PROCEDIMIENTOS ALM DE BD FITFA PRUEBA 22-09-01

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

ALTER Procedure [dbo].[Activar\_InstructorInvitado]

@usuario\_doc int, ---es el documento del instructor nuevo

@Instructor\_id int, ---es el id de maestro del instructor nuevo---o sea galarraga

@institucion\_id int

as

declare @usuario\_id int

select @usuario\_id = usuario\_id from usuario where usuario.usuario\_doc = @usuario\_doc

--select @Institucion\_id = institucion\_id from instructor where instructor\_id=@Instructor\_id

declare @nuevo\_instructor int

insert

instructor (usuario\_id, centro\_id)

values

(@usuario\_id, '')

select @nuevo\_instructor = @@IDENTITY

insert institucion\_x\_instructor (instructor\_id,institucion\_id)values

(@nuevo\_instructor,@institucion\_id)

GO

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

ALTER Procedure [dbo].[Activar\_Usuario]

@Usuario\_id int, ---es el id del alumno q sera activado

@Instructor\_id int, ---ojo este es id de usuario no instructor

@usuario\_tipo varchar(50),

@institucion\_id int

as

if @usuario\_tipo ='instructor'

begin

--declare @Institucion\_id int

--select @Institucion\_id = institucion\_id from instructor where instructor\_id=@Instructor\_id

declare @nuevo\_instructor int

A

insert

instructor (usuario\_id, centro\_id)

values

(@Usuario\_id, '')

select @nuevo\_instructor = @@IDENTITY

insert

institucion\_x\_instructor (instructor\_id,institucion\_id)

values

(@nuevo\_instructor,@institucion\_id)

end

update Usuario set

usuario\_estado='activo'

where usuario\_id=@Usuario\_id

declare @id\_instructor int ---este es el q se loguea, el instructor que va a habilitar al alumno

select @id\_instructor = instructor.instructor\_id from instructor where instructor.usuario\_id=@instructor\_id ---instructor\_id es usuario\_id q traigo del login de la app

update alumnos\_x\_instructor set alumno\_x\_instructor\_estado='activo' where usuario\_id=@Usuario\_id and alumnos\_x\_instructor.instructor\_id=@id\_instructor

GO

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

ALTER procedure [dbo].[alumuno\_x\_instructor\_Actulizar]

@instructor\_id int,

@usuario\_id INT

as

update alumnos\_x\_instructor set instructor\_id=@instructor\_id where usuario\_id=@usuario\_id

GO

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

ALTER procedure [dbo].[area\_alta]

@area\_descripcion varchar(max),

@evento\_id int

as

insert Area (Area\_descripcion , evento\_id ) values (@area\_descripcion, @evento\_id )

GO

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

ALTER procedure [dbo].[area\_obtener\_asignadas]

@evento\_id int,

@area\_id int

as

select Area.Area\_id as 'id', Area.Area\_descripcion as 'Area' from Area where Area.evento\_id=@evento\_id order by Area.Area\_id asc

---esta uso cuando accedo desde el usuario torneo, solo me muestra 1 area en particular para ese evento

select Area.Area\_id as 'id', Area.Area\_descripcion as 'Area' from Area where Area.evento\_id=@evento\_id and Area.Area\_id = @area\_id order by Area.Area\_id asc

GO

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

ALTER procedure [dbo].[Carga\_Competidor\_Guardar]

@usuario\_apellido varchar(50),

@usuario\_nombre varchar(50),

@instructor\_id int,

@institucion\_id int,

@evento\_id int,

@inscripcion\_fechahora datetime,

@inscripcion\_peso decimal(5,2),

@categoria\_id int

as

declare @usuario\_id int

insert dbo.usuario (

usuario\_apellido,

usuario\_nombre,

usuario\_estado,

Prueba

)

values (

@usuario\_apellido,

@usuario\_nombre,

'activo',

'Borrar')

select @usuario\_id = @@IDENTITY

--insertar alumno x instructor

insert dbo.alumnos\_x\_instructor (usuario\_id, instructor\_id, alumno\_x\_instructor\_estado, institucion\_id, Prueba) values (@usuario\_id, @instructor\_id, 'activo', @institucion\_id,'Borrar')

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

--alta inscripcion

insert inscripcion (usuario\_id, evento\_id, inscripcion\_fechahora, inscripcion\_peso, inscripcion\_instructor\_id ) values(@usuario\_id, @evento\_id, @inscripcion\_fechahora, @inscripcion\_peso , @instructor\_id)

---recupero el tipoevento, me sirve para validar si es necesario hacer un alta en "inscripciones\_x\_torneo"

declare @inscripcion\_id int

SELECT @inscripcion\_id =@@IDENTITY

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

insert inscripciones\_x\_torneo (inscripcion\_id, categoria\_id, en\_llave) values (@inscripcion\_id, @categoria\_id, 'no')

GO

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

ALTER Procedure [dbo].[Carga\_Competidor\_Obtener\_Categorias]

@sexo as varchar(50),

@tipo as varchar(50)

as

select \* from categoria where categoria\_sexo=@sexo and categoria\_tipo=@tipo

select \* from graduacion order by graduacion.graduacion\_id asc

GO

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

ALTER procedure [dbo].[Categoria\_buscar]

@categoria\_sexo varchar(50),

@categoria\_tipo varchar(50),

@categoria\_gradinicial varchar(50),

@categoria\_gradfinal varchar(50),

@categoria\_edadinicial varchar(50),

@categoria\_edadfinal varchar(50),

@categoria\_peso\_inicial varchar(50),

@categoria\_peso\_Final varchar(50)

as

select \* from categoria where categoria\_sexo = @categoria\_sexo and categoria\_tipo = @categoria\_tipo

and categoria\_gradinicial = @categoria\_gradinicial and categoria\_gradfinal = @categoria\_gradfinal

and categoria\_edadinicial = @categoria\_edadinicial and categoria\_edadfinal = @categoria\_edadfinal

and categoria\_peso\_inical = @categoria\_peso\_inicial and categoria\_peso\_Final = @categoria\_peso\_Final

GO

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

ALTER procedure [dbo].[Categoria\_obtener\_info]

@categoria\_id int

as

select

categoria\_id,

categoria\_sexo,

categoria\_gradinicial,

categoria\_gradfinal,

categoria\_edadinicial,

categoria\_edadfinal,

categoria\_peso\_inical,

categoria\_peso\_Final,

categoria\_tipo

from categoria where categoria.categoria\_id = @categoria\_id

GO

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

ALTER procedure [dbo].[Costos\_modificar]

@Costos\_id int,

@Costos\_monto decimal(38,2)

as

update Costos set Costos\_monto = @Costos\_monto where Costos.Costos\_id = @Costos\_id

GO

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

ALTER procedure [dbo].[Costos\_obtener]

as

--0) recupero los costos solo de examenes.

select Costos\_id, Costos\_descripcion, Costos\_tipo, Costos\_monto, graduacion\_id

from Costos where Costos.Costos\_tipo = 'Examen' order by Costos.Costos\_id asc

--1) recupero los otros costos

select Costos\_id, Costos\_descripcion, Costos\_tipo, Costos\_monto, graduacion\_id

from Costos where Costos.Costos\_tipo = 'Otro' order by Costos.Costos\_descripcion asc

GO

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

ALTER procedure [dbo].[Curso\_recuperar\_inscriptos]

@evento\_id int

as

select \* from(SELECT dbo.evento.evento\_id, dbo.inscripcion.inscripcion\_id, dbo.inscripcion.usuario\_id, dbo.usuario.graduacion\_id, dbo.evento.evento\_descripcion, dbo.evento.evento\_fecha, dbo.evento.evento\_costo, dbo.evento.evento\_direccion,

usuario.usuario\_apellido+', '+usuario.usuario\_nombre as 'Alumno', dbo.graduacion.graduacion\_desc, usuario.usuario\_doc

FROM dbo.evento INNER JOIN

dbo.inscripcion ON dbo.evento.evento\_id = dbo.inscripcion.evento\_id INNER JOIN

dbo.usuario ON dbo.inscripcion.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id

where evento.evento\_id=@evento\_id) T1

full outer join

(select inscripcion\_id, instructor.instructor\_id, usuario.usuario\_apellido+', '+usuario.usuario\_nombre as 'instructor'

from inscripcion inner join instructor on inscripcion.inscripcion\_instructor\_id = instructor.instructor\_id inner join usuario

on instructor.usuario\_id = usuario.usuario\_id where inscripcion.evento\_id = @evento\_id) T2 ON T1.inscripcion\_id = T2.inscripcion\_id

ORDER BY T2.instructor\_id asc

GO

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

ALTER Procedure [dbo].[Datos\_Personales\_Actualizar\_Datos]

@usuario\_id int,

@usuario\_nombre varchar(50),

@usuario\_apellido varchar(50),

@usuario\_fechanac date,

@usuario\_nacionalidad varchar(50),

@usuario\_sexo varchar(50),

@estadocivil\_id int,

@usuario\_profesion varchar(50),

@usuario\_domicilio varchar(50),

@usuario\_codigopostal int,

@provincia\_id int,

@ciudad\_id int,

@usuario\_telefono varchar(50),

@usuario\_mail varchar(50),

@usuario\_nrolibreta varchar(50),

@graduacion\_id int,

@usuario\_usuario varchar(50),

@usuario\_password varchar(50)

as

update Usuario set

usuario\_nombre = @usuario\_nombre,

usuario\_apellido = @usuario\_apellido,

usuario\_fechanac = @usuario\_fechanac,

usuario\_nacionalidad = @usuario\_nacionalidad,

usuario\_sexo = @usuario\_sexo,

estadocivil\_id = @estadocivil\_id,

usuario\_profesion = @usuario\_profesion,

usuario\_domicilio = @usuario\_domicilio,

usuario\_codigopostal = @usuario\_codigopostal,

provincia\_id = @provincia\_id,

ciudad\_id = @ciudad\_id,

usuario\_telefono = @usuario\_telefono,

usuario\_mail = @usuario\_mail,

usuario\_nrolibreta = @usuario\_nrolibreta,

graduacion\_id = @graduacion\_id,

usuario\_usuario = @usuario\_usuario,

usuario\_password=@usuario\_password

where usuario\_id=@Usuario\_id

go

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

ALTER Procedure [dbo].[Datos\_Personales\_Obtener\_Datos\_Usuarios]

@Us\_id as int

as

set dateformat dmy

SELECT dbo.usuario.usuario\_nombre, dbo.usuario.usuario\_apellido, Convert(varchar(10),dbo.usuario.usuario\_fechanac,103) as fechaNac, dbo.usuario.usuario\_nacionalidad, dbo.usuario.usuario\_sexo, dbo.estado\_civil.estadocivil\_id,

dbo.ciudad.ciudad\_desc, dbo.provincia.provincia\_desc, dbo.usuario.usuario\_domicilio, dbo.usuario.usuario\_profesion, dbo.usuario.usuario\_codigopostal, dbo.usuario.usuario\_telefono,

dbo.usuario.usuario\_mail, dbo.usuario.usuario\_tipo, dbo.usuario.usuario\_id, dbo.usuario.provincia\_id, dbo.usuario.ciudad\_id, usuario.usuario\_foto,

dbo.usuario.graduacion\_id , usuario.usuario\_doc,

dbo.usuario.usuario\_nrolibreta, usuario.usuario\_usuario,usuario.usuario\_password

FROM dbo.usuario INNER JOIN

dbo.estado\_civil ON dbo.usuario.estadocivil\_id = dbo.estado\_civil.estadocivil\_id INNER JOIN

dbo.provincia ON dbo.usuario.provincia\_id = dbo.provincia.provincia\_id INNER JOIN

dbo.ciudad ON dbo.usuario.ciudad\_id = dbo.ciudad.ciudad\_id

where

usuario\_id=@Us\_id

select usuario\_usuario from usuario where usuario\_id=@Us\_id

select instructor\_id, usuario\_id from instructor where usuario\_id = @Us\_id ---esto lo uso para validar el master -choco:18-07-2019

select instructor\_id from alumnos\_x\_instructor where usuario\_id = @Us\_id

go

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

ALTER procedure [dbo].[Datos\_Personales\_Validar\_libreta]

@usuario\_nrolibreta varchar(max)

as

select

usuario.usuario\_id

from usuario where usuario.usuario\_nrolibreta = @usuario\_nrolibreta

GO

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

ALTER Procedure [dbo].[Desactivar\_Usuario]

@Usuario\_id int

as

update Usuario set

usuario\_estado='Inactivo'

where usuario\_id=@Usuario\_id

GO

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

ALTER procedure [dbo].[Estado\_civil\_obtener]

as

select

estadocivil\_id, estadocivil\_desc

from estado\_civil

GO

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

ALTER Procedure [dbo].[Evento\_Actualizar]

@evento\_id int,

@evento\_descripcion varchar(50),

@evento\_foto image,

@evento\_fecha date,

@evento\_fechacierre datetime,

@evento\_tipoevento varchar(50),

@evento\_costo decimal,

@evento\_cap\_max\_insc int,

@evento\_direccion varchar(max)

as

update evento set

evento\_descripcion=@evento\_descripcion,

evento\_foto =@evento\_foto,

evento\_fecha =@evento\_fecha,

evento\_fechacierre =@evento\_fechacierre,

evento\_tipoevento =@evento\_tipoevento,

evento\_costo =@evento\_costo,

evento\_cap\_max\_insc = @evento\_cap\_max\_insc,

evento\_direccion = @evento\_direccion

where evento\_id=@evento\_id

GO

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

ALTER procedure [dbo].[Evento\_eliminar]

@evento\_id int

as

delete from evento where evento.evento\_id = @evento\_id

GO

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

ALTER procedure [dbo].[Evento\_inscripcion\_cargar]

--este procedimiento trae la informacion del usuario para que sea cargada en el form "Evento\_datos"

@usuario\_id int

as

--esta variable la voy a usar para obtener la graduacion actual, anterior y siguiente si es posible

declare @id\_graduacion int

select @id\_graduacion = graduacion\_id from usuario where usuario.usuario\_id=@usuario\_id

select usuario.usuario\_id, usuario\_doc as 'dni',

usuario\_apellido +', '+usuario\_nombre as 'apenom',

usuario.usuario\_sexo as 'sexo',

(cast(datediff(dd, usuario.usuario\_fechanac ,GETDATE()) / 365.25 as int)) as Edad,

graduacion.graduacion\_desc as 'graduacion'

FROM dbo.usuario INNER JOIN

dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id

where usuario.usuario\_estado='activo' and usuario.usuario\_id = @usuario\_id

declare @id\_instructor int

select @id\_instructor=instructor\_id from alumnos\_x\_instructor where alumnos\_x\_instructor.usuario\_id=@usuario\_id and alumno\_x\_instructor\_estado = 'activo'

--aqui recupero el id\_instructor necesario para buscar la institucion en la tabla institucion x instructor

SELECT provincia.provincia\_desc as 'provincia', institucion.institucion\_abreviacion as 'abreviacion',

institucion.institucion\_descripcion as 'institucion',

usuario.usuario\_apellido+', '+usuario.usuario\_nombre as 'instructor'

