

**Enunciado**

Un parque ecológico quiere informatizar la gestión de su sistema de información para poder obtener datos más concluyentes sobre las especies migratorias que se establecen en su territorio. El parque desea clasificar todas las especies animales y contabilizar el número de individuos de cada especie que se establecen en el territorio en cada época del año. Se desea que la base de datos almacene datos de nuevas especies y nuevos individuos de cada especie animal. Para cada especie, se requiere el nombre de la especie, características generales de un individuo tipo y sus periodos migratorios. Para cada individuo se tendrá en cuenta el peso, dimensiones, y un código que identifique de forma única, irá almacenada en un pequeño dispositivo con una batería autónoma implantada en el animal, y que servirá para detectar cuando el individuo sale o entra en el territorio gracias a unas torretas de control instaladas en el perímetro del parque. Estas torretas informan vía inalámbrica al sistema de las idas y venidas de cada individuo. De esta manera, se pueden contabilizar sus migraciones y los posibles descontroles que sufran en el periodo migratorio. Toda la información sobre migraciones de individuos de determinadas especies será enviada cada tres meses a un grupo de expertos biológicos, encargados de hacer una valoración sobre futuros periodos migratorios y posibles alteraciones del comportamiento de las especies. De los biólogos, se quiere almacenar, el nombre, la dirección, provincia y fecha de su titulación.

**Entidades y atributos.****Especies.**

- Cod. Especie.
- Nombre de la especie.
- Características generales.
- Tipo.
- Periodos migratorios.

**Individuos.**

- Cod. Individuo.
- Peso.
- Dimensiones.
- Código

**Torreta.**

- Cod. Torreta.
- Ubicación.
- Nombre.

**Biólogos.**

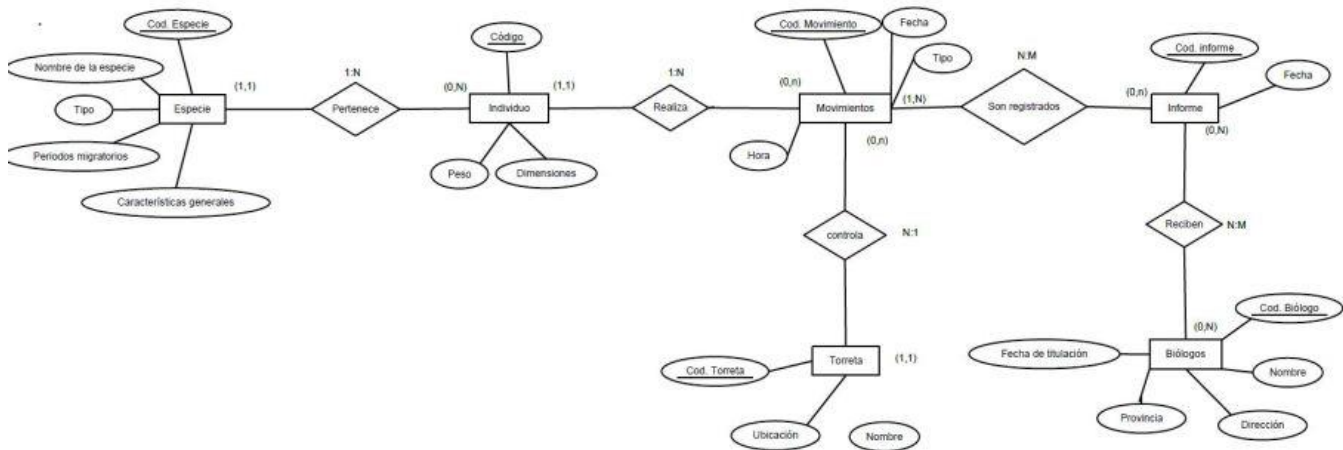
- Cod. Biólogo.
- Nombre.
- Dirección.
- Provincia.
- Fecha de titulación.

**Movimiento.**

- Cod. Movimiento.
- Fecha.
- Hora.
- Tipo.

**Informe**

- Cod. Informe
- Fecha

**Modelo entidad-relación.**

**Relación especie pertenece a individuo:** He puesto 0,n porque una especie puede estar compuesta mínimo por 0 individuos y máximo por varios, y 1,1 por que un individuo solo puede pertenecer a una especie.

**Relación individuo realiza movimientos:** He puesto 0,n por que un individuo puede realizar mínimo 0 movimientos y máximo varios movimientos, por otro lado he puesto 1,1 por que un movimiento solo puede ser realizado por un individuo.

