



INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO



Database Selected Topics

Alumno:

Meza Vargas Brandon David

Boleta:

2020630288

Grupo:

3CM16

Profesor:

Galeana Chávez María Del Rosario

Fecha:

24/04/23

Actividad 4

1. Seleccionar el nombre de todos los tipos de aviones con más de 200 asientos

Query

```
SELECT tipoAvion as nombre FROM [dbo].[TipoAvion] WHERE numAsientos > 200
```

Resultado

Results Messages	
	nombre
1	Airbus A320neo
2	A350 XWB
3	Tu-244

2. Seleccionar el nombre de todos los tipos de aviones con más de 200 asientos, con la siguiente leyenda: "El avión" <tipo avión> "tiene una capacidad de " <num asientos> "asientos"

Query

```
SELECT CONCAT('El avión ', tipoAvion, ' tiene una capacidad de ', numAsientos, ' asientos')  
FROM [dbo].[TipoAvion]  
WHERE numAsientos > 200
```

Resultado

	(No column name)
1	El avión Airbus A320neo tiene una capacidad de 220 asientos
2	El avión A350 XWB tiene una capacidad de 300 asientos
3	El avión Tu-244 tiene una capacidad de 269 asientos

3. Seleccionar el nombre, apellido paterno y apellido materno de todos los empleados de las aerolíneas

Query

```
SELECT nombre, paterno, materno FROM Persona  
JOIN Empleado ON Empleado.IdPersona = Persona.IdPersona
```

Resultado

	nombre	paterno	materno
1	Eduardo	Ortiz	NULL
2	Martín	Carmona	Bautista
3	Victor	Sandoval	Escud...
4	Carlos	Tadeo	Almeida
5	Elsa	Fernán...	Cabrera
6	Yuridia	Salazar	Figuer...
7	Edith	Parez	Grana...
8	Beatriz	Pedraza	López

4. Seleccionar el nombre, apellido paterno y apellido materno de todos los empleados de las aerolíneas concatenados en una sola columna con el alias nombre del empleado.

Query

```
SELECT CONCAT(nombre, ' ', paterno, ' ', materno) as 'nombre del empleado' FROM Persona  
JOIN Empleado ON Empleado.IdPersona = Persona.IdPersona
```

Resultado

	nombre del empleado
1	Eduardo Ortiz
2	Martín Carmona Bautista
3	Victor Sandoval Escudero
4	Carlos Tadeo Almeida
5	Elsa Fernández Cabrera
6	Yuridia Salazar Figueroa
7	Edith Parez Granados
8	Beatriz Pedraza López

5. Seleccionar el nombre, apellido paterno y apellido materno de todos los empleados de las aerolíneas con la siguiente leyenda "bienvenido, " <nombre completo de la persona> ", hoy es " <fecha actual>

Query

```
SELECT CONCAT('Bienvenido ', nombre, ' ', paterno, ' ', materno, ' hoy es ', CONVERT(date, GETDATE())) as 'bienvenida' FROM Persona  
JOIN Empleado ON Empleado.IdPersona = Persona.IdPersona
```

Resultado

	bienvenida
1	Bienvenido Eduardo Ortiz hoy es 2023-04-23
2	Bienvenido Martín Carmona Bautista hoy es 2023-04...
3	Bienvenido Victor Sandoval Escudero hoy es 2023-0...
4	Bienvenido Carlos Tadeo Almeida hoy es 2023-04-23
5	Bienvenido Elsa Fernández Cabrera hoy es 2023-04...
6	Bienvenido Yuridia Salazar Figueroa hoy es 2023-04...
7	Bienvenido Edith Parez Granados hoy es 2023-04-23
8	Bienvenido Beatriz Pedraza López hoy es 2023-04-23

6. Seleccionar el nombre completo de todas las personas en la base de datos que cumpla años el 21/04/2023 y mostrarlo como "Felices " <edad> " años " <nombre completo> "!!!"

