

EVALUACION	Parcial 2	GRUPO	M3B	FECHA	25/06/2019
MATERIA	Base de Datos 2				
CARRERA	Analista Programador/Analista en Tecnologías de la Información				
CONDICIONES	Puntos 30 Sin material Duración 2.5 horas				

Una empresa lleva registro de viajes aéreos de las personas, para realizar su gestión implementó un sistema que utiliza las siguientes tablas:

PERSONA (Pasaporte, Nacionalidad, Nombre, CI)

Registra los datos de las Personas las cuales se identifican con el Pasaporte, y la CI es una clave alterna.

AVION (AviMod, AviNro, AviFchFab, AviAsi, AviCntVue)

Registra los datos de Aviones los cuales se identifican por su Modelo y su Número. Además se guarda la fecha de Fabricación, la Cantidad de asientos y la Cantidad de vuelos que ha realizado el avión.

AEREOPUERTOS (APid, APNombre, Pais)

Registra los datos de los Aereopuertos. Identificados por un número *APid*. También se guarda el nombre del aereopuerto y el país en cual se encuentra

VUELO (Vueld, AviMod, AviNro, CapPasaporte, VueFch, APId_S, APId_L, VueMillas)

Registra los datos de los Vuelos identificados por un número *Vueld*. Para cada vuelo se guarda el Avión que realiza el vuelo, el pasaporte del capitán encargado del vuelo, la fecha del vuelo, los aeropuertos de Salida y de Llegada, además de la cantidad de millas del vuelo.

PASAJE (PasiNro, Pasaporte, Vueld, Clase, NroAsi, Precio, Descuento)

Se registran los datos de los pasajes los cuales se identifican por un número, y se registra el pasaporte de la persona, el Vuelo, la clase (pasaje es de Primera o Económica), el Nro de Asientos, el Precio y el Descuento realizado si corresponde.

VIAJE (Viajeld, Vueld, Pasaporte, Motivo)

Se registran los datos de los viajes realizado por las personas Y los vuelos que componen cada viaje los tramos que componen el viaje de una persona. También se registra el motivo que puede ser: "Turismo", "Trabajo", "Otros"

Se pide:

- 1) Implementar una función que dado un aeropuerto, una persona y una fecha devuelva un indicador de si la persona ha pasado por el aeropuerto en la fecha indicada (*Puntos 3*)

```
Create Function Ej1(@APId int, @Pasap char(10), @Fch date) returns char(1)
as
begin
declare @paso char(1);

set @paso = 'N'
if Exists( select * from Vuelo V, Pasaje P
           where P.VueId = V.vueId and V.VueFch = @Fch and
                 P.pasaporte = @Pasap and
                 (V.ApId_S = @APId or V.ApId_L = @APId )
         )
    set @paso = 'S'
end
return @paso
end;end;
```

- 2) Implementar una función que dado la CI de un capitán y un país devuelva la cantidad de veces que el capitán a pasado por ese país, como capitán de vuelos. (*Puntos 4*)

```
Create Function Ej2(@ciCap char(11), @pais varchar(20)) returns int
as
begin
declare @cant int;
declare @cantLlegSal int;

-- cant de veces que paso como capitan de vuelos que SALIAN/LLEGABAN del pais
select @cant = count(*)
from Persona P, Vuelo V, Aereopuerto A
where P.pasaporte = V.CapPasaporte and P.ci = @ciCap and A.Pais = @pais
      and (V.APId_S = A.APId or V.APId_L = A.APId )

-- cant. de vuelos locales(pais de llegada igual al de salida), de ese pais
-- en estos casos se deberia contar una sola vez, en la consulta anterior se
-- contaron dos veces
select @cantLlegSal = count(*)
from Persona P, Vuelo V, Aereopuerto APS , Aeropuerto APL
where P.pasaporte = V.CapPasaporte and V.APId_L = APS.APId and
      V.APId_L = APL.APId and P.ci = @ciCap and APS.Pais = APL.Pais
      and APS.Pais = @pais

return @cant - @cantLlegSal
end
```

- 3) Implementar un procedimiento que dado un avión devuelva el promedio de millas que ha volado, y el año que ha volado más veces, si hay más de un año en estas condiciones devolver cualquiera de ellos. (Puntos 3)

```
Create Procedure Ej3 @avMod char(10), @avNro int, @cantMi int output,
                    @mejorAño int output
as
begin
    select @cantMi = avg(V.VueMillas)
    from Vuelo V where V.aviMod = @avMod and V.aviNro = @avNro ;

    Select @mejorAño = year(V.VueFch)
    from Vuelo V where V.aviMod = @avMod and V.aviNro = @avNro
    group by year(V.VueFch)
    order by count(*) asc;
end;
```

