

**Universidad Rafael Landívar**  
**Facultad de Ingeniería**  
**Base de Datos I**



**SEGUNDO PROYECTO**  
**CONSULTAS**

**Andrea Guadalupe Alvarez Ramos 1095718**  
**José Carlos Girón Márquez 1064718**

**Guatemala, 06 de mayo del 2020**

## CONSULTAS

1. El listado de empleados que han fungido como gerentes de turno de la sede donde más socios se han inscrito.

```
SELECT COUNT(C.Ubicacion) AS CANTIDAD, C.Ubicacion
INTO #CantidadxSede
FROM Contrato C
GROUP BY C.Ubicacion

SELECT * FROM #CantidadxSede

SELECT P.Codigo, P.Nombre + ' ' + P.Apellido AS NombreCompleto
FROM PERSONAL P INNER JOIN
(
    SELECT R1.Ubicacion
    FROM
    (
        SELECT * FROM #CantidadxSede CS
    ) R1
    WHERE R1.CANTIDAD IN
    (
        SELECT MAX(R2.CANTIDAD)
        FROM
        (
            SELECT * FROM #CantidadxSede
        ) R2
    )
) R3
ON P.CodigoSede=R3.Ubicacion
WHERE P.Codigo=P.CodigoJefe
```

2. La información de los socios que realizaron más de 15 visitas en el último mes a cualquier sede.

```
SELECT Codigo, Nombre, Apellido, NIT, Direccion, Telefono, Cantidad_Visitas
FROM Cliente C
INNER JOIN
(
    SELECT COUNT(V.CodigoCliente) AS Cantidad_Visitas, V.CodigoCliente
    FROM Visita V
    WHERE DATEPART(month, V.Fecha) = DATEPART(month, GETDATE()) - 1
          AND DATEPART(year, V.Fecha) = DATEPART(year, GETDATE())
    GROUP BY V.CodigoCliente
    HAVING COUNT(V.CodigoCliente) > 15
) R1
ON C.Codigo=R1.CodigoCliente
```

**3. El detalle del empleado de atención al cliente que más veces se ha reubicado entre sedes.**

```
SELECT CodigoEmpleado, COUNT(1) as Cantidad
INTO #CantidadReubicación
FROM Reubicacion
Inner Join Personal On Personal.Codigo = CodigoEmpleado
WHERE Puesto = 'Servicio al Cliente'
GROUP BY CodigoEmpleado

SELECT *
FROM Personal P
INNER JOIN
(
    SELECT R1.CodigoEmpleado
    FROM
    (
        SELECT * FROM #CantidadReubicación
    ) R1
    WHERE R1.CANTIDAD
    IN
    (
        SELECT MAX(R2.CANTIDAD)
        FROM
        (
            SELECT * FROM #CantidadReubicación
        ) R2
    )
) R3
ON P.Codigo=R3.CodigoEmpleado
```

**4. El promedio de deuda total pendiente por sede de los contratos terminados.**

```
SELECT S.Codigo, ROUND(AVG(TP.Precio), 2) AS DeudaPromedio
FROM Contrato C
INNER JOIN CancelacionContrato CC
    ON C.Numero=CC.Numero
INNER JOIN TipoContrato TP
    ON C.TipoContrato=TP.Nombre
INNER JOIN Sede S
    ON C.Ubicacion=S.Codigo
GROUP BY S.Codigo
```

## 5. El detalle de los 5 ejercicios que más veces aparecen en los entrenamientos.

```
SELECT Nombre, Descripcion, Complejidad, TipoMaquina, Cantidad
FROM Ejercicio EJ INNER JOIN
(
    SELECT TOP 5 COUNT(R.NombreEjercicio) AS Cantidad, R.NombreEjercicio
    FROM Rutina R
    INNER JOIN Ejercicio EJ
        ON R.NombreEjercicio=EJ.Nombre
    GROUP BY R.NombreEjercicio
    ORDER BY CANTIDAD DESC
) R1
ON EJ.Nombre=R1.NombreEjercicio
```

## 6. Los datos del instructor con menos certificaciones que ha dado más clases grupales en cada sede.

```
SELECT DPI, Nombre, Apellido, Fecha_Nacimiento
FROM Instructor I
INNER JOIN
(
    SELECT TOP 1 COUNT(CL.DPI_Instructor) AS Total_Clases, CL.DPI_Instructor, Cantidad_Certificaciones
    FROM CLASE CL
    INNER JOIN
    (
        SELECT COUNT(CER.DPI_Instructor) AS Cantidad_Certificaciones, CER.DPI_Instructor
        FROM Instructor I
        INNER JOIN Certificacion CER
            ON I.DPI=CER.DPI_Instructor
        GROUP BY CER.DPI_Instructor
    ) R1
    ON CL.DPI_Instructor=R1.DPI_Instructor
    GROUP BY CL.DPI_Instructor, Cantidad_Certificaciones
    ORDER BY Cantidad_Certificaciones ASC, Total_Clases DESC
) R2
ON I.DPI=R2.DPI_Instructor
```

