



# Registro de Actividades Hípicas

---

**Materia:** 75.15 Bases de Datos

**Grupo:** 10

**Fecha de entrega:** 29 de mayo del 2013

**Integrantes:**

- Fraiman, Yamila (90708)
- Goldberg, Juan Sebastián (82078)
- Mezzina, Agustín (89637)
- Ramos, Milton (91110)



## Contents

Hipotesis y Supuestos .....	2
Validaciones a realizar.....	2
Diagrama Entidad Interrelación .....	4
Modelo Relacional.....	5
Diagrama del modelo relacional .....	8
Diccionario de datos.....	9
Consultas .....	13
Condición A .....	13
Condición B.....	13
Condición C.....	14
Condición D .....	15
Condición E.....	16
Condición F.....	18
Reporte.....	19

## Hipotesis y Supuestos

1. Puede existir más de un dueño por equino.
2. Puede existir más de un cuidador por equino.
3. Puede existir más de un entrenador por equino.
4. La madre y/o el padre de un equino pueden ser desconocidos.

## Validaciones a realizar

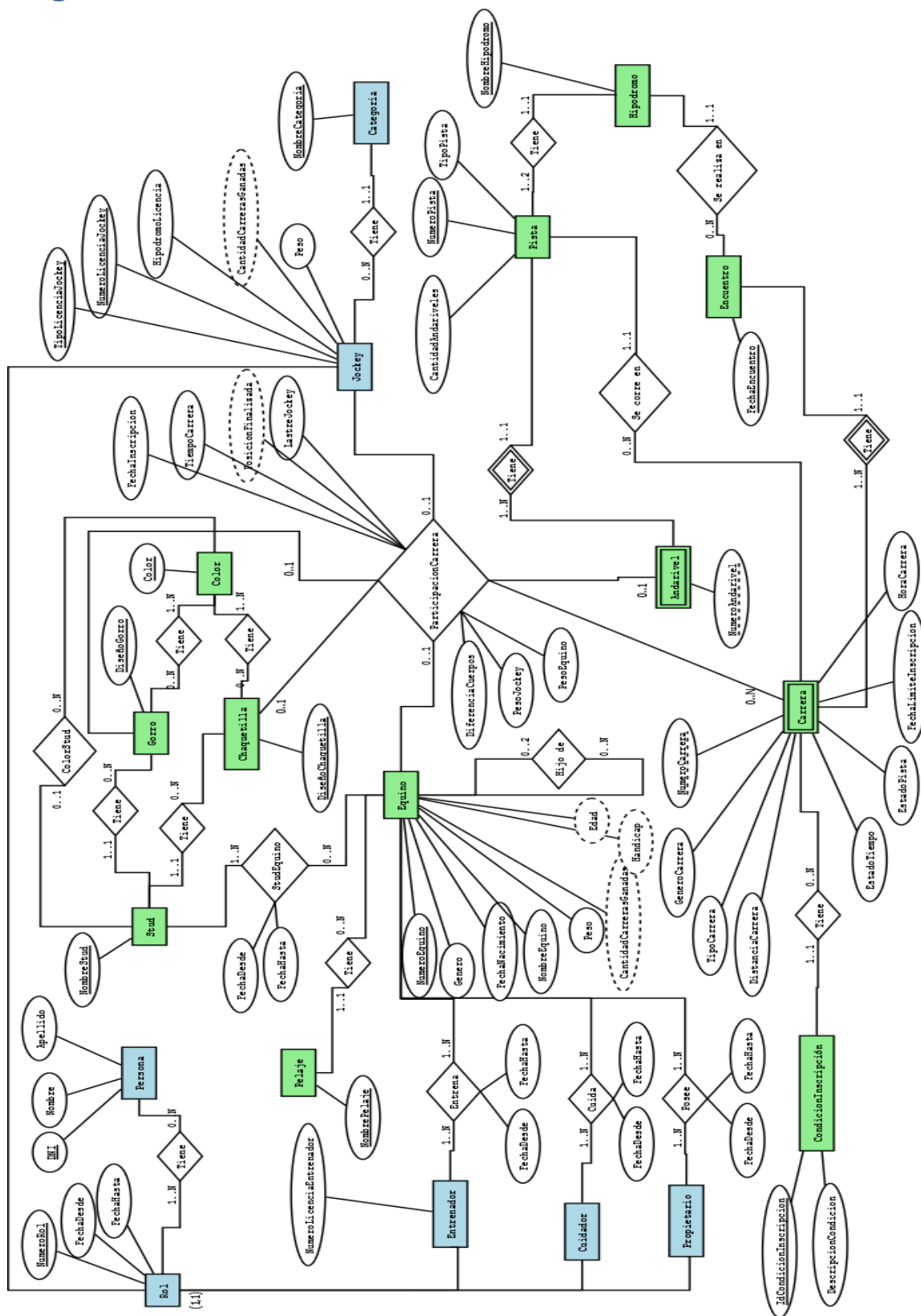
1. Un equino solo puede participar en una única carrera por encuentro.
2. Una carrera puede ser solo de caballos o solo de yeguas.
3. En una carrera solo pueden correr equinos que clasifiquen según el tipo de carrera definido.
4. Para todo Período, habrá que validar que no exista solapamiento entre entre fechas de una misma relación.



5. Un jockey debe utilizar por carrera un gorro y una chaquetilla que pertenezca al stud del caballo que monta.
6. Un equino no puede ser antecesor de sí mismo en el árbol genealógico.
7. La madre de un equino debe ser hembra y el padre macho.
8. El peso de los Jockeys para una inscripción es el máximo posible.



## Diagrama Entidad Interrelación





## Modelo Relacional

A continuación se definen las relaciones correspondientes al modelo presentado en los diagramas entidad interrelación.