FROM dbo.instructor INNER JOIN

dbo.usuario ON dbo.instructor.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.alumnos\_x\_instructor ON dbo.instructor.instructor\_id = dbo.alumnos\_x\_instructor.instructor\_id INNER JOIN

dbo.institucion ON dbo.alumnos\_x\_instructor.institucion\_id = dbo.institucion.institucion\_id INNER JOIN

dbo.provincia ON dbo.institucion.provincia\_id = dbo.provincia.provincia\_id

where alumnos\_x\_instructor.alumno\_x\_instructor\_estado='activo' and

alumnos\_x\_instructor.instructor\_id=@id\_instructor and alumnos\_x\_instructor.usuario\_id=@usuario\_id

select top 1 graduacion\_id, graduacion\_desc from graduacion where graduacion.graduacion\_id < @id\_graduacion order by graduacion\_id desc

select graduacion\_id, graduacion\_desc from graduacion where graduacion.graduacion\_id=@id\_graduacion

select top 1 graduacion\_id, graduacion\_desc from graduacion where graduacion.graduacion\_id > @id\_graduacion order by graduacion\_id asc

GO

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

ALTER Procedure [dbo].[Evento\_ObetenerEvento\_ID]

@ID int

as

set dateformat dmy

select evento\_id,

evento\_descripcion,

evento\_foto,

evento\_fecha,

CONVERT(VARCHAR(10),evento.evento\_fechacierre,103) AS fechacierre,

RIGHT(evento.evento\_fechacierre, 7) AS horacierre,

evento.evento\_tipoevento,

evento.evento\_costo,

evento.evento\_direccion,

evento.evento\_cap\_max\_insc

from evento where evento.evento\_id=@ID

select \* from ExamenTurno where ExamenTurno.evento\_id = @ID

--tambien recupero los inscriptos, si hay inscriptos. no puedo modificar

select \* from inscripcion where inscripcion.evento\_id = @ID

GO

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

ALTER Procedure [dbo].[Evento\_ObetenerEventos]

as

select evento\_descripcion, evento\_id from evento

--aqui pongo otra consulta pero q uso en inscripciones, para ello voy a validar la fecha limite asi como la hora.

declare @hoy as datetime

set @hoy=getdate()

--select @hoy as fechahoy 'esto es para probar q si trae la fecha y hora

select evento.evento\_id as 'id', evento.evento\_descripcion as 'desc', evento.evento\_fechacierre as 'fech\_y\_horalimite' from evento

where evento.evento\_fechacierre > @hoy

order by evento.evento\_descripcion asc

select evento.evento\_id as 'id',

evento.evento\_descripcion as 'Descripción',

evento.evento\_tipoevento as 'Tipo',

CONVERT(VARCHAR(10), evento.evento\_fecha, 103) as 'Fecha' ,

evento.evento\_fechacierre as 'Cierre de inscripción'

from evento

where evento.evento\_fechacierre > @hoy

order by evento.evento\_descripcion asc

GO

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

ALTER Procedure [dbo].[Evento\_obtener\_torneos]

as

--solo trae los torneos para generar las llaves

--aqui pongo otra consulta pero q uso en inscripciones, para ello voy a validar la fecha limite asi como la hora.

declare @hoy as datetime

set @hoy=getdate()

select evento.evento\_id as 'id',

evento.evento\_descripcion as 'Descripción',

evento.evento\_tipoevento as 'Tipo',

CONVERT(VARCHAR(10), evento.evento\_fecha, 103) as 'Fecha' ,

evento.evento\_fechacierre as 'Cierre de inscripción'

from evento

where evento.evento\_tipoevento='Torneo'

order by evento.evento\_descripcion asc

--aqui esta la misma rutina pero validando q solo muestre aquellos torneos q aun se puedan inscribir

--select evento.evento\_id as 'id',

--evento.evento\_descripcion as 'Descripción',

--evento.evento\_tipoevento as 'Tipo',

-- CONVERT(VARCHAR(10), evento.evento\_fecha, 103) as 'Fecha' ,

--evento.evento\_fechacierre as 'Cierre de inscripción'

--from evento

--where (evento.evento\_fechacierre > @hoy) and evento.evento\_tipoevento='Torneo'

--order by evento.evento\_descripcion asc

GO

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

ALTER procedure [dbo].[Evento\_obtenerEventos\_inscripto]

--consulto que tipo de evento es y si el usuario esta inscripto

@usuario\_id int

as

select inscripcion.usuario\_id,

inscripcion.evento\_id as 'id',

evento.evento\_descripcion as 'desc'

from inscripcion inner join evento on inscripcion.evento\_id = evento.evento\_id

where inscripcion.usuario\_id=@usuario\_id

GO

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

ALTER Procedure [dbo].[Evento\_Seleccionar\_Curso]

as

Select \* from evento

GO

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

ALTER Procedure [dbo].[Evento\_Seleccionar\_Examen]

as

Select \* from evento where evento.evento\_tipoevento='Examen'

GO

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

ALTER Procedure [dbo].[Evento\_Seleccionar\_torneo]

as

Select evento\_id, evento\_descripcion, CONVERT(varchar, evento\_fecha, 103) as 'evento\_fecha' from evento where evento.evento\_tipoevento='Torneo'

order by evento\_fecha asc

GO

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

ALTER Procedure [dbo].[Eventos\_Alta]

@evento\_descripcion varchar(50),

@evento\_foto image,

@evento\_fecha date,

@evento\_fechacierre datetime,

@evento\_tipoevento varchar(50),

@evento\_costo decimal,

@evento\_direccion varchar(max),

@evento\_cap\_max\_insc int

as

insert dbo.evento(

evento\_descripcion,

evento\_foto,

evento\_fecha,

evento\_fechacierre,

evento\_tipoevento,

evento\_costo,

evento\_direccion,

evento\_cap\_max\_insc)

values

(@evento\_descripcion,

@evento\_foto,

@evento\_fecha,

@evento\_fechacierre,

@evento\_tipoevento,

@evento\_costo,

@evento\_direccion,

@evento\_cap\_max\_insc )

---recupero el ID del evento nuevo que acabo de insertar

select @@IDENTITY as 'evento\_id'

GO

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

ALTER procedure [dbo].[Eventos\_validar]

@evento\_id int

as

select \* from evento where evento.evento\_id = @evento\_id

GO

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

ALTER procedure [dbo].[Examen\_liquidacion\_obtener\_inscriptos]

@evento\_id int

as

SELECT ' ' as 'Nro',

dbo.usuario.usuario\_apellido +' '+dbo.usuario.usuario\_nombre as 'ApeyNom',

dbo.usuario.usuario\_doc as 'Dni',

dbo.usuario.usuario\_nrolibreta as 'Libreta',

' ' as 'Grad.Rendir',---esta la tengo q obtener en base al id de la graduacion actual del alumno.

(cast(datediff(dd, usuario.usuario\_fechanac ,GETDATE()) / 365.25 as int)) as Edad,

' ' as 'Graduacion',

inscripciones\_x\_examen.graduacion\_actual\_id as 'graduacion\_id', ---esta es la graduacion es la que va a obtener o tiene en primera instancia, ya que (rinde y obtenida) son iguales

' ' as 'Exam.Anterior', ---esto debo consultar cual es la anterior a la fecha de este examen.

dbo.alumnos\_x\_instructor.instructor\_id, ---este el id del instructor para recuperar nombre

' ' AS 'Instructor',

dbo.usuario.usuario\_id, ---este es el id del usuario inscripto

dbo.evento.evento\_id ,

dbo.evento.evento\_descripcion,

dbo.ExamenTurno.ExamenTurno\_desc as 'Turno',

dbo.inscripciones\_x\_examen.Inscexamen\_id --me sirve para mover de turno si es q el op del sistema lo desea.

FROM dbo.instructor INNER JOIN

dbo.alumnos\_x\_instructor ON dbo.instructor.instructor\_id = dbo.alumnos\_x\_instructor.instructor\_id INNER JOIN

dbo.usuario ON dbo.alumnos\_x\_instructor.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.institucion ON dbo.alumnos\_x\_instructor.institucion\_id = dbo.institucion.institucion\_id INNER JOIN

dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id inner join

dbo.inscripcion on dbo.usuario.usuario\_id = inscripcion.usuario\_id inner join

dbo.evento on dbo.inscripcion.evento\_id = dbo.evento.evento\_id inner join

dbo.inscripciones\_x\_examen on dbo.inscripcion.inscripcion\_id = dbo.inscripciones\_x\_examen.inscripcion\_id inner join

dbo.ExamenTurno on dbo.inscripciones\_x\_examen.ExamenTurno\_id = ExamenTurno.ExamenTurno\_id

where evento.evento\_id = @evento\_id and usuario.usuario\_estado='activo' order by alumnos\_x\_instructor.instructor\_id asc

---1) recupero las graduaciones

select \* from graduacion order by graduacion.graduacion\_id asc

GO

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

ALTER procedure [dbo].[Examen\_recuperar\_fecha]

@usuario\_id int

as

select

usuario.usuario\_id,

usuario.usuario\_doc,

evento.evento\_fecha,

evento.evento\_tipoevento,

evento.evento\_id

from usuario inner join inscripcion on usuario.usuario\_id = inscripcion.usuario\_id

inner join evento on inscripcion.evento\_id = evento.evento\_id

where evento.evento\_tipoevento = 'Examen' and inscripcion.usuario\_id = @usuario\_id order by evento.evento\_fecha desc

GO

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

ALTER procedure [dbo].[Examen\_recuperar\_inscriptos]

@evento\_id int

as

SELECT ' ' as 'Nro',

dbo.usuario.usuario\_apellido +' '+dbo.usuario.usuario\_nombre as 'ApeyNom',

dbo.usuario.usuario\_doc as 'Dni',

dbo.usuario.usuario\_nrolibreta as 'Libreta',

' ' as 'Grad.Rendir',---esta la tengo q obtener en base al id de la graduacion actual del alumno.

(cast(datediff(dd, usuario.usuario\_fechanac ,GETDATE()) / 365.25 as int)) as Edad,

' ' as 'Graduacion',

inscripciones\_x\_examen.graduacion\_actual\_id as 'graduacion\_id', ---esta es la graduacion es la que va a obtener o tiene en primera instancia, ya que (rinde y obtenida) son iguales

' ' as 'Exam.Anterior', ---esto debo consultar cual es la anterior a la fecha de este examen.

dbo.alumnos\_x\_instructor.instructor\_id, ---este el id del instructor para recuperar nombre

' ' AS 'Instructor',

dbo.usuario.usuario\_id, ---este es el id del usuario inscripto

dbo.evento.evento\_id ,

dbo.evento.evento\_descripcion,

dbo.ExamenTurno.ExamenTurno\_desc as 'Turno',

dbo.inscripciones\_x\_examen.Inscexamen\_id --me sirve para mover de turno si es q el op del sistema lo desea.

FROM dbo.instructor INNER JOIN

dbo.alumnos\_x\_instructor ON dbo.instructor.instructor\_id = dbo.alumnos\_x\_instructor.instructor\_id INNER JOIN

dbo.usuario ON dbo.alumnos\_x\_instructor.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.institucion ON dbo.alumnos\_x\_instructor.institucion\_id = dbo.institucion.institucion\_id INNER JOIN

dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id inner join

dbo.inscripcion on dbo.usuario.usuario\_id = inscripcion.usuario\_id inner join

dbo.evento on dbo.inscripcion.evento\_id = dbo.evento.evento\_id inner join

dbo.inscripciones\_x\_examen on dbo.inscripcion.inscripcion\_id = dbo.inscripciones\_x\_examen.inscripcion\_id inner join

dbo.ExamenTurno on dbo.inscripciones\_x\_examen.ExamenTurno\_id = ExamenTurno.ExamenTurno\_id

where evento.evento\_id = @evento\_id and usuario.usuario\_estado='activo' order by ExamenTurno.ExamenTurno\_desc, usuario.usuario\_apellido asc

---1) recupero las graduaciones

select \* from graduacion order by graduacion.graduacion\_id asc

--2) recupero turnos de un evento puntual

select \* from dbo.ExamenTurno where ExamenTurno.evento\_id = @evento\_id order by ExamenTurno.ExamenTurno\_desc asc

--3) recupero info de examen

select

evento\_id,

evento\_descripcion,

evento\_fecha,

evento\_fechacierre,

evento\_direccion,

evento\_cap\_max\_insc

from evento where evento.evento\_id = @evento\_id

--4) traer solo los inscriptos sin evaluar

SELECT ' ' as 'Nro',

dbo.usuario.usuario\_apellido +' '+dbo.usuario.usuario\_nombre as 'ApeyNom',

dbo.usuario.usuario\_doc as 'Dni',

dbo.usuario.usuario\_nrolibreta as 'Libreta',

' ' as 'Grad.Rendir',---esta la tengo q obtener en base al id de la graduacion actual del alumno.

(cast(datediff(dd, usuario.usuario\_fechanac ,GETDATE()) / 365.25 as int)) as Edad,

dbo.graduacion.graduacion\_desc as 'Graduacion',

inscripciones\_x\_examen.graduacion\_actual\_id as 'graduacion\_id', ---esta es la graduacion actual del alumno

' ' as 'Exam.Anterior', ---esto debo consultar cual es la anterior a la fecha de este examen.

dbo.alumnos\_x\_instructor.instructor\_id, ---este el id del instructor para recuperar nombre

' ' AS 'Instructor',

dbo.usuario.usuario\_id, ---este es el id del usuario inscripto

dbo.evento.evento\_id ,

dbo.evento.evento\_descripcion,

dbo.ExamenTurno.ExamenTurno\_desc as 'Turno',

dbo.inscripciones\_x\_examen.Inscexamen\_id --me sirve para mover de turno si es q el op del sistema lo desea.

FROM dbo.instructor INNER JOIN

dbo.alumnos\_x\_instructor ON dbo.instructor.instructor\_id = dbo.alumnos\_x\_instructor.instructor\_id INNER JOIN

dbo.usuario ON dbo.alumnos\_x\_instructor.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.institucion ON dbo.alumnos\_x\_instructor.institucion\_id = dbo.institucion.institucion\_id INNER JOIN

dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id inner join

dbo.inscripcion on dbo.usuario.usuario\_id = inscripcion.usuario\_id inner join

dbo.evento on dbo.inscripcion.evento\_id = dbo.evento.evento\_id inner join

dbo.inscripciones\_x\_examen on dbo.inscripcion.inscripcion\_id = dbo.inscripciones\_x\_examen.inscripcion\_id inner join

dbo.ExamenTurno on dbo.inscripciones\_x\_examen.ExamenTurno\_id = ExamenTurno.ExamenTurno\_id

where evento.evento\_id = @evento\_id and usuario.usuario\_estado='activo' and inscripciones\_x\_examen.inscexamen\_resultado = 'Sin Evaluar' order by ExamenTurno.ExamenTurno\_desc, usuario.usuario\_apellido asc

--5) traer solo los inscriptos sin evaluar

SELECT ' ' as 'Nro',

dbo.usuario.usuario\_apellido +' '+dbo.usuario.usuario\_nombre as 'ApeyNom',

dbo.usuario.usuario\_doc as 'Dni',

dbo.usuario.usuario\_nrolibreta as 'Libreta',

' ' as 'Grad.Rendir',---esta la tengo q obtener en base al id de la graduacion actual del alumno.

