



Tecnológico de Estudios Superiores de Ecatepec

**División de Ingeniería en Sistemas
Computacionales**

Academia en Ciencias de la Ingeniería

Materia

Administración de Base de Datos

Grupo 5701

Alumno

Campero Granados Luis Daniel

Profesor

Jiménez Alfaro Abraham Jorge

Practica 1

Practica 1

Abrimos SQL Server Management Studio. Una vez abierto empezaremos con nuestras actividades en un nuevo Query.

Tendremos unas dos tablas cada una con su clave primaria, pero en el caso de la segunda tabla tendrá una clave foránea que será la primaria de la primera.

Las tablas las definimos con *Not Null*, debemos especificarlo siempre en la lista de campos a insertar. Pasado las inserciones realizaremos los procedimientos almacenados, los procedimientos almacenados de SQL pueden ser instrucciones de tipo Transact-SQL o referencias a un método de Common Runtime Language (CLR) de .NET. Dichas instrucciones se encuentran almacenadas de forma física con un nombre dentro de la base de datos.

Código del Query

```
Create Database Escuela
```

```
Use Escuela
```

```
Create Table Carreras(
```

```
Id_Carrera Int Not Null,
```

```
Nombre_Carrera Varchar(50) Not Null,
```

```
Titulo Varchar(50) Null,
```

```
Constraint Pk_Carreras Primary Key Clustered (Id_Carrera Asc))
```

```
Go
```

```
Create Table Sexos(
```

```
Id_Sexo Int Not Null,
```

```
N_Sexo Varchar(20) Not Null,
```

```
Constraint Pk_Sexos Primary Key Clustered (Id_Sexo Asc))
```

```
Go
```

```
Create Table Estados_Cursados(
```

```
Id_Estado Int Not Null,  
N_Estado Varchar(30) Not Null,  
Constraint Pk_Estados_Cursados Primary Key Clustered (Id_Estado Asc))  
Go
```

```
Create Table Barrios(  
Id_Barrio Int Not Null,  
N_Barrio Varchar(50) Not Null,  
Constraint Pk_Barrios Primary Key Clustered (Id_Barrio Asc))  
Go
```

```
Create Table Tipo_Notas(  
Id_Tipo_Nota Int Not Null,  
N_Tipo_Nota Varchar(20) Not Null,  
Constraint Pk_Tipo_Nota Primary Key Clustered (Id_Tipo_Nota Asc))  
Go
```

```
Create Table Tipo_Documentos(  
Id_Tipo_Documento Int Not Null,  
N_Tipo_Documento Varchar(20) Not Null,  
Constraint Pk_Tipo_Documentos Primary Key Clustered (Id_Tipo_Documento Asc))  
Go
```

```
Create Table Materias(  
Id_Materia Int Not Null,  
Nombre Varchar(50) Not Null,  
Constraint Pk_Materias Primary Key Clustered (Id_Materia Asc))  
Go
```

```
Create Table Alumnos_Cursando_Materias(  
Id_Materia Int Not Null,  
Id_Carrera Int Not Null,  
Id_Estudiante Int Not Null,  
Fecha_Inscripcion Datetime Not Null,
```

Id_Estado Int Not Null,

Fecha_Estado Datetime Not Null,

Constraint Pk_Alumnos_Cursando_Materias Primary Key Clustered (Id_Materia Asc, Id_Carrera Asc, Id_Estudiante Asc, Fecha_Inscripcion Asc))

Go

Create Table Alumnos_Examenes(

Id_Materia Int Not Null,

Id_Carrera Int Not Null,

Id_Estudiante Int Not Null,

Fecha_Del_Examen Datetime Not Null,

Fecha_Inscripcion_A_Examen Datetime Not Null,

Nota Tinyint Null,

Constraint Pk_Alumnos_Examenes Primary Key Clustered (Id_Materia Asc, Id_Carrera Asc, Id_Estudiante Asc, Fecha_Del_Examen Asc))

Go

Create Table Alumnos_Cursando_Notas(

Id_Materia Int Not Null,

Id_Carrera Int Not Null,

Id_Estudiante Int Not Null,

Fecha_Inscripcion Datetime Not Null,

Fecha_Nota Datetime Not Null,

Id_Tipo_Nota Int Not Null,

Nota Tinyint Null,

Constraint Pk_Alumnos_Cursando_Notas Primary Key Clustered (Id_Materia Asc, Id_Carrera Asc, Id_Estudiante Asc, Fecha_Inscripcion Asc, Fecha_Nota Asc))