**Relación torreta controla movimientos:** He puesto 1,1 por que un movimiento solo puede ser capturado por una torreta, por otro lado e puesto 0,n porque una torreta puede capturar mínimo 0 movimientos y máximo varios.

**Relación movimientos son registrados en informe:** He puesto 1,n porque en un informe pueden ser registrados mínimo 1 y máximo varios movimientos.

**Relación informe reciben biólogos:** he puesto 0,n por que los biólogos pueden recibir mínimo 0 y máximo varios informes.

Paso a tablas.

- ESPECIE (COD.ESPECIE, NOMBREDELAESPECIE, TIPO, PERIDOSMIGRATORIOS, CARACTERISTICASGENERALES)
- INDIVIDUO (CODIGO.INDIVIDUO, COD.ESPECIE, PESO, DIMENSIONES)
- TORRETA (COD.TORRETA, UBICACIÓN, NOMBRE)
- MOVIMIENTOS (COD.MOVIMIENTO, CODIGO.INDIVIDUO, COD.TORRETA, FECHA, HORA, TIPO)
- INFORME (COD.INFORME, FECHA)
- BIOLOGOS (COD.BIOLOGO, NOMBRE, DIRECCION, PROVINCIA, FECHADETITULACION)
- SON REGISTRADOS (COD.MOVIMIENTO, COD.INFORME)
- RECIBEN (COD.INFORME, COD.BIOLOGO)

**Código SQL.**

```
create table especie (  
    cod.especie varchar (30) not null,  
    nombreespecie varchar (20) not null,  
    tipo varchar (10) not null,  
    periodosmigratorios varchar (30),  
    caracteristicasgenerales varchar (30) not null,  
    primary key (cod.especie)  
);  
  
create table individuo (  
    codigo.individuo varchar (20) not null,  
    cod.especie varchar (30) not null,  
    peso varchar (5) not null,  
    dimensiones varchar (20),  
    primary key (codigoindividuo),  
    foreign key (cod.especie) references especie(cod.especie)  
);  
  
create table torreta (  
    cod.torreta varchar (20) not null,  
    ubicacion varchar (30),  
    nombre varchar (20) not null,  
    primary key (cod.torreta)  
);  
  
create table movimientos (  
    cod.movimiento varchar (20) not null,  
    codigo.individuo varchar (20) not null,  
    cod.torreta varchar (20) not null,  
    fecha date not null,  
    hora varchar (10) not null,  
    tipo varchar (10) not null,  
    primary key (cod.movimiento),  
    foreign key (codigo.individuo) references individuo(codigo.individuo),  
    foreign key (cod.torreta) references torreta(cod.torreta)  
);  
  
create table informe (  
    cod.informe varchar (20) not null,  
    fecha date not null,  
    primary key (cod.informe)  
);  
  
create table biologos (  
    cod.biologo varchar (20) not null,  
    nombre varchar (15) not null,  
    direccion varchar (20) not null,  
    provincia varchar (15),  
    fechadetitulacion date not null,  
);
```

```
create table sonregistrados (  
    cod.movimiento varchar (20) not null,  
    cod.informe varchar (20) not null,  
    foreign key (cod.movimiento) references movimientos(cod.movimiento),  
    foreign key (cod.informe) references informe(cod.informe)  
);  
  
create table reciben (  
    cod.informe varchar (20) not null,  
    cod.biologo varchar (20) not null,  
    foreign key (cod.informe) references informe(cod.informe),  
    foreign key (cod.biologo) references biologos(cod.biologo)  
);
```

**Ejercicio 2**

Se va a crear una página web de contactos en el dominio BuscoPareja.net. Cuando un usuario se registra en el sistema, se almacenan sus datos personales, concretamente su email, su nombre, dirección, ciudad, país, su sexo y orientación sexual, su foto y una password que utilizará junto con su email para acceder al sistema. El usuario a continuación utilizará junto con su email para acceder al sistema. El usuario a continuación rellena una lista de preferencias o gustos. De cada gusto o preferencia se almacenará el tipo (Deporte, Música, Evento Social), la fecha de la última que hizo una actividad de ese estilo, y si le gustaría o no que se le pareara tuviera la misma preferencia. A parte de esta información se organizan citas entre los contactos en distintas ubicaciones. De Se desea registrar las citas entre los contactos almacenando quién se cita con quién, en qué lugar y a qué hora, y si la cita fracasó.