## 7. El listado de cuántas máquinas hay en cada sede por músculo principal

```
SELECT DISTINCT M.NombreSede, R1.MusculoPrincipal, R1.CANTIDAD
FROM Maquina M
INNER JOIN
(
    SELECT COUNT(M.MusculoPrincipal) AS CANTIDAD, M.MusculoPrincipal
    FROM Maquina M
    GROUP BY M.MusculoPrincipal
) R1
ON M.MusculoPrincipal=R1.MusculoPrincipal
ORDER BY M.NombreSede
```

**8. Top 3 de los empleados de ventas que más han devengado en comisiones de ventas por nuevas inscripciones.**

```
SELECT TOP 3 C.CodigoVendedor, ROUND(SUM(TC.Precio) * 0.1, 2) AS Comisiones
FROM Contrato C
INNER JOIN Personal P
    ON C.CodigoVendedor=P.Codigo
INNER JOIN TipoContrato TC
    ON C.TipoContrato=TC.Nombre
WHERE P.Puesto='Ventas'
GROUP BY C.CodigoVendedor
ORDER BY Comisiones DESC
```

**9. El número de socios activos por membresía que hay en cada sede.**

```
SELECT C.Ubicacion, C.TipoContrato, COUNT(C.CodigoCliente) AS NoSocios
FROM Contrato C
LEFT JOIN CancelacionContrato CC
    ON C.Numero=CC.Numero
WHERE CC.Numero IS NULL
GROUP BY C.TipoContrato, C.Ubicacion
```

**10. Identificar el mes en que más socios terminan su contrato.**

```
Select Top 1 DATEPART(month, Fecha_Fin) As Mes
From CancelacionContrato
Group by DATEPART(month, Fecha_Fin)
Order by COUNT(1) DESC
```

**11. El promedio de duración de los entrenamientos asignados por sede.**

```
Select CodigoSede, AVG(Convert (int, Duracion)) As Promedio
From Catalogo
Inner Join Entrenamiento ON Entrenamiento.Nombre = Catalogo.NombreEntrenamiento
Group by CodigoSede
```

**12. Los detalles de la sede que ha impartido más clase grupales pero tiene menos duchas.**

```
Select Top 1 Sede.Codigo, Nombre, Direccion, Telefono, MaxCapacidad,
    Lockers, Duchas, COUNT(Clase.Codigo) As Clases
From Sede
Inner Join Clase ON Clase.CodigoSede = Sede.Codigo
Group by Sede.Codigo, Nombre, Direccion, Telefono,
MaxCapacidad,Lockers, Duchas
Order by Clases DESC, Duchas ASC
```



### 13. La información de los socios con membresías plus que más sedes visitaron el último mes junto con el número de sedes visitadas.

```
3Select Top 3 Nombre, Apellido, TipoContrato, NIT, Direccion, Telefono, COUNT(DISTINCT CodigoSede) As SedesVisitadas
From Visita
Inner Join Cliente ON Cliente.Codigo = Visita.CodigoCliente
Inner Join Contrato ON Contrato.CodigoCliente = Cliente.Codigo
Where TipoContrato = 'Premium'
      AND DATEPART(month, Fecha) = DATEPART(month, GETDATE()) - 1
      AND DATEPART(year, Fecha) = DATEPART(year, GETDATE())
Group by Nombre, Apellido, TipoContrato, NIT, Direccion, Telefono
Order by SedesVisitadas DESC
```

### 14. El detalle de los entrenadores que están de turno en la hora en que más socios llegan a entrenar por sede.

```
Select HoraVisita As Hora, COUNT(HoraVisita) As Visitas, CodigoSede
Into #VisitasPorHora
From(
    Select DATEPART(HOUR, Hora) As HoraVisita, CodigoSede
    From Visita
)as data
Group by HoraVisita, CodigoSede

Select data.CodigoSede, Hora as HoraOcupada
Into #VisitasMaxima
From (
    Select CodigoSede, MAX(Visitas) As MaxVisitas
    From #VisitasPorHora
    Group by CodigoSede
)as data
Inner Join #VisitasPorHora ON #VisitasPorHora.Visitas = MaxVisitas AND #VisitasPorHora.CodigoSede = data.CodigoSede

Select Distinct Personal.Codigo, data.CodigoSede, Nombre, Apellido, FechaNacimiento, DPI, Salario
From(
    Select DATEPART(HOUR, Hora) As HoraVisita, CodigoSede, CodigoEntrenador
    From Visita
)as data
Inner Join Personal ON Personal.Codigo = data.CodigoEntrenador
Inner Join #VisitasMaxima ON #VisitasMaxima.CodigoSede = data.CodigoSede
      AND #VisitasMaxima.HoraOcupada = HoraVisita
Where Puesto = 'Entrenador'
```