Go

Create Table Inscripciones_A_Carreras(

Id_Estudiante Int Not Null,

Id_Carrera Int Not Null,

Fecha Datetime Not Null,

Constraint Pk_Plan_De_Carrera Primary Key Clustered (Id_Estudiante Asc, Id_Carrera Asc))

Go

```
Create Table Plan_De_Carrera(  
    Id_Materia Int Not Null,  
    Id_Carrera Int Not Null,  
    Anio_Cursado Int Null,  
    Cuatrimestre_Anio Int Null,  
    Constraint Pk_Pla_De_Carrera Primary Key Clustered (Id_Materia Asc, Id_Carrera Asc))  
Go
```

```
Create Table Estudiantes(  
    Id_Estudiante Int Not Null,  
    Apellido Varchar(30) Not Null,  
    Nombres Varchar(100) Not Null,  
    Id_Tipo_Documento Int Not Null,  
    Documento Bigint Not Null,  
    Id_Sexo Int Null,  
    Calle Varchar(60) Not Null,  
    Calle_Numero Varchar(30) Not Null,  
    Id_Barrio Int Null,  
    Constraint Pk_Estudiantes Primary Key Clustered (Id_Estudiante Asc))  
Go
```

```
Create Table Materias_X_Turnos_Examenes(  
    Id_Turno Int Not Null,  
    Id_Materia Int Not Null,  
    Id_Carrera Int Not Null,  
    Fecha_Del_Examen Datetime Not Null,  
    Constraint Pk_Materias_X_Turnos_Examenes Primary Key Clustered (Id_Materia Asc, Id_Carrera Asc, Fecha_Del_Examen Asc)  
)  
Go
```

```
Alter Table Alumnos_Cursando_Materias Add Constraint Fk_Alumnos_Cursando_Materias_Estados Foreign Key(Id_Estado)  
References Estados_Cursados (Id_Estado)
```

Go

```
Alter Table Alumnos_Cursando_Materias Add Constraint Fk_Alumnos_Cursando_Materias_Estudiantes Foreign Key(Id_Estudiante)
```

References Estudiantes (Id_Estudiante)

Go

```
Alter Table Alumnos_Cursando_Materias Add Constraint Fk_Alumnos_Cursando_Materias_Plan_Carrera Foreign Key(Id_Materia, Id_Carrera)
```

References Plan_De_Carrera (Id_Materia, Id_Carrera)

Go

```
Alter Table Alumnos_Examenes Add Constraint Fk_Alumnos_Examenes_Estudiantes Foreign Key(Id_Estudiante)
```

References Estudiantes (Id_Estudiante)

Go

```
Alter Table Alumnos_Examenes Add Constraint Fk_Alumnos_Examenes_Materias_X_Turnos_Examenes Foreign
```

Key(Id_Materia, Id_Carrera, Fecha_Del_Examen)

References Materias_X_Turnos_Examenes (Id_Materia, Id_Carrera, Fecha_Del_Examen)

Go

```
Alter Table Alumnos_Cursando_Notas Add Constraint Fk_Alumnos_Cursando_Notas_Tipo_Notas Foreign Key(Id_Tipo_Nota)
```

References Tipo_Notas (Id_Tipo_Nota)

Go

```
Alter Table Alumnos_Cursando_Notas Add Constraint Fk_Aiumnos_Cursando_Notas_Alumnos_Cursando_Materias
```

Foreign Key(Id_Materia, Id_Carrera, Id_Estudiante, Fecha_Inscripcion)

References Alumnos_Cursando_Materias (Id_Materia, Id_Carrera, Id_Estudiante, Fecha_Inscripcion)

Go

```
Alter Table Inscripciones_A_Carreras Add Constraint Fk_Inscripciones_A_Carreras_Carreras Foreign Key(Id_Carrera)
```

References Carreras (Id_Carrera)

Go

--

Alter Table Inscripciones_A_Carreras Add Constraint Fk_Inscripciones_A_Carreras_Estudiantes Foreign Key(Id_Estudiante)

References Estudiantes (Id_Estudiante)

Go

Alter Table Plan_De_Carrera Add Constraint Fk_Plan_De_Carrera_Carreras Foreign Key(Id_Carrera)

References Carreras (Id_Carrera)

Go

Alter Table Plan_De_Carrera Add Constraint Fk_Plan_De_Carrera_Materias Foreign Key(Id_Materia)

References Materias (Id_Materia)

Go

Alter Table Estudiantes Add Constraint Fk_Estudiantes_Barrios Foreign Key(Id_Barrio)

References Barrios (Id_Barrio)

