



INSTITUTO POLITÉCNICO NACIONAL

UNIDAD PROFESIONAL INTERDISCIPLINARIA
EN INGENIERIA Y TECNOLOGÍAS AVANZADAS

BASES DE DATOS DISTRIBUIDAS

GRUPO: 3TM3

PRÁCTICA 1

PROFESOR: DE LA CRUZ SOSA CARLOS

INTEGRANTES EQUIPO 2:

OCAMPO CABRERA ASael	2022640070
ROBERT ROA KARLA GUADALUPE	2022640710
SANCHEZ VILLAGRANA OSMAR ROBERTO	2022640514

Numero de consulta:	Número 1. Listar el top 5 de las entidades con más casos confirmados por cada uno de los años registrados en la base de datos.
Requisitos:	Se usó la tabla datoscovid, filtrando solo casos confirmados (CLASIFICACION_FINAL = 1, 2, 3). También se utilizó cat_entidades para obtener los nombres de las entidades. Se empleó YEAR(FECHA_INGRESO) para agrupar los datos y una subconsulta para seleccionar las cinco entidades con más casos por año.
Significado de los valores de los catálogos:	CLASIFICACION_FINAL: Filtra casos confirmados (1: asociación clínica-epidemiológica, 2: comité de dictaminación, 3: laboratorio). ENTIDAD_RES: Clave de la entidad donde se registró el caso. FECHA_INGRESO: Fecha de registro del caso. Total_casos: Número de casos confirmados por entidad y año.
Responsable de la consulta:	Ocampo Cabrera Asael
Comentarios:	Se utilizaron instrucciones vistas en clase como COUNT (), JOIN, WHERE para seleccionar el top 5. Sin embargo, esta vez se hizo uso de YEAR () para filtrar el año de una fecha la consulta deseada.

Results		Messages	
	Año	Estado	Confrimados
1	2020	CIUDAD DE MÉXICO	362275
2	2020	MÉXICO	161809
3	2020	GUANAJUATO	86268
4	2020	NUEVO LEÓN	85561
5	2020	SONORA	56700
6	2021	CIUDAD DE MÉXICO	653053
7	2021	MÉXICO	234438
8	2021	NUEVO LEÓN	126299
9	2021	GUANAJUATO	119429
10	2021	JALISCO	112295
11	2022	CIUDAD DE MÉXICO	351093
12	2022	MÉXICO	132309
13	2022	NUEVO LEÓN	97680
14	2022	GUANAJUATO	71416
15	2022	SAN LUIS POTOSÍ	70474

✓ Query execut... | LAP_ASA\SQLEXPRESS (16.0 RTM) | LAP_ASA\ocamp (70) | Covid_Hist | 00:24:34 | 15 rows

Numero de consulta:	Número 2. Listar el municipio con más casos confirmados recuperados por estado y por año.
Requisitos:	Se empleó la tabla datoscovid, filtrando solo casos confirmados (CLASIFICACION_FINAL = 1, 2, 3). Se utilizó YEAR(FECHA_INGRESO) para agrupar los datos por año y MUNICIPIO_RES para contar casos a nivel municipal. Se realizó una subconsulta con MAX() para seleccionar el municipio con más casos por estado y año.
Significado de los valores de los catálogos:	CLASIFICACION_FINAL: Filtra casos confirmados (1: asociación clínica epidemiológica, 2: comité de dictaminación, 3: laboratorio). ENTIDAD_RES: Clave de la entidad donde se registró el caso. MUNICIPIO_RES: Clave del municipio donde se registró el caso. FECHA_INGRESO: Fecha en que se registró el caso. total_casos: Número de casos confirmados por municipio, entidad y año.
Responsable de la consulta:	Ocampo Cabrera Asael
Comentarios:	Al igual que en otras consultas, se hizo uso de múltiples instrucciones vistas previamente en clase, sin embargo, se hizo uso de YEAR () filtrar el año de una fecha en nuestra consulta.

