SELECT

product\_name,

list\_price

FROM

production.products

ORDER BY

product\_name;

CREATE PROCEDURE uspProductList

AS

BEGIN

SELECT

product\_name,

list\_price

FROM

production.products

ORDER BY

product\_name;

END;

The uspProductList is the name of the stored procedure.

The AS keyword separates the heading and the body of the stored procedure.

If the stored procedure has one statement, the BEGIN and END keywords surrounding the statement are optional. However, it is a good practice to include them to make the code clear.

to the CREATE PROCEDURE keywords, you can use the CREATE PROC keywords to make the statement shorter.

to execute the uspProductList stored procedure, you use the following statement:

Deleting a stored procedure

To delete a stored procedure, you use the DROP PROCEDURE or DROP PROC statement:

1

DROP PROCEDURE sp\_name;

or

1

DROP PROC sp\_name;

select \* from employee1

By using "SET NOCOUNT ON" within your stored procedure you can shut off these messages and reduce some of the traffic.

declare @logmsg varchar(100)

set @logmsg = suser\_sname() + ': Tried to access the dotnet system.'

exec xp\_logevent 50005, @logmsg

print @logmsg

select \* from EmployeeDB.dbo.Employees

CREATE PROCEDURE SelectEmployeestabledata

AS

Begin

End

GO

USE [EmployeeDB]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[SelectEmployeestabledata] Script Date: 20-07-2017 03:58:29 \*\*\*\*\*\*/

[SelectEmployeestabledata] Ben

exec dbo.SelectEmployeestabledata

--exec SelectEmployeestabledata

--exec SelectEmployeestabledata @FirstName='Ben'

ALTER PROCEDURE [dbo].[SelectEmployeestabledata]

AS

SELECT \* FROM EmployeeDB.dbo.Employees

Go

ALTER PROCEDURE [dbo].[SelectEmployeestabledata]

@FirstNam varchar(100)

AS

SET NOCOUNT ON

SELECT \* FROM EmployeeDB.dbo.Employees

where FirstName=@FirstNam

PRINT @@ROWCOUNT

GO

Api sql

select \* from aspnet\_Users

select \* from tblPriceSheetDetail

select count(\*) as TablesCount from sys.tables

select count(\*) as ProceduresCount from sys.procedures

By using "SET NOCOUNT ON" within your stored procedure you can shut off these messages and reduce some of the traffic.

declare @logmsg varchar(100)

set @logmsg = suser\_sname() + ': Tried to access the dotnet system.'

exec xp\_logevent 50005, @logmsg

print @logmsg

select \* from EmployeeDB.dbo.Employees

CREATE PROCEDURE SelectEmployeestabledata

AS

Begin

End

GO

USE [EmployeeDB]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[SelectEmployeestabledata] Script Date: 20-07-2017 03:58:29 \*\*\*\*\*\*/

--[SelectEmployeestabledata] Ben

--exec dbo.SelectEmployeestabledata

--exec SelectEmployeestabledata

--exec SelectEmployeestabledata @FirstName='Ben'

--ALTER PROCEDURE [dbo].[SelectEmployeestabledata]

--AS

--SELECT \* FROM EmployeeDB.dbo.Employees

--Go

ALTER PROCEDURE [dbo].[SelectEmployeestabledata]

@FirstNam varchar(100)

AS

SET NOCOUNT ON

SELECT \* FROM EmployeeDB.dbo.Employees

where FirstName=@FirstNam

PRINT @@ROWCOUNT

GO