Go

Alter Table Estudiantes Add Constraint Fk_Estudiantes_Sexos Foreign Key(Id_Sexo)

References Sexos (Id_Sexo)

Go

Alter Table Estudiantes Add Constraint Fk_Estudiantes_Tipo_Documentos Foreign Key(Id_Tipo_Documento)

References Tipo_Documentos (Id_Tipo_Documento)

Go

Alter Table Materias_X_Turnos_Examenes Add Constraint Fk_Materias_X_Turnos_Examenes_Plan_Carrera

Foreign Key(Id_Materia, Id_Carrera)

References Plan_De_Carrera (Id_Materia, Id_Carrera)

Go

Insert Into Barrios (Id_Barrio,N_Barrio) Values (1,'Alto Aiberdi')

Insert Into Barrios (Id_Barrio,N_Barrio) Values (2,'Yofie')

Insert Into Barrios (Id_Barrio,N_Barrio) Values (3,'Nueva Cordoba')

Insert Into Barrios (Id_Barrio,N_Barrio) Values (4,'Centro')

Insert Into Barrios (Id_Barrio,N_Barrio) Values (5,'Los Boulevares')

Insert Into Barrios (Id_Barrio,N_Barrio) Values (6,'Guemes')

Insert Into Barrios (Id_Barrio,N_Barrio) Values (7,'Ipona')

Insert Into Barrios (Id_Barrio,N_Barrio) Values (8,'Alta Cordoba')

Insert Into Barrios (Id_Barrio,N_Barrio) Values (9,'General Paz')

Insert Into Tipo_Notas(Id_Tipo_Nota, N_Tipo_Nota) Values (1,'Parcial 1')

Insert Into Tipo_Notas(Id_Tipo_Nota, N_Tipo_Nota) Values (2,'Parcial 2')

Insert Into Tipo_Notas(Id_Tipo_Nota, N_Tipo_Nota) Values (3,'Final')

Insert Into Tipo_Notas(Id_Tipo_Nota, N_Tipo_Nota) Values (4,'Trabajo Practico')

Insert Into Tipo_Documentos(Id_Tipo_Documento,N_Tipo_Documento) Values (1,'Dni')

Insert Into Tipo_Documentos(Id_Tipo_Documento,N_Tipo_Documento) Values (2,'Lc')

Insert Into Carreras(Id_Carrera,Nombre_Carrera,Titulo) Values (1, 'Ingeniería En Sistemas De Informacion',
'Ingeniero En Sistemas De Informacion')

Insert Into Sexos(Id_Sexo,N_Sexo) Values (1,'Masculino')

Insert Into Sexos(Id_Sexo,N_Sexo) Values (2,'Femenino')

Insert Into Estados_Cursados (Id_Estado,N_Estado) Values (1,'Inscripto')

Insert Into Estados_Cursados (Id_Estado,N_Estado) Values (2,'Regular')

Insert Into Estados_Cursados (Id_Estado,N_Estado) Values (3,'Libre')

Insert Into Materias(Id_Materia,Nombre) Values (1,'Algoritmos Y Estructuras De Datos')

Insert Into Materias(Id_Materia,Nombre) Values (2,'Analisis Matematico I')

Insert Into Materias(Id_Materia,Nombre) Values (3,'Analisis Matematico II')

Insert Into Materias(Id_Materia,Nombre) Values (4,'Algebra Y Geometria Analitica')

Insert Into Materias(Id_Materia,Nombre) Values (5,'Matematica Discreta')

Insert Into Materias(Id_Materia,Nombre) Values (6,'Sistemas Y Organizaciones')

Insert Into Materias(Id_Materia,Nombre) Values (7,'Ingenieria Y Sociedad')

Insert Into Materias(Id_Materia,Nombre) Values (8,'Paradigmas De Programacion')

Insert Into Plan_De_Carrera(Id_Materia,Id_Carrera,Anio_Cursado,Cuatrimestre_Anio) Values (1,1,1,1)

Insert Into Plan_De_Carrera(Id_Materia,Id_Carrera,Anio_Cursado,Cuatrimestre_Anio) Values (2,1,1,1)

Insert Into Plan_De_Carrera(Id_Materia,Id_Carrera,Anio_Cursado,Cuatrimestre_Anio) Values (3,1,1,1)

Insert Into Plan_De_Carrera(Id_Materia,Id_Carrera,Anio_Cursado,Cuatrimestre_Anio) Values (4,1,1,1)

Insert Into Plan_De_Carrera(Id_Materia,Id_Carrera,Anio_Cursado,Cuatrimestre_Anio) Values (5,1,1,2)