Por cada relación se indican sus campos y su PK. Luego, por relación se indican las FKs y validaciones a realizar:

- Hipodromo(NombreHipodromo)
- Pista(NumeroPista, NombreHipodromo, TipoPista, CantidadAndariveles)
  - FKs:
    - (NombreHipodromo)
- Andarivel(NumeroPista, NumeroAndarivel)
  - FKs:
    - (NumeroPista)
  - Validaciones:
    - NumeroAndarivel debe ser mayor igual a 1 y menor igual a CantidadAndariveles de la pista a la que pertenece.
- Encuentro(FechaEncuentro, NombreHipodromo)
  - FKs:
    - (NombreHipodromo)
- CondicionInscripción(IdCondicionInscripcion, DescripcionCondicion)
- Carrera(FechaEncuentro, NumeroCarrera, NumeroPista, GeneroCarrera, TipoCarrera, DistanciaCarrera, EstadoTiempo, EstadoPista, FechaLimiteInscripcion, IdCondicionInscripcion, HoraCarrera)
  - FKs:
    - (FechaEncuentro)
    - (NumeroPista)
    - (IdCondicionInscripcion)
  - Validaciones:
    - El hipódromo del encuentro debe coincidir con el hipódromo de la pista
  - Campos que pueden ser nulos:
    - EstadoTiempo
    - EstadoPista
- Pelaje(NombrePelaje)
- Equino(NumeroEquino, Genero, NombrePelaje, FechaNacimiento, Peso, NombreEquino)
  - FKs:
    - (NombrePelaje)
- HijoDe(NumeroEquino, Padre, Madre)
  - FKs:
    - (Hijo): (NumeroEquino)
    - (Padre): (NumeroEquino)
    - (Madre): (NumeroEquino)
  - Validaciones:
    - Un equino no puede ser antecesor de sí mismo en el árbol genealógico.
    - La madre debe ser una hembra.
    - El padre debe ser un macho.
  - Campos que pueden ser nulos:
    - Padre
    - Madre
- Stud(NombreStud)
- Color(Color)
- ColorStud(NombreStud, Color)
  - FKs:
    - (NombreStud)
    - (Color)
- Gorro(DiseñoGorro, NombreStud)
  - FKs:
    - (NombreStud)
- ColorGorro(DiseñoGorro, Color)
  - FKs:
    - (NombreStud, DiseñoGorro)
    - (Color)
- Chaquetilla(DiseñoChaquetilla, NombreStud)
  - FKs:
    - (NombreStud)



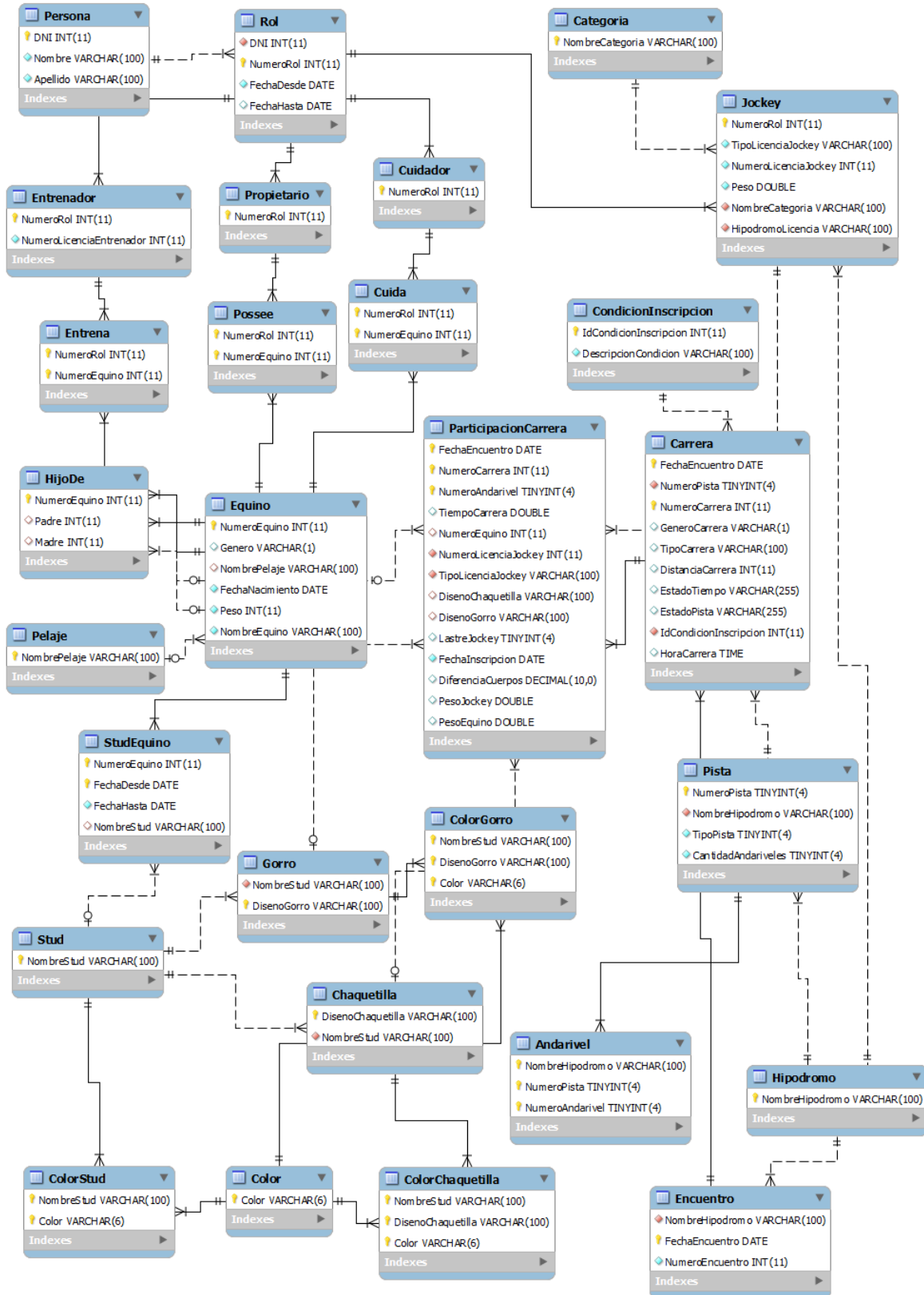
- ColorChaquetilla(DiseñoChaquetilla, Color)
  - FKs:
    - (NombreStud, DiseñoChaquetilla)
    - (Color)
- StudEquino(NumeroEquino, FechaDesde, FechaHasta, NombreStud)
  - Claves Candidatas:
    - (NumeroEquino, FechaDesde)
    - (NumeroEquino, FechaHasta)
  - FKs:
    - (NumeroEquino)
  - Validaciones:
    - No puede haber solapamiento de periodos (FechaDesde, FechaHasta) para un mismo NumeroEquino.
- ParticipacionCarrera(FechaEncuentro, NumeroCarrera, NumeroAndarivel, TiempoCarrera, LastreJockey, NumeroEquino, NumeroLicenciaJockey, TipoLicenciaJockey, DiseñoChaquetilla, DiseñoGorro, FechaInscripcion, DiferenciaCuerpos, PesoJockey, PesoEquino)
  - Claves Candidatas:
    - (FechaEncuentro, NumeroCarrera, NumeroAndarivel)
    - (FechaEncuentro, NumeroCarrera, NumeroEquino)
    - (FechaEncuentro, NumeroCarrera, NumeroLicenciaJockey, TipoLicenciaJockey)
  - Campos que pueden ser nulos:
    - TiempoCarrera
    - LastreJockey
    - DiseñoChaquetilla
    - DiseñoGorro
    - DiferenciaCuerpos
    - PesoJockey
    - PesoEquino
  - FKs:
    - (FechaEncuentro, NumeroCarrera)
    - (NumeroEquino)
    - (NumeroLicenciaJockey, TipoLicenciaJockey)
    - (DiseñoChaquetilla)
    - (DiseñoGorro)
  - Validaciones:
    - En un encuentro (el encuentro al que pertenece la carrera) un equino solo puede participar una única vez.
    - En una carrera (FechaEncuentro, NumeroCarrera) solo pueden participar equinos (NumeroEquino) del mismo género.
    - En una carrera (FechaEncuentro, NumeroCarrera) solo pueden participar equinos (NumeroEquino) para los que se corresponda su Handicap.
    - El Stud del equino, de la chaquetilla y del gorro deben coincidir.
    - El número de andarivel debe existir para la pista en la que se realiza la carrera.
    - Se debe validar que se cumplan los requisitos descriptos según la condición de inscripción (IdCondicionInscripcion) de la carrera (FechaEncuentro, NumeroCarrera).
- Persona(DNI, Nombre, Apellido)
- Rol(NumeroRol, DNI, FechaDesde, FechaHasta)
  - FKs:
    - (DNI)
- Propietario(NumeroRol)
  - FKs:
    - (NumeroRol)
  - Validaciones:
    - Para un mismo DNI, no debe haber solapamiento de periodos en la relación Rol.
- Cuidador(NumeroRol)
  - FKs:
    - (NumeroRol)
  - Validaciones:
    - Para un mismo DNI, no debe haber solapamiento de periodos en la relación Rol.
- Entrenador(NumeroRol, NumeroLicenciaEntrenador)
  - FKs:
    - (NumeroRol)
  - Validaciones:
    - Para un mismo DNI, no debe haber solapamiento de periodos en la relación Rol.
    - Para distintos DNI no puede haber un mismo NumeroLicenciaEntrenador.
- Categoria(NombreCategoria)



