### **Consulta sobre una tabla**

Considere los esquemas de las bases de datos de **Voluntarios** (**unc\_esq\_voluntarios**) y de **Películas** (**unc\_esq\_peliculas**) cuyos diagramas se muestran en la Figura 1 y 2 respectivamente.

La base de datos de **Voluntarios** representa la información de personas voluntarias que trabajan para instituciones realizando determinadas tareas. De las instituciones se conoce su dirección y el país y continente al cual pertenecen. De cada voluntario se conoce su voluntario coordinador y las tareas e instituciones que ha realizado a lo largo de su historial.

La base de datos de **Películas** mantiene información de un centro de distribución de las películas que comercializa, los distribuidores y las entregas que deben realizar o ha realizado a cada video-club.

Se incluyen datos de la empresa productora de cada película, los distribuidores de las mismas (clasificados como nacionales o internacionales) y además los empleados que trabajan para cada distribuidor, las tareas a realizar y a qué departamento pertenecen.

**Bibliografía recomendada:** Documentacion PostgreSQL a consultar: 7. Queries

[https://www.postgresql.org/docs/11/static/queries.html](https://www.postgresql.org/docs/9.5/static/queries.html)

***Identifique el esquema, la tabla y resuelva las siguientes consultas SQL:***

1. Seleccione el identificador y nombre de todas las instituciones que son Fundaciones.(V)

SELECT id\_institucion, nombre\_institucion FROM unc\_esq\_voluntario.institucion;

1. Seleccione el identificador de distribuidor, identificador de departamento y nombre de todos los departamentos.(P)

SELECT id\_distribuidor, id\_departamento, nombre FROM unc\_esq\_peliculas.departament;

1. Muestre el nombre, apellido y el teléfono de todos los empleados cuyo id\_tarea sea 7231, ordenados por apellido y nombre.(P)

SELECT apellido, nombre, telefono FROM unc\_esq\_peliculas.empleado WHERE id\_tarea='7231' ORDER BY apellido ASC , nombre ASC;

1. Muestre el apellido e identificador de todos los empleados que no cobran porcentaje de comisión.(P)

SELECT apellido, id\_empleado FROM unc\_esq\_peliculas.empleado WHERE porc\_comision IS NULL;

1. Muestre el apellido y el identificador de la tarea de todos los voluntarios que no tienen coordinador.(V)

SELECT apellido, id\_tarea FROM unc\_esq\_voluntario.voluntario WHERE id\_coordinador IS NULL;

1. Muestre los datos de los distribuidores internacionales que no tienen registrado teléfono. (P)

SELECT \* FROM unc\_esq\_peliculas.internacional JOIN unc\_esq\_peliculas.distribuidor ON telefono IS NULL;

1. Muestre los apellidos, nombres y mails de los empleados con cuentas de gmail y cuyo sueldo sea superior a $ 1000. (P)

SELECT apellido, nombre, e\_mail FROM unc\_esq\_peliculas.empleado WHERE e\_mail LIKE '%@gmail.com' AND sueldo>1000;

1. Seleccione los diferentes identificadores de tareas que se utilizan en la tabla empleado. (P)

SELECT id\_tarea FROM unc\_esq\_peliculas.empleado;

1. Muestre el apellido, nombre y mail de todos los voluntarios cuyo teléfono comienza con +51. Coloque el encabezado de las columnas de los títulos 'Apellido y Nombre' y 'Dirección de mail'. (V)

SELECT apellido, nombre, e\_mail FROM unc\_esq\_voluntario.voluntario WHERE telefono LIKE '+51%';

1. Hacer un listado de los cumpleaños de todos los empleados donde se muestre el nombre y el apellido (concatenados y separados por una coma) y su fecha de cumpleaños (solo el día y el mes), ordenado de acuerdo al mes y día de cumpleaños en forma ascendente. (P)

SELECT nombre||', '||apellido||'cumple el dia: ', *EXTRACT*(MONTH FROM fecha\_nacimiento) ||'-'|| *EXTRACT*(DAY FROM fecha\_nacimiento) FROM unc\_esq\_peliculas.empleado ORDER BY fecha\_nacimiento ASC;