Insert Into Plan_De_Carrera(Id_Materia,Id_Carrera,Anio_Cursado,Cuatrimestre_Anio) Values (6,1,1,2)

Insert Into Plan_De_Carrera(Id_Materia,Id_Carrera,Anio_Cursado,Cuatrimestre_Anio) Values (7,1,1,2)

Insert Into Plan_De_Carrera(Id_Materia,Id_Carrera,Anio_Cursado,Cuatrimestre_Anio) Values (8,1,1,2)

Insert
Estudiantes(Id_Estudiante,Apellido,Nombres,Id_Tipo_Documento,Documento,Id_Sexo,Calle,Calle_Numero,Id_Barrio) Into
Values (1,'Abrutsky','Maximiliano Adrian',1,23852963,1,'Av. Colon','1035',1)

Insert
Estudiantes(Id_Estudiante,Apellido,Nombres,Id_Tipo_Documento,Documento,Id_Sexo,Calle,Calle_Numero,Id_Barrio) Into
Values (2,'Damiano','Luis',1,19147258,1,'Dean Funes','903',1)

Insert Into Inscripciones_A_Carreras(Id_Estudiante,Id_Carrera,Fecha) Values (1,1,Convert(Datetime,'26/01/2009',103))

Insert Into Alumnos_Cursando_Materias (Id_Materia,Id_Carrera,Id_Estudiante,Fecha_Inscripcion,Id_Estado,Fecha_Estado)
Values (1,1,1,Convert(Datetime,'01/02/2009',103),1,Convert(Datetime,'01/02/2009',103))

Insert Into Alumnos_Cursando_Materias (Id_Materia,Id_Carrera,Id_Estudiante,Fecha_Inscripcion,Id_Estado,Fecha_Estado)
Values (2,1,1,Convert(Datetime,'01/02/2009',103),1,Convert(Datetime,'01/02/2009',103))

Insert Into Alumnos_Cursando_Materias (Id_Materia,Id_Carrera,Id_Estudiante,Fecha_Inscripcion,Id_Estado,Fecha_Estado)
Values (3,1,1,Convert(Datetime,'01/02/2009',103),1,Convert(Datetime,'01/02/2009',103))

Insert Into Alumnos_Cursando_Materias (Id_Materia,Id_Carrera,Id_Estudiante,Fecha_Inscripcion,Id_Estado,Fecha_Estado)
Values (4,1,1,Convert(Datetime,'01/02/2009',103),1,Convert(Datetime,'01/02/2009',103))

Insert
Alumnos_Cursando_Notas(Id_Materia,Id_Carrera,Id_Estudiante,Fecha_Inscripcion,Fecha_Nota,Id_Tipo_Nota,Nota) Into
Values (1,1,1,Convert(Datetime,'01/02/2009',103),Convert (Datetime,'15/03/2009',103),1,5)

Insert
Alumnos_Cursando_Notas(Id_Materia,Id_Carrera,Id_Estudiante,Fecha_Inscripcion,Fecha_Nota,Id_Tipo_Nota,Nota) Into
Values (1,1,1,Convert(Datetime,'01/02/2009',103),Convert (Datetime,'26/05/2009',103),2,7)

Insert
Alumnos_Cursando_Notas(Id_Materia,Id_Carrera,Id_Estudiante,Fecha_Inscripcion,Fecha_Nota,Id_Tipo_Nota,Nota) Into
Values (2,1,1,Convert(Datetime,'01/02/2009',103),Convert (Datetime,'18/03/2009',103),1,3)

Insert
Alumnos_Cursando_Notas(Id_Materia,Id_Carrera,Id_Estudiante,Fecha_Inscripcion,Fecha_Nota,Id_Tipo_Nota,Nota) Into

Values (2,1,1,Convert(Datetime,'01/02/2009',103),Convert (Datetime,'29/05/2009',103),2,4)

Go

Create Function Obtener_Nombre_Completo(@Id_Estudiante Int)

Returns Varchar(150)

As

Begin

Declare @Nombrecompleto Varchar(150)

Select @Nombrecompleto = Upper(Left(Apellido, 1)) + Lower(Right(Apellido, Len(Apellido)-1))

+','

+ Lower(Nombres)

From Estudiantes

Where Id_Estudiante = @Id_Estudiante

Return @Nombrecompleto

End

Go

Select Id_Estudiante, Apellido, Nombres, Dbo.Obtener_Nombre_Completo(Id_Estudiante)