- 4) Implementar una función o procedimiento que reciba como parámetros un rango de fechas y devuelva la cantidad de aviones con antigüedad mayor a 10 años, y que han realizado más de 50 vuelos largos en el rango de fechas indicado. Un vuelo largo es aquel de más de 2000 millas (Puntos 3)

```
Create Procedure Ej4 @f1 date, @f2 date, @cntAvi int output
as
begin
    select @cntAvi = count(*)
    from Avion A
    where A.AviFchFab < year(getdate()) - 10 and
        (select count(*) from Vuelo V where V.aviMod = A.aviMod
         and V.aviNro = A.aviNro and V.vueFch >= @f1 and V.vueFch <= @f2
         and V.vueMillas > 2000
        )
        > 50
end;
```

- 5) Implementar un disparador que controle que para un vuelo el capitán no puede ser pasajero (es pasajero si tiene pasaje para ese vuelo). Considerar tanto inserciones como modificaciones. Considerar Múltiples registros, y de tal forma que debe afecte los registros que cumplan la condición. (Puntos 4)

```
Create Trigger Ej5 on Vuelo
Instead Of Insert, Update
as
begin
    if Exists(select * from Inserted ) and
    Not Exists(select * from Deleted)
    begin --operacion disparadora Insert

        insert into Vuelo select I.* from Inserted I
        where not exists(select * from Pasaje P where P.vueId = I.vueId
                        and I.capPasaporte = P.Pasaporte)
```

```
end
Else -- operacion disparadora Update
begin
update Vuelo
set AviMod = I.AviMod, AviNro = I.AviNro ,
    capPasaporte = I.capPasaporte, vueFch = I.VueFch,
    APId_S = I.APId_S, APId_L = I.APId_L,
    vueMillas = I.vueMillas
from vuelo V, Inserted I
where V.vueId = I.VueId and
    not exists(select * from Pasaje P where P.vueId = I.vueId
        and I.capPasaporte = P.Pasaporte)
end
end;
```

- 6) Implementar un disparador que cada vez que se cambie el descuento o el precio de un pasaje registre en la tabla LogCtrlPasajes los datos correspondientes

LOGCTRLPASAJES(fecha, usuario, Pasaje, PrecioAnt, PrecioNue, DescuentoAnt, DescuentoNue)

Considerar Múltiples registros. (Puntos 3)

```
Create Trigger Ej6 on Pasaje
After Update
as
begin
insert into LogCtrlPasajes
select getdate(), user, I.PasjNro, D.Precio, I.Precio, D.Descuento,
I.Descuento from Inserted I, Deleted D where I.PasjNro = D.PasjNro and
(I.Descuento <> D.Descuento or I.Precio <> D.Precio )
End
```

- 7) Implementar un disparador que cada vez que se agregue o elimine un vuelo actualice la cantidad de vuelos que ha realizado el avión. Considerar Múltiples registros. (Puntos 3)

```
create Trigger Ej7 on Vuelo
After Insert, Delete
as
begin
if Exists(select * from Inserted) and Not exists(select * from Deleted)
begin --operacion disparadora es insert
update avion set AviCntVue = AviCntVue + 1
from Avion A, Inserted I
where A.aviNro = I.AviNro and A.aviMod = I.AviMod
end
else -- operacion disparadora es delete
begin
update avion set AviCntVue = AviCntVue - 1
from Avion A, Inserted I
where A.aviNro = I.AviNro and A.aviMod = I.AviMod
end
end
```

- 8) Implementar un disparador que controle que no se pueda agregar un pasaje si el vuelo ya tiene sus asientos ocupados. Considere que se agrega de a un registro por vez. (Puntos 4)

```
create Trigger Ej8 on Pasaje
after insert
as
begin
    declare @AMod char(10);
    declare @ANro int;
    declare @vue int;

    select @AMod = V.aviMod, @ANro = V.aviNro, @vue = I.vueId
    from Inserted I, Vuelo V where I.vueId = V.vueId ;

    if (select count(*) from Pasaje where VueId = @vue) >
        (select AviAsi from Avion where AviMod = @AMod and AviNro = @ANro)
        Rollback;

End
```

- 9) Implementar una vista que para cada persona con más de 10 vuelos devuelva la CI, el nombre, cantidad de millas que ha volado, y fecha del último vuelo. Solo deben aparecer personas que con más de 3 vuelos (Puntos 3)

```
create view Ej10
as (
    select P.ci, P.nombre, sum(v.vueMillas) as CantMillas, max(V.vueFch) as
    UlimoVuelo
    from Velo V, Pasaje PJ, Persona P
    where V.vueid = PJ.vueId and PJ.pasaporte = P.pasaporte
    group by P.ci, P.nombre
    having count(V.vueId) > 10
)
```