Query

```
SELECT CONCAT('Felices ', DATEDIFF(YEAR, nacimiento, '2023-04-21'), ' años ', nombre, ' ', paterno, ' ', materno, '!!!')  
FROM Persona  
WHERE MONTH(nacimiento) = 4 AND DAY(nacimiento) = 21
```

Resultado

Results	Messages
	(No column name)
1	Felices 0 años Roberto Martinez Soriano!!!
2	Felices 0 años Ceferino De la O Dominguez!!!
3	Felices 0 años Mario Carranza Gutierrez!!!
4	Felices 0 años Fidencio Galeana Hernestrosa!!!
5	Felices 0 años Juvenal Pastrana Villalba!!!

7. mostrar la antigüedad en años de todos los aviones almacenados en la base de datos

Query

```
SELECT DATEDIFF(YEAR, fechaFabricacion, GETDATE()) antigüedad, matricula FROM Avion
```

Resultado

results		messages
	antigüedad	matricula
1	13	234312
2	12	234320
3	13	234323
4	10	432734
5	9	432888
6	13	437239
7	11	734592
8	12	738492
9	13	738493

8. mostrar la fecha exacta del siguiente mantenimiento de los aviones tomando como referencia la fecha del último, ordenándolos por el que está más próximo a vencer

Query

```
SELECT matricula, DATEADD(MONTH, mesesMantenimiento, UltimoMantenimiento) 'mantenimiento proximo' FROM Avion
ORDER BY 'mantenimiento proximo' ASC
```

Resultado

matricula		mantenimiento proximo
1	437239	2013-04-01 00:00:00.000
2	234320	2013-07-15 00:00:00.000
3	234312	2014-04-01 00:00:00.000
4	738492	2014-06-30 00:00:00.000
5	234323	2014-07-06 00:00:00.000
6	734592	2014-07-23 00:00:00.000
7	432888	2015-01-01 00:00:00.000
8	738493	2015-02-03 00:00:00.000
9	432734	2015-06-10 00:00:00.000

9. mostrar el tiempo en días que falta para el mantenimiento de los aviones tomando como referencia la fecha del último, ordenándolos por el que está más próximo a vencer

Query

```
SELECT matricula, DATEDIFF(DAY, UltimoMantenimiento, DATEADD(MONTH, mesesMantenimiento, UltimoMantenimiento))
'mantenimiento proximo en días' FROM Avion
ORDER BY 'mantenimiento proximo en días' ASC
```

Resultado

	matricula	mantenimiento proximo en dias
1	234320	365
2	432888	365
3	734592	365
4	738492	546
5	234312	547
6	432734	547
7	234323	730
8	738493	730
9	437239	1096

10. mostrar el tiempo en días por el que se atrás o adelante el último mantenimiento de los aviones tomando como referencia la fecha de adquisición.

Query

```
SELECT DATEDIFF(DAY, fechaFabricacion, UltimoMantenimiento) dias FROM Avion
```

Resultado

	dias
1	914
2	366
3	731
4	0
5	0
6	0
7	365
8	730
9	1096

11. mostrar todos los vuelos y ordenarlos por los que tengan están más próximos y tengan más lugares vacíos

Query

```
WITH asientos AS (
    SELECT ve.idVuelo, vu.matriculaAvion, vu.precioPorAsiento, vu.fechaVuelo, SUM(numAsientos) as asientosOcupados
    FROM Venta ve
    INNER JOIN Vuelo vu ON vu.idVuelo = ve.idVuelo
    GROUP BY ve.idVuelo, vu.matriculaAvion, vu.precioPorAsiento, vu.fechaVuelo
)
SELECT asi.idVuelo, asi.matriculaAvion, asi.precioPorAsiento,
asi.fechaVuelo, ta.numAsientos - asi.asientosOcupados as asientosVacios
FROM asientos asi
INNER JOIN Avion a ON asi.matriculaAvion = a.matricula
INNER JOIN TipoAvion ta ON ta.idTipoAvion = a.idTipoAvion
ORDER BY asi.fechaVuelo ASC, asientosVacios DESC
```