As Nombrecompleto

From Estudiantes

Go

Create Procedure Actualizar_Estado_Materia

@Id_Estudiante Int,@Id_Materia Int,@Id_Carrera Int

As

Declare @Fecha_Inscripcion_Cursado Datetime

Select @Fecha_Inscripcion_Cursado = Max(Fecha_Inscripcion)

From Alumnos_Cursando_Materias

Where Id_Materia = @Id_Materia

And Id_Estudiante = @Id_Estudiante

If @Fecha_Inscripcion_Cursado Is Null

Print 'El Alumno No Se Ha Inscripto A La Materia-Carrera Especifica'

Else

Begin

Declare @Cant_Parciales Int

Declare @Cant_Parciales_Aprobados Int

Declare @Cant_Parciales_Reprobados Int

Select @Cant_Parciales = Count(*)

From Alumnos_Cursando_Notas Notas

Inner Join Materias Mat On Notas.Id_Materia = Mat.Id_Materia

Inner Join Tipo_Notas Tip On Notas.Id_Tipo_Nota = Tip.Id_Tipo_Nota

Where Notas.Id_Estudiante = @Id_Estudiante

And Notas.Id_Materia = @Id_Materia

And Notas.Id_Carrera = @Id_Carrera

And Notas.Fecha_Inscripcion = @Fecha_Inscripcion_Cursado

And Lower(Tip.N_Tipo_Nota) Like '%Parcial%'

Select @Cant_Parciales_Aprobados = Count(*)

From Alumnos_Cursando_Notas Notas

Inner Join Materias Mat On Notas.Id_Materia = Mat.Id_Materia

Inner Join Tipo_Notas Tip On Notas.Id_Tipo_Nota = Tip.Id_Tipo_Nota

Where Notas.Id_Estudiante = @Id_Estudiante

And Notas.Id_Materia = @Id_Materia

And Notas.Id_Carrera = @Id_Carrera

And Notas.Fecha_Inscripcion = @Fecha_Inscripcion_Cursado

And Lower(Tip.N_Tipo_Nota) Like '%Parcial%'

And Notas.Nota >= 4

Set @Cant_Parciales_Reprobados = @Cant_Parciales - @Cant_Parciales_Aprobados

Print 'Cantidad De Parciales Rendidos =' + Cast(@Cant_Parciales As Char)

```

Print 'Cantidad De Parciales Aprobados =' + Cast(@Cant_Parciales_Aprobados As Char)
Print 'Cantidad De Parciales Reprobados =' + Cast(@Cant_Parciales_Reprobados As Char)

If @Cant_Parciales < 2
    Print 'La Cantidad Rendida De Parciales Es Menor A 2, No Se Cambia El
Estado'
Else If @Cant_Parciales_Reprobados >= 2
    Begin
        Print '2 Parciales Aprobados, Regular'
        Update Alumnos_Cursando_Materias
        Set Id_Estado = (Select Id_Estado From
Estados_Cursados Where N_Estado Like 'Regular')

        Where Id_Estudiante = @Id_Estudiante
        And Id_Materia = @Id_Materia
        And Id_Carrera = @Id_Carrera
        And Fecha_Inscripcion =
@Fecha_Inscripcion_Cursado
    End
Else
    Begin
        Print 'Menos De 2 Parciales Aprobados, Libre'
        -- Actualización Al Estado Libre
        Update Alumnos_Cursando_Materias
        Set Id_Estado = (Select Id_Estado From
Estados_Cursados Where N_Estado Like 'Libre')

        Where Id_Estudiante = @Id_Estudiante
        And Id_Materia = @Id_Materia
        And Id_Carrera = @Id_Carrera
        And Fecha_Inscripcion =
@Fecha_Inscripcion_Cursado
    End
End

--Invocacion: 1
Exec Actualizar_Estado_Materia 1,8,1

```

--Resultado: El Alumno No Se Ha Inscripto A La Materia — Carrera Especificada

--Invocacion: 2

Exec Actualizar_Estado_Materia 1,2,1

/*Resultado:

Cantidad De Parciales Rendidos = 2

Cantidad De Parciales Aprobados = 1

Cantidad De Parciales Reprobados = 1

Mano: De 2 Parciales Aprobados, Libre

*/

--Invocacion: 3

Exec Actualizar_Estado_Materia 1,1,1

/*Resultado:

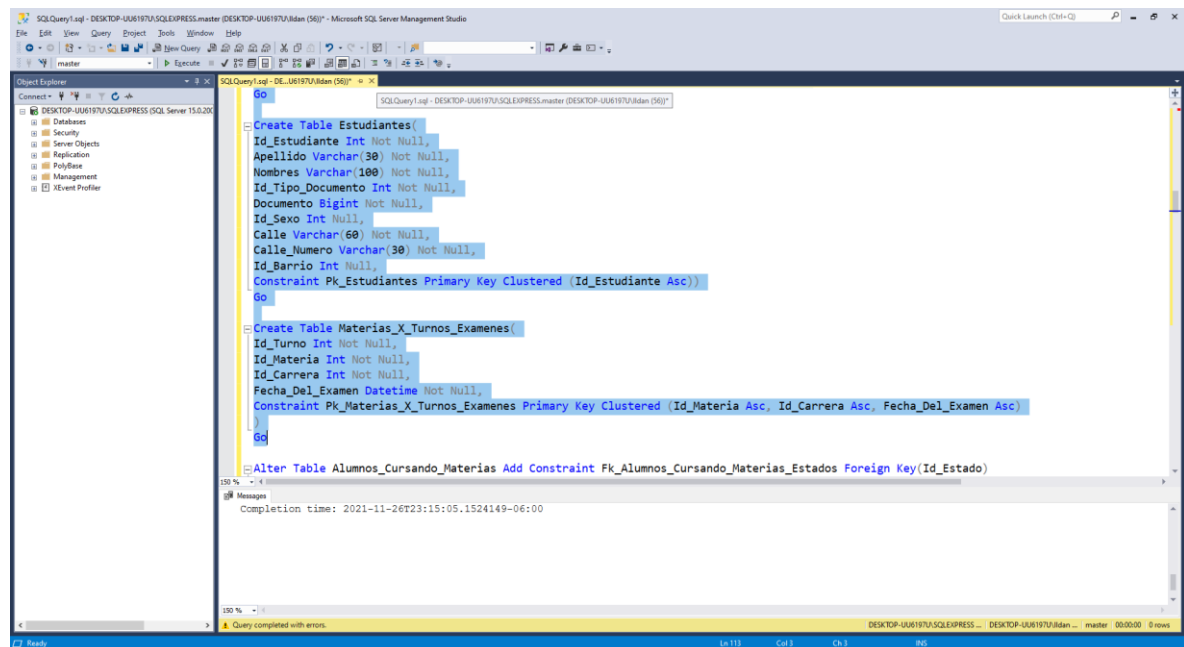
Cantidad De Parciales Rendidos = 2

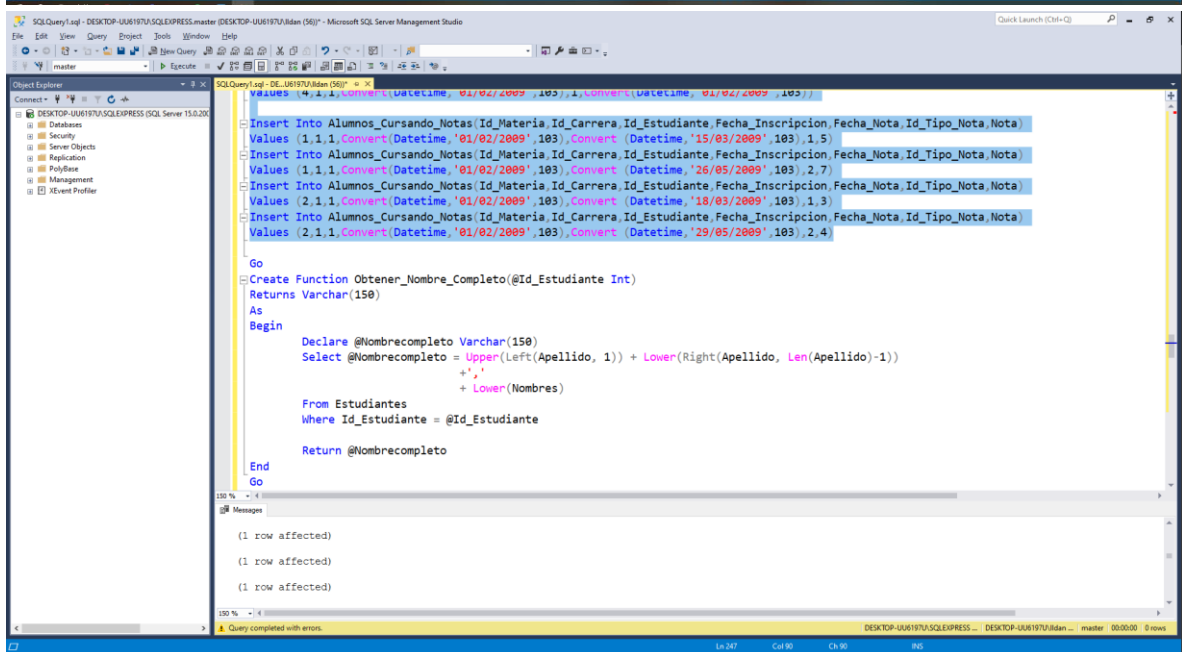
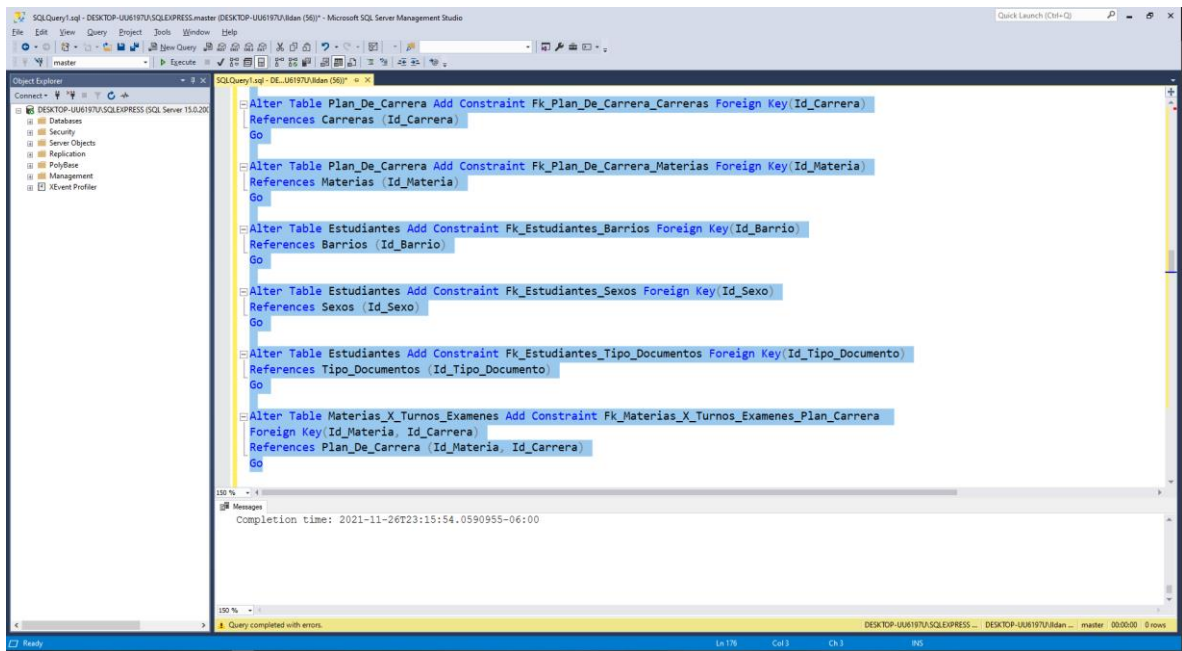
Cantidad De Parciales Aprobados = 2

Cantidad De Parciales Reprobados = 0

2 Parciales Aprobados, Regular

*/





SQLQuery1.sql - DESKTOP-U06197U\SQLEXPRESS-master (DESKTOP-U06197U) - Microsoft SQL Server Management Studio

Object Explorer

- DESKTOP-U06197U\SQLEXPRESS (SQL Server 15.0.2005.5)
- Database
- Security
- Server Objects
- Replication
- PolyBase
- Management
- Event Viewer

```
Insert Into Alumnos_Cursando_Notas (Id_Materia, Id_Carrera, Id_Estudiante, Fecha_Inscripcion, Fecha_Nota, Id_Tipo_Nota, Nota)
Values (2,1,1,Convert(Datetime,'01/02/2009'),103,Convert (Datetime,'18/03/2009'),103),1,3)
Insert Into Alumnos_Cursando_Notas (Id_Materia, Id_Carrera, Id_Estudiante, Fecha_Inscripcion, Fecha_Nota, Id_Tipo_Nota, Nota)
Values (2,1,1,Convert(Datetime,'01/02/2009'),103),Convert (Datetime,'29/05/2009'),103),2,4)

Go

Create Function Obtener_Nombre_Completo (@Id_Estudiante Int)
Returns Varchar(150)
As
Begin
    Declare @NombreCompleto Varchar(150)
    Select @NombreCompleto = Upper(Left(Apellido, 1)) + Lower(Right(Apellido, Len(Apellido)-1))
    + ' '
    + Lower(Nombres)
    From Estudiantes
    Where Id_Estudiante = @Id_Estudiante

    Return @NombreCompleto
End
Go

Select Id_Estudiante, Apellido, Nombres, Dbo.Obtener_Nombre_Completo(Id_Estudiante)
As NombreCompleto
From Estudiantes
```

Completion time: 2021-11-26T23:17:02.6720715-06:00

Query completed with errors.