### 15. El detalle de la sede donde más socios han terminado su contrato.

```
Select Codigo, Nombre, Direccion, Telefono, MaxCapacidad, Lockers, Duchas
From
(
    Select Top 1 Ubicacion, COUNT(Ubicacion) As Cancelaciones
    From Contrato
    Inner Join CancelacionContrato ON Contrato.Numero = CancelacionContrato.Numero
    Group by Ubicacion
    Order by Cancelaciones DESC
) As data
Inner Join Sede ON Sede.Codigo = Ubicacion
```

**16. El listado de usuarios que tienen asignado el entrenamiento con mayor número de ejercicios.**

```
select CodigoCliente, Nombre + ' ' + Apellido As NombreCompleto, NIT, Direccion, Telefono
from (
    select Top 1 NombreEntrenamiento As Entrenamiento, COUNT(NombreEjercicio) As Ejercicios
    from Rutina
    group by NombreEntrenamiento
    order by Ejercicios DESC
)As data
Inner Join Visita On Visita.NombreEntrenamiento = data.Entrenamiento
Inner Join Cliente On Cliente.Codigo = CodigoCliente
```

**17. El detalle de la sede con más personal de recepción pero menos número de lockers.**

```
select Top 1 CodigoSede, Sede.Nombre, Direccion, Telefono, MaxCapacidad,
            Lockers, Duchas, COUNT(1) As Personal
from Personal
Inner Join Sede ON Sede.Codigo = CodigoSede
Where Puesto = 'Recepcion'
Group by CodigoSede, Sede.Nombre, Direccion, Telefono,
        MaxCapacidad, Lockers, Duchas
Order by Personal DESC, Lockers ASC
```

**18. El número de clases que se han impartido de cada disciplina, en cada sede.**

```
select CodigoSede, Tipo, COUNT(1) As Clases
from Clase
group by CodigoSede, Tipo
order by CodigoSede
```

**19. El entrenador que más socios atiende por turno (matutina, mixto, vespertina, nocturno) por sede.**

```
select CodigoEntrenador, Personal.CodigoSede, Turno, COUNT(1) As Visitas
into #Entrenadores
from Visita
Inner Join Personal ON Personal.Codigo = CodigoEntrenador
Where Puesto = 'Entrenador'
Group by CodigoEntrenador, Personal.CodigoSede, Turno

select Personal.CodigoSede, Personal.Turno, Codigo, Nombre, Apellido, FechaNacimiento, DPI, Salario
from (
    select Turno, CodigoSede As Sede, MAX(Visitas) As MaxVisitas
    from #Entrenadores
    group by CodigoSede, Turno
)as data
Inner Join #Entrenadores On CodigoSede = Sede AND MaxVisitas = Visitas
Inner Join Personal On CodigoEntrenador = Codigo

Drop Table #Entrenadores
```

## 20. La máquina por sede que ha tenido más mantenimientos durante todo su tiempo de uso.

```
Select CodigoMaquina, NombreSede, COUNT(1) As Mantenimientos
Into #Mantenimiento
From Mantenimiento
Inner Join Maquina ON Maquina.Codigo = CodigoMaquina
Group by CodigoMaquina, NombreSede

Select Maquina.NombreSede, Maquina.Codigo, MusculoPrincipal, Ubicacion
From (
    Select NombreSede, MAX(Mantenimientos) As MaxMantenimientos
    From #Mantenimiento
    Group by NombreSede
)as data
Inner Join #Mantenimiento ON #Mantenimiento.NombreSede = data.NombreSede AND Mantenimientos = MaxMantenimientos
Inner Join Maquina ON Maquina.Codigo = CodigoMaquina

Drop Table #Mantenimiento
```