- Jockey(NumeroRol, TipoLicenciaJockey, NumeroLicenciaJockey, Peso, NombreCategoria, HipodromoLicencia)
  - Claves Candidatas:
    - (NumeroRol)
    - (TipoLicenciaJockey, NumeroLicenciaJockey)
  - FKs:
    - (NumeroRol)
    - (NombreCategoria)
    - (HipodromoLicencia)
  - Validaciones:
    - Para un mismo DNI, no debe haber solapamiento de períodos en la relación Rol.
    - Para distintos DNI no puede haber un mismo (TipoLicenciaJockey, NumeroLicenciaJockey).
- Posee(NumeroRol, NumeroEquino, FechaDesde, FechaHasta)
  - FKs:
    - (NumeroRol) (respecto de la relación Propietario)
    - (NumeroEquino)
  - Validaciones:
    - El período de posesión debe encontrarse dentro del período del rol.
    - Para un mismo (DNI, NumeroEquino) no debe haber solapamiento de períodos.
- Cuida(NumeroRol, NumeroEquino, FechaDesde, FechaHasta)
  - FKs:
    - (NumeroRol) (respecto de la relación Cuidador)
    - (NumeroEquino)
  - Validaciones:
    - El período de cuidado debe encontrarse dentro del período del rol.
    - Para un mismo (DNI, NumeroEquino) no debe haber solapamiento de períodos.
- Entrena(NumeroRol, NumeroEquino, FechaDesde, FechaHasta)
  - FKs:
    - (NumeroRol) (respecto de la relación Entrenador)
    - (NumeroEquino)
  - Validaciones:
    - El período de entrenamiento debe encontrarse dentro del período del rol.
    - Para un mismo (DNI, NumeroEquino) no debe haber solapamiento de períodos.