SQLQuery1.sql - DESKTOP-U06197U\SQLEXPRESS-master (DESKTOP-U06197U) - Microsoft SQL Server Management Studio

Object Explorer

- DESKTOP-U06197U\SQLEXPRESS (SQL Server 15.0.2005.5)
- Database
- Security
- Server Objects
- Replication
- PolyBase
- Management
- Event Viewer

```
FROM Estudiantes
Where Id_Estudiante = @Id_Estudiante

Return @NombreCompleto
End
Go

Select Id_Estudiante, Apellido, Nombres, Dbo.Obtener_Nombre_Completo Id_Estudiante
As NombreCompleto
From Estudiantes

Go

Create Procedure Actualizar_Estado_Materia
@Id_Estudiante Int, @Id_Materia Int, @Id_Carrera Int
As
Declare @Fecha_Inscripcion_Cursado Datetime

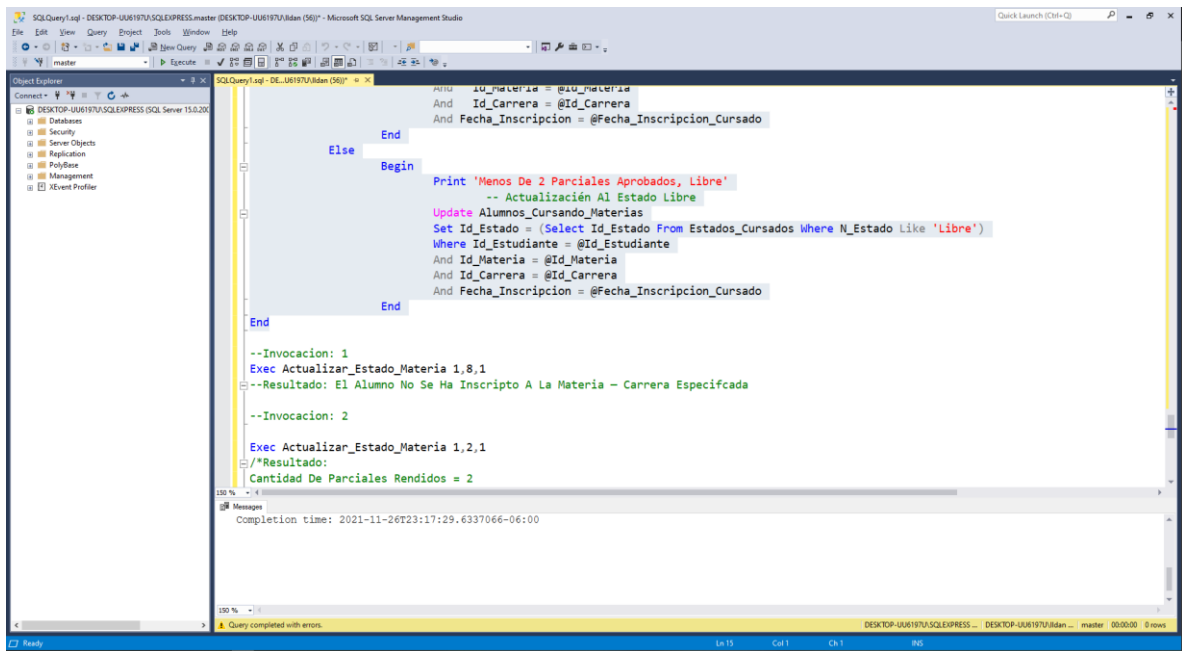
Select @Fecha_Inscripcion_Cursado = Max(Fecha_Inscripcion)
From Alumnos_Cursando_Materias
Where Id_Materia = @Id_Materia
And Id_Estudiante = @Id_Estudiante

If @Fecha_Inscripcion_Cursado Is Null
Print 'El Alumno No Se Ha Inscripto A La Materia-Carrera Especifica'
Else
```

Results

Id_Estudiante	Apellido	Nombres	NombreCompleto
1	Rodriguez	Manuelito Adrian	Rodriguez manuelito adrian
2	Daniels	Luis	Daniels Luis

Query executed successfully.



Resultados