Results		Messages			
	año	entidad	municipio_res	total_casos	
1	2022	AGUASCALIENTES	001	19448	
2	2020	AGUASCALIENTES	001	14689	
3	2021	AGUASCALIENTES	001	18604	
4	2020	BAJA CALIFORNIA	002	15671	
5	2021	BAJA CALIFORNIA	004	21172	
6	2022	BAJA CALIFORNIA	004	17669	
7	2021	BAJA CALIFORNIA SUR	008	20785	
8	2022	BAJA CALIFORNIA SUR	003	18721	
9	2020	BAJA CALIFORNIA SUR	003	8957	
10	2021	CAMPECHE	002	8303	

Query execut... | LAP_ASA\SQLEXPRESS (16.0 RTM) | LAP_ASA\ocamp (70) | Covid_Hist | 00:00:30 | 96 rows

Numero de consulta:	Número 3. Listar el porcentaje de casos confirmados en cada una de las siguientes morbilidades a nivel nacional: diabetes, obesidad e hipertensión.
Requisitos:	Se utilizó la tabla principal datoscovid para contar los casos confirmados (CLASIFICACION_FINAL en 1, 2, 3) en personas con diabetes, obesidad e hipertensión. Se creó una vista llamada casos_morbilidades, donde se cuentan los casos para cada morbilidad y el total de casos confirmados. Posteriormente, la consulta calcula el porcentaje de cada morbilidad dividiendo el número de casos por el total y multiplicando por 100, redondeando el resultado a dos decimales.
Significado de los valores de los catálogos:	CLASIFICACION_FINAL (1, 2, 3): Indica que un caso está confirmado. SI_NO:() Define si la persona tiene diabetes, obesidad o hipertensión, teniendo (1 = Sí, 2 = No, 97 = No aplica, 98 = Se ignora, 99 = No especificado). total_casos: Cantidad total de casos confirmados a nivel nacional. Porcentajes calculados: Representan el porcentaje de personas con cada morbilidad respecto al total de casos confirmados.
Responsable de la consulta:	Ocampo Cabrera Asael
Comentarios:	Para esta consulta se hizo uso de CASE WHEN para dar condiciones de selección en nuestra vista, además, se implementó el uso de ROUND () para hacer el redondeo del total de los porcentajes de nuestra consulta y CAST () para mejorar la precisión y presentación de los resultados, al hacer uso de decimales.

Results Messages			
	porcentaje_diabetes	porcentaje_obesidad	porcentaje_hipertension
1	9.62	10.57	12.79

Query executed successfully. | LAP_ASA\SQLEXPRESS (16.0 RTM) | LAP_ASA\ocamp (70) | Covid_Hist | 00:00:08 | 1 rows

Numero de consulta:	Número 4. Listar los municipios que no tengan casos confirmados en todas las morbilidades: hipertensión, obesidad, diabetes, tabaquismo.
Requisitos:	Se utilizó la tabla principal datoscovid, aplicando condiciones en la cláusula WHERE para excluir los casos confirmados (CLASIFICACION_FINAL no en 1, 2, 3). Además, se filtraron registros donde todas las morbilidades (diabetes, obesidad, hipertension, tabaquismo) sean iguales a 1, lo que indica que la persona padece esas condiciones. Finalmente, se agruparon los resultados por entidad y municipio para identificar aquellos sin casos confirmados bajo estos criterios.
Significado de los valores de los catálogos:	CLASIFICACION_FINAL (1, 2, 3): Casos confirmados de COVID-19. SI_NOO): Indica si la persona tiene la morbilidad correspondiente, teniendo (1 = Sí, 2 = No, 97 = No aplica, 98 = Se ignora, 99 = No especificado). ENTIDAD_RES y MUNICIPIO_RES: Representan el estado y el municipio del paciente, con sus claves disponibles en los catálogos de entidades y municipios.
Responsable de la consulta:	Ocampo Cabrera Asael
Comentarios:	Al igual que en varias consultas, se hizo uso de funciones previamente vistas en clase, en este caso se utilizó funciones de agregación y filtrado los municipios sin casos confirmados en personas con todas las morbilidades mencionadas.

Results		Messages
	ENTIDAD_RES	Municipio
1	28	002
2	16	052
3	07	106
4	16	015
5	18	001
6	17	002
7	04	008
8	13	029
9	14	105
10	08	049

Query exec... LAP_ASA\SQLEXPRESS (16.0 RTM) LAP_ASA\ocamp (70) Covid_Hist 00:00:07 841 rows