Resultado

Results		Messages			
	idVuelo	matriculaAvion	precioPorAsiento	fechaVuelo	asientosVacios
1	13	432734	5000.00	2014-01-01 00:00:00.000	176
2	10	234323	2500.00	2014-01-02 00:00:00.000	218
3	14	432734	4730.00	2014-01-15 00:00:00.000	178
4	15	432888	4032.00	2014-02-01 00:00:00.000	163
5	12	234323	3000.00	2014-03-01 00:00:00.000	215
6	19	734592	6340.00	2014-08-20 00:00:00.000	147
7	8	234320	2590.00	2014-09-11 00:00:00.000	79
8	7	234320	3670.00	2014-09-12 00:00:00.000	94
9	9	234320	6000.00	2014-09-15 00:00:00.000	90
10	20	734592	1324.00	2014-10-02 00:00:00.000	147
11	2	234312	3000.00	2014-12-01 00:00:00.000	96

12. mostrar los vuelos que tengan un precio por asiento mayor a 1000 pesos, ordenarlos de manera ascendente

Query

```
SELECT * FROM Vuelo WHERE precioPorAsiento > 1000 ORDER BY precioPorAsiento ASC
```

Resultado

Results		Messages			
	idVuelo	matriculaAvion	precioPorAsiento	fechaVuelo	
1	20	734592	1324.00	2014-10-02 00:00:00.000	
2	6	234312	2000.00	2014-08-25 00:00:00.000	
3	16	738492	2340.00	2014-12-01 00:00:00.000	
4	10	234323	2500.00	2014-01-02 00:00:00.000	
5	8	234320	2590.00	2014-09-11 00:00:00.000	
6	12	234323	3000.00	2014-03-01 00:00:00.000	
7	2	234312	3000.00	2014-12-01 00:00:00.000	
8	7	234320	3670.00	2014-09-12 00:00:00.000	
9	4	234312	3700.00	2014-07-02 00:00:00.000	
10	15	432888	4032.00	2014-02-01 00:00:00.000	
11	14	432734	4730.00	2014-01-15 00:00:00.000	
12	13	432734	5000.00	2014-01-01 00:00:00.000	
13	9	234320	6000.00	2014-09-15 00:00:00.000	
14	3	234312	6000.00	2014-12-23 00:00:00.000	
15	19	734592	6340.00	2014-08-20 00:00:00.000	
16	17	738492	7000.00	2014-12-30 00:00:00.000	
17	18	738492	7302.00	2014-12-31 00:00:00.000	

13. Mostrar todos los empleados que no tengan jefe

Query

```
SELECT * FROM Empleado WHERE idJefe IS NULL
```

Resultado

Results		Messages					
	idEmpleado	idPersona	idTipoEmpleado	Aerolineaid	fechaIngreso	suelo	idJefe
1	5	13	5	3	2008-09-07 00:00:00.000	45700.00	NULL
2	6	14	5	4	2008-01-01 00:00:00.000	49200.00	NULL
3	7	20	5	5	2005-11-15 00:00:00.000	45900.00	NULL

14. mostrar el día de la semana en que le toca mantenimiento a cada uno de los aviones

Query

```
SELECT matricula, DATEPART(WEEKDAY, DATEADD(MONTH, mesesMantenimiento, UltimoMantenimiento)) dia FROM Avion
```

Resultado

	matricula	dia
1	234312	3
2	234320	2
3	234323	1
4	432734	4
5	432888	5
6	437239	2
7	734592	4
8	738492	2
9	738493	3

15. Mostrar las aerolíneas extranjeras, con el texto, la aerolínea <nombre> es extranjera

Query

```
SELECT 'La aerolinea ' + Nombre + ' es extranjera'
FROM Aerolinea WHERE nacional = 0
```

