

```
begin tran
commit tran
```

```
begin tran
rollback tran
```

```
begin tran
commit tran
rollback tran
```

```
drop table TestTbl
begin tran
create table TestTbl
(
    id int,
    name varchar(50)
)
commit tran
```

```
drop table TestTbl
begin tran
create table TestTbl
(
    id int,
    name varchar(50)
)
--commit tran
```

```
insert into testtbl values (1,'Hasan')
select * from testtbl
delete from testtbl
```

```
begin transaction
insert into testtbl values (1,'Labib')
rollback transaction
```

```
begin transaction
insert into testtbl values (1,'Habib')
commit transaction
```

```
begin transaction
insert into testtbl values (1,'Rabbi')
```

commit transaction

begin transaction

insert into testtbl values (1,'Fuad')

commit transaction

begin transaction

insert into testtbl values (1,'Masud')

select * from testtbl

rollback transaction

select * from testtbl

declare @t varchar(50)='testT'

begin transaction @t

insert into testtbl values (1,'Masud')

select * from testtbl

rollback transaction @t

select * from testtbl

declare @t varchar(50)='testT'

begin transaction @t

insert into testtbl values (1,'Masud')

select * from testtbl

Commit transaction @t

select * from testtbl

-----error handling

--RAISERROR

--THROW

--TRY_CONVERT and TRY_parse

--@@ERROR

--Structured Error handling Using TRY/CaTCh

select 10/0

RAISERROR('This is a test',16,1)

RAISERROR('This is a test',10,1)

RAISERROR('This is a test',19,1) with log

RAISERROR('This is a test',20,1) with log

Begin try

select 10/0

end try

begin catch

```
    print 'before throw';
    throw;
    print 'after throw';
end catch
```

```
Begin try
    select 10/0
end try
begin catch
    print 'before throw';
    declare @errmsg varchar(4000),
            @errseverity int,
            @errstate int
    set @errmsg=error_message();
    set @errseverity=ERROR_SEVERITY ();
    set @errstate=ERROR_STATE ();
    select @errmsg,@errseverity,@errstate
    RAISERROR('This is a test',@errseverity,@errstate)
    print 'after throw';
end catch
```

```
begin try
    select 10/0
end try
begin catch
    declare @errNumber int
    set @errNumber=@@error
    if @errNumber=8134
    begin
        print 'this is a div errr'
    end
    else
    begin
        print 'not div error'
    end
end catch
```

```
select * from testtbl
begin try
    begin transaction
    insert into testtbl values (1,'Masud3')
    select 10/0
    commit tran
```

```
end try
begin catch
    rollback
    raiserror ('this is a div error',16,1)
end catch
```

```
SELECT TRY_CONVERT(DATETIME, '1752-12-31');
SELECT TRY_CONVERT(DATETIME, '1753-01-01');
select TRY_CONVERT(int,'01a')
```

```
select try_parse('01/Feb/2014' as date)
```

```
-----dynamic sql
select * from testtbl
select * from sqltest
```

```
declare @tblName varchar(50)='hr.Employees'
execute ('select * from ' + @tblName)
```

```
declare @tblName varchar(50)='sales.MyOrders'
exec ('select * from ' + @tblName)
```

```
declare @tblName varchar(50)='testtbl'
set @tblName='select * from '+@tblName
exec (@tblName)
```

```
create proc testProc
as
begin
declare @tblName varchar(50)='testtbl'
set @tblName='select * from '+@tblName
exec (@tblName)
end
```

```
exec testProc
```

```
select * from testtbl where name='Masud'
select ('select * from testtbl where name="Masud"')
exec ('select * from testtbl where name="Masud"')
```

```
declare @testName varchar(50)
set @testName=""Masud""
--select ('select * from testtbl where name='+@testName+')

```

```
exec ('select * from testtbl where name='+@testName+')
select * from testtbl where name='Masud'
select * from testtbl where name='Masud'
```

```
declare @testName varchar(50)
set @testName=""Masud" or 1=1'
exec ('select * from testtbl where name='+@testName+')
```

```
EXECUTE sys.sp_executesql
    @statement =N'select * from testtbl where name=@testName;',
    @params = N'@testName varchar(50)',
    @testName = 'Masud or 1=1';
```