

# T-SQL STORED PROCEDURES CHEATSHEET

-Bikram

## SYNTAX:

```
CREATE PROCEDURE/ PROC sp_name (parameters) AS  
BEGIN  
    <Statement Block>  
END
```

**Database** - AdventureWorks2014

**Tables** — PurchaseOrderHeader & Vendor in Purchasing schema

USE AdventureWorks2014

```
SELECT * FROM Purchasing.PurchaseOrderHeader  
SELECT * FROM Purchasing.Vendor
```

## Stored procedure with multiple input parameters:

```
CREATE PROC usp_VendorAndEmployee(@VendID  
INT, @EmpID INT)  
AS  
BEGIN  
    SELECT SUM(SubTotal) AS TotalAmount FROM  
    Purchasing.PurchaseOrderHeader  
    WHERE VendorID = @VendID AND  
    EmployeeID = @EmpID  
END
```

## Stored procedure with input-text parameter:

```
CREATE PROC usp_NameLike(@NameText  
NVARCHAR(MAX))  
AS  
BEGIN  
    SELECT * FROM Purchasing.Vendor  
    WHERE NAME LIKE '%' + @NameText + '%'  
END
```

## Stored procedure without any parameters/variables:

```
CREATE PROC usp_Top10TaxPO  
AS  
BEGIN  
    SELECT TOP 10 * FROM  
    Purchasing.PurchaseOrderHeader  
    ORDER BY TaxAmt DESC  
END
```

## Stored procedure with a single input parameter:

```
CREATE PROC  
usp_YearModified(@ModYear INT)  
AS  
BEGIN  
    SELECT * FROM  
    Purchasing.PurchaseOrderHeader  
    WHERE YEAR(ModifiedDate)  
    = @ModYear  
END
```

## Stored procedure with input parameter having default value:

```
CREATE PROC  
usp_PurchaseOrderDetails(@PONumber INT =  
15)  
AS  
BEGIN  
    SELECT * FROM  
    Purchasing.PurchaseOrderHeader  
    WHERE PurchaseOrderID =  
    @PONumber  
END
```

## Stored procedure using Input and Output parameters:

```
CREATE PROC usp_VendorTaxAmt(@VendID INT,  
@VendTaxAmt DECIMAL(10,2) OUTPUT)  
AS  
BEGIN  
    SELECT @VendTaxAmt = SUM(TaxAmt) FROM  
    Purchasing.PurchaseOrderHeader  
    WHERE VendorID = @VendID  
END
```

## Execution of stored procedure with output parameter:

```
DECLARE @VendTax DECIMAL(10,2)  
EXEC usp_VendorTaxAmt 1496, @VendTax OUTPUT  
PRINT @VendTax
```

## Stored procedure using variables:

```
CREATE PROC usp_CompanyList(@CreditRt INT)  
AS  
BEGIN  
    DECLARE @FirmList NVARCHAR(MAX)  
    SET @FirmList = ''  
    SELECT @FirmList=@FirmList + Name + CHAR(10)  
    FROM Purchasing.Vendor  
    WHERE CreditRating= @CreditRt  
    PRINT @FirmList  
END
```

## Stored procedure with the use of RETURN clause:

```
CREATE PROC usp_VendorCountByYear2(@ModYear INT)  
AS  
BEGIN  
    RETURN  
    (SELECT COUNT(VendorID) FROM  
    Purchasing.PurchaseOrderHeader  
    WHERE YEAR(ModifiedDate) = @ModYear)  
END
```

```
DECLARE @VendorCount INT  
EXEC @VendorCount = usp_VendorCountByYear2 2015  
SELECT @VendorCount
```

## Above Stored procedure without using RETURN clause:

```
CREATE PROC usp_VendorCountByYear(@ModYear  
INT,@VendCount INT OUT)  
AS  
BEGIN  
    SELECT @VendCount = COUNT(VendorID) FROM  
    Purchasing.PurchaseOrderHeader  
    WHERE YEAR(ModifiedDate) = @ModYear  
END
```

```
DECLARE @VendorCount INT  
EXEC usp_VendorCountByYear 2015, @VendorCount OUT  
SELECT @VendorCount AS 'Number of vendors'
```