## Diagrama del modelo relacional







## Diccionario de datos

```
CREATE TABLE Hipodromo (
    NombreHipodromo varchar(100) NOT NULL,
    PRIMARY KEY (NombreHipodromo)
);

CREATE TABLE Pista (
    NumeroPista tinyint NOT NULL,
    NombreHipodromo varchar(100) NOT NULL,
    TipoPista tinyint NOT NULL,
    CantidadAndariveles tinyint NOT NULL,
    PRIMARY KEY (NumeroPista),
    CONSTRAINT FK_Pista_NombreHipodromo FOREIGN KEY (NombreHipodromo) REFERENCES Hipodromo(NombreHipodromo)
);

CREATE TABLE Andarivel (
    NombreHipodromo varchar(100) NOT NULL,
    NumeroPista tinyint NOT NULL,
    NumeroAndarivel tinyint NOT NULL,
    PRIMARY KEY (NombreHipodromo, NumeroPista, NumeroAndarivel),
    CONSTRAINT FK_Andarivel_NombreHipodromoNumeroPista FOREIGN KEY (NombreHipodromo, NumeroPista) REFERENCES
Pista(NombreHipodromo, NumeroPista)
);

CREATE TABLE Encuentro (
    NombreHipodromo varchar(100) NOT NULL,
    FechaEncuentro date NOT NULL,
    NumeroEncuentro int NOT NULL,
    INDEX (FechaEncuentro),
    PRIMARY KEY (FechaEncuentro),
    UNIQUE (NombreHipodromo, NumeroEncuentro),
    CONSTRAINT FK_Encuentro_NombreHipodromo FOREIGN KEY (NombreHipodromo) REFERENCES Hipodromo(NombreHipodromo)
);

CREATE TABLE CondicionInscripcion (
    IdCondicionInscripcion int NOT NULL,
    DescripcionCondicion varchar(100) NOT NULL,
    PRIMARY KEY(IdCondicionInscripcion)
);

CREATE TABLE Carrera (
    FechaEncuentro date NOT NULL,
    NumeroPista tinyint NOT NULL,
    NumeroCarrera int NOT NULL,
    GeneroCarrera varchar(1) NOT NULL,
    TipoCarrera varchar(100) NOT NULL,
    DistanciaCarrera int NOT NULL,
    EstadoTiempo varchar(255) NULL,
    EstadoPista varchar(255) NULL,
    IdCondicionInscripcion int NOT NULL,
    HoraCarrera time NOT NULL,
    PRIMARY KEY (FechaEncuentro, NumeroCarrera),
    CONSTRAINT FK_Carrera_FechaEncuentro FOREIGN KEY (FechaEncuentro) REFERENCES Encuentro(FechaEncuentro),
    CONSTRAINT FK_Carrera_NumeroPista FOREIGN KEY (NumeroPista) REFERENCES Pista(NumeroPista),
    CONSTRAINT FK_Carrera_IdCondicionInscripcion FOREIGN KEY (IdCondicionInscripcion) REFERENCES
CondicionInscripcion(IdCondicionInscripcion)
);

CREATE TABLE Pelaje (
    NombrePelaje varchar(100) NOT NULL,
    PRIMARY KEY (NombrePelaje)
);

CREATE TABLE Equino (
```



```
NumeroEquino int NOT NULL,
Genero varchar(1) NOT NULL,
NombrePelaje varchar(100) NOT NULL,
NombreEquino varchar(100) NOT NULL,
FechaNacimiento date NOT NULL,
Peso int NOT NULL,
PRIMARY KEY (NumeroEquino),
CONSTRAINT FK_Equino_NombrePelaje FOREIGN KEY (NombrePelaje) REFERENCES Pelaje(NombrePelaje)
);

CREATE TABLE HijoDe (
    NumeroEquino int NOT NULL,
    Padre int NULL,
    Madre int NULL,
    PRIMARY KEY (NumeroEquino),
    CONSTRAINT FK_HijoDe_NumeroEquino FOREIGN KEY (NumeroEquino) REFERENCES Equino(NumeroEquino),
    CONSTRAINT FK_HijoDe_Padre FOREIGN KEY (Padre) REFERENCES Equino(NumeroEquino),
    CONSTRAINT FK_HijoDe_Madre FOREIGN KEY (Madre) REFERENCES Equino(NumeroEquino)
);

CREATE TABLE Stud (
    NombreStud varchar(100) NOT NULL,
    PRIMARY KEY (NombreStud)
);

CREATE TABLE Color (
    Color varchar(6) NOT NULL,
    PRIMARY KEY (Color)
);

CREATE TABLE ColorStud (
    NombreStud varchar(100) NOT NULL,
    Color varchar(6) NOT NULL,
    PRIMARY KEY (NombreStud, Color),
    CONSTRAINT FK_ColorStud_NombreStud FOREIGN KEY (NombreStud) REFERENCES Stud(NombreStud),
    CONSTRAINT FK_ColorStud_Color FOREIGN KEY (Color) REFERENCES Color(Color)
);

CREATE TABLE Gorro (
    NombreStud varchar(100) NOT NULL,
    DisenoGorro varchar(100) NOT NULL,
    PRIMARY KEY (DisenoGorro),
    CONSTRAINT FK_Gorro_NombreStud FOREIGN KEY (NombreStud) REFERENCES Stud(NombreStud)
);

CREATE TABLE ColorGorro (
    NombreStud varchar(100) NOT NULL,
    DisenoGorro varchar(100) NOT NULL,
    Color varchar(6) NOT NULL,
    PRIMARY KEY (NombreStud, DisenoGorro, Color),
    CONSTRAINT FK_ColorGorro_NombreStudDisenoGorro FOREIGN KEY (NombreStud, DisenoGorro) REFERENCES Gorro(NombreStud, DisenoGorro),
    CONSTRAINT FK_ColorGorro_Color FOREIGN KEY (Color) REFERENCES Color(Color)
);

CREATE TABLE Chaquetilla (
    DisenoChaquetilla varchar(100) NOT NULL,
    NombreStud varchar(100) NOT NULL,
    PRIMARY KEY (DisenoChaquetilla),
    CONSTRAINT FK_Chaquetilla_NombreStud FOREIGN KEY (NombreStud) REFERENCES Stud(NombreStud)
);

CREATE TABLE ColorChaquetilla (
    NombreStud varchar(100) NOT NULL,
    DisenoChaquetilla varchar(100) NOT NULL,
```



```
Color varchar(6) NOT NULL,
PRIMARY KEY (NombreStud, DisenoChaquetilla, Color),
CONSTRAINT FK_ColorChaquetilla_NombreStudDisenoChaquetilla FOREIGN KEY (NombreStud, DisenoChaquetilla) REFERENCES
Chaquetilla(NombreStud, DisenoChaquetilla),
CONSTRAINT FK_ColorChaquetilla_Color FOREIGN KEY (Color) REFERENCES Color(Color)
);

CREATE TABLE StudEquino (
NumeroEquino int NOT NULL,
FechaDesde date NOT NULL,
FechaHasta date NOT NULL,
NombreStud varchar(100) NOT NULL,
PRIMARY KEY (NumeroEquino, FechaDesde),
UNIQUE (NumeroEquino, FechaHasta),
CONSTRAINT FK_StudEquino_NumeroEquino FOREIGN KEY (NumeroEquino) REFERENCES Equino(NumeroEquino),
CONSTRAINT FK_StudEquino_NombreStud FOREIGN KEY (NombreStud) REFERENCES Stud(NombreStud)
);

CREATE TABLE Persona (
DNI int NOT NULL,
Nombre varchar(100) NOT NULL,
Apellido varchar (100) NOT NULL,
PRIMARY KEY (DNI)
);

CREATE TABLE Rol (
DNI int NOT NULL,
NumeroRol int NOT NULL,
FechaDesde date NOT NULL,
FechaHasta date NULL,
PRIMARY KEY (NumeroRol),
CONSTRAINT FK_Rol_DNI FOREIGN KEY (DNI) REFERENCES Persona(DNI)
);

CREATE TABLE Propietario (
NumeroRol int NOT NULL,
PRIMARY KEY (NumeroRol),
CONSTRAINT FK_Propietario_NumeroRol FOREIGN KEY (NumeroRol) REFERENCES Rol(NumeroRol)
);

CREATE TABLE Cuidador (
NumeroRol int NOT NULL,
PRIMARY KEY (NumeroRol),
CONSTRAINT FK_Cuidador_NumeroRol FOREIGN KEY (NumeroRol) REFERENCES Rol(NumeroRol)
);

CREATE TABLE Entrenador (
NumeroRol int NOT NULL,
NumeroLicenciaEntrenador int NOT NULL,
PRIMARY KEY (NumeroRol),
UNIQUE (NumeroLicenciaEntrenador),
CONSTRAINT FK_Entrenador_NumeroRol FOREIGN KEY (NumeroRol) REFERENCES Rol(NumeroRol)
);

CREATE TABLE Categoria (
NombreCategoria varchar(100) NOT NULL,
PRIMARY KEY (NombreCategoria)
);

CREATE TABLE Jockey (
NumeroRol int NOT NULL,
TipolicenciaJockey varchar(100) NOT NULL,
NumeroLicenciaJockey int NOT NULL,
Peso double NOT NULL,
NombreCategoria varchar(100) NOT NULL,
```



```
Hipodromolicencia varchar(100) NOT NULL,
UNIQUE (TipoLicenciaJockey, NumeroLicenciaJockey),
PRIMARY KEY (NumeroRol),
CONSTRAINT FK_Jockey_NumeroRol FOREIGN KEY (NumeroRol) REFERENCES Rol(NumeroRol),
CONSTRAINT FK_Jockey_NombreCategoria FOREIGN KEY (NombreCategoria) REFERENCES Categoria(NombreCategoria),
CONSTRAINT FK_Jockey_HipodromoLicencia FOREIGN KEY (Hipodromolicencia) REFERENCES Hipodromo(NombreHipodromo)
);

CREATE TABLE Posee (
NumeroRol int NOT NULL,
NumeroEquino int NOT NULL,
PRIMARY KEY (NumeroRol, NumeroEquino),
CONSTRAINT FK_Posee_NumeroRol FOREIGN KEY (NumeroRol) REFERENCES Propietario(NumeroRol),
CONSTRAINT FK_Posee_NumeroEquino FOREIGN KEY (NumeroEquino) REFERENCES Equino (NumeroEquino)
);

CREATE TABLE Cuida (
NumeroRol int NOT NULL,
NumeroEquino int NOT NULL,
PRIMARY KEY (NumeroRol, NumeroEquino),
CONSTRAINT FK_Cuida_NumeroRol FOREIGN KEY (NumeroRol) REFERENCES Cuidador(NumeroRol),
CONSTRAINT FK_Cuida_NumeroEquino FOREIGN KEY (NumeroEquino) REFERENCES Equino (NumeroEquino)
);

CREATE TABLE Entrena (
NumeroRol int NOT NULL,
NumeroEquino int NOT NULL,
PRIMARY KEY (NumeroRol, NumeroEquino),
CONSTRAINT FK_Entrena_NumeroRol FOREIGN KEY (NumeroRol) REFERENCES Entrenador(NumeroRol),
CONSTRAINT FK_Entrena_NumeroEquino FOREIGN KEY (NumeroEquino) REFERENCES Equino (NumeroEquino)
);

CREATE TABLE ParticipacionCarrera (
FechaEncuentro date NOT NULL,
NumeroCarrera int NOT NULL,
NumeroAndarivel tinyint NOT NULL,
TiempoCarrera double NULL,
NumeroEquino int NULL,
NumeroLicenciaJockey int NOT NULL,
TipoLicenciaJockey varchar(100) NOT NULL,
DisenoChaquetilla varchar(100) NULL,
DisenoGorro varchar(100) NULL,
LastreJockey tinyint NULL,
FechaInscripcion date NOT NULL,
DiferenciaCuerpos decimal NULL,
PesoJockey double NULL,
PesoEquino double NULL,
PRIMARY KEY (FechaEncuentro, NumeroCarrera, NumeroAndarivel),
UNIQUE (FechaEncuentro, NumeroCarrera, NumeroLicenciaJockey, TipoLicenciaJockey),
UNIQUE (FechaEncuentro, NumeroCarrera, NumeroEquino),
CONSTRAINT FK_ParticipacionCarrera_Carrera FOREIGN KEY (FechaEncuentro, NumeroCarrera) REFERENCES Carrera(FechaEncuentro,
NumeroCarrera),
CONSTRAINT FK_ParticipacionCarrera_NumeroEquino FOREIGN KEY (NumeroEquino) REFERENCES Equino(NumeroEquino),
CONSTRAINT FK_ParticipacionCarrera_Jockey FOREIGN KEY (TipoLicenciaJockey, NumeroLicenciaJockey) REFERENCES
Jockey(TipoLicenciaJockey, NumeroLicenciaJockey),
CONSTRAINT FK_ParticipacionCarrera_DisenChaquetilla FOREIGN KEY (DisenoChaquetilla) REFERENCES
Chaquetilla(DisenoChaquetilla),
CONSTRAINT FK_ParticipacionCarrera_DisenGorro FOREIGN KEY (DisenoGorro) REFERENCES Gorro(DisenoGorro)
);
```