(cast(datediff(dd, usuario.usuario\_fechanac ,GETDATE()) / 365.25 as int)) as Edad,

dbo.graduacion.graduacion\_desc as 'Graduacion',

inscripciones\_x\_examen.graduacion\_obtenida\_id as 'graduacion\_id', ---esta es la graduacion actual del alumno

' ' as 'Exam.Anterior', ---esto debo consultar cual es la anterior a la fecha de este examen.

dbo.alumnos\_x\_instructor.instructor\_id, ---este el id del instructor para recuperar nombre

' ' AS 'Instructor',

dbo.usuario.usuario\_id, ---este es el id del usuario inscripto

dbo.evento.evento\_id ,

dbo.evento.evento\_descripcion,

dbo.ExamenTurno.ExamenTurno\_desc as 'Turno',

dbo.inscripciones\_x\_examen.Inscexamen\_id --me sirve para mover de turno si es q el op del sistema lo desea.

FROM dbo.instructor INNER JOIN

dbo.alumnos\_x\_instructor ON dbo.instructor.instructor\_id = dbo.alumnos\_x\_instructor.instructor\_id INNER JOIN

dbo.usuario ON dbo.alumnos\_x\_instructor.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.institucion ON dbo.alumnos\_x\_instructor.institucion\_id = dbo.institucion.institucion\_id INNER JOIN

--dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id inner join

dbo.inscripcion on dbo.usuario.usuario\_id = inscripcion.usuario\_id inner join

dbo.evento on dbo.inscripcion.evento\_id = dbo.evento.evento\_id inner join

dbo.inscripciones\_x\_examen on dbo.inscripcion.inscripcion\_id = dbo.inscripciones\_x\_examen.inscripcion\_id inner join

dbo.ExamenTurno on dbo.inscripciones\_x\_examen.ExamenTurno\_id = ExamenTurno.ExamenTurno\_id inner join

dbo.graduacion ON dbo.inscripciones\_x\_examen.graduacion\_obtenida\_id = dbo.graduacion.graduacion\_id

where evento.evento\_id = @evento\_id and usuario.usuario\_estado='activo' and inscripciones\_x\_examen.inscexamen\_resultado = 'Desaprobado' order by ExamenTurno.ExamenTurno\_desc, usuario.usuario\_apellido asc

--6) traer solo los inscriptos sin evaluar

SELECT ' ' as 'Nro',

dbo.usuario.usuario\_apellido +' '+dbo.usuario.usuario\_nombre as 'ApeyNom',

dbo.usuario.usuario\_doc as 'Dni',

dbo.usuario.usuario\_nrolibreta as 'Libreta',

' ' as 'Grad.Rendir',---esta la tengo q obtener en base al id de la graduacion actual del alumno.

(cast(datediff(dd, usuario.usuario\_fechanac ,GETDATE()) / 365.25 as int)) as Edad,

dbo.graduacion.graduacion\_desc as 'Graduacion',

--dbo.usuario.graduacion\_id, ---esta es la graduacion actual del alumno

inscripciones\_x\_examen.graduacion\_obtenida\_id as 'graduacion\_id',

' ' as 'Exam.Anterior', ---esto debo consultar cual es la anterior a la fecha de este examen.

dbo.alumnos\_x\_instructor.instructor\_id, ---este el id del instructor para recuperar nombre

' ' AS 'Instructor',

dbo.usuario.usuario\_id, ---este es el id del usuario inscripto

dbo.evento.evento\_id ,

dbo.evento.evento\_descripcion,

dbo.ExamenTurno.ExamenTurno\_desc as 'Turno',

dbo.inscripciones\_x\_examen.Inscexamen\_id --me sirve para mover de turno si es q el op del sistema lo desea.

FROM dbo.instructor INNER JOIN

dbo.alumnos\_x\_instructor ON dbo.instructor.instructor\_id = dbo.alumnos\_x\_instructor.instructor\_id INNER JOIN

dbo.usuario ON dbo.alumnos\_x\_instructor.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.institucion ON dbo.alumnos\_x\_instructor.institucion\_id = dbo.institucion.institucion\_id INNER JOIN

--dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id inner join

dbo.inscripcion on dbo.usuario.usuario\_id = inscripcion.usuario\_id inner join

dbo.evento on dbo.inscripcion.evento\_id = dbo.evento.evento\_id inner join

dbo.inscripciones\_x\_examen on dbo.inscripcion.inscripcion\_id = dbo.inscripciones\_x\_examen.inscripcion\_id inner join

dbo.ExamenTurno on dbo.inscripciones\_x\_examen.ExamenTurno\_id = ExamenTurno.ExamenTurno\_id inner join

dbo.graduacion ON dbo.inscripciones\_x\_examen.graduacion\_obtenida\_id = dbo.graduacion.graduacion\_id

where evento.evento\_id = @evento\_id and usuario.usuario\_estado='activo' and inscripciones\_x\_examen.inscexamen\_resultado = 'Aprobado' order by ExamenTurno.ExamenTurno\_desc, usuario.usuario\_apellido asc

--7) traer solo los inscriptos sin evaluar

SELECT ' ' as 'Nro',

dbo.usuario.usuario\_apellido +' '+dbo.usuario.usuario\_nombre as 'ApeyNom',

dbo.usuario.usuario\_doc as 'Dni',

dbo.usuario.usuario\_nrolibreta as 'Libreta',

' ' as 'Grad.Rendir',---esta la tengo q obtener en base al id de la graduacion actual del alumno.

(cast(datediff(dd, usuario.usuario\_fechanac ,GETDATE()) / 365.25 as int)) as Edad,

dbo.graduacion.graduacion\_desc as 'Graduacion',

--dbo.usuario.graduacion\_id, ---esta es la graduacion actual del alumno

inscripciones\_x\_examen.graduacion\_obtenida\_id as 'graduacion\_id',

' ' as 'Exam.Anterior', ---esto debo consultar cual es la anterior a la fecha de este examen.

dbo.alumnos\_x\_instructor.instructor\_id, ---este el id del instructor para recuperar nombre

' ' AS 'Instructor',

dbo.usuario.usuario\_id, ---este es el id del usuario inscripto

dbo.evento.evento\_id ,

dbo.evento.evento\_descripcion,

dbo.ExamenTurno.ExamenTurno\_desc as 'Turno',

dbo.inscripciones\_x\_examen.Inscexamen\_id --me sirve para mover de turno si es q el op del sistema lo desea.

FROM dbo.instructor INNER JOIN

dbo.alumnos\_x\_instructor ON dbo.instructor.instructor\_id = dbo.alumnos\_x\_instructor.instructor\_id INNER JOIN

dbo.usuario ON dbo.alumnos\_x\_instructor.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.institucion ON dbo.alumnos\_x\_instructor.institucion\_id = dbo.institucion.institucion\_id INNER JOIN

--dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id inner join

dbo.inscripcion on dbo.usuario.usuario\_id = inscripcion.usuario\_id inner join

dbo.evento on dbo.inscripcion.evento\_id = dbo.evento.evento\_id inner join

dbo.inscripciones\_x\_examen on dbo.inscripcion.inscripcion\_id = dbo.inscripciones\_x\_examen.inscripcion\_id inner join

dbo.ExamenTurno on dbo.inscripciones\_x\_examen.ExamenTurno\_id = ExamenTurno.ExamenTurno\_id inner join

dbo.graduacion ON dbo.inscripciones\_x\_examen.graduacion\_obtenida\_id = dbo.graduacion.graduacion\_id

where evento.evento\_id = @evento\_id and usuario.usuario\_estado='activo' and inscripciones\_x\_examen.inscexamen\_resultado = 'Doble Promoción' order by ExamenTurno.ExamenTurno\_desc, usuario.usuario\_apellido asc

GO

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

ALTER PROCEDURE [dbo].[ExamenCertificacion\_alta]

@usuario\_id int,

@graduacion\_id int,

@evento\_id int

as

insert ExamenCertificacion (usuario\_id , graduacion\_id, evento\_id)values(@usuario\_id, @graduacion\_id,@evento\_id )

GO

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

ALTER procedure [dbo].[ExamenCostos\_recuperar]

@evento\_id int

as

select \* from ExamenCostos where ExamenCostos.evento\_id = @evento\_id order by ExamenCostos.ExamenCostos\_id asc

GO

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

ALTER procedure [dbo].[ExamenTurno\_alta]

@evento\_id int,

@ExamenTurno\_desc varchar(max)

as

insert ExamenTurno ( evento\_id, ExamenTurno\_desc ) values (@evento\_id, @ExamenTurno\_desc)

GO

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

ALTER procedure [dbo].[ExamenTurno\_eliminar]

@evento\_id int

as

delete from ExamenTurno where ExamenTurno.evento\_id = @evento\_id

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

ALTER PROCEDURE [dbo].[Generar\_ciudad\_pruebas\_lote]

AS

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\buenos aires\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\catamarca\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\cba\_localidades.txt.'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\Chaco\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\chubut\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\corrientes\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\entre rios\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\formosa\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\jujuy\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\la pampa\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\la rioja\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\mendoza\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\misiones\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\neuquen\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\rio negro\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\salta\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\san juan\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\san luis\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\santa cruz\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\santa fe\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\sgo del estero\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

BULK

INSERT dbo.ciudad

FROM 'C:\downloads\localidades\tierra del fuego\_localidades.txt'--Ruta archivo

WITH

( FIELDTERMINATOR = ',', --separa campos

ROWTERMINATOR = '.') --separa filas

GO

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

ALTER procedure [dbo].[Graduacion\_obtener\_todo]

as

select \* from graduacion order by graduacion.graduacion\_id asc

GO

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

ALTER procedure [dbo].[HAMERSOFT\_consulta\_inscriptos]

as

select \* from usuario where usuario.usuario\_apellido='coria'

select \* from usuario where usuario.usuario\_apellido='carabajal'

select\* from inscripcion

select inscripciones\_x\_torneo.torneo\_id, inscripciones\_x\_torneo.inscripcion\_id, inscripciones\_x\_torneo.categoria\_id , categoria\_tipo, categoria\_edadinicial, categoria\_edadfinal,

categoria\_gradinicial, categoria\_gradfinal, categoria\_sexo, categoria\_peso\_inical, categoria\_peso\_Final from inscripciones\_x\_torneo inner join categoria on inscripciones\_x\_torneo.categoria\_id=categoria.categoria\_id

GO

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

ALTER procedure [dbo].[imagen\_obtener]

as

select \* from Imagenes

GO

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

ALTER procedure [dbo].[imagen\_subir]

@imagen image

as

insert Imagenes (imagen ) values (@imagen)

GO

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

ALTER procedure [dbo].[Inscripcion\_alta\_categorias]

@inscripcion\_id int,

@categoria\_id int

as

insert inscripciones\_x\_torneo (inscripcion\_id, categoria\_id, en\_llave) values (@inscripcion\_id, @categoria\_id, 'no' )

GO

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

ALTER procedure [dbo].[Inscripcion\_alta\_masiva]

--CHOCOLONEA PABLO:

--esta rutina es para inscripciones masivas de alumnos realizadas por el usuario instructor.

--por ello trae 2 id q se aclaran a continuacion.

@USU\_id\_alumno int, --este es el del alumno

@evento\_id int,

@inscripcion\_fechahora datetime,

@inscripcion\_peso decimal(5,2),

@instructor\_id int

as

--alta inscripcion

insert inscripcion (usuario\_id, evento\_id, inscripcion\_fechahora, inscripcion\_peso, inscripcion\_instructor\_id ) values(@USU\_id\_alumno, @evento\_id, @inscripcion\_fechahora, @inscripcion\_peso , @instructor\_id)

---recupero el tipoevento, me sirve para validar si es necesario hacer un alta en "inscripciones\_x\_torneo"

SELECT @@IDENTITY AS 'inscripcion\_id'

select evento.evento\_tipoevento as 'tipo\_evento' from evento where evento.evento\_id=@evento\_id

GO

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

ALTER procedure [dbo].[Inscripcion\_alta\_usuario]

--CHOCOLONEA PABLO:

--esta rutina es para inscripciones hechas por el alumno.

--por ello solo trae la id del usuario q se obtiene de la variable de session del login.

@USU\_id int,

@evento\_id int,

@inscripcion\_fechahora datetime,

@inscripcion\_peso decimal(5,2)

as

--debo recuperar el id del instructor del cual dependo, para ello hago lo siguiente

declare @instructor\_id int

--se valida que el usuario este en la tabla alumnos x instructor y q ademas este en estado activo

--estado activo: puede estar activo para un instructor e inactivo para otro.

select @instructor\_id = alumnos\_x\_instructor.instructor\_id from alumnos\_x\_instructor where alumnos\_x\_instructor.usuario\_id= @USU\_id and alumnos\_x\_instructor.alumno\_x\_instructor\_estado='activo'

--alta inscripcion

insert inscripcion (usuario\_id, evento\_id, inscripcion\_fechahora, inscripcion\_peso, inscripcion\_instructor\_id ) values(@USU\_id, @evento\_id, @inscripcion\_fechahora, @inscripcion\_peso , @instructor\_id)

---recupero el tipoevento, me sirve para validar si es necesario hacer un alta en "inscripciones\_x\_torneo"

SELECT @@IDENTITY AS 'inscripcion\_id'

select evento.evento\_tipoevento as 'tipo\_evento' from evento where evento.evento\_id=@evento\_id

GO

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

ALTER procedure [dbo].[Inscripcion\_borrar\_alumno]

--Chocolonea Pablo:

--esta rutina borra la inscripcion realizada por el alumno

@usuario\_id int,

@evento\_id int

as

declare @inscripcion\_id int

select @inscripcion\_id = inscripcion\_id from inscripcion where inscripcion.usuario\_id=@usuario\_id and inscripcion.evento\_id=@evento\_id

delete from inscripcion where inscripcion.usuario\_id=@usuario\_id and inscripcion.evento\_id=@evento\_id

delete from inscripciones\_x\_torneo where inscripcion\_id = @inscripcion\_id

delete from inscripciones\_x\_examen where inscripcion\_id = @inscripcion\_id

GO

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

ALTER procedure [dbo].[Inscripcion\_consultar\_alumno\_inscripto]

--consulto que tipo de evento es y si el usuario esta inscripto

@evento\_id int,

@usuario\_id int

as

select inscripcion.usuario\_id, inscripcion.evento\_id from inscripcion where inscripcion.evento\_id = @evento\_id and inscripcion.usuario\_id=@usuario\_id

GO

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

ALTER procedure [dbo].[Inscripcion\_consultar\_evento]

--consulto que tipo de evento es

@evento\_id int

as

select evento.evento\_tipoevento as 'tipo\_evento' from evento where evento.evento\_id=@evento\_id

select evento\_id,

evento\_descripcion,

evento\_foto,

evento\_fecha,

evento\_fechacierre,

evento\_tipoevento,

evento\_costo,

evento\_estado,

evento\_direccion,

evento\_cap\_max\_insc

from evento where evento.evento\_id = @evento\_id

select \* from ExamenTurno where ExamenTurno.evento\_id = @evento\_id

GO

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

ALTER procedure [dbo].[Inscripcion\_eliminar\_alumno\_torneo]

@inscripcion\_id int,

@categoria\_id int

as

delete from inscripciones\_x\_torneo where inscripcion\_id = @inscripcion\_id and categoria\_id = @categoria\_id

---y ademas voy a listar

select \* from inscripciones\_x\_torneo where inscripcion\_id = @inscripcion\_id

GO

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

ALTER procedure [dbo].[Inscripcion\_eliminar\_masivo]

