

Práctica 8

1. Inicie el programa de Aplicación **MySQL Workbench**.
2. Verifique si están corriendo el servicio **MySQL**, en caso contrario oprima el botón **Start** en el Control Panel de XAMPP.
3. Junto con el documento **PDF** de esta práctica descargue el archivo que viene anexo llamado **abogado.sql**, en la carpeta **Mis documentos**.
4. En la ventana de **MySQL Workbench** haga clic sobre el recuadro gris que dice **localhost**, debajo del título **MySQL Connections**.
5. Establecida la conexión, abra el menú **Server** y seleccione la opción **Data import**.
6. Seleccione la opción **Import from Self-Contained file**.
7. Oprima el botón de la derecha para localizar la carpeta **Mis documentos** y seleccionar el archivo **abogado.sql**, oprima el botón **Abrir**.
8. En **default schema** no necesita crear ni seleccionar ninguno, pues automáticamente se va a crear, oprima el botón de **Start Import**.
9. Si es necesario proporcione la contraseña de **root** y oprima **OK**.
10. Para saber el total de horas que tiene asignadas el abogado **Luis Domínguez**, puede escribir la siguiente consulta en la que mediante una subconsulta se obtiene la clave de este abogado:

```
SELECT SUM(horas) AS TotalHoras FROM asignacion where ClaveA = (SELECT ClaveA  
FROM abogado where NombreA="Luis Domínguez");
```

11. Ahora, si se desea saber los nombres de los casos en los que participa el abogado **Luis Domínguez** tiene que escribir la siguiente consulta con subconsultas anidadas:

```
SELECT NombreCaso FROM caso where NumCaso IN (SELECT NumCaso FROM asignacion  
where ClaveA = (SELECT ClaveA FROM abogado where NombreA="Luis Domínguez"));
```

12. Ejecute la consulta y observe que **Luis Domínguez** participa en 4 diferentes casos, por esta razón en la consulta se tuvo que utilizar una subconsulta mediante la cláusula **IN**, verifique que otra alternativa equivalente sería utilizar **= ANY** en vez de **IN**.
13. Ahora se requiere listar los nombres de los abogados que trabajan en el departamento que dirige **Carolina Torres**.

```
SELECT NombreA FROM abogado where ClaveD = (SELECT ClaveD FROM departamento  
where Responsable ="Carolina Torres");
```

14. En la siguiente consulta se pide el número total de horas asignadas para cada clave de abogado, para ello se agrupan las asignaciones por clave y se hace la suma de horas:

```
SELECT ClaveA, SUM(Horas) FROM asignacion GROUP BY ClaveA;
```
15. Para saber el nombre del empleado tiene el mayor número de horas asignadas se debe hacer una subconsulta que indique las sumas de horas por clave de abogado, luego otra consulta que extraiga la clave del abogado que tiene el mayor número de horas, para luego, en la consulta principal obtener el nombre correspondiente a esa clave.

```
SELECT NombreA FROM abogado where ClaveA IN (SELECT ClaveA FROM asignacion
GROUP BY ClaveA HAVING SUM(Horas) >= ALL (SELECT SUM(Horas) FROM asignacion
GROUP BY ClaveA));
```

16. En este caso conviene usar la cláusula **IN** para prevenir un error en caso de que hubiera dos o más empleados empatados con el mismo número máximo de horas.
17. En esta consulta se obtienen los nombres de los empleados que no son responsables de algún departamento:

```
SELECT NombreA FROM abogado where NombreA NOT IN (SELECT Responsable FROM
departamento);
```

18. Ahora se pide hacer una consulta que busque si hay una diferencia del departamento que dirige un responsable respecto al departamento en el que trabaja.

```
SELECT * FROM departamento where (Responsable, ClaveD) NOT IN (SELECT NombreA,
ClaveD FROM abogado);
```

19. Este ejemplo se le llama *subconsulta de registro* y solamente en algunas versiones de gestores de SQL es posible hacer, en caso de que no fuera posible, tendríamos que usar otra estrategia que se conoce como *subconsulta correlacionada*, tal como se ve a continuación:

```
SELECT * FROM departamento d where Responsable NOT IN (SELECT NombreA FROM
abogado a WHERE a.ClaveD=d.ClaveD);
```

20. Observe que por conveniencia se han usado los alias d y a para las tablas departamento y abogado respectivamente. Además en la subconsulta se hace referencia a un valor de la consulta principal, de ahí el nombre de este tipo de consultas.
21. Genere una ventana para consultas y escriba la siguiente consulta para obtener los datos del abogado **54** y de sus asignaciones, dado que se trata de dos tablas es necesario el empleo de la cláusula INNER JOIN:

```
SELECT * FROM abogado a INNER JOIN asignacion s ON a.ClaveA=s.ClaveA where
a.ClaveA = 54;
```

22. Ejecute la consulta y observe que hay 4 registros conteniendo los datos personales y las asignaciones del empleado Luis Domínguez.
23. Genere otra ventana para consultas y escriba la siguiente consulta que muestra el nombre del empleado y el total de horas asignadas a cada uno:

```
SELECT NombreA, SUM(Horas) FROM abogado a INNER JOIN asignacion s ON
e.ClaveA=s.ClaveA GROUP BY a.ClaveA;
```