## Consultas

### Condición A

```
/*Yeguas de 5 y 6 años que no hayan ganado. Peso 5 años 57 kilos,  
6 años 55 kilos.*/  
select 'Pasa condicion'  
from Equino as E  
where  
    E.NumeroEquino = @NumeroEquino and  
    E.Genero = 'H' and  
    (  
        /*Peso 5 años 57 kilos*/  
        ( FLOOR(DATEDIFF(NOW(), E.FechaNacimiento) / 365) = 5 and  
        exists(  
            select *  
            from Jockey J1  
            where  
                J1.TipoLicenciaJockey = @TipoLicenciaJockey and  
                J1.NumeroLicenciaJockey = @NumeroLicenciaJockey and  
                J1.Peso <= 57  
        ) ) or  
        /*6 años 55 kilos.*/  
        ( FLOOR(DATEDIFF(NOW(), E.FechaNacimiento) / 365) = 6 and  
        exists(  
            select *  
            from Jockey J1  
            where  
                J1.TipoLicenciaJockey = @TipoLicenciaJockey and  
                J1.NumeroLicenciaJockey = @NumeroLicenciaJockey and  
                J1.Peso <= 55  
        ) )  
    ) and  
/*que no hayan ganado*/  
not exists(  
    select 'Ganó'  
    from ParticipacionCarrera PC0  
    where PC0.NumeroEquino = E.NumeroEquino and  
    /*0 caballos tienen mejor tiempo que el caballo en estudio  
    en la carrera en estudio: Esto significa que ganó.*/  
    0 = (  
        select count(*)  
        from ParticipacionCarrera PC1  
        where  
            PC1.FechaEncuentro = PC0.FechaEncuentro and  
            PC1.NumeroCarrera = PC0.NumeroCarrera and  
            PC1.NumeroEquino != PC0.NumeroEquino and  
            PC1.TiempoCarrera > PC0.TiempoCarrera  
    )  
)
```