**Entidades y atributos.****Usuario.**

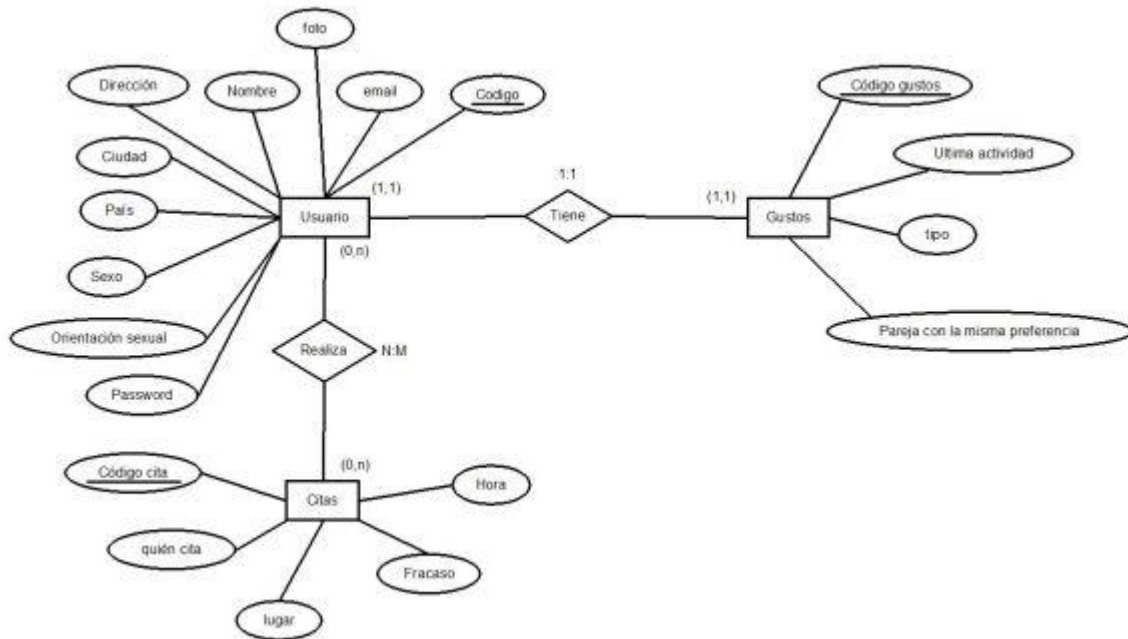
- Código.
- Email
- Nombre
- Dirección.
- Ciudad.
- País.
- Sexo.
- Orientación sexual
- Foto
- Password

**Gustos.**

- Código gustos.
- Tipo.
- Última actividad.
- Pareja con la misma preferencia.

**Citas.**

- Código cita.
- Quién cita.
- Lugar.
- Hora
- Fracaso.

**Modelo entidad-relación.**

- **Relación usuario tiene gustos:** He puesto 1,1 por que según lo que e entendido en el enunciado todos los usuarios deben elegir un gusto, mínimo uno y máximo uno.
- **Relación usuario realiza citas:** He puesto 0,n por que un usuario mínimo puede realizar 0 citas y máximo varias citas.

**Paso a tablas.**

- USUARIO (CÓDIGO, EMAIL, NOMBRE, DIRECCION, CIUDAD, PAIS, SEXO, ORIENTACIONSEXUAL, PASSWORD)
- GUSTOS (CODIGO, ULTIMAACTIVIDAD, TIPO, PAREJA CON LA MISMA PREFERENCIA)
- CITAS (CÓDIGO, QUIEN CITA, LUGAR, HORA, FRACASO)
- REALIZA (CODIGO-GUSTOS, CODIGO-CITA)

**Código SQL.**

```
create table usuario(  
    codigo varchar (20) not null,  
    email varchar (30) not null,  
    nombre varchar (15) not null,  
    direccion varchar (30) not null,  
    ciudad varchar (15) not null,  
    pais varchar (15) not null,  
    sexo varchar (10) not null,  
    orientacionsexual varchar (10),  
    password varchar (30) not null,  
    primary key (codigo)  
);  
  
create table gustos(  
    codigo varchar (20) not null,  
    ultimaactividad varchar (20) not null,  
    tipo varchar (20) not null,  
    parejaconlamismapreferencia varchar (7) not null,  
    primary key (codigo)  
);  
  
create table citas (  
    codigo varchar (20) not null,  
    quien cita varchar (20),  
    lugar varchar (20),  
    hora varchar (7),  
    fracaso varchar (2),  
    primary key (codigo)  
);  
  
create table realiza (  
    codigo-gustos varchar (20) not null,  
    codigo-cita varchar (20) not null,  
    foreign key (codigo-gustos) references gustos(codigo),  
    foreign key (codigo-cita) references citas(codigo)  
);
```