@inscripcion\_id int

as

delete from inscripciones\_x\_torneo where inscripcion\_id = @inscripcion\_id

delete from inscripciones\_x\_examen where inscripcion\_id = @inscripcion\_id

delete from inscripcion where inscripcion.inscripcion\_id = @inscripcion\_id

delete from inscripcion\_qr where inscripcion\_id = @inscripcion\_id

GO

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

ALTER procedure [dbo].[inscripcion\_imagenQR\_Alta]

@inscripcion\_id int,

@imagenQR image

as

insert inscripcion\_qr (inscripcion\_id,inscripcion\_qr\_imagen) values (@inscripcion\_id,@imagenQR)

GO

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

ALTER procedure [dbo].[Inscripcion\_modificar]

@inscripcion\_id int,

@inscripcion\_peso decimal(5,2),

@categoria\_id\_vieja int,

@categoria\_id\_nueva int

as

--update inscripcion set inscripcion\_peso = @inscripcion\_peso where inscripcion.inscripcion\_id = @inscripcion\_id

--choco: 13-05-2022 no quiero alterar el peso, pero si moverlo a la categoria q indique.

update inscripciones\_x\_torneo set categoria\_id = @categoria\_id\_nueva where inscripcion\_id = @inscripcion\_id and categoria\_id = @categoria\_id\_vieja

GO

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

ALTER procedure [dbo].[Inscripcion\_obtener\_categoria]

as

select

categoria\_id, categoria\_sexo, categoria\_gradinicial, categoria\_gradfinal, categoria\_edadinicial, categoria\_edadfinal, categoria\_peso\_inical, categoria\_peso\_Final, categoria\_tipo

from categoria where categoria.categoria\_tipo='Lucha'

select

categoria\_id, categoria\_sexo, categoria\_gradinicial, categoria\_gradfinal, categoria\_edadinicial, categoria\_edadfinal, categoria\_peso\_inical, categoria\_peso\_Final, categoria\_tipo

from categoria where categoria.categoria\_tipo='Forma'

GO

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

ALTER procedure [dbo].[Inscripcion\_obtener\_categoria\_roturaespecial]

as

select

categoria\_id, categoria\_sexo, categoria\_gradinicial, categoria\_gradfinal, categoria\_edadinicial, categoria\_edadfinal, categoria\_peso\_inical, categoria\_peso\_Final, categoria\_tipo

from categoria where categoria.categoria\_tipo='Rotura Especial'

GO

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

ALTER procedure [dbo].[Inscripcion\_obtener\_categoria\_roturapoder]

as

select

categoria\_id, categoria\_sexo, categoria\_gradinicial, categoria\_gradfinal, categoria\_edadinicial, categoria\_edadfinal, categoria\_peso\_inical, categoria\_peso\_Final, categoria\_tipo

from categoria where categoria.categoria\_tipo='Rotura de Poder'

GO

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

ALTER procedure [dbo].[inscripcion\_recuperar\_ID]

@usuario\_id int

as

--recupero id de inscripcion de un usuario determinado.

select inscripcion.inscripcion\_id from inscripcion inner join usuario on inscripcion.usuario\_id = usuario.usuario\_id

where inscripcion.usuario\_id = @usuario\_id

GO

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

ALTER procedure [dbo].[Inscripcion\_validar\_alta]

@inscripcion\_id int

as

select \* from inscripcion where inscripcion.inscripcion\_id = @inscripcion\_id

GO

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

ALTER procedure [dbo].[Inscripciones\_categoria\_consultar]

---consulto la cant de inscriptos en una categoria determinada, de un evento puntual

@evento\_id int,

@categoria\_id int

as

select \* from inscripciones\_x\_torneo inner join inscripcion on inscripciones\_x\_torneo.inscripcion\_id = inscripcion.inscripcion\_id

where evento\_id = @evento\_id and categoria\_id = @categoria\_id

GO

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

ALTER procedure [dbo].[Inscripciones\_credenciales\_obtener]

@evento\_id int

as

select

evento.evento\_id,

evento.evento\_descripcion,

evento.evento\_foto,

evento.evento\_fecha,

usuario.usuario\_id ,

usuario.usuario\_doc,

usuario.usuario\_apellido,

usuario.usuario\_nombre,

graduacion.graduacion\_desc as 'graduacion',

usuario.usuario\_telefono,

'INSTRUCTOR' as 'instructor',

usuario.usuario\_foto,

instructor.instructor\_id,

inscripcion.inscripcion\_id

from evento inner join inscripcion on evento.evento\_id= inscripcion.evento\_id

inner join usuario on inscripcion.usuario\_id= usuario.usuario\_id

inner join instructor on inscripcion.inscripcion\_instructor\_id= instructor.instructor\_id

inner join graduacion on usuario.graduacion\_id=graduacion.graduacion\_id where evento.evento\_id=@evento\_id

select inscripcion.usuario\_id as 'ID',

usuario.usuario\_doc as 'dni',

usuario.usuario\_apellido+' '+usuario\_nombre as 'ApellidoyNombre',

institucion.institucion\_abreviacion as 'Institucion\_abreviatura',

institucion.institucion\_descripcion as 'Institucion',

provincia.provincia\_desc as 'Provincia'

---esta la uso en el modulo de generacion de credenciales

from inscripcion inner join evento on inscripcion.evento\_id=evento.evento\_id

inner join usuario on inscripcion.usuario\_id= usuario.usuario\_id

inner join alumnos\_x\_instructor on usuario.usuario\_id = alumnos\_x\_instructor.usuario\_id

inner join institucion on alumnos\_x\_instructor.institucion\_id= institucion.institucion\_id

inner join provincia on institucion.provincia\_id = provincia.provincia\_id

where inscripcion.evento\_id=@evento\_id

--este era el from anterior con un inner join que me repetia las inscripciones, si un usuario se anotaba en lucha y forma

--from inscripcion inner join inscripciones\_x\_torneo on inscripcion.inscripcion\_id = inscripciones\_x\_torneo.inscripcion\_id

--inner join categoria on inscripciones\_x\_torneo.categoria\_id=categoria.categoria\_id

--inner join evento on inscripcion.evento\_id=evento.evento\_id

--inner join usuario on inscripcion.usuario\_id= usuario.usuario\_id

--inner join alumnos\_x\_instructor on usuario.usuario\_id = alumnos\_x\_instructor.usuario\_id

--inner join institucion on alumnos\_x\_instructor.institucion\_id= institucion.institucion\_id

--inner join provincia on institucion.provincia\_id = provincia.provincia\_id

--where inscripcion.evento\_id=@evento\_id

GO

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

ALTER procedure [dbo].[inscripciones\_x\_examen\_alta]

@inscripcion\_id int,

@ExamenTurno\_id int,

@graduacion\_actual\_id int

as

insert inscripciones\_x\_examen (inscripcion\_id, ExamenTurno\_id, inscexamen\_resultado, graduacion\_actual\_id , graduacion\_obtenida\_id )

values

(@inscripcion\_id, @ExamenTurno\_id, 'Sin Evaluar', @graduacion\_actual\_id , @graduacion\_actual\_id )

GO

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

ALTER procedure [dbo].[inscripciones\_x\_examen\_eliminar]

@Inscexamen\_id int

as

--se elimina en "inscripciones x examen" y en "inscripciones"

declare @inscripcion\_id int

select @inscripcion\_id = inscripcion\_id from inscripciones\_x\_examen where inscripciones\_x\_examen.Inscexamen\_id = @Inscexamen\_id

delete from inscripcion where inscripcion.inscripcion\_id = @inscripcion\_id

delete from inscripciones\_x\_examen where inscripciones\_x\_examen.Inscexamen\_id = @Inscexamen\_id

GO

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

ALTER procedure [dbo].[inscripciones\_x\_examen\_modificar]

@Inscexamen\_id int,

@ExamenTurno\_id int

as

update inscripciones\_x\_examen set ExamenTurno\_id = @ExamenTurno\_id where Inscexamen\_id= @Inscexamen\_id

GO

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

ALTER procedure [dbo].[inscripciones\_x\_examen\_modificar\_resultado]

@Inscexamen\_id int,

@ExamenTurno\_id int,

@resultado varchar(50),

@graduacion\_obtenida\_id int

as

update inscripciones\_x\_examen set inscexamen\_resultado = @resultado, graduacion\_obtenida\_id = @graduacion\_obtenida\_id where Inscexamen\_id= @Inscexamen\_id

--faltaria cambiarlo de graduacion al inscripto.

GO

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

ALTER procedure [dbo].[inscripciones\_x\_examen\_validar]

@evento\_id int,

@ExamenTurno\_id int

as

--recupero para consultar cant max x turno

select \* from evento where evento.evento\_id = @evento\_id

--recupero para obtener la cantidad de inscriptos a q un turno en particular

select \* from inscripciones\_x\_examen where ExamenTurno\_id = @ExamenTurno\_id

GO

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

ALTER procedure [dbo].[institucion\_actualizar]

@institucion\_id int,

@provincia\_id int,

@institucion\_descripcion varchar(50),

@institucion\_abreviacion varchar(50),

@institucion\_logo image

as

update institucion set provincia\_id=@provincia\_id ,

institucion\_descripcion=@institucion\_descripcion,

institucion\_abreviacion=@institucion\_abreviacion,

institucion\_logo=@institucion\_logo

where institucion.institucion\_id=@institucion\_id

GO

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

ALTER procedure [dbo].[institucion\_actualizar\_sin\_foto]

@institucion\_id int,

@provincia\_id int,

@institucion\_descripcion varchar(50),

@institucion\_abreviacion varchar(50)

as

update institucion set provincia\_id=@provincia\_id ,

institucion\_descripcion=@institucion\_descripcion,

institucion\_abreviacion=@institucion\_abreviacion

where institucion.institucion\_id=@institucion\_id

GO

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

ALTER procedure [dbo].[Institucion\_alta]

@provincia\_id int,

@institucion\_descripcion as varchar(50),

@institucion\_abreviacion varchar(50),

@institucion\_logo image,

@instructor\_id int

as

insert dbo.institucion (provincia\_id, institucion\_descripcion, institucion\_abreviacion, institucion\_logo)

values

(@provincia\_id, @institucion\_descripcion, @institucion\_abreviacion, @institucion\_logo )

--aqui valido, si @instructor\_id = 0, significa que solo estoy agregando una institucion, pero no vinculo a un instructor.

if @instructor\_id <> 0

begin

insert dbo.institucion\_x\_instructor (instructor\_id, institucion\_id )values (@instructor\_id, @@IDENTITY)

end

GO

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

ALTER procedure [dbo].[Institucion\_asignar]

@instructor\_id int,

@institucion\_id int

as

insert dbo.institucion\_x\_instructor (instructor\_id, institucion\_id )values (@instructor\_id, @institucion\_id)

GO

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

ALTER procedure [dbo].[institucion\_buscar]

@institucion\_id int

as

select institucion\_id, provincia\_id,

institucion\_descripcion, institucion\_abreviacion,

institucion\_estado, institucion\_logo from institucion where institucion.institucion\_id=@institucion\_id

GO

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

ALTER procedure [dbo].[Institucion\_desvincular]

@instructor\_id int,

@institucion\_id int

as

delete from institucion\_x\_instructor where instructor\_id=@instructor\_id and institucion\_id = @institucion\_id

GO

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

ALTER procedure [dbo].[institucion\_instructores\_obtenertodos]

as

SELECT dbo.instructor.instructor\_id,

dbo.usuario.usuario\_doc,

dbo.usuario.usuario\_apellido + ', '+dbo.usuario.usuario\_nombre as 'Apellido\_y\_Nombre',

dbo.provincia.provincia\_desc

FROM dbo.provincia INNER JOIN

dbo.usuario ON dbo.provincia.provincia\_id = dbo.usuario.provincia\_id INNER JOIN

dbo.instructor ON dbo.usuario.usuario\_id = dbo.instructor.usuario\_id

GO

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

ALTER procedure [dbo].[institucion\_obtener\_de\_instructor]

@instructor\_id int

as

SELECT dbo.institucion.institucion\_id, dbo.institucion.institucion\_descripcion, dbo.institucion.institucion\_abreviacion, dbo.provincia.provincia\_desc

FROM dbo.institucion\_x\_instructor INNER JOIN

dbo.institucion ON dbo.institucion\_x\_instructor.institucion\_id = dbo.institucion.institucion\_id INNER JOIN

dbo.provincia ON dbo.institucion.provincia\_id = dbo.provincia.provincia\_id

where institucion\_x\_instructor.instructor\_id=@instructor\_id

select

usuario.usuario\_doc , usuario.usuario\_apellido+', '+usuario.usuario\_nombre

from instructor inner join usuario on instructor.usuario\_id=usuario.usuario\_id where instructor.instructor\_id=@instructor\_id

GO

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

ALTER procedure [dbo].[Institucion\_obtener\_instructor\_alumnos]

@institucion\_id int,

@instructor\_id int

as

select alumno\_x\_instructor\_id, usuario\_id, instructor\_id from alumnos\_x\_instructor where institucion\_id = @institucion\_id and instructor\_id = @instructor\_id

GO

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

SELECT dbo.institucion.institucion\_id,

dbo.institucion.provincia\_id,

dbo.provincia.provincia\_desc,

dbo.institucion.institucion\_descripcion,

dbo.institucion.institucion\_abreviacion

FROM dbo.institucion INNER JOIN

dbo.provincia ON dbo.institucion.provincia\_id = dbo.provincia.provincia\_id order by provincia.provincia\_desc asc

GO

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

ALTER procedure [dbo].[Instructor\_borrar\_todos\_alumnos]

@instructor\_id int

as

--hace un select primero de los alumnos del instructor

SELECT \* FROM alumnos\_x\_instructor where alumnos\_x\_instructor.instructor\_id='3'

delete from alumnos\_x\_instructor where instructor\_id= '3'

---despues viendo el select pone en los delete los id\_usuarios

delete from usuario where usuario.usuario\_id='13'

delete from usuario where usuario.usuario\_id='14'

delete from usuario where usuario.usuario\_id='15'

delete from usuario where usuario.usuario\_id='16'

delete from usuario where usuario.usuario\_id='17'

delete from usuario where usuario.usuario\_id='41'

-- pone en los delete de los instructores el id\_usuario

select \* from instructor order by instructor.usuario\_id asc

delete from instructor where instructor.usuario\_id= '14'

delete from instructor where instructor.usuario\_id= '15'

delete from instructor where instructor.usuario\_id= '15'

delete from instructor where instructor.usuario\_id= '16'

delete from instructor where instructor.usuario\_id= '17'

delete from instructor where instructor.usuario\_id= '41'

--ahora hago un select de institucion x instructor

select \* from institucion\_x\_instructor

delete from institucion\_x\_instructor where instructor\_id='4'

delete from institucion\_x\_instructor where instructor\_id='24'

delete from institucion\_x\_instructor where instructor\_id='28'

delete from institucion\_x\_instructor where instructor\_id='29'

delete from institucion\_x\_instructor where instructor\_id='27'

delete from institucion\_x\_instructor where instructor\_id='30'

delete from institucion\_x\_instructor where instructor\_id='26'

delete from institucion\_x\_instructor where instructor\_id='25'

GO

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

ALTER procedure [dbo].[Instructor\_buscar\_alumno]

@usuario\_id int,

@alumno\_dni int

as

declare @id\_instructor int

select @id\_instructor= instructor\_id from instructor where instructor.usuario\_id = @usuario\_id