### Condición B

```
/* Todo caballo de 5 y 6 años que no hayan ganado. Peso 5 años 57 kilos,  
6 años 55 kilos.  
(Para jockeys y jockeys aprendices con licencia de Palermo o San Isidro que  
hayan ganado hasta 20 carreras desde el 1o de abril de 2012).*/
```

```
SELECT  
    'Pasa condicion'  
FROM  
    Equino, Jockey  
WHERE  
    Equino.NumeroEquino = @NumeroEquino  
    AND Jockey.NumeroLicenciaJockey = @NumeroLicenciaJockey
```



```

AND Jockey.TipoLicenciaJockey = @TipoLicenciaJockey
AND Equino.NumeroEquino NOT IN (
    SELECT ParticipacionCarrera2.NumeroEquino
    FROM ParticipacionCarrera AS ParticipacionCarrera2
    WHERE TiempoCarrera = (
        SELECT MAX(ParticipacionCarrera3.TiempoCarrera)
        FROM ParticipacionCarrera AS ParticipacionCarrera3
        WHERE
            ParticipacionCarrera2.FechaEncuentro = ParticipacionCarrera3.FechaEncuentro
            AND ParticipacionCarrera2.NumeroCarrera = ParticipacionCarrera3.FechaEncuentro
    )
)
AND (Jockey.Peso > 57 AND FLOOR(DATEDIFF(NOW(), Equino.FechaNacimiento) / 365) = 5 OR (Jockey.peso > 55 AND
FLOOR(DATEDIFF(NOW(), Equino.FechaNacimiento) / 365) = 6))
AND (Jockey.HipodromoLicencia = "Palermo" OR Jockey.HipodromoLicencia != "San Isidro")
AND Jockey.NombreCategoria = "Aprendiz"
AND EXISTS (
    SELECT ParticipacionCarrera.NumeroLicenciaJockey, ParticipacionCarrera.TipoLicenciaJockey
    FROM ParticipacionCarrera, Carrera
    WHERE
        TiempoCarrera = (
            SELECT MIN(ParticipacionCarrera4.TiempoCarrera)
            FROM ParticipacionCarrera AS ParticipacionCarrera4
            WHERE
                ParticipacionCarrera.FechaEncuentro = ParticipacionCarrera4.FechaEncuentro
                AND ParticipacionCarrera.NumeroCarrera = ParticipacionCarrera4.FechaEncuentro
        )
        AND ParticipacionCarrera.NumeroLicenciaJockey = Jockey.NumeroLicenciaJockey
        AND ParticipacionCarrera.TipoLicenciaJockey = Jockey.TipoLicenciaJockey
        AND Carrera.NumeroCarrera = ParticipacionCarrera.NumeroCarrera
        AND Carrera.FechaEncuentro = ParticipacionCarrera.FechaEncuentro
        AND Carrera.FechaEncuentro <= date('2013-04-12')
    GROUP BY ParticipacionCarrera.NumeroLicenciaJockey, ParticipacionCarrera.TipoLicenciaJockey
    HAVING COUNT(*) <= 20
)
)

```

## Condición C

/\*  
c) Yeguas de 4 y 5 años perdedoras, que habiendo corrido no hayan clasificado 2a, 3a ó 4a en Los Hipódromos de Palermo ó San Isidro en sus tres últimas actuaciones. (Pueden participar las S.P.C. que hayan corrido esta categoría siempre y cuando no hayan obtenido esos puestos en condicionales u otro tipo de carreras, también quedan excluidos los ejemplares que en sus últimas cinco (5) actuaciones hayan participado en carreras clásicas). Peso 4 años 57 kilos; 5 años 55 kilos.  
\*/

```

SELECT
    'Pasa condicion'
FROM
    Equino, Jockey
WHERE
    Equino.NumeroEquino = @NumeroEquino
    AND Jockey.NumeroLicenciaJockey = @NumeroLicenciaJockey
    AND Jockey.TipoLicenciaJockey = @TipoLicenciaJockey
    AND Equino.Genero = 'H'
    AND (Jockey.Peso > 57 AND FLOOR(DATEDIFF(NOW(), Equino.FechaNacimiento) / 365) = 4 OR (Jockey.peso > 55 AND
FLOOR(DATEDIFF(NOW(), Equino.FechaNacimiento) / 365) = 5))
    AND NOT EXISTS(
        SELECT
            'Termino 1º, 2º, 3º ó 4º',
            'En alguno de los últimos 3 encuentros',
            'En Palermo o San Isidro'
        FROM ParticipacionCarrera PC0, Encuentro EN
        WHERE
            PC0.NumeroEquino = Equino.NumeroEquino
            AND PC0.FechaEncuentro = EN.FechaEncuentro
    )

```



```

AND EN.NombreHipodromo IN ('Palermo', 'San Isidro')
AND 3 >= (
    SELECT COUNT(*)
    FROM ParticipacionCarrera PC2
    WHERE
        PC2.FechaEncuentro >= PC0.FechaEncuentro AND
        PC2.NumeroEquino = PC0.NumeroEquino
)
AND 3 >= (
    SELECT COUNT(*)
    FROM ParticipacionCarrera PC1
    WHERE
        PC1.FechaEncuentro = PC0.FechaEncuentro AND
        PC1.NumeroCarrera = PC0.NumeroCarrera AND
        PC1.NumeroEquino != PC0.NumeroEquino AND
        PC1.TiempoCarrera > PC0.TiempoCarrera
)
)
)