Numero de consulta:	Número 5. Listar los estados con más casos recuperados con neumonía.
Requisitos:	Se utilizó la tabla datoscovid para identificar los casos de pacientes con neumonía (neumonia = 1). La consulta une esta tabla con el catálogo de entidades cat_entidades mediante la clave de entidad (ENTIDAD_UM = clave) para obtener el nombre del estado. Se aplicó COUNT(*) para contar los casos de neumonía por estado y se usó GROUP BY para agrupar los resultados por entidad. Finalmente, los resultados se ordenaron en orden descendente según el número de casos de neumonía.
Significado de los valores de los catálogos:	neumonia: Indica si el paciente presentó neumonía. cat_entidades: Contiene la clave y el nombre de cada estado. ENTIDAD_UM: Representa la entidad donde se atendió al paciente. Total_con_neumonia: Cantidad de casos de neumonía recuperados por estado.
Responsable de la consulta:	Ocampo Cabrera Asael
Comentarios:	Para esta consulta no se hizo uso de ninguna nueva función diferente a las previamente vistas en clase.

Results		Messages
	Estado	Total_con_neumonia
1	CIUDAD DE MÉXICO	157835
2	MÉXICO	89093
3	PUEBLA	45007
4	VERACRUZ DE IGNACIO DE LA LLAVE	30846
5	GUANAJUATO	30567
6	JALISCO	29907
7	BAJA CALIFORNIA	27039
8	NUEVO LEÓN	26598
9	SONORA	21991
10	CHIHUAHUA	21553

Query execut... LAP_ASA\SQLEXPRESS (16.0 RTM) LAP_ASA\ocamp (70) Covid_Hist 00:00:06 32 rows

Numero de consulta:	Número 6. Listar el total de casos confirmados/sospechosos por estado en cada uno de los años registrados en la base de datos.
Requisitos:	Se utilizó la tabla datoscovid para obtener el número de casos confirmados (CLASIFICACION_FINAL en 1, 2, 3) y sospechosos (CLASIFICACION_FINAL = 6). Se creó una vista llamada casos_confirmados_sospechosos, donde se agruparon los datos por año (FECHA_INGRESO) y estado (ENTIDAD_UM), calculando el total de casos por clasificación. Posteriormente, se unió esta vista con el catálogo de entidades cat_entidades para obtener el nombre del estado y se ordenaron los resultados en orden ascendente por año y estado.
Significado de los valores de los catálogos:	<p>CLASIFICACION_FINAL: Se utilizó (1, 2, 3) para casos confirmados de COVID-19, y (6) para casos sospechosos.</p> <p>FECHA_INGRESO: Fecha de ingreso del paciente al sistema de salud.</p> <p>ENTIDAD_UM: Representa la entidad donde se atendió al paciente.</p> <p>cat_entidades: Contiene la clave y el nombre de cada estado.</p> <p>Confirmados: Total de casos confirmados por estado y año.</p> <p>Sospechosos: Total de casos sospechosos por estado y año.</p>
Responsable de la consulta:	Ocampo Cabrera Asael
Comentarios:	Al igual que en previas consultas, se hizo uso CASE WHEN para usar una condición en la selección de nuestra vista, así como, YEAR () para filtrar por año una fecha de la consulta.

Results		Messages		
	año	Estado	Confirmados	Sospechosos
1	2020	AGUASCALIENTES	18071	6846
2	2020	BAJA CALIFORNIA	36965	28530
3	2020	BAJA CALIFORNIA SUR	18135	907
4	2020	CAMPECHE	7886	4216
5	2020	CHIAPAS	8057	29234
6	2020	CHIHUAHUA	39749	2753
7	2020	CIUDAD DE MÉXICO	428012	38439
8	2020	COAHUILA DE ZARAGOZA	52945	8086
9	2020	COLIMA	8140	1745
10	2020	DURANGO	26244	1893

Query execut... | LAP_ASA\SQLEXPRESS (16.0 RTM) | LAP_ASA\ocamp (70) | Covid_Hist | 00:00:24 | 96 rows