1. Recupere la cantidad mínima, máxima y promedio de horas aportadas por los voluntarios nacidos desde 1990. (V)

SELECT *MAX*(horas\_aportadas), *MIN*(horas\_aportadas), *count*(nro\_voluntario), *sum*(horas\_aportadas)/*count*(nro\_voluntario) PROMEDIO

FROM unc\_esq\_voluntario.voluntario WHERE *EXTRACT*(YEAR FROM fecha\_nacimiento)>=1990;

SELECT *MAX*(horas\_aportadas), *MIN*(horas\_aportadas), *AVG* (horas\_aportadas) PROMEDIO

FROM unc\_esq\_voluntario.voluntario WHERE *EXTRACT*(YEAR FROM fecha\_nacimiento)>=1990;

1. Listar la cantidad de películas que hay por cada idioma. (P)

SELECT idioma ||':' IDIOMA, *count*(titulo) CANTIDAD FROM unc\_esq\_peliculas.pelicula GROUP BY idioma;

1. Calcular la cantidad de empleados por departamento. (P)

SELECT id\_departamento ||': ' DEPARTAMENTO, *count*(id\_empleado) CANT\_EMPLEADOS FROM unc\_esq\_peliculas.empleado GROUP BY id\_departamento;

1. Mostrar los códigos de películas que han recibido entre 3 y 5 entregas. (veces entregadas, NO cantidad de películas entregadas).

SELECT codigo\_pelicula, cantidad FROM unc\_esq\_peliculas.renglon\_entrega WHERE cantidad>=3 AND cantidad<=5;

1. ¿Cuántos cumpleaños de voluntarios hay cada mes? (V) 

SELECT *EXTRACT*(MONTH FROM fecha\_nacimiento) MES, *count*(fecha\_nacimiento) FROM unc\_esq\_voluntario.voluntario GROUP BY *EXTRACT*(MONTH FROM fecha\_nacimiento);

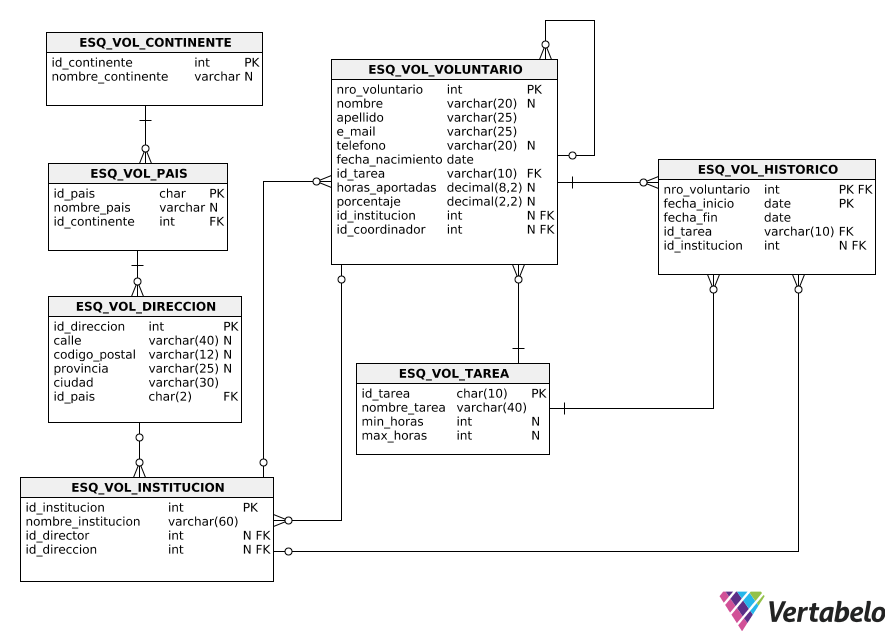
1. ¿Cuáles son las 2 instituciones que más voluntarios tienen? (V)

SELECT id\_institucion, *count*(id\_institucion) CANTIDAD FROM unc\_esq\_voluntario.voluntario GROUP BY id\_institucion ORDER BY *count*(id\_institucion) DESC LIMIT 2;

1. ¿Cuáles son los id de ciudades que tienen más de un departamento? 

SELECT id\_ciudad FROM unc\_esq\_peliculas.departamento GROUP BY id\_ciudad HAVING *COUNT*(id\_departamento)>1;

**FIGURA 1**

****

**FIGURA 2**