```

## Condición D

```

/*
Caballos de 4 y 5 años
(con exclusión de yeguas)
perdedores
que habiendo corrido no hayan clasificado 2º, 3º ó 4º en Los Hipódromos de
Palermo ó San Isidro en sus tres últimas actuaciones. (Pueden participar Los
S.P.C. que hayan corrido esta categoría siempre y cuando no hayan obtenido
esos puestos en condicionales u otro tipo de carreras, también quedan
excluidos Los ejemplares que en sus
últimas cinco (5) actuaciones hayan participado en carreras clásicas).
Peso 4 años 57 kilos; 5 años 55 kilos.
*/

set @NumeroEquino = 0;
set @TipoLicenciaJockey = '';
set @NumeroLicenciaJockey= '';

select 'Pasa condición'
from Equino as E
where
    E.NumeroEquino = @NumeroEquino and
    (
        /*Peso 4 años 57 kilos*/
        ( FLOOR(DATEDIFF(NOW(), E.FechaNacimiento) / 365) = 4 and
        exists(
            select *
            from Jockey J1
            where
                J1.TipoLicenciaJockey = @TipoLicenciaJockey and
                J1.NumeroLicenciaJockey = @NumeroLicenciaJockey and
                J1.Peso <= 57
            ) ) or
        /*5 años 55 kilos.*/
        ( FLOOR(DATEDIFF(NOW(), E.FechaNacimiento) / 365) = 5 and
        exists(
            select *
            from Jockey J1
            where
                J1.TipoLicenciaJockey = @TipoLicenciaJockey and
                J1.NumeroLicenciaJockey = @NumeroLicenciaJockey and
                J1.Peso <= 55
            ) )
        ) and
        /*(con exclusión de yeguas)*/
        E.Genero = 'M' and

```



```

/*no hayan clasificado 1º, 2º, 3º o 4º en Los Hipódromos de Palermo ó
San Isidro en sus tres últimas actuaciones.*/
not exists(
  select
    'Termino 1º, 2º, 3º o 4º',
    'En alguno de los últimos 3 encuentros',
    'En Palermo o San Isidro'
  from ParticipacionCarrera PC0, Encuentro EN
  where
    PC0.NumeroEquino = E.NumeroEquino and
    PC0.FechaEncuentro = EN.FechaEncuentro and
    EN.NombreHipodromo in ('Palermo', 'San Isidro') and
    /*La cantidad de carreras en las que participó el Equino con fecha
    igual o posterior a la fecha del encuentro de la carrera estudiada sea
    como mucho 3*/
    3 >= (
      select count(*)
      from ParticipacionCarrera PC2
      where
        PC2.FechaEncuentro >= PC0.FechaEncuentro and
        PC2.NumeroEquino = PC0.NumeroEquino
    ) and
    /*Como mucho 3 caballos tienen mejor tiempo que el caballo en estudio
    en la carrera en estudio: Esto significa que está entre Los mejores 4*/
    3 >= (
      select count(*)
      from ParticipacionCarrera PC1
      where
        PC1.FechaEncuentro = PC0.FechaEncuentro and
        PC1.NumeroCarrera = PC0.NumeroCarrera and
        PC1.NumeroEquino != PC0.NumeroEquino and
        PC1.TiempoCarrera > PC0.TiempoCarrera
    )
  ) and
  /*quedan excluidos los ejemplares que en sus últimas cinco (5) actuaciones
  hayan participado en carreras clásicas)*/
not exists(
  select
    'Participo en carrera Clasica',
    'En sus últimas cinco (5) actuaciones'
  from ParticipacionCarrera PC10, Carrera CA
  where
    PC10.FechaEncuentro = CA.FechaEncuentro and
    PC10.NumeroCarrera = CA.NumeroCarrera and
    PC10.NumeroEquino = E.NumeroEquino and
    CA.TipoCarrera = 'Clásica' and
    PC10.FechaEncuentro = CA.FechaEncuentro and
    /*La cantidad de carreras en las que participó el Equino con fecha
    igual o posterior a la fecha del encuentro de la carrera estudiada sea
    como mucho 5*/
    5 >= (
      select count(*)
      from ParticipacionCarrera PC11
      where
        PC11.FechaEncuentro >= PC10.FechaEncuentro and
        PC11.NumeroEquino = PC10.NumeroEquino
    )
  )
)

```

## Condición E

```

/*
e) Yeguas de 3 años que no hayan ganado. Peso 56 kilos. (No podrán
inscribirse las ejemplares pertenecientes a las caballerizas que ocupen
los 12 primeros puestos en la estadística por sumas ganadas desde el 1o de
marzo de 2012)

```





```

*/

set @NumeroEquino = 0;
set @TipoLicenciaJockey = '';
set @NumeroLicenciaJockey= '';

select 'Pasa condición'
from Equino as E
where
    E.NumeroEquino = @NumeroEquino and
    /*Peso 3 años 56 kilos*/
    ( FLOOR(DATEDIFF(NOW(), E.FechaNacimiento) / 365) = 3 and
    exists(
        select *
        from Jockey J1
        where
            J1.TipoLicenciaJockey = @TipoLicenciaJockey and
            J1.NumeroLicenciaJockey = @NumeroLicenciaJockey and
            J1.Peso <= 56
        ) ) and
    /*yeguas*/
    E.Genero = 'H' and
    /*que no hayan ganado*/
    not exists(
        select 'Primer puesto'
        from ParticipacionCarrera PC0
        where
            PC0.NumeroEquino = E.NumeroEquino and
            /*0 caballos tienen mejor tiempo que el caballo en estudio
            en la carrera en estudio: Esto significa que ganó.*/
            not exists(
                select 'Tiempo mejor de otro Equino'
                from ParticipacionCarrera PC1
                where
                    PC1.FechaEncuentro = PC0.FechaEncuentro and
                    PC1.NumeroCarrera = PC0.NumeroCarrera and
                    PC1.NumeroEquino != PC0.NumeroEquino and
                    PC1.TiempoCarrera > PC0.TiempoCarrera
            )
        ) and
    /*
    No podrán inscribirse las ejemplares pertenecientes a las caballerizas
    que ocupen los 12 primeros puestos en la estadística por sumas ganadas
    desde el 1o de marzo de 2012
    Es decir el equino no pertenece a una caballeriza que se encuentre
    entre las primeras 12.
    Es decir la caballeriza a la que pertenece el equino tiene por encima
    12 caballerizas o más que lo superan en puntos.
    */
    12 <= (
        /*Se obtiene la cantidad de caballerizas que superan la
        caballeriza del equino */
        select count(SE1.NombreStud)
        from ParticipacionCarrera PC2, StudEquino SE1
        where
            PC2.FechaEncuentro >= date('2012-03-01') and
            PC2.NumeroEquino = SE1.NumeroEquino and
            PC2.FechaEncuentro >= SE1.FechaDesde and
            PC2.FechaEncuentro <= SE1.FechaHasta and
            /*No se tiene en cuenta el stud del equino analizado.*/
            SE1.NombreStud not in (
                select SE2.NombreStud
                from StudEquino SE2
                where SE2.NumeroEquino = E.NumeroEquino
            ) and