24. También se puede obtener el total de horas por caso de manera semejante, pero agregue la condición de que los casos tengan más de 80 horas asignadas en total y más de 2 asignaciones:

```
SELECT NombreCaso, SUM(Horas) FROM caso c INNER JOIN asignacion s ON
c.NumCaso=s.NumCaso GROUP BY c.NumCaso HAVING SUM(Horas)>80 AND COUNT(*)>2;
```

25. Considere ahora la siguiente consulta con subconsultas para obtener los nombres de los empleados que tienen una tarifa más alta que **Carlos Cuevas**:

```
SELECT NombreA, Tarifa FROM abogado WHERE Tarifa > (SELECT Tarifa FROM abogado
```

```
WHERE NombreA = 'Carlos Cuevas');
```

26. Ahora haga una consulta equivalente empleando INNER JOIN, observe que la condición puesta en la cláusula ON es una desigualdad:

```
SELECT a.NombreA, a.Tarifa FROM abogado a INNER JOIN abogado f ON e.Tarifa > f.Tarifa WHERE f.NombreA = 'Carlos Cuevas';
```

27. Genere otra ventana para consultas y escriba la siguiente consulta para determinar los casos que no tienen asignaciones:

```
SELECT NombreCaso FROM caso WHERE NumCaso NOT IN (SELECT NumCaso FROM asignacion);
```

28. Es posible utilizar una sentencia LEFT JOIN para conseguir el mismo resultado:

```
SELECT NombreCaso FROM caso c LEFT JOIN asignacion s ON c.NumCaso = s.NumCaso WHERE s.NumCaso IS NULL;
```

29. Al usar la condición **s.NumCaso IS NULL** se están solicitando los casos sin asignaciones.

30. Genere otra ventana para consultas y escriba la siguiente consulta para saber quiénes son los abogados con asignaciones en el caso “**Casa Blanca**”.

```
SELECT NombreA FROM abogado WHERE ClaveA IN (SELECT ClaveA FROM asignacion WHERE NumCaso = (SELECT NumCaso FROM caso WHERE NombreCaso='Casa Blanca'));
```

31. Ahora la debe reescribir mediante el empleo de INNER JOINs:

```
SELECT a.NombreA FROM (abogado a INNER JOIN asignacion s ON a.ClaveA=s.ClaveA) INNER JOIN Caso c ON c.NumCaso=s.NumCaso WHERE NombreCaso='Casa Blanca'));
```

32. Construya las consultas SQL que a continuación se solicitan, usando Subconsultas:

- Obtener clave, nombre y teléfono de los abogados que trabajan en el Departamento llamado **Litigios**.
- Listar los nombres de los abogados que están asignados al Caso **Dr. Simi**, ordenados alfabéticamente.
- Listar el nombre del Departamento y del Responsable para los departamentos en que haya abogados tienen más de una Asignación con más de 30 horas.
- Obtener la clave del caso, clave del abogado y las horas asignadas para todas las asignaciones en algún caso cuyo nombre inicia con la letra **L**.
- Obtener clave del caso y las horas asignadas para cada una de las asignaciones del abogado **Luis Domínguez**.
- Obtener la lista del nombre de los abogados ordenados alfabéticamente y la ciudad donde radican, para todos los abogados que tienen una Tarifa mayor que la tarifa más alta de los abogados de Uruapan.

33. Construya la consulta que a continuación se solicita, usando HAVING:

- Listar la clave del abogado y el total de horas que tiene asignadas, para aquéllos que sumen más de **50** horas en total por todas sus asignaciones.

34. Construya las consultas SQL que a continuación se solicitan, usando subconsultas y HAVING:

- Obtener el Nombre del Departamento en donde se paga la mayor cantidad total a

sus abogados (multiplique la tarifa por las horas asignadas).

- i) Obtener el Nombre del caso y la duración en días, para todos casos que tengan más de dos Asignaciones. (Sugerencia: utilice la función DATEDIFF)

35. Realice las consultas SQL que a continuación se solicitan usando INNER JOIN.

- j) Listar el nombre del caso, el nombre del abogado y las horas asignadas para todos los casos iniciados el año **2014**.
- k) Listar los nombres de los casos que no tengan más de dos asignaciones.
- l) Listar el nombre del abogado y el del departamento donde trabaja para todos los abogados que participan en el caso **Lago Contaminado**.
- m) Listar todos los datos de los **abogados** que trabajen en el departamento que dirige **Adela Quezada**.
- n) Listar el nombre del abogado y el total de horas que tiene asignadas, ordenados en forma descendente.
- o) Listar el nombre de los departamentos que tengan dos o más abogados asignados a algún caso cuyo nombre inicie con la letra **P**.
- p) Liste los nombres de los abogados que NO sean de la misma ciudad y NO trabajen en el mismo departamento que ningún otro abogado.

36. Muestre las consultas al instructor.

37. Fin de la Práctica.