SELECT dbo.usuario.usuario\_id as 'ID',

dbo.usuario.usuario\_doc as 'Documento',

dbo.usuario.usuario\_apellido +', '+dbo.usuario.usuario\_nombre as 'Apellido y Nombre', (cast(datediff(dd, usuario.usuario\_fechanac ,GETDATE()) / 365.25 as int)) as Edad,

dbo.usuario.usuario\_telefono as 'Teléfono',

dbo.graduacion.graduacion\_desc as 'Graduación',

dbo.usuario.usuario\_tipo as 'Tipo',

dbo.usuario.usuario\_sexo as 'Sexo',

dbo.usuario.usuario\_foto as 'foto'

FROM dbo.instructor INNER JOIN

dbo.alumnos\_x\_instructor ON dbo.instructor.instructor\_id = dbo.alumnos\_x\_instructor.instructor\_id INNER JOIN

dbo.usuario ON dbo.alumnos\_x\_instructor.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.institucion ON dbo.alumnos\_x\_instructor.institucion\_id = dbo.institucion.institucion\_id INNER JOIN

dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id

where alumnos\_x\_instructor.instructor\_id= @id\_instructor and usuario.usuario\_estado='activo' and usuario.usuario\_doc = @alumno\_dni order by usuario.usuario\_doc asc

select usuario.usuario\_apellido+', '+usuario\_nombre as 'ApellidoyNombre' from usuario where usuario.usuario\_id=@usuario\_id

SELECT dbo.usuario.usuario\_id as 'ID',

dbo.usuario.usuario\_doc as 'Documento',

dbo.usuario.usuario\_apellido +', '+dbo.usuario.usuario\_nombre as 'Apellido y Nombre', (cast(datediff(dd, usuario.usuario\_fechanac ,GETDATE()) / 365.25 as int)) as Edad,

dbo.usuario.usuario\_telefono as 'Teléfono',

dbo.graduacion.graduacion\_desc as 'Graduación',

dbo.usuario.usuario\_tipo as 'Tipo'

FROM dbo.instructor INNER JOIN

dbo.alumnos\_x\_instructor ON dbo.instructor.instructor\_id = dbo.alumnos\_x\_instructor.instructor\_id INNER JOIN

dbo.usuario ON dbo.alumnos\_x\_instructor.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.institucion ON dbo.alumnos\_x\_instructor.institucion\_id = dbo.institucion.institucion\_id INNER JOIN

dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id

where alumnos\_x\_instructor.instructor\_id= @id\_instructor and usuario.usuario\_estado<>'pendiente' and usuario.usuario\_doc = @alumno\_dni order by usuario.usuario\_doc asc

GO

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

ALTER procedure [dbo].[Instructor\_buscar\_alumno\_recursivo]

@usuario\_id int

as

declare @id\_instructor int

select @id\_instructor= instructor\_id from instructor where instructor.usuario\_id = @usuario\_id

declare @instructor\_nombre as varchar(max)

select @instructor\_nombre = usuario.usuario\_apellido+' '+usuario.usuario\_nombre from usuario where usuario.usuario\_id = @usuario\_id

SELECT dbo.usuario.usuario\_id as 'ID',

dbo.usuario.usuario\_doc as 'Documento',

dbo.usuario.usuario\_apellido +' '+dbo.usuario.usuario\_nombre as 'Apellido y Nombre',

(cast(datediff(dd, usuario.usuario\_fechanac ,GETDATE()) / 365.25 as int)) as Edad,

dbo.usuario.usuario\_telefono as 'Teléfono',

dbo.graduacion.graduacion\_desc as 'Graduación',

dbo.usuario.usuario\_tipo as 'Tipo',

dbo.usuario.usuario\_sexo as 'Sexo',

dbo.usuario.usuario\_foto as 'foto',

@instructor\_nombre as 'Instructor',

graduacion.graduacion\_id AS 'graduacion\_id',

dbo.usuario.usuario\_nrolibreta as 'NroLibreta'

FROM dbo.instructor INNER JOIN

dbo.alumnos\_x\_instructor ON dbo.instructor.instructor\_id = dbo.alumnos\_x\_instructor.instructor\_id INNER JOIN

dbo.usuario ON dbo.alumnos\_x\_instructor.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.institucion ON dbo.alumnos\_x\_instructor.institucion\_id = dbo.institucion.institucion\_id INNER JOIN

dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id

where alumnos\_x\_instructor.instructor\_id= @id\_instructor and usuario.usuario\_estado='activo' order by usuario.usuario\_doc asc

select usuario.usuario\_apellido+', '+usuario\_nombre as 'ApellidoyNombre' from usuario where usuario.usuario\_id=@usuario\_id

--esto si quiero traer todo sin importar el estado

--SELECT dbo.usuario.usuario\_id as 'ID',

--dbo.usuario.usuario\_doc as 'Documento',

--dbo.usuario.usuario\_apellido +', '+dbo.usuario.usuario\_nombre as 'Apellido y Nombre', (cast(datediff(dd, usuario.usuario\_fechanac ,GETDATE()) / 365.25 as int)) as Edad,

--dbo.usuario.usuario\_telefono as 'Teléfono',

--dbo.graduacion.graduacion\_desc as 'Graduación',

--dbo.usuario.usuario\_tipo as 'Tipo'

--FROM dbo.instructor INNER JOIN

-- dbo.alumnos\_x\_instructor ON dbo.instructor.instructor\_id = dbo.alumnos\_x\_instructor.instructor\_id INNER JOIN

-- dbo.usuario ON dbo.alumnos\_x\_instructor.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

-- dbo.institucion ON dbo.alumnos\_x\_instructor.institucion\_id = dbo.institucion.institucion\_id INNER JOIN

-- dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id

-- where alumnos\_x\_instructor.instructor\_id= @id\_instructor and usuario.usuario\_estado<>'pendiente' and usuario.usuario\_doc = @alumno\_dni order by usuario.usuario\_doc asc

GO

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

ALTER Procedure [dbo].[Instructor\_Buscar\_Inscriptos]

@evento\_id as int,

@usuario\_id\_Instructor int--Id\_Instrucor

as

declare @instructor\_id int

select @instructor\_id= instructor\_id from instructor where instructor.usuario\_id=@usuario\_id\_Instructor

select inscripcion.usuario\_id as 'Inscr\_Usu\_ID' from inscripcion

where inscripcion.evento\_id=@evento\_id and inscripcion\_instructor\_id=@instructor\_id

GO

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

ALTER procedure [dbo].[instructor\_modificar\_porcentaje]

@instructor\_id int,

@instructor\_porcentaje decimal(38,2)

as

update instructor set instructor\_porcentaje = @instructor\_porcentaje where instructor\_id = @instructor\_id

GO

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

ALTER procedure [dbo].[Instructor\_obtener\_alumnos]

@usuario\_id int

as

declare @id\_instructor int

select @id\_instructor= instructor\_id from instructor where instructor.usuario\_id = @usuario\_id

declare @instructor\_nombre as varchar(max)

select @instructor\_nombre = usuario.usuario\_apellido+' '+usuario.usuario\_nombre from usuario where usuario.usuario\_id = @usuario\_id

SELECT dbo.usuario.usuario\_id as 'ID',

dbo.usuario.usuario\_doc as 'Documento',

dbo.usuario.usuario\_apellido +' '+dbo.usuario.usuario\_nombre as 'Apellido y Nombre', (cast(datediff(dd, usuario.usuario\_fechanac ,GETDATE()) / 365.25 as int)) as Edad,

dbo.usuario.usuario\_telefono as 'Teléfono',

dbo.graduacion.graduacion\_desc as 'Graduación',

dbo.usuario.usuario\_tipo as 'Tipo',

dbo.usuario.usuario\_sexo as 'Sexo',

dbo.usuario.usuario\_foto as 'foto',

graduacion.graduacion\_id as 'graduacion\_id',

@instructor\_nombre as 'Instructor',

dbo.usuario.usuario\_nrolibreta as 'NroLibreta'

FROM dbo.instructor INNER JOIN

dbo.alumnos\_x\_instructor ON dbo.instructor.instructor\_id = dbo.alumnos\_x\_instructor.instructor\_id INNER JOIN

dbo.usuario ON dbo.alumnos\_x\_instructor.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.institucion ON dbo.alumnos\_x\_instructor.institucion\_id = dbo.institucion.institucion\_id INNER JOIN

dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id

where alumnos\_x\_instructor.instructor\_id= @id\_instructor and usuario.usuario\_estado='activo' order by usuario.usuario\_doc asc

select usuario.usuario\_apellido+', '+usuario\_nombre as 'ApellidoyNombre' from usuario where usuario.usuario\_id=@usuario\_id

SELECT dbo.usuario.usuario\_id as 'ID',

dbo.usuario.usuario\_doc as 'Documento',

dbo.usuario.usuario\_apellido +' '+dbo.usuario.usuario\_nombre as 'Apellido y Nombre', (cast(datediff(dd, usuario.usuario\_fechanac ,GETDATE()) / 365.25 as int)) as Edad,

dbo.usuario.usuario\_telefono as 'Teléfono',

dbo.graduacion.graduacion\_desc as 'Graduación',

dbo.usuario.usuario\_tipo as 'Tipo',

dbo.usuario.usuario\_estado as 'Estado',

@instructor\_nombre as 'Instructor',

dbo.usuario.usuario\_nrolibreta as 'NroLibreta'

FROM dbo.instructor INNER JOIN

dbo.alumnos\_x\_instructor ON dbo.instructor.instructor\_id = dbo.alumnos\_x\_instructor.instructor\_id INNER JOIN

dbo.usuario ON dbo.alumnos\_x\_instructor.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.institucion ON dbo.alumnos\_x\_instructor.institucion\_id = dbo.institucion.institucion\_id INNER JOIN

dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id

where alumnos\_x\_instructor.instructor\_id= @id\_instructor and usuario.usuario\_estado<>'pendiente' order by usuario.usuario\_doc asc

GO

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

ALTER procedure [dbo].[Instructor\_obtener\_id]

--obtengo instructor\_id a partir del id del alumno

@usuario\_id int

as

--select instructor\_id from alumnos\_x\_instructor where alumnos\_x\_instructor.usuario\_id= @usuario\_id and alumnos\_x\_instructor.alumno\_x\_instructor\_estado='activo'

SELECT dbo.usuario.usuario\_id as 'ID',

dbo.usuario.usuario\_doc as 'Documento',

dbo.usuario.usuario\_apellido +' '+dbo.usuario.usuario\_nombre as 'Apellido y Nombre',

(cast(datediff(dd, usuario.usuario\_fechanac ,GETDATE()) / 365.25 as int)) as Edad,

dbo.usuario.usuario\_telefono as 'Teléfono',

dbo.graduacion.graduacion\_desc as 'Graduación',

dbo.usuario.usuario\_tipo as 'Tipo',

dbo.usuario.usuario\_sexo as 'Sexo',

dbo.usuario.usuario\_foto as 'foto',

graduacion.graduacion\_id AS 'graduacion\_id',

instructor.instructor\_id

FROM dbo.instructor INNER JOIN

dbo.alumnos\_x\_instructor ON dbo.instructor.instructor\_id = dbo.alumnos\_x\_instructor.instructor\_id INNER JOIN

dbo.usuario ON dbo.alumnos\_x\_instructor.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.institucion ON dbo.alumnos\_x\_instructor.institucion\_id = dbo.institucion.institucion\_id INNER JOIN

dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id

where usuario.usuario\_id=@usuario\_id and usuario.usuario\_estado='activo' order by usuario.usuario\_doc asc

GO

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

ALTER procedure [dbo].[Instructor\_obtener\_INFO]

@instructor\_id int

as

--recupera la informacion del instructor

select

usuario.usuario\_apellido+', '+usuario.usuario\_nombre as 'ApellidoyNombre',

usuario.usuario\_doc, instructor.instructor\_porcentaje

from instructor inner join usuario on instructor.usuario\_id = usuario.usuario\_id

where instructor.instructor\_id= @instructor\_id

GO

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

ALTER procedure [dbo].[Instructor\_obtener\_institucion\_id]

@instructor\_id int

as

select \* from institucion\_x\_instructor where instructor\_id = @instructor\_id

GO

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

ALTER procedure [dbo].[Instructor\_obtener\_instructores]

@usuario\_id int

as

declare @id\_instructor int

select @id\_instructor= instructor\_id from instructor where instructor.usuario\_id = @usuario\_id

select

alumnos\_x\_instructor.alumno\_x\_instructor\_id, usuario.usuario\_id, usuario.usuario\_apellido+', '+usuario.usuario\_nombre as 'Apenom',

alumnos\_x\_instructor.alumno\_x\_instructor\_estado

from alumnos\_x\_instructor inner join usuario on alumnos\_x\_instructor.usuario\_id=usuario.usuario\_id

where alumnos\_x\_instructor.instructor\_id = @id\_instructor and

usuario.usuario\_estado='activo' and

usuario.usuario\_tipo='instructor' order by usuario.usuario\_apellido asc

GO

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

ALTER procedure [dbo].[Instructor\_obtener\_invitado]

as

select

instructor.instructor\_id,

usuario.usuario\_id,

usuario.tipodoc\_id,

usuario.usuario\_doc,

usuario\_apellido+','+usuario.usuario\_nombre+'.(dni:'+convert(varchar,usuario\_doc)+')' as 'Nombre',

institucion\_x\_instructor.institucion\_id

from instructor inner join usuario on instructor.usuario\_id = usuario.usuario\_id

inner join institucion\_x\_instructor on institucion\_x\_instructor.instructor\_id = instructor.instructor\_id

where usuario.usuario\_estado = 'activo' and usuario.usuario\_obs = 'invitado' order by usuario.usuario\_apellido asc

GO

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

ALTER procedure [dbo].[Instructor\_Obtener\_Mi\_ID\_de\_Instructor]

@usuario\_id int

as

--recupera la informacion del instructor

select

usuario.usuario\_apellido+', '+usuario.usuario\_nombre as 'ApellidoyNombre',

usuario.usuario\_doc,

instructor.instructor\_id

from instructor inner join usuario on instructor.usuario\_id = usuario.usuario\_id

where instructor.usuario\_id= @usuario\_id

GO

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

ALTER procedure [dbo].[Instructor\_obtener\_solo\_alumnos\_INSTRUCTORES]

---lo uso para traer alumnos instructores.