Resultado

	(No column name)
1	La aerolinea American Airlines es extranjera

16. Selecciona la tabla de clientes ordenándola por la fecha de registro de manera descendente

Query

```
SELECT * FROM Cliente ORDER BY fechaRegistro DESC
```

Resultado

	clienteId	personalId	idTipoCliente	fechaRegistro
1	3	18	1	2014-08-05 00:00:00.000
2	2	7	1	2014-08-01 00:00:00.000
3	1	5	1	2014-07-01 00:00:00.000
4	11	17	3	2011-12-10 00:00:00.000
5	9	10	3	2011-12-09 00:00:00.000
6	4	9	2	2011-06-05 00:00:00.000
7	6	15	2	2010-12-20 00:00:00.000
8	5	12	2	2010-12-12 00:00:00.000
9	10	4	3	2010-07-30 00:00:00.000
10	14	2	3	2010-02-03 00:00:00.000
11	15	6	3	2010-02-03 00:00:00.000
12	12	19	3	2010-01-01 00:00:00.000
13	7	16	2	2009-12-01 00:00:00.000
14	13	14	3	2008-01-01 00:00:00.000
15	8	8	3	2002-05-01 00:00:00.000

17. Selecciona al cliente tipo Oro que tenga más antigüedad de registro

Query

```
SELECT TOP 1 c.clienteId, c.personaId, c.idTipoCliente, c.fechaRegistro, p.nombre, p.paterno, p.materno
FROM Cliente c
INNER JOIN TipoCliente tc ON tc.idTipoCliente = c.idTipoCliente
INNER JOIN Persona p ON c.personaId = p.IdPersona
WHERE tipoCliente = 'Oro'
ORDER BY fechaRegistro
```

Resultado

	clienteId	personaId	idTipoCliente	fechaRegistro	nombre	paterno	materno
1	8	8	3	2002-05-01 00:00:00.000	Eleonora	Solis	NULL

18. Selecciona todos los domicilios y Ordénalos por código postal, colonia y calle

Query

```
SELECT * FROM Direccion Order BY codigoPostal, colonia, calle
```

Resultado

	direccionId	calle	numero	colonia	codigoPostal
1	10	Zempoaltecas	s/n	Exhacienda el Rosario	02420
2	18	Insurgentes Sur	234	Nochebuena	03720
3	4	Av. H. Escuela Naval Militar	83	Presidentes Ejidales	04470
4	12	Arcos de Belén	345	Centro	06010
5	8	Av. 20 de Noviembre	2	Centro	06060
6	21	Luis Moya	12	Centro	06070
7	7	Alfonso Reyes	30	Hipódromo Condesa	06140
8	3	Insurgentes Centro	765	San Rafael	06470
9	11	Serapio Rendón	4	San Rafael	06470
10	9	Río Amazonas	456	Cuauhtémoc	06500
11	5	Río Atoyac	12	Cuauhtémoc	06500
12	15	Río Tigris	45	Cuauhtémoc	06500
13	13	Av. Paseo de la Reforma	124	Juárez	06600
14	6	Lieja	3	Juárez	06600
15	19	Álvaro Obregón	34	Roma	06700
16	2	Canela	123	Granjias	08400
17	17	Montes Urales	83	Lomas Virreyes	11000
18	22	Bvd. Miguel de Cervante...	384	Ampliación Granada	11529

19. Selecciona todos los domicilios cuyo número de casa no sea numérico

Query

```
SELECT * FROM Direccion WHERE ISNUMERIC(numero) <> 1
```

Resultado

	direccionId	calle	numero	colonia	codigoPostal
1	10	Zempoaltecas	s/n	Exhacienda el Rosario	02420

20. Selecciona el sueldo diario de los empleados considerando que todos los meses son de 30 días, agregar el alias "sueldo diario" y ordenarlo de manera descendente

