into student(rollno, name, batch,

marks, coach, course) values (7,'Ajay','B006'

,56,'Kapil','C++')

End

exec InsertStudent

Procedure with parameters

alter procedure InsertStudent (@rollno int , @name varchar(20), @batch char(4),

@marks int , @coach varchar(20) , @course varchar(10))

As

Begin

Insert into student(rollno, name, batch,

marks, coach, course) values (@rollno, @name, @batch,@marks,@coach,@course)

End

exec InsertStudent 8 , 'Kapil', 'B009',78,'Ajeet','Java'

Updating Records

create proc UpdateStudent (@rollno int, @batch char(4),

@marks int , @coach varchar(20), @course varchar(10))

As

Begin

update student set batch = @batch , marks = @marks,

coach = @coach, course = @course where rollno = @rollno

ENd

exec UpdateStudent 3 , 'B009', 100 , 'Lalit', 'Java'

exec GetStudents

Delete Records

create proc DeleteStudent (@rollno int)

As

Begin

Delete student where rollno = @rollno

ENd

exec DeleteStudent 3

Can procedure return some values ? Yes

How

1. By using return statement (We can only return 1 value and that too integer)
2. By using output parameters , we can return more than 1 value

Find out maximum marks thru a procedure

Procedure returning value

create proc GetMaxMarks

as

begin

declare @max int

Select @max= max(marks) from student

return @max

End

* To call it

declare @max int

exec @max = GetMaxMarks

print 'Max Marks are ' + Convert(varchar,@max)

sp\_helptext InsertStudent: used to return statements of a procedure

Write a procedure to Insert a Record, If such rollno exists , return 1 else return 0

alter procedure InsertStudent (@rollno int , @name varchar(20), @batch char(4),

@marks int , @coach varchar(20) , @course varchar(10))

As

Begin

if exists(Select \* from student where rollno=@rollno)

return 1

else

begin

Insert into student(rollno, name, batch,

marks, coach, course) values (@rollno, @name, @batch,@marks,@coach,@course)

return 0

end

End

declare @flag int

exec @flag = InsertStudent 10 , 'Kapil', 'B009',78,'Ajit','Java'

if @flag = 1

print 'Record with this RollNo already exist'

else

print 'Record inserted'

We will pass RollNo and we need corresponding values

We will use output parameters now

create proc GetStudentDetails (@rollno int, @name varchar(20) output ,

@batch char(4) output , @marks int output)

AS

Begin

Select @name = name , @marks = marks , @batch = batch

from student where rollno = @rollno

end

* To Call it

declare @name varchar(20), @batch char(4), @marks int

exec GetStudentDetails 10 , @name output , @batch output , @marks output

print 'Details of RollNo 4 are ' + @name + ' ' + @batch + ' '+

convert(varchar, @marks)

Functions are used to perform some specific task

Functionc cud be

System defined , user defined

System defined

String numeric General , date functions

Len () , lower() upper() trim()

Max() min() sum() power() sin()

Convert ()

Dateadd() , datename() , datediff()

User Defined functions : created by user

Syntax

Create function <functionname> (parameters)

Returns <returntype>

As

Begin

Statements

Return value

end

create function Func1()

returns int

as

begin

return 1

end

select dbo.Func1()

--------------------------------------------------------

create function Func2()

returns varchar(20)

AS

Begin

return 'Hello World'

end

select dbo.FUnc2()

Differences in Stored Procedure & Functions

|  |  |
| --- | --- |
| Stored Procedures | Functions |
| Stored Procedures may or may not return a value | Functions have to always return a value |
| Stored Procedures are stored in a pre-compiled form | Functions are always compiled |
| Stored Procedures can have all DDL / DML / DQL statements | Functions can only have DQL(select  Statements) |
| They can call Functions | Functions can not call sp |
| They can have Exception Handling | No Exception Handling |
| Sp can have both input & output parameters | Functions only work with input parameters. They don’t have output parameters |
|  |  |

create function CombineDetails(@value1 varchar(20),

@value2 varchar(20))

returns varchar(max)

As

Begin

return @value1 + ' ' + @value2

end

select count(\*), max(marks) from student

select dbo.Combinedetails(name, batch ) from student

Create function Fun\_EmployeeInformation()

returns table

as

return(select \* from Student )

select \* from dbo.Fun\_EmployeeInformation()