@usuario\_id int

as

declare @id\_instructor int

select @id\_instructor= instructor\_id from instructor where instructor.usuario\_id = @usuario\_id

select instructor.instructor\_id as 'ID',

usuario.usuario\_doc as 'Documento',

usuario.usuario\_apellido +', '+usuario.usuario\_nombre as 'ApeNomb',

usuario.usuario\_nrolibreta as 'Libreta',

graduacion.graduacion\_desc as 'Graduacion',

instructor.instructor\_porcentaje as 'Porcentaje',

usuario.usuario\_id,

usuario.graduacion\_id

from alumnos\_x\_instructor inner join instructor on alumnos\_x\_instructor.usuario\_id = instructor.usuario\_id

inner join usuario on alumnos\_x\_instructor.usuario\_id = usuario.usuario\_id

inner join graduacion on usuario.graduacion\_id = graduacion.graduacion\_id

where alumnos\_x\_instructor.instructor\_id = @id\_instructor order by graduacion.graduacion\_id, usuario.usuario\_apellido+', '+usuario.usuario\_nombre asc

GO

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

ALTER procedure [dbo].[Llave\_alta]

@evento\_id int,

@categoria\_id int,

@Llave\_cantidad int,

@area\_id int

as

insert Llave (evento\_id, categoria\_id, Llave\_cantidad, Area\_id ) values (@evento\_id, @categoria\_id, @Llave\_cantidad, @area\_id )

SELECT @@IDENTITY AS 'Llave\_id'

GO

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

ALTER procedure [dbo].[Llave\_deshacer\_llave]

@usuario\_id int,

@evento\_id int,

@categoria\_id int

as

---aqui vamos a poner el campo en\_llano en no...esto borra la inscripcion, solo cambia de estado ya q no estará asignada aun a una llave en particular

declare @inscripcion\_id int

select @inscripcion\_id = inscripcion.inscripcion\_id from inscripcion where inscripcion.evento\_id=@evento\_id and inscripcion.usuario\_id=@usuario\_id

--ahora lo busco en la tabla inscripcions x torneo

update inscripciones\_x\_torneo set en\_llave='no' where inscripciones\_x\_torneo.inscripcion\_id=@inscripcion\_id and inscripciones\_x\_torneo.categoria\_id=@categoria\_id

GO

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

ALTER procedure [dbo].[llave\_eliminar]

@llave\_id int

as

delete from Llave\_item where Llave\_item.Llave\_id = @llave\_id

delete from Llave where Llave.Llave\_id=@llave\_id

GO

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

ALTER Procedure [dbo].[Llave\_item\_actualizar]

@LLave\_item\_id int

as

update Llave\_item set Llave\_item\_enlazado = 'enlazado' where Llave\_item.LLave\_item\_id = @LLave\_item\_id

GO

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

ALTER procedure [dbo].[Llave\_item\_actualizar\_orden]

--nota: modificacion fecha 2022-09-01

@LLave\_item\_id int,

@Llave\_item\_usuario\_id int

as

update Llave\_item set Llave\_item\_usuario\_id = @Llave\_item\_usuario\_id where LLave\_item\_id = @LLave\_item\_id

GO

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

ALTER procedure [dbo].[Llave\_item\_actualizar\_progreso]

@LLave\_item\_id int,

@Llave\_item\_usuario\_id int

as

update Llave\_item set Llave\_item\_usuario\_id=@Llave\_item\_usuario\_id where Llave\_item.LLave\_item\_id = @LLave\_item\_id

GO

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

ALTER Procedure [dbo].[Llave\_item\_actualizar\_raiz]

@LLave\_item\_id int

as

update Llave\_item set Llave\_item\_enlazado = 'enlazado listo' where Llave\_item.LLave\_item\_id = @LLave\_item\_id

GO

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

ALTER Procedure [dbo].[Llave\_item\_actualizar\_usuario]

@LLave\_item\_id int,

@Llave\_item\_usuario\_id int

as

update Llave\_item set Llave\_item\_enlazado = 'enlazado', Llave\_item\_usuario\_id=@Llave\_item\_usuario\_id where Llave\_item.LLave\_item\_id = @LLave\_item\_id

-----------toda esta seccion es para poner en estado "si" el campo "en\_llave" que es necesario para filtras las inscripciones q ya estan en una llave

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

--select \* from Llave\_item where Llave\_item.LLave\_item\_id =@LLave\_item\_id

declare @llave\_id int

select @llave\_id = Llave\_item.Llave\_id from Llave\_item where Llave\_item.LLave\_item\_id=@LLave\_item\_id

--obtengo el evento

--select \* from Llave where Llave.Llave\_id=@llave\_id

declare @evento\_id int

select @evento\_id = Llave.evento\_id from Llave where Llave.Llave\_id=@llave\_id

declare @categoria\_id int

select @categoria\_id = Llave.categoria\_id from Llave where Llave.Llave\_id = @llave\_id

declare @inscripcion\_id int

select @inscripcion\_id = inscripcion.inscripcion\_id from inscripcion where inscripcion.evento\_id=@evento\_id and inscripcion.usuario\_id=@Llave\_item\_usuario\_id

--SELECT \* FROM inscripciones\_x\_torneo where

--inscripciones\_x\_torneo.inscripcion\_id= @inscripcion\_id and inscripciones\_x\_torneo.categoria\_id=@categoria\_id

update inscripciones\_x\_torneo set en\_llave='si' where

inscripciones\_x\_torneo.inscripcion\_id= @inscripcion\_id and inscripciones\_x\_torneo.categoria\_id=@categoria\_id

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

GO

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

ALTER procedure [dbo].[Llave\_item\_alta]

@Llave\_item\_usuario\_id int,

@Llave\_item\_PIzq int,

@Llave\_item\_PDerecho int,

@Llave\_item\_nivel int,

@Llave\_item\_Numero int,

@estado varchar(30),

@Llave\_id int

as

insert Llave\_item (Llave\_item\_usuario\_id, Llave\_item\_PIzq,

Llave\_item\_PDerecho,

Llave\_item\_nivel,

Llave\_item\_Numero, Llave\_item\_enlazado, Llave\_id) values (

@Llave\_item\_usuario\_id,

@Llave\_item\_PIzq,

@Llave\_item\_PDerecho,

@Llave\_item\_nivel,

@Llave\_item\_Numero, @estado, @Llave\_id

)

GO

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

ALTER procedure [dbo].[Llave\_item\_borrar\_hoja]

@LLave\_item\_id int

as

delete from Llave\_item where Llave\_item.LLave\_item\_id=@LLave\_item\_id

GO

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

ALTER procedure [dbo].[Llave\_item\_consulta]

@Llave\_id int

as

select

LLave\_item\_id,

Llave\_item\_usuario\_id,

Llave\_item\_PIzq,

Llave\_item\_PDerecho,

Llave\_item\_nivel,

Llave\_item\_Numero, Llave\_item\_enlazado

from Llave\_item where Llave\_id =@Llave\_id order by Llave\_item.Llave\_item\_Numero asc

GO

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

ALTER procedure [dbo].[Llave\_item\_consulta\_nivel]

@Llave\_item\_nivel int,

@Llave\_id int

as

select

LLave\_item\_id,

Llave\_item\_usuario\_id,

Llave\_item\_PIzq,

Llave\_item\_PDerecho,

Llave\_item\_nivel,

Llave\_item\_Numero, Llave\_item\_enlazado

from Llave\_item where Llave\_item\_nivel=@Llave\_item\_nivel and llave\_id = @Llave\_id order by Llave\_item.Llave\_item\_Numero asc

GO

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

ALTER Procedure [dbo].[Llave\_item\_quitar\_enlace]

@LLave\_item\_id int

as

update Llave\_item set Llave\_item\_PIzq='0', Llave\_item\_PDerecho='0' where Llave\_item.LLave\_item\_id = @LLave\_item\_id

go

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

ALTER procedure [dbo].[LLave\_obtener\_inscriptos]

@evento\_id int

as

select evento.evento\_id,

evento.evento\_descripcion,

inscripcion.inscripcion\_id,

inscripciones\_x\_torneo.torneo\_id ,

inscripciones\_x\_torneo.categoria\_id,

categoria.categoria\_tipo,

categoria.categoria\_gradinicial,

categoria.categoria\_gradfinal,

categoria.categoria\_edadinicial,

categoria.categoria\_edadfinal,

categoria.categoria\_peso\_inical,

categoria.categoria\_peso\_Final,

categoria.categoria\_sexo

from inscripcion inner join inscripciones\_x\_torneo on inscripcion.inscripcion\_id = inscripciones\_x\_torneo.inscripcion\_id

inner join categoria on inscripciones\_x\_torneo.categoria\_id=categoria.categoria\_id

inner join evento on inscripcion.evento\_id=evento.evento\_id

where inscripcion.evento\_id=@evento\_id

order by inscripciones\_x\_torneo.categoria\_id asc

select \* from graduacion order by graduacion.graduacion\_id asc

go

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

ALTER procedure [dbo].[LLave\_obtener\_inscriptos\_categoria]

@evento\_id int,

@categoria\_id int

as

---esta rutina la uso para cargar inicialmente los inscriptos en los nodos de la llave segun su categoria.

select inscripcion.usuario\_id, 'no' as 'agregado'

from inscripcion inner join inscripciones\_x\_torneo on inscripcion.inscripcion\_id = inscripciones\_x\_torneo.inscripcion\_id

inner join categoria on inscripciones\_x\_torneo.categoria\_id=categoria.categoria\_id

inner join evento on inscripcion.evento\_id=evento.evento\_id

where inscripcion.evento\_id=@evento\_id and inscripciones\_x\_torneo.categoria\_id=@categoria\_id

order by inscripciones\_x\_torneo.categoria\_id asc

go

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

ALTER procedure [dbo].[LLave\_obtener\_inscriptos\_filtrados]

@evento\_id int,

@categoria\_tipo varchar(50)

as

--recupera todos los inscriptos filtrados por evento y modalidad lucha o formas

select evento.evento\_id,

evento.evento\_descripcion,

inscripcion.inscripcion\_id,

inscripciones\_x\_torneo.torneo\_id ,

inscripciones\_x\_torneo.categoria\_id,

categoria.categoria\_tipo,

categoria.categoria\_gradinicial,

categoria.categoria\_gradfinal,

categoria.categoria\_edadinicial,

categoria.categoria\_edadfinal,

categoria.categoria\_peso\_inical,

categoria.categoria\_peso\_Final,

categoria.categoria\_sexo

from inscripcion inner join inscripciones\_x\_torneo on inscripcion.inscripcion\_id = inscripciones\_x\_torneo.inscripcion\_id

inner join categoria on inscripciones\_x\_torneo.categoria\_id=categoria.categoria\_id

inner join evento on inscripcion.evento\_id=evento.evento\_id

where inscripcion.evento\_id=@evento\_id and categoria.categoria\_tipo=@categoria\_tipo and inscripciones\_x\_torneo.en\_llave = 'no'

order by inscripciones\_x\_torneo.categoria\_id asc

select \* from graduacion order by graduacion.graduacion\_id asc

go

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

ALTER procedure [dbo].[LLave\_obtener\_inscriptos\_sin\_llave]

@evento\_id int,

@categoria\_id int

as

select inscripcion.usuario\_id as 'ID',

usuario.usuario\_doc as 'dni',

usuario.usuario\_apellido+' '+usuario\_nombre as 'ApellidoyNombre',

institucion.institucion\_abreviacion as 'Institucion\_abreviatura',

institucion.institucion\_descripcion as 'Institucion',

provincia.provincia\_desc as 'Provincia',

inscripcion.inscripcion\_instructor\_id as 'instructor\_id',

inscripcion.inscripcion\_peso as 'Peso', inscripcion.inscripcion\_id

from inscripcion inner join inscripciones\_x\_torneo on inscripcion.inscripcion\_id = inscripciones\_x\_torneo.inscripcion\_id

inner join categoria on inscripciones\_x\_torneo.categoria\_id=categoria.categoria\_id

inner join evento on inscripcion.evento\_id=evento.evento\_id

inner join usuario on inscripcion.usuario\_id= usuario.usuario\_id

inner join alumnos\_x\_instructor on usuario.usuario\_id = alumnos\_x\_instructor.usuario\_id

inner join institucion on alumnos\_x\_instructor.institucion\_id= institucion.institucion\_id

inner join provincia on institucion.provincia\_id = provincia.provincia\_id

where inscripcion.evento\_id=@evento\_id and inscripciones\_x\_torneo.categoria\_id=@categoria\_id and inscripciones\_x\_torneo.en\_llave='no'

order by inscripcion.inscripcion\_peso asc

go

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

ALTER procedure [dbo].[LLave\_obtener\_llavegenerada\_etc]

@evento\_id int,

@categoria\_id int

as

select evento.evento\_id,

evento.evento\_descripcion,

inscripcion.inscripcion\_id,

inscripciones\_x\_torneo.torneo\_id ,

inscripciones\_x\_torneo.categoria\_id,

categoria.categoria\_tipo,

categoria.categoria\_gradinicial,

categoria.categoria\_gradfinal,

categoria.categoria\_edadinicial,

categoria.categoria\_edadfinal,

categoria.categoria\_sexo,

evento.evento\_fecha,

evento.evento\_fechacierre,

categoria.categoria\_peso\_inical,

categoria.categoria\_peso\_Final

from inscripcion inner join inscripciones\_x\_torneo on inscripcion.inscripcion\_id = inscripciones\_x\_torneo.inscripcion\_id

inner join categoria on inscripciones\_x\_torneo.categoria\_id=categoria.categoria\_id

inner join evento on inscripcion.evento\_id=evento.evento\_id

where inscripcion.evento\_id=@evento\_id and inscripciones\_x\_torneo.categoria\_id=@categoria\_id

order by inscripciones\_x\_torneo.categoria\_id asc

select \* from graduacion order by graduacion.graduacion\_id asc

declare @llave\_id int

select @llave\_id = llave\_id from Llave where Llave.evento\_id=@evento\_id and Llave.categoria\_id=@categoria\_id

select

\*

from Llave\_item where Llave\_item.Llave\_id = @llave\_id order by Llave\_item.Llave\_item\_Numero asc

--aun no la uso

select

Llave\_item.Llave\_item\_usuario\_id,usuario.usuario\_id,

usuario.usuario\_apellido +', '+usuario.usuario\_nombre as 'apenom',

Llave\_item\_Numero

from Llave\_item inner join usuario on Llave\_item.Llave\_item\_usuario\_id=usuario.usuario\_id where Llave\_item.Llave\_id = @llave\_id order by Llave\_item.Llave\_item\_Numero asc

go

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

ALTER procedure [dbo].[LLave\_obtener\_llavegenerada\_etc\_2]

@Llave\_id int

as

---esta version trae toda la info de la llave generada solo que ahora la busca por llave\_id, ya que no me sirve buscar por evento\_id y categoria\_id (OJO puede haber mas de 1 llave por categoria, x eso la correccion)

select

evento.evento\_id,

evento.evento\_descripcion,

Llave.categoria\_id,

categoria.categoria\_tipo,

categoria.categoria\_gradinicial,

categoria.categoria\_gradfinal,

categoria.categoria\_edadinicial,

categoria.categoria\_edadfinal,

categoria.categoria\_sexo,

evento.evento\_fecha,

evento.evento\_fechacierre,

categoria.categoria\_peso\_inical,

categoria.categoria\_peso\_Final,

Llave.Llave\_cantidad as 'inscriptos'

from Llave inner join evento on Llave.evento\_id= evento.evento\_id

inner join categoria on Llave.categoria\_id=categoria.categoria\_id

where Llave.Llave\_id = @Llave\_id

select \* from graduacion order by graduacion.graduacion\_id asc

select \* from Llave\_item where Llave\_item.Llave\_id = @llave\_id order by Llave\_item.Llave\_item\_Numero asc

--aun no la uso

select

Llave\_item.Llave\_item\_usuario\_id,usuario.usuario\_id,

usuario.usuario\_apellido +', '+usuario.usuario\_nombre as 'apenom',

Llave\_item\_Numero,

alumnos\_x\_instructor.instructor\_id as 'instructor\_id'

from Llave\_item inner join usuario on Llave\_item.Llave\_item\_usuario\_id=usuario.usuario\_id

inner join alumnos\_x\_instructor on usuario.usuario\_id = alumnos\_x\_instructor.usuario\_id

where Llave\_item.Llave\_id = @llave\_id order by Llave\_item.Llave\_item\_Numero asc

go

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

ALTER procedure [dbo].[Llave\_obtener\_llaves\_generadas\_info]