Query

```
SELECT sueldo/30 as 'sueldo diario' FROM Empleado
```

Resultado

Results Messages	
	sueldo diario
1	1583.3333
2	1301.6666
3	1333.3333
4	1523.3333
5	1640.00
6	1530.00
7	566.6666
8	490.00

21. Realiza una consulta a los tipos de clientes que muestre el siguiente texto "al ser cliente "<tipo cliente> " tienes un descuento de " <porcentaje descuento>

Query

```
SELECT CONCAT('Al ser cliente ', tipoCliente, ' tienes un descuento de ', descuento, '%') FROM TipoCliente
```

Resultado

Results Messages	
	(No column name)
1	Al ser cliente Nuevo tienes un descuento de 0%
2	Al ser cliente Plata tienes un descuento de 5%
3	Al ser cliente Oro tienes un descuento de 10%

22. selecciona los vuelos del mes de diciembre de 2014 que cuesten menos de 4000 pesos y ordenalos de forma ascendente en costo

Query

```
SELECT * FROM Vuelo WHERE MONTH(fechaVuelo) = 12 AND YEAR(fechaVuelo) = 2014 AND precioPorAsiento < 4000  
ORDER BY precioPorAsiento asc
```

Resultado

Results

Messages

	idVuelo	matriculaAvion	precioPorAsiento	fechaVuelo
1	16	738492	2340.00	2014-12-01 00:00:00.000
2	2	234312	3000.00	2014-12-01 00:00:00.000

23. selecciona los clientes tipo oro que tengan menos de 5 años de antigüedad

Query

```
SELECT c.clienteId, c.personaId, c.idTipoCliente, c.fechaRegistro, p.nombre, p.paterno, p.materno  
FROM Cliente c  
INNER JOIN TipoCliente tc ON tc.idTipoCliente = c.idTipoCliente  
INNER JOIN Persona p ON c.personaId = p.IdPersona  
WHERE tipoCliente = 'Oro' AND DATEDIFF(YEAR, fechaRegistro, GETDATE()) < 5
```

Resultado

100 %

Results

Messages

clienteId	personalId	idTipoCliente	fechaRegistro	nombre	paterno	materno
-----------	------------	---------------	---------------	--------	---------	---------

24. muestra una lista de todos los tipos de aviones ordenándolos de manera ascendente por el nombre del fabricante y descendente por la cantidad de asientos

Query

```
SELECT * FROM TipoAvion ORDER BY fabricante ASC, numAsientos DESC
```

Resultado

Results		Messages		
	idTipoAvion	tipoAvion	fabricante	numAsientos
1	4	Bombardier BD-500-1A10	Bombardier Aerospace	100
2	6	MRJ 90ER	Mitsubishi Aircraft Corporation	96
3	2	A350 XWB	Airbus	300
4	1	Airbus A320neo	Airbus	220
5	3	Boeing 737 MAX 9	Boeing Commercial Airplanes	180
6	5	COMAC C919	COMAC	190
7	7	Tu-244	Túpolev	269
8	8	MS-21-200	United Aircraft Corporation	150

25. selecciona los tipos de empleado de vuelo con la siguiente leyenda "el tipo de empleado " + <nombre del tipo> + " requiere estar en vuelo"

Query

```
SELECT CONCAT('El tipo de empleado ', tipoEmpleado, ' requiere estar en vuelo')
FROM TipoEmpleado WHERE esDeVuelo = 1
```

Resultado

	Results	Messages
	(No column name)	
1	El tipo de empleado Piloto requiere estar en vuelo	
2	El tipo de empleado Aeromosa requiere estar en vuelo	

26. selecciona los vuelos que se realizaron esta última semana

Query

```
SELECT * FROM Vuelo
WHERE fechaVuelo BETWEEN DATEADD(DAY, -7, GETDATE()) AND GETDATE()
```