## VISTAS

**Nombre de la Vista:** SociosFrecuentes

**Enunciado:** La información de los socios que realizaron más de 15 visitas en el último mes a cualquier sede.

**Consulta SQL:**

```
CREATE VIEW SociosFrecuentes
([Codigo], [Nombre], [Apellido], [NIT], [Direccion], [Telefono], [Cantidad_Visitas])
WITH ENCRYPTION
AS
SELECT Codigo, Nombre, Apellido, NIT, Direccion, Telefono, Cantidad_Visitas
FROM Cliente C
INNER JOIN
(
    SELECT COUNT(V.CodigoCliente) AS Cantidad_Visitas, V.CodigoCliente
    FROM Visita V
    WHERE DATEPART(month, V.Fecha) = DATEPART(month, GETDATE()) - 1
        AND DATEPART(year, V.Fecha) = DATEPART(year, GETDATE())
    GROUP BY V.CodigoCliente
    HAVING COUNT(V.CodigoCliente) > 15
) R1
ON C.Codigo=R1.CodigoCliente
```

**Justificación:** Se implementó la vista de socios más frecuentes debido a que es una consulta que se realiza con frecuencia y es necesario tener un rápido acceso a la información. Es de conveniencia para algunos departamentos del gimnasio como el de ventas para ofrecerles ofertas especiales a los clientes para incentivar que sigan llegando.

**Nombre de la Vista:** DeudaContratosTerminados

**Enunciado:** El promedio de deuda total pendiente por sede de los contratos terminados.

**Consulta SQL:**

```
CREATE VIEW DeudaContratosTerminados
([Codigo], [DeudaPromedio])
WITH ENCRYPTION
AS
SELECT S.Codigo, ROUND(AVG(TP.Precio), 2) AS DeudaPromedio
FROM Contrato C
INNER JOIN CancelacionContrato CC
    ON C.Numero=CC.Numero
INNER JOIN TipoContrato TP
    ON C.TipoContrato=TP.Nombre
INNER JOIN Sede S
    ON C.Ubicacion=S.Codigo
GROUP BY S.Codigo
```

**Justificación:** Se eligió esta consulta como vista debido a que se seleccionan aquellas columnas de las cuales usuarios específicos pueden tener acceso por cuestiones de seguridad al trabajar con cantidades de dinero. Inclusive es una vista encriptada para salvaguardar los datos.

**Nombre de la Vista:** DatosInstructor

**Enunciado:** Los datos del instructor con menos certificaciones que ha dado más clases grupales en cada sede.

**Consulta SQL:**

```
CREATE VIEW DatosInstructor
([DPI], [Nombre], [Apellido], [Fecha_Nacimiento])
WITH ENCRYPTION
AS
SELECT DPI, Nombre, Apellido, Fecha_Nacimiento
FROM Instructor I
INNER JOIN
(
    SELECT TOP 1 COUNT(CL.DPI_Instructor) AS Total_Clases, CL.DPI_Instructor, Cantidad_Certificaciones
    FROM CLASE CL
    INNER JOIN
    (
        SELECT COUNT(CER.DPI_Instructor) AS Cantidad_Certificaciones, CER.DPI_Instructor
        FROM Instructor I
        INNER JOIN Certificacion CER
        ON I.DPI=CER.DPI_Instructor
        GROUP BY CER.DPI_Instructor
    ) R1
    ON CL.DPI_Instructor=R1.DPI_Instructor
    GROUP BY CL.DPI_Instructor, Cantidad_Certificaciones
    ORDER BY Cantidad_Certificaciones ASC, Total_Clases DESC
) R2
ON I.DPI=R2.DPI_Instructor
```

**Justificación:** Se implementó la vista para que se una estructura de apoyo, debido a que es una consulta con múltiples tablas, lo que la hace compleja. Además es conveniente tener un acceso rápido a los instructores.

**Nombre de la Vista:** Top3EmpleadosVentas

**Enunciado:** Top 3 de los empleados de ventas que más han devengado en comisiones de ventas por nuevas inscripciones.

**Consulta SQL:**

```
CREATE VIEW Top3EmpleadosVentas
([CodigoVendedor], [Comisiones])
WITH ENCRYPTION
AS
SELECT TOP 3 C.CodigoVendedor, ROUND(SUM(TC.Precio) * 0.1, 2) AS Comisiones
FROM Contrato C
INNER JOIN Personal P
    ON C.CodigoVendedor=P.Codigo
INNER JOIN TipoContrato TC
    ON C.TipoContrato=TC.Nombre
WHERE P.Puesto='Ventas'
GROUP BY C.CodigoVendedor
ORDER BY Comisiones DESC
```

**Justificación:** Se realizó una vista de esta consulta por cuestiones de seguridad debido a que cuando se consultan cantidades monetarias es necesario seleccionar solo aquellas que se desea mostrar al usuario final.