---esta rutina trae un listado con las llaves que se han generado con su informacion particular como el nombre y la cantidad de inscriptos

@evento\_id int

as

select Llave.Llave\_id as 'ID',

Llave.Llave\_cantidad as 'inscriptos',

Llave.categoria\_id,

categoria.categoria\_tipo,

categoria.categoria\_gradinicial,

categoria.categoria\_gradfinal,

categoria.categoria\_edadinicial,

categoria.categoria\_edadfinal,

categoria.categoria\_peso\_inical,

categoria.categoria\_peso\_Final,

categoria.categoria\_sexo,

Area.Area\_descripcion as 'Area'

from Llave inner join categoria on Llave.categoria\_id=categoria.categoria\_id

inner join evento on Llave.evento\_id=evento.evento\_id

inner join Area on Llave.Area\_id = Area.Area\_id

where Llave.evento\_id = @evento\_id order by categoria.categoria\_id asc

select \* from graduacion order by graduacion.graduacion\_id asc

go

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

ALTER procedure [dbo].[Llave\_obtener\_llaves\_generadas\_infoArea]

---esta rutina trae un listado con las llaves que se han generado con su informacion particular como el nombre y la cantidad de inscriptos

@evento\_id int,

@area\_id int

as

select Llave.Llave\_id as 'ID',

Llave.Llave\_cantidad as 'inscriptos',

Llave.categoria\_id,

categoria.categoria\_tipo,

categoria.categoria\_gradinicial,

categoria.categoria\_gradfinal,

categoria.categoria\_edadinicial,

categoria.categoria\_edadfinal,

categoria.categoria\_peso\_inical,

categoria.categoria\_peso\_Final,

categoria.categoria\_sexo,

Area.Area\_descripcion as 'Area'

from Llave inner join categoria on Llave.categoria\_id=categoria.categoria\_id

inner join evento on Llave.evento\_id=evento.evento\_id

inner join Area on Llave.Area\_id = Area.Area\_id

where Llave.evento\_id = @evento\_id and Area.Area\_id = @area\_id order by categoria.categoria\_id asc

select \* from graduacion order by graduacion.graduacion\_id asc

go

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

ALTER procedure [dbo].[Llave\_para\_reordenar]

@Llave\_id int

as

select LLave\_item\_id, Llave\_item\_usuario\_id, Llave\_item\_PIzq, Llave\_item\_PDerecho, Llave\_item\_nivel, Llave\_item\_Numero, Llave\_item\_enlazado, Llave\_id,

usuario\_apellido +','+usuario\_nombre as 'USUARIO', alumnos\_x\_instructor.instructor\_id

from Llave\_item inner join usuario on usuario.usuario\_id = Llave\_item\_usuario\_id

inner join alumnos\_x\_instructor on alumnos\_x\_instructor.usuario\_id = usuario.usuario\_id where Llave\_id = @Llave\_id order by LLave\_item\_id asc

select \* from Llave\_item where Llave\_id = @Llave\_id

go

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

ALTER Procedure [dbo].[Llave\_Verificar\_existencia]

@cat\_id int,

@evento\_id int

as

select \* from Llave where Llave.categoria\_id=@cat\_id and Llave.evento\_id=@evento\_id

go

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

ALTER procedure [dbo].[Llaves\_Finalizadas\_filtro]

---Busco el nivel 0 de cada categoria si

@Llave\_id int

as

select Llave\_item\_usuario\_id from Llave\_item where Llave\_id=@Llave\_id and Llave\_item\_nivel=0

go

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

ALTER PROCEDURE [dbo].[Mantenimiento\_actualizar\_inscripcion]

---se actualiza categoria\_id en la tabla inscripcion\_torneo

@torneo\_id int,

@categoria\_id INT

as

UPDATE inscripciones\_x\_torneo set categoria\_id = @categoria\_id where torneo\_id = @torneo\_id

Go

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

ALTER PROCEDURE [dbo].[Mantenimiento\_ObtenerInscriptos\_AmbosSexosLucha]

@evento\_id int ---trae inscriptos solo de 1 evento puntual

AS

select

inscripcion.evento\_id,

inscripcion.inscripcion\_id,

inscripciones\_x\_torneo.categoria\_id,

categoria\_sexo,

categoria\_gradinicial,

categoria\_gradfinal,

categoria\_edadinicial,

categoria\_edadfinal,

categoria\_peso\_inical,

categoria\_peso\_Final,

categoria\_tipo,

inscripcion.inscripcion\_peso,

inscripciones\_x\_torneo.torneo\_id from inscripciones\_x\_torneo INNER JOIN inscripcion ON inscripciones\_x\_torneo.inscripcion\_id = inscripcion.inscripcion\_id

INNER JOIN categoria on inscripciones\_x\_torneo.categoria\_id = categoria.categoria\_id

where inscripcion.evento\_id = @evento\_id and categoria.categoria\_sexo = 'AMBOS SEXOS' and categoria.categoria\_tipo = 'Lucha'

go

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

ALTER PROCEDURE [dbo].[Mantenimiento\_ObtenerInscriptos\_todo]

@evento\_id int ---trae inscriptos solo de 1 evento puntual

AS

select

inscripcion.evento\_id,

inscripcion.inscripcion\_id,

inscripciones\_x\_torneo.categoria\_id,

categoria\_sexo,

categoria\_gradinicial,

categoria\_gradfinal,

categoria\_edadinicial,

categoria\_edadfinal,

categoria\_peso\_inical,

categoria\_peso\_Final,

categoria\_tipo,

inscripcion.inscripcion\_peso,

inscripciones\_x\_torneo.torneo\_id from inscripciones\_x\_torneo INNER JOIN inscripcion ON inscripciones\_x\_torneo.inscripcion\_id = inscripcion.inscripcion\_id

INNER JOIN categoria on inscripciones\_x\_torneo.categoria\_id = categoria.categoria\_id

where inscripcion.evento\_id = @evento\_id and categoria.categoria\_tipo = 'Lucha'

go

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

ALTER procedure [dbo].[Mensaje\_Nuevo\_Registro]

@usuario\_id as int

as

SELECT

(usuario\_apellido+ ' '+ usuario\_nombre) AS Nombre,

usuario\_domicilio,

provincia.provincia\_desc, usuario\_sexo,usuario\_doc, institucion.institucion\_descripcion, usuario.usuario\_tipo, alumnos\_x\_instructor.institucion\_id, usuario.usuario\_foto, usuario.usuario\_fecha\_registro

FROM dbo.alumnos\_x\_instructor INNER JOIN

dbo.institucion ON dbo.alumnos\_x\_instructor.institucion\_id = dbo.institucion.institucion\_id INNER JOIN

dbo.usuario ON dbo.alumnos\_x\_instructor.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.provincia ON dbo.usuario.provincia\_id = dbo.provincia.provincia\_id

where usuario\_estado='pendiente' and usuario.usuario\_id=@usuario\_id

go

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

ALTER procedure [dbo].[Miembros\_obtener]

@usuario\_id int

as

declare @id\_instructor int

select @id\_instructor = instructor.instructor\_id from instructor where instructor.usuario\_id= @usuario\_id

SELECT dbo.usuario.usuario\_doc, dbo.usuario.usuario\_apellido, dbo.usuario.usuario\_nombre, dbo.usuario.graduacion\_id, dbo.graduacion.graduacion\_desc,

dbo.usuario.usuario\_fechanac, dbo.usuario.usuario\_estado, dbo.usuario.usuario\_tipo

FROM dbo.alumnos\_x\_instructor INNER JOIN

dbo.usuario ON dbo.alumnos\_x\_instructor.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id

where alumnos\_x\_instructor.instructor\_id=@id\_instructor

go

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

ALTER procedure [dbo].[Miembros\_obtener\_datos\_personales]

@usuario\_id int

as

select usuario\_apellido +', '+usuario\_nombre as 'apellido y nombre',

usuario\_doc as 'dni', usuario\_fechanac as 'fecha de nacimiento',

usuario\_telefono as 'telefono', usuario\_mail as 'correo',

usuario\_domicilio as 'direccion', provincia.provincia\_desc as 'provincia',

ciudad\_desc as 'ciudad', usuario\_nacionalidad as 'nacionalidad',

estadocivil\_desc as 'estado civil', usuario\_profesion as 'profesion',

usuario\_foto as 'foto', usuario\_tipo, graduacion\_desc as 'graduacion',

usuario.usuario\_estado as 'estado'

from dbo.usuario INNER JOIN

dbo.provincia ON dbo.usuario.provincia\_id = dbo.provincia.provincia\_id INNER JOIN

dbo.ciudad ON dbo.usuario.ciudad\_id = dbo.ciudad.ciudad\_id INNER JOIN

dbo.estado\_civil ON dbo.usuario.estadocivil\_id = dbo.estado\_civil.estadocivil\_id

INNER JOIN

dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id

where usuario.usuario\_id=@usuario\_id

declare @id\_instructor int

select @id\_instructor=instructor\_id from alumnos\_x\_instructor where alumnos\_x\_instructor.usuario\_id=@usuario\_id and alumnos\_x\_instructor.alumno\_x\_instructor\_estado='activo'

--aqui recupero el id\_instructor necesario para buscar la institucion en la tabla institucion x instructor

SELECT provincia.provincia\_desc as 'provincia', institucion.institucion\_abreviacion as 'abreviacion',

institucion.institucion\_descripcion as 'institucion',

usuario.usuario\_apellido+', '+usuario.usuario\_nombre as 'instructor'

FROM dbo.instructor INNER JOIN

dbo.usuario ON dbo.instructor.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.alumnos\_x\_instructor ON dbo.instructor.instructor\_id = dbo.alumnos\_x\_instructor.instructor\_id INNER JOIN

dbo.institucion ON dbo.alumnos\_x\_instructor.institucion\_id = dbo.institucion.institucion\_id INNER JOIN

dbo.provincia ON dbo.institucion.provincia\_id = dbo.provincia.provincia\_id

where alumnos\_x\_instructor.alumno\_x\_instructor\_estado='activo' and

alumnos\_x\_instructor.instructor\_id=@id\_instructor and alumnos\_x\_instructor.usuario\_id=@usuario\_id

go

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

ALTER procedure [dbo].[Provincias\_generar]

--se genera la tabla desde cero con las provincias

as

INSERT INTO provincia VALUES ('1', 'Buenos Aires');

INSERT INTO provincia VALUES ('2', 'Capital Federal');

INSERT INTO provincia VALUES ('3', 'Catamarca');

INSERT INTO provincia VALUES ('4', 'Chaco');

INSERT INTO provincia VALUES ('5', 'Chubut');

INSERT INTO provincia VALUES ('6', 'Cordoba');

INSERT INTO provincia VALUES ('7', 'Corrientes');

INSERT INTO provincia VALUES ('8', 'Entre Rios');

INSERT INTO provincia VALUES ('9', 'Formosa');

INSERT INTO provincia VALUES ('10', 'Jujuy');

INSERT INTO provincia VALUES ('11', 'La Pampa');

INSERT INTO provincia VALUES ('12', 'La Rioja');

INSERT INTO provincia VALUES ('13', 'Mendoza');

INSERT INTO provincia VALUES ('14', 'Misiones');

INSERT INTO provincia VALUES ('15', 'Neuquen');

INSERT INTO provincia VALUES ('16', 'Rio Negro');

INSERT INTO provincia VALUES ('17', 'Salta');

INSERT INTO provincia VALUES ('18', 'San Juan');

INSERT INTO provincia VALUES ('19', 'San Luis');

INSERT INTO provincia VALUES ('20', 'Santa Cruz');

INSERT INTO provincia VALUES ('21', 'Santa Fe');

INSERT INTO provincia VALUES ('22', 'Santiago del Estero');

INSERT INTO provincia VALUES ('23', 'Tierra del Fuego');

INSERT INTO provincia VALUES ('24', 'Tucuman');

Go

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

ALTER Procedure [dbo].[Reporte\_Inscripcion\_Persona]

@usarioid as int,

@eventoid as int

as

select

dbo.inscripcion.usuario\_id,

dbo.usuario.usuario\_apellido + ', '+ dbo.usuario.usuario\_nombre as 'apellido y nombre',

dbo.usuario.usuario\_doc,

dbo.inscripcion.evento\_id,

dbo.evento.evento\_descripcion,

dbo.inscripcion.inscripcion\_id,

dbo.inscripciones\_x\_torneo.categoria\_id,

dbo.categoria.categoria\_tipo,

dbo.inscripcion\_qr.inscripcion\_qr\_imagen,

dbo.usuario.graduacion\_id,

dbo.graduacion.graduacion\_desc,

inscripcion.inscripcion\_peso

from inscripcion inner join inscripcion\_qr on inscripcion.inscripcion\_id=inscripcion\_qr.inscripcion\_id

inner join usuario on inscripcion.usuario\_id=usuario.usuario\_id

inner join evento on inscripcion.evento\_id= evento.evento\_id

inner join inscripciones\_x\_torneo on inscripcion.inscripcion\_id=inscripciones\_x\_torneo.inscripcion\_id

inner join categoria on inscripciones\_x\_torneo.categoria\_id=categoria.categoria\_id

inner join graduacion on usuario.graduacion\_id=graduacion.graduacion\_id

where inscripcion.usuario\_id=@usarioid and inscripcion.evento\_id=@eventoid order by usuario.usuario\_id

go

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

ALTER procedure [dbo].[Reporte\_listado\_inscriptos]

@instructor\_id int,

@evento\_id int

as

select

usuario.usuario\_id,

usuario.usuario\_apellido +', '+ usuario.usuario\_nombre as 'ApellidoyNombre',

instructor.instructor\_id,

evento.evento\_descripcion as 'Evento',

evento.evento\_fecha,

inscripciones\_x\_torneo.categoria\_id,

categoria.categoria\_tipo,

categoria.categoria\_gradinicial,

categoria.categoria\_gradfinal,

categoria.categoria\_edadinicial,

categoria.categoria\_edadfinal,

categoria.categoria\_peso\_inical,

categoria.categoria\_peso\_Final,

categoria.categoria\_sexo

from inscripcion inner join inscripciones\_x\_torneo on inscripcion.inscripcion\_id=inscripciones\_x\_torneo.inscripcion\_id

inner join usuario on inscripcion.usuario\_id=usuario.usuario\_id

inner join instructor on inscripcion.inscripcion\_instructor\_id = instructor.instructor\_id

inner join evento on inscripcion.evento\_id = evento.evento\_id

inner join categoria on inscripciones\_x\_torneo.categoria\_id=categoria.categoria\_id

where evento.evento\_id = @evento\_id

select \* from graduacion order by graduacion.graduacion\_id asc

select

usuario.usuario\_apellido+usuario.usuario\_nombre as 'Instructor\_apenom',

usuario.usuario\_doc,

graduacion.graduacion\_desc

from instructor inner join usuario on instructor.usuario\_id=usuario.usuario\_id

inner join graduacion on usuario.graduacion\_id=graduacion.graduacion\_id

where instructor.instructor\_id = instructor\_id

go

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

ALTER procedure [dbo].[Reporte\_obtener\_Inscriptos\_Toreno\_Instructor]

