



Time to complete: 101:27 Points: 6/10

1. ¿Cuántos clientes hay en San Justo?

1 / 1 pt

```
SELECT COUNT(*) AS cliente_En_SanJusto
FROM Clientes c JOIN Localidad l ON c.cod_loc=l.cod_loc
WHERE l.descripcion='San Justo';
```

2. Dar de alta en la base de datos al técnico Pablo Marquez, con el legajo 1234 y un sueldo de \$10000.

1 / 1 pt

```
INSERT INTO Tecnico,
(legajo, nombre, sueldo)
VALUES (1234, "Pablo Marquez", 10000);
```

3. Efectuar un reporte de servicios del mes de enero de 2015, indicando por cada una de ellos: fecha de servicio, patente, nombre de cliente, descripción de la localidad en donde vive, nombre del técnico y costo de reparación.

0 / 1 pt

```
SELECT s.fecha, s.patente, c.nombre, l.descripcion, t.nombre, s.costo
FROM servicio JOIN tecnico ON s.legajo_tecnico = t.legajo
JOIN auto ON s.patente = a.patente
JOIN cliente ON a.cod_cli = c.cod_cli
JOIN localidad ON c.cod_loc = l.cod_loc
WHERE s.fecha = '2015-01'
GROUP BY s.fecha, s.patente, c.nombre, l.descripcion, t.nombre, s.costo;
```

Incorrecto uso de filtro de fechas y agrupamiento.

4. Listar patente y cantidad de servicios realizados durante el año 2013. Sólo mostrar aquellos registros donde dicha cantidad sea superior a dos.

0.5 / 1 pt

```
SELECT a.patente, COUNT(*) as CANT_SERV
FROM Auto a JOIN Servicio s ON a.patente=s.patente
WHERE s.fecha BETWEEN '20130101' AND '20131231'
GROUP BY s.patente
HAVING COUNT(*)>2;
```

Si la fecha es de tipo datetime el between va a dejar afuera a los servicios realizados el último día. Para que utiliza la tabla auto?

5. Bonificar un 10% el costo de las reparaciones de aquellos autos cuyo modelo contenga la palabra "Focus".

0 / 1 pt

```
UPDATE s.costo
SET s.costo= s.costo - (costo*0,10)
WHERE patente IN (SELECT * FROM AUTO
WHERE LIKE '%FOCUS%');
```

Debe indicar la tabla sobre la que quiere realizar el update. En la subconsulta debe obtener solamente las patentes.

6. Indicar el costo promedio de servicio por localidad de cliente (mostrando la descripción de la localidad, no el código).

0.5 / 1 pt

```
SELECT l.descripcion, AVG(s.costo)
FROM Servicio s JOIN Auto a ON s.patente = a.patente
JOIN Cliente c ON a.cod_cli=c.cod_cli
JOIN Localidad l ON c.cod_loc=l.cod_loc
GROUP BY l.descripcion;
```

Debe agrupar también por la clave de localidad.

7. Listar el nombre y sueldo de todos los técnicos que no hayan efectuado servicios con costo superior a \$1000.

0 / 1 pt

```
SELECT t.nombre, t.sueldo
FROM Tecnico t JOIN Servicio s ON t.legajo_tecnico=s.legajo_tecnico
WHERE s.costo<1000;
```

Esta solución no tiene en cuenta a técnicos que no hayan efectuado ningún servicio.

8. Mostrar patente y modelo, de aquellos autos que fueron atendidos por todos los técnicos. 1 / 1 pt

```
SELECT a.patente, a.modelo  
FROM Auto a  
WHERE NOT EXISTS  
( SELECT 1  
FROM Tecnico t  
WHERE NOT EXISTS ( SELECT 1 FROM Servicio s  
WHERE a.patente = s.patente AND s.legajo_tecnico = t.legajo_tecnico));
```

9. Eliminar los servicios efectuados por técnicos que cobran más de \$5000. 1 / 1 pt

```
DELETE FROM Servicio  
WHERE legajo_tecnico IN ( SELECT t.legajo  
FROM Tecnico t  
WHERE t.sueldo > 5000);
```

10. Escriba la sentencia DDL de creación de la tabla Tecnico, determinando los tipos de datos y restricciones de integridad según su criterio. 1 / 1 pt

```
CREATE TABLE Tecnico(  
legajo SMALLINT NOT NULL PRIMARY KEY,  
nombre VARCHAR(40) NOT NULL,  
sueldo SMALLINT NOT NULL);  
  
INSERT INTO Tecnico (legajo, nombre, sueldo) VALUES  
(1,'Leandro', 10000);
```