

# Introducción a Bases de Datos y SQL

Resolución de proyecto integrador - 5.1

## Resolución de la etapa 5.1

1. Utilizando la tabla **PRODUCTOS\_NEPTUNO**, obtener una lista de todos aquellos productos cuyo **precio supere el precio promedio**. Esta debe contener todos los campos de la tabla. Por último, ordenar **alfabéticamente** el resultado según los nombres de los productos.

```
SELECT * FROM PRODUCTOS_NEPTUNO
WHERE PRECIOUNIDAD >
(SELECT AVG(PRECIOUNIDAD) FROM PRODUCTOS_NEPTUNO)
ORDER BY NOMBREPRODUCTO;
```

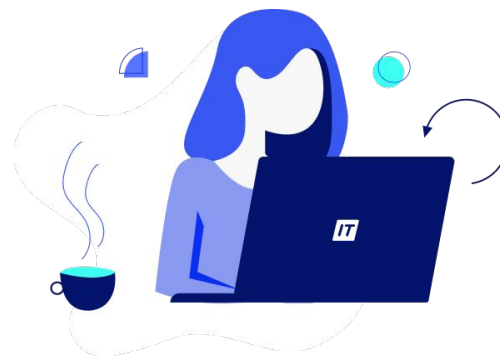


2. Toma la tabla **PRODUCTOS\_NEPTUNO** y obtén una lista de todos aquellos productos cuyo **precio** sea superior al producto **más caro** de la tabla **PRODUCTOS\_SUSPENDIDOS**. Esta debe contener **todos los campos** de la tabla. Luego, ordenar el resultado de **mayor a menor** de acuerdo con los precios obtenidos.

```
SELECT * FROM PRODUCTOS_NEPTUNO
WHERE PRECIO >
(SELECT MAX(PRECIO) FROM PRODUCTOS_SUSPENDIDOS)
ORDER BY PRECIO DESC;
```

3. Utilizando la tabla **VARONES**, obtener una lista de todos aquellos bebés que hayan nacido con una cantidad de **semanas de gestación menor** que el bebé de **sexo indeterminado con menor gestación**. La lista debe mostrar **todos los campos** de la tabla.

```
SELECT * FROM VARONES  
WHERE SEMANAS <  
(SELECT MIN(SEMANAS) FROM INDETERMINADOS);
```

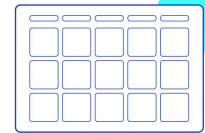


4. Dada la tabla **PRODUCTOS\_NEPTUNO**, obtener una lista de todos los productos cuyo nombre comience con la **inicial del apellido del empleado cuyo IDEMPLEADO es el número 8**. Esta debe mostrar **todos los campos** de la tabla **PRODUCTOS\_NEPTUNO** y se debe ordenar **alfabéticamente** según los nombres de los **productos**.

```
SELECT * FROM PRODUCTOS_NEPTUNO
WHERE LEFT(NOMBREPRODUCTO, 1) =
(SELECT LEFT(NOMBRE_EMPLEADO, 1) FROM EMPLEADOS
WHERE IDEMPLEADO = 8)
ORDER BY NOMBREPRODUCTO;
```

5. Utilizando la tabla **PRODUCTOS\_NEPTUNO**, obtener una lista de todos los productos que pertenezcan al **proveedor con el ID más alto**. La lista debe mostrar todos los campos de la tabla **PRODUCTOS\_NEPTUNO** y debe ordenarse **alfabéticamente** por los nombres de los **productos**.

```
SELECT * FROM PRODUCTOS_NEPTUNO
WHERE IDPROVEEDOR =
(SELECT MAX(IDPROVEEDOR) FROM PROVEEDORES)
ORDER BY NOMBREPRODUCTO;
```



6. Dada la tabla **PRODUCTOS\_NEPTUNO**, extraer una lista de todos aquellos productos que pertenezcan a la categoría **BEBIDAS** y cuyo **precio sea superior al producto más caro de la categoría CONDIMENTOS**. La lista debe mostrar **todos los campos** de la tabla.

```
SELECT * FROM PRODUCTOS_NEPTUNO
WHERE NOMBRECATEGORIA = 'BEBIDAS' AND
PRECIOUNIDAD >
(SELECT MAX(PRECIOUNIDAD) FROM PRODUCTOS_NEPTUNO
WHERE NOMBRECATEGORIA = 'CONDIMENTOS');
```

7. A partir de la tabla **MUJERES**, obtener una lista de todas aquellas **bebas** que hayan nacido de **madres con una edad superior a la madre más longeva** que figure en la tabla **VARONES**. La lista debe mostrar **todos los campos** de la tabla **MUJERES**.

```
SELECT * FROM MUJERES  
WHERE EDAD_MADRE >  
(SELECT MAX(EDAD_MADRE) FROM VARONES);
```





8. Utilizando la tabla **CLIENTES\_NEPTUNO**, extraer una lista de todos los clientes que hayan realizado **compras por un cargo superior a 500 dólares**. La lista debe mostrar los campos **NOMBRECOMPANIA**, **CIUDAD** y **PAÍS** y debe estar ordenada **alfabéticamente** por los nombres de las **compañías**.

```
SELECT * FROM PRODUCTOS_NEPTUNO
WHERE NOMBRECATEGORIA = 'BEBIDAS' AND
PRECIOUNIDAD >
(SELECT MAX(PRECIOUNIDAD) FROM PRODUCTOS_NEPTUNO
WHERE NOMBRECATEGORIA = 'CONDIMENTOS');
```

**¡Sigamos  
trabajando!**