@evento\_id int,

@instructor\_id int

as

select

evento.evento\_id,

evento.evento\_descripcion,

evento.evento\_foto,

evento.evento\_fecha,

usuario.usuario\_id ,

usuario.usuario\_doc,

usuario.usuario\_apellido,

usuario.usuario\_nombre,

graduacion.graduacion\_desc as 'graduacion',

usuario.usuario\_telefono,

'INSTRUCTOR' as 'instructor',

usuario.usuario\_foto,

instructor.instructor\_id

from evento inner join inscripcion on evento.evento\_id= inscripcion.evento\_id

inner join usuario on inscripcion.usuario\_id= usuario.usuario\_id

inner join instructor on inscripcion.inscripcion\_instructor\_id= instructor.instructor\_id

inner join graduacion on usuario.graduacion\_id=graduacion.graduacion\_id

where evento.evento\_id=@evento\_id and instructor.instructor\_id=@instructor\_id

go

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

ALTER procedure [dbo].[Reporte\_obtener\_instructores\_y\_eventos]

as

select

instructor.instructor\_id,

usuario.usuario\_apellido+', '+usuario.usuario\_nombre as 'apenom'

from instructor inner join usuario on instructor.usuario\_id=usuario.usuario\_id

select

evento.evento\_id , evento.evento\_descripcion

from evento order by evento.evento\_descripcion asc

go

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

ALTER procedure [dbo].[Solicitudes\_Pendientes]

@usuario\_id int

as

--Select

--usuario\_id,

--(usuario\_apellido+ ' '+ usuario\_nombre) AS Nombre,

--usuario\_domicilio as Mensaje,

--usuario\_codigopostal as Fecha

--From usuario

--where usuario\_estado='Pendiente'

declare @id\_instructor int

select @id\_instructor= instructor\_id from instructor where instructor.usuario\_id=@usuario\_id

SELECT dbo.usuario.usuario\_id ,

(dbo.usuario.usuario\_apellido+ ' '+ dbo.usuario.usuario\_nombre) AS Nombre,

dbo.usuario.usuario\_estado as Mensaje, Convert(varchar(10),dbo.usuario.usuario\_fecha\_registro,103) as Fecha, usuario\_tipo, alumnos\_x\_instructor.institucion\_id

FROM dbo.alumnos\_x\_instructor INNER JOIN

dbo.usuario ON dbo.alumnos\_x\_instructor.usuario\_id = dbo.usuario.usuario\_id

where usuario\_estado='Pendiente' and instructor\_id=@id\_instructor order by usuario.usuario\_fecha\_registro asc

---Choco, existe el campo institucion\_id x si queremos hacer un select discriminado por institucion, ya que el instructor puede pertenecer a varias

Go

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

ALTER procedure [dbo].[Torneo\_recuperar\_inscriptos]

@evento\_id int

as

select evento\_id, evento\_descripcion, CONVERT(varchar, evento\_fecha, 103) as 'evento\_fecha',

evento\_direccion from evento where evento\_id = @evento\_id

select \* from inscripcion inner join inscripciones\_x\_torneo on inscripcion.inscripcion\_id = inscripciones\_x\_torneo.inscripcion\_id

inner join categoria on inscripciones\_x\_torneo.categoria\_id = categoria.categoria\_id

where evento\_id = @evento\_id order by categoria.categoria\_tipo asc , categoria\_gradinicial asc

select \* from graduacion order by graduacion.graduacion\_id asc

select \* from inscripcion where inscripcion.evento\_id = @evento\_id

go

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

ALTER procedure [dbo].[Torneo\_recuperar\_inscriptos\_categoria]

@evento\_id int,

@categoria\_id int

as

select \* from(SELECT dbo.evento.evento\_id, dbo.inscripcion.inscripcion\_id, dbo.inscripcion.usuario\_id, dbo.usuario.graduacion\_id, dbo.evento.evento\_descripcion, dbo.evento.evento\_fecha, dbo.evento.evento\_costo, dbo.evento.evento\_direccion,

usuario.usuario\_apellido+', '+usuario.usuario\_nombre as 'Alumno', dbo.graduacion.graduacion\_desc, usuario.usuario\_doc

FROM dbo.evento INNER JOIN

dbo.inscripcion ON dbo.evento.evento\_id = dbo.inscripcion.evento\_id INNER JOIN

dbo.usuario ON dbo.inscripcion.usuario\_id = dbo.usuario.usuario\_id INNER JOIN

dbo.graduacion ON dbo.usuario.graduacion\_id = dbo.graduacion.graduacion\_id INNER JOIN

dbo.inscripciones\_x\_torneo on dbo.inscripcion.inscripcion\_id = dbo.inscripciones\_x\_torneo.inscripcion\_id

where evento.evento\_id=@evento\_id and inscripciones\_x\_torneo.categoria\_id = @categoria\_id) T1

full outer join

(select inscripcion.inscripcion\_id, instructor.instructor\_id, usuario.usuario\_apellido+', '+usuario.usuario\_nombre as 'instructor'

from inscripcion inner join instructor on inscripcion.inscripcion\_instructor\_id = instructor.instructor\_id inner join usuario

on instructor.usuario\_id = usuario.usuario\_id

inner join inscripciones\_x\_torneo on inscripcion.inscripcion\_id = inscripciones\_x\_torneo.inscripcion\_id

where inscripcion.evento\_id = @evento\_id and inscripciones\_x\_torneo.categoria\_id = @categoria\_id ) T2 ON T1.inscripcion\_id = T2.inscripcion\_id

ORDER BY 1,2

Go

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

ALTER procedure [dbo].[Usuario\_actualizar\_contraseña]

@usuario\_password varchar(50),

@usuario\_id int

as

update usuario set usuario\_password=@usuario\_password where usuario\_id=@usuario\_id

go

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

ALTER procedure [dbo].[Usuario\_alta]

@usuario\_foto image,

@usuario\_apellido varchar(50),

@usuario\_nombre varchar(50),

@tipodoc\_id int,

@usuario\_doc int,

@usuario\_sexo varchar(50),

@usuario\_nacionalidad varchar(50),

@estadocivil\_id int,

@usuario\_profesion varchar(50),

@usuario\_fechanac date,

@usuario\_domicilio varchar(50),

@usuario\_codigopostal int,

@provincia\_id int,

@ciudad\_id int,

@usuario\_telefono varchar(50),

@usuario\_mail varchar(50),

@graduacion\_id int,

@usuario\_password varchar(50),

@usuario\_fecha\_registro datetime,

@instructor\_id int,

@usuario\_tipo varchar(50),

@usuario\_usuario varchar(50),

@institucion\_id int,

@usuario\_nrolibreta varchar(50)

as

declare @usuario\_id int

--insertar usuario, el estado va a estar en inactivo hasta que se apruebe la peticion.

insert dbo.usuario (usuario\_foto,

usuario\_apellido,

usuario\_nombre,

tipodoc\_id,

usuario\_doc,

usuario\_sexo,

usuario\_nacionalidad,

estadocivil\_id,

usuario\_profesion,

usuario\_fechanac,

usuario\_domicilio,

usuario\_codigopostal,

provincia\_id,

ciudad\_id,

usuario\_telefono,

usuario\_mail,

graduacion\_id,

usuario\_password,

usuario\_estado,

usuario\_fecha\_registro, usuario\_tipo, usuario\_usuario, usuario\_nrolibreta

)

values (@usuario\_foto,

@usuario\_apellido,

@usuario\_nombre,

@tipodoc\_id,

@usuario\_doc,

@usuario\_sexo,

@usuario\_nacionalidad,

@estadocivil\_id,

@usuario\_profesion,

@usuario\_fechanac,

@usuario\_domicilio,

@usuario\_codigopostal,

@provincia\_id,

@ciudad\_id,

@usuario\_telefono,

@usuario\_mail,

@graduacion\_id,

@usuario\_password,

'pendiente',

@usuario\_fecha\_registro, @usuario\_tipo, @usuario\_usuario, @usuario\_nrolibreta)

select @usuario\_id = @@IDENTITY

--insertar alumno x instructor

insert dbo.alumnos\_x\_instructor (usuario\_id, instructor\_id, alumno\_x\_instructor\_estado, institucion\_id) values (@usuario\_id, @instructor\_id, 'pendiente', @institucion\_id)

go

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

ALTER procedure [dbo].[Usuario\_alta\_invitado]

@usuario\_foto image,

@usuario\_apellido varchar(50),

@usuario\_nombre varchar(50),

@tipodoc\_id int,

@usuario\_doc int,

@usuario\_sexo varchar(50),

@usuario\_nacionalidad varchar(50),

@estadocivil\_id int,

@usuario\_profesion varchar(50),

@usuario\_fechanac date,

@usuario\_domicilio varchar(50),

@usuario\_codigopostal int,

@provincia\_id int,

@ciudad\_id int,

@usuario\_telefono varchar(50),

@usuario\_mail varchar(50),

@graduacion\_id int,

@usuario\_password varchar(50),

@usuario\_fecha\_registro datetime,

@instructor\_id int,

@usuario\_tipo varchar(50),

@usuario\_usuario varchar(50),

@institucion\_id int,

@usuario\_nrolibreta varchar(50),

@usuario\_obs varchar(max)

as

declare @usuario\_id int

--insertar usuario, el estado va a estar en inactivo hasta que se apruebe la peticion.

insert dbo.usuario (usuario\_foto,

usuario\_apellido,

usuario\_nombre,

tipodoc\_id,

usuario\_doc,

usuario\_sexo,

usuario\_nacionalidad,

estadocivil\_id,

usuario\_profesion,

usuario\_fechanac,

usuario\_domicilio,

usuario\_codigopostal,

provincia\_id,

ciudad\_id,

usuario\_telefono,

usuario\_mail,

graduacion\_id,

usuario\_password,

usuario\_estado,

usuario\_fecha\_registro, usuario\_tipo, usuario\_usuario, usuario\_nrolibreta, usuario\_obs

)

values (@usuario\_foto,

@usuario\_apellido,

@usuario\_nombre,

@tipodoc\_id,

@usuario\_doc,

@usuario\_sexo,

@usuario\_nacionalidad,

@estadocivil\_id,

@usuario\_profesion,

@usuario\_fechanac,

@usuario\_domicilio,

@usuario\_codigopostal,

@provincia\_id,

@ciudad\_id,

@usuario\_telefono,

@usuario\_mail,

@graduacion\_id,

@usuario\_password,

'activo',

@usuario\_fecha\_registro, @usuario\_tipo, @usuario\_usuario, @usuario\_nrolibreta, @usuario\_obs)

select @usuario\_id = @@IDENTITY

--insertar alumno x instructor

insert dbo.alumnos\_x\_instructor (usuario\_id, instructor\_id, alumno\_x\_instructor\_estado, institucion\_id) values (@usuario\_id, @instructor\_id, 'activo', @institucion\_id)

go

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

ALTER procedure [dbo].[Usuario\_filtrarciudades\_x\_Provincias]

@provincia\_id int

as

select ciudad\_id, ciudad\_desc from ciudad where ciudad.provincia\_id=@provincia\_id order by ciudad.ciudad\_desc asc

go

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

ALTER procedure [dbo].[Usuario\_imagen\_actualizar]

@usuario\_id int,

@usuario\_imagen image

as

update usuario set usuario\_foto=@usuario\_imagen where usuario.usuario\_id=@usuario\_id

go

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

ALTER procedure [dbo].[Usuario\_modificar\_graduacion]

@usuario\_id int,

@graduacion\_id int

as

update usuario set graduacion\_id = @graduacion\_id where usuario.usuario\_id = @usuario\_id

go

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

ALTER procedure [dbo].[Usuario\_ObtenerGraduaciones]

as

select graduacion.graduacion\_id, graduacion.graduacion\_desc from graduacion order by graduacion.graduacion\_id asc

go

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

ALTER procedure [dbo].[Usuario\_ObtenerInstituciones\_x\_provincia]

@provincia\_id int

as

select institucion\_id, institucion\_abreviacion from institucion where institucion.provincia\_id = @provincia\_id order by institucion.institucion\_descripcion asc

go

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

ALTER Procedure [dbo].[Usuario\_ObtenerInstructor]

@institucion\_id int

as

--SELECT (dbo.usuario.usuario\_apellido + ','+dbo.usuario.usuario\_nombre) as Nombre, dbo.instructor.instructor\_id

--FROM dbo.instructor INNER JOIN

-- dbo.usuario ON dbo.instructor.usuario\_id = dbo.usuario.usuario\_id

-- where instructor.institucion\_id= @institucion\_id order by usuario.usuario\_apellido asc

SELECT (usuario\_apellido + ',' + usuario\_nombre) as Nombre, dbo.instructor.instructor\_id

FROM dbo.instructor INNER JOIN

dbo.institucion\_x\_instructor ON dbo.instructor.instructor\_id = dbo.institucion\_x\_instructor.instructor\_id INNER JOIN

dbo.usuario ON dbo.instructor.usuario\_id = dbo.usuario.usuario\_id

where institucion\_id=@institucion\_id order by usuario.usuario\_apellido

go

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

ALTER procedure [dbo].[Usuario\_ObtenerProvincias]

as

select provincia.provincia\_id, provincia.provincia\_desc from provincia order by provincia.provincia\_desc asc

go

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

ALTER Procedure [dbo].[Usuario\_sesion]

@USU\_usuario varchar(max),

@USU\_contr varchar(max)

as

select usuario.usuario\_id,

usuario.usuario\_nombre,

usuario.usuario\_apellido,

usuario.usuario\_password,

usuario.usuario\_estado,

usuario.usuario\_tipo from usuario where usuario.usuario\_usuario=@USU\_usuario and usuario.usuario\_password=@USU\_contr

go

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

ALTER procedure [dbo].[Usuario\_validar\_contraseña]

@usuario\_id int,

@usuario\_password varchar(50)

as

select usuario\_id from dbo.usuario where usuario\_id=@usuario\_id and usuario\_password=@usuario\_password

go

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

ALTER procedure [dbo].[Usuario\_validar\_DNI]

@Usuario\_doc int

as

select \* from usuario where usuario.usuario\_doc = @Usuario\_doc

go

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

ALTER procedure [dbo].[Usuario\_validar\_registro]

@usuario\_usuario varchar(50),

@usuario\_doc int

as

select usuario\_id from usuario where usuario.usuario\_usuario = @usuario\_usuario

select usuario\_id from usuario where usuario.usuario\_doc = @usuario\_doc

go

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

ALTER procedure [dbo].[UsuarioEvento\_alta]

@evento\_id int,

@usuario\_usuario varchar(50),

@usuario\_password varchar(50)

as

insert usuario ( usuario\_usuario , usuario\_password, usuario\_tipo, usuario\_estado ) values (@usuario\_usuario, @usuario\_password, 'Torneo', 'activo')

declare @id\_usuario int

select @id\_usuario = @@IDENTITY

insert UsuarioEvento (evento\_id , usuario\_id ) values (@evento\_id , @id\_usuario )

go

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

ALTER procedure [dbo].[UsuarioEvento\_Obtener]

@usuario\_id int

as

select \* from UsuarioEvento where usuario\_id=@usuario\_id

go

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

ALTER procedure [dbo].[UsuarioEvento\_obtener\_evento]

@evento\_id int

as

--choco: lo uso para recuperar los usuarios vinculados al evento en la tabla usuarioevento

select UsuarioEvento.usuario\_id as id, usuario.usuario\_usuario as Usuario,

usuario.usuario\_password as contraseña from UsuarioEvento

inner join usuario on UsuarioEvento.usuario\_id = usuario.usuario\_id

where evento\_id = @evento\_id

go

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