```



```

/*y que el equino obtenido sea ganador (no se discrimina por genero)*/
exists(
  select 'Primer puesto'
  from ParticipacionCarrera PC3
  where
    PC3.NumeroEquino = PC2.NumeroEquino and
    /*0 caballos tienen mejor tiempo que el caballo en estudio
    en la carrera en estudio: Esto significa que ganó.*/
    not exists(
      select 'Tiempo mejor de otro Equino'
      from ParticipacionCarrera PC4
      where
        PC4.FechaEncuentro = PC3.FechaEncuentro and
        PC4.NumeroCarrera = PC3.NumeroCarrera and
        PC4.NumeroEquino != PC3.NumeroEquino and
        PC4.TiempoCarrera > PC3.TiempoCarrera
    )
)
group by SE1.NombreStud
having
  /*Se pide que la cantidad de carreras ganadas por stud sea mayor que la
  cantidad de carreras ganada por el stud del equino analizado.*/
  count(*) > (
    select count(*)
    from ParticipacionCarrera PC5, StudEquino SE2
    where
      PC5.FechaEncuentro >= date('2012-03-01') and
      PC5.NumeroEquino = SE2.NumeroEquino and
      PC5.FechaEncuentro >= SE2.FechaDesde and
      PC5.FechaEncuentro <= SE2.FechaHasta and
      /*se tiene en cuenta solo el stud del equino analizado.*/
      SE1.NombreStud in (
        select SE2.NombreStud
        from StudEquino SE2
        where SE2.NumeroEquino = E.NumeroEquino
      ) and
      /*y que el equino obtenido sea ganador (no se discrimina por genero)*/
      exists(
        select 'Primer puesto'
        from ParticipacionCarrera PC6
        where
          PC6.NumeroEquino = PC5.NumeroEquino and
          /*0 caballos tienen mejor tiempo que el caballo en estudio
          en la carrera en estudio: Esto significa que ganó.*/
          not exists(
            select 'Tiempo mejor de otro Equino'
            from ParticipacionCarrera PC7
            where
              PC7.FechaEncuentro = PC6.FechaEncuentro and
              PC7.NumeroCarrera = PC6.NumeroCarrera and
              PC7.NumeroEquino != PC6.NumeroEquino and
              PC7.TiempoCarrera > PC6.TiempoCarrera
          )
        )
      )
  )
)
)
)

```

## Condición F

*/\*Caballos de 3 años (con exclusión de yeguas) que no hayan ganado.  
Peso 56 kilos\*/*

```

set @NumeroEquino = 0;
set @TipoLicenciaJockey = '';
set @NumeroLicenciaJockey= '';

```



```
select 'Pasa condición'
from Equino as E
where
    E.NumeroEquino = @NumeroEquino and
    /*Peso 3 años 56 kilos*/
    ( FLOOR(DATEDIFF(NOW(), E.FechaNacimiento) / 365) = 3 and
    exists(
        select *
        from Jockey J1
        where
            J1.TipolicenciaJockey = @TipolicenciaJockey and
            J1.NumeroLicenciaJockey = @NumeroLicenciaJockey and
            J1.Peso <= 56
        ) and
    /*(con exclusión de yeguas)*/
    E.Genero = 'M' and
    /*que no hayan ganado*/
    not exists(
        select 'Primer puesto'
        from ParticipacionCarrera PC0
        where
            PC0.NumeroEquino = E.NumeroEquino and
            /*0 caballos tienen mejor tiempo que el caballo en estudio
            en la carrera en estudio: Esto significa que ganó.*/
            not exists(
                select 'Tiempo mejor de otro Equino'
                from ParticipacionCarrera PC1
                where
                    PC1.FechaEncuentro = PC0.FechaEncuentro and
                    PC1.NumeroCarrera = PC0.NumeroCarrera and
                    PC1.NumeroEquino != PC0.NumeroEquino and
                    PC1.TiempoCarrera > PC0.TiempoCarrera
            )
        )
    )
```

## Reporte

```
SELECT
    Andarivel.NumeroAndarivel,
    Equino.NombreEquino,
    ParticipacionCarrera.PesoEquino,
    PersonaJockey.Nombre,
    Jockey.Peso,
    ParticipacionCarrera.DiferenciaCuerpos,
    PersonaCuidador.Nombre,
    Stud.NombreStud
FROM
    Equino, Andarivel, Jockey, Cuida, Persona AS PersonaJockey, Rol AS RolCuidador, Rol AS RolJockey, Persona AS
    PersonaCuidador, Stud, ParticipacionCarrera, StudEquino
WHERE
    Equino.NumeroEquino = ParticipacionCarrera.NumeroEquino
    AND (
        Jockey.NumeroLicenciaJockey = ParticipacionCarrera.NumeroLicenciaJockey
        AND Jockey.TipolicenciaJockey = ParticipacionCarrera.TipolicenciaJockey
    )
    AND RolJockey.DNI = PersonaJockey.DNI
    AND Jockey.NumeroRol = RolJockey.NumeroRol
    AND RolCuidador.DNI = PersonaCuidador.DNI
    AND (RolCuidador.NumeroRol = Cuida.NumeroRol AND RolCuidador.FechaDesde IS NULL)
    AND Cuida.NumeroEquino = Equino.NumeroEquino
    AND (Andarivel.NumeroAndarivel = ParticipacionCarrera.NumeroAndarivel)
    AND (StudEquino.NumeroEquino = Equino.NumeroEquino AND StudEquino.FechaHasta IS NULL)
    AND StudEquino.NombreStud = Stud.NombreStud
```