Resultado

Results		Messages		
	idVuelo	matriculaAvion	precioPorAsiento	fechaVuelo

27. selecciona los vuelos que se compraron durante el último mes

Query

```
SELECT v.idVuelo, v.matriculaAvion FROM Vuelo v INNER JOIN Venta ve ON v.idVuelo = ve.idVuelo
WHERE ve.fechaCompra BETWEEN DATEADD(MONTH, -1, GETDATE()) AND GETDATE()
```

Resultado

Results	Messages
idVuelo	matriculaAvion

28. selecciona a las personas cuya longitud del teléfono no sea exactamente igual a 8

Query

```
SELECT * FROM Persona WHERE LEN(telefono) <> 8
```

Resultado

Results		Messages						
	IdPersona	nombre	paterno	materno	direccionId	telefono	nacimiento	sexo
1	1	Edgardo	Hernández	Madrigal	2	5561283734	1980-08-30 00:00:00.000	0
2	4	Eduardo	Ortiz	NULL	8	53123212231	1965-09-27 00:00:00.000	0
3	9	Marco	Ramírez	Ortiz	11	53247845232	1987-02-09 00:00:00.000	0
4	10	Elisa	Romero	Ayala	12	535221	1979-08-30 00:00:00.000	1
5	15	Monica	Juárez	NULL	17	591203	1979-12-09 00:00:00.000	1
6	22	Roberto	Martinez	Soriano	2	5561283734	2023-04-21 11:59:32.720	0
7	23	Ceferino	De la O	Dominguez	2	53123212231	2023-04-21 11:59:36.230	0

29. Selecciona las compras con una antigüedad mayor a un año

Query

```
SELECT * FROM Venta WHERE DATEDIFF(YEAR, fechaCompra, GETDATE()) > 1
```

Resultado

Results Messages

	idVenta	idCliente	fechaCompra	numAsientos	idVuelo	idVendedor
1	1	1	2014-07-01 00:00:00.000	2	2	1
2	2	1	2014-08-07 00:00:00.000	1	7	2
3	3	1	2014-05-07 00:00:00.000	4	9	2
4	4	2	2014-03-24 00:00:00.000	2	2	4
5	6	3	2014-02-26 00:00:00.000	1	7	4
6	7	3	2014-02-26 00:00:00.000	2	10	4
7	8	4	2014-01-01 00:00:00.000	4	8	2
8	9	4	2014-02-20 00:00:00.000	2	9	1
9	11	5	2014-12-23 00:00:00.000	1	12	1
10	12	6	2013-09-11 00:00:00.000	2	14	1
11	13	7	2014-08-30 00:00:00.000	4	13	1
12	14	8	2014-02-10 00:00:00.000	2	15	2
13	15	6	2014-03-02 00:00:00.000	2	15	1
14	16	9	2014-05-09 00:00:00.000	4	15	2
15	17	10	2014-08-30 00:00:00.000	4	15	2
16	18	10	2014-07-23 00:00:00.000	4	12	2
17	19	11	2014-01-01 00:00:00.000	4	8	1
18	20	12	2013-11-05 00:00:00.000	2	8	1

Query executed successfully.

DESKTOP-1ARV6MM\SQLEXPRESS ...DESKTOP-1ARV6MM\BrandM...Aerolineas00:00:0023 ro

Query executed successfully. DESKTOP-1ARV6MM\SQLEXPRESS ... DESKTOP-1ARV6MM\BrandM... Aerolíneas 00:00:00 23 rows

30. selecciona el día de la semana en que saldrán todos los vuelos.

Query

```
SELECT DATEPART(WEEKDAY, fechaVuelo) 'dia vuelo' FROM Vuelo
```

Resultado

Results Messages	
	dia vuelo
1	2
2	3
3	4
4	2
5	6
6	5
7	2
8	5
9	7
10	4
11	4
12	7
13	2
14	3
15	4
16	4
17	5