Numero de consulta:	Número 7. Para el año 2020 y 2021 cuál fue el mes con más casos registrados, confirmados, sospechosos, por estado registrado en la base de datos.
Requisitos:	Se utilizó la tabla datoscovid para obtener el número de casos registrados, confirmados (CLASIFICACION_FINAL en 1, 2, 3) y sospechosos (CLASIFICACION_FINAL = 6). Se creó la vista casos_por_mes, donde se agruparon los datos por año, mes y estado (ENTIDAD_UM), calculando el total de casos para cada clasificación. Luego, se aplicó una consulta con RANK() en Max_casos, que ordenó los meses de cada estado según el número de casos registrados en orden descendente. Finalmente, se filtraron los registros con RANK() = 1, asegurando que solo se mostrara el mes con más casos por estado.
Significado de los valores de los catálogos:	<p>CLASIFICACION_FINAL: Se utilizó (1, 2, 3) para casos confirmados de COVID-19, y (6) para casos sospechosos.</p> <p>FECHA_INGRESO: Fecha de ingreso del paciente al sistema de salud.</p> <p>ENTIDAD_UM: Representa la entidad donde se atendió al paciente.</p> <p>cat_entidades: Contiene la clave y el nombre de cada estado.</p> <p>Registrados: Total de casos registrados en un mes por estado.</p> <p>Confirmados: Total de casos confirmados en un mes por estado.</p> <p>Sospechosos: Total de casos sospechosos en un mes por estado.</p> <p>mes: Mes del año en el que se registraron los casos.</p>
Responsable de la consulta:	Ocampo Cabrera Asael
Comentarios:	Para esta consulta se hizo uso de la cláusula WITH previamente vista en clase, para ser referenciada en nuestra consulta los valores de la tabla temporal creada. Así mismo, dentro de la misma cláusula se utilizó RANK() para la asignación de rango en las filas y obtener los valores de rango deseados, y OVER(PARTITION BY) para particionar el conjunto de valores y encontrar el mes con más casos por estado.

Results		Messages				
	año	Estado	mes	Registrados	Confirmados	Sospechosos
1	2020	AGUASCALIENTES	11	10879	4709	879
2	2021	AGUASCALIENTES	1	11456	3825	926
3	2020	BAJA CALIFORNIA	12	16987	7895	4705
4	2021	BAJA CALIFORNIA	11	29417	11789	700
5	2020	BAJA CALIFORNIA SUR	12	10330	3374	32
6	2021	BAJA CALIFORNIA SUR	12	22812	7272	79
7	2020	CAMPECHE	7	5090	2462	1207
8	2021	CAMPECHE	8	11362	5553	397
9	2020	CHIAPAS	6	14125	2697	10409
10	2021	CHIAPAS	8	14822	4406	5740

Query execut... | LAP_ASA\SQLEXPRESS (16.0 RTM) | LAP_ASA\ocamp (70) | Covid_Hist | 00:00:39 | 64 rows

Numero de consulta:	Número 8. Listar el municipio con menos defunciones en el mes con más casos confirmados con neumonía en los años 2020 y 2021.
Requisitos:	Crear una consulta que identifique el municipio con menos defunciones en el mes con más casos confirmados de COVID-19 con neumonía en los años 2020 y 2021.
Significado de los valores de los catálogos:	<p>CLASIFICACION_FINAL IN ('1', '2', '3'): Filtra los casos confirmados de COVID-19 según el catálogo cat_clasificacion_final.</p> <p>NEUMONIA = 1: Indica que el caso confirmado presentó neumonía.</p> <p>FECHA_INGRESO: Fecha de ingreso del caso.</p> <p>FECHA_DEF: Fecha de defunción del caso (no nula para considerar defunciones).</p> <p>MUNICIPIO_RES: Municipio de residencia del paciente.</p> <p>anio: Año de ingreso del caso.</p> <p>mes: Mes de ingreso del caso.</p> <p>total_casos: Total de casos confirmados con neumonía por mes y año.</p> <p>total_defunciones: Total de defunciones por municipio, mes y año.</p>
Responsable de la consulta:	Sánchez Villagrana Osmar Roberto
Comentarios:	<p>- Se utilizaron subconsultas con WITH para identificar el mes con más casos confirmados con neumonía por año y luego encontrar el municipio con menos defunciones en ese mes. La función ROW_NUMBER() se utilizó para asignar un número único a cada mes dentro de cada año, ordenado por el número de casos en orden descendente. Esto permitió identificar el mes con más casos (el que tiene ROW_NUMBER = 1) para cada año. La función RANK() se utilizó para asignar un rango a cada municipio dentro de cada año y mes, ordenado por el número de defunciones en orden ascendente. Esto permitió identificar los municipios con menos defunciones (los que tienen RANK = 1) en el mes con más casos. se aplican funciones de ventana (ROW_NUMBER y RANK) para ordenar y filtrar los resultados de manera eficiente.</p> <p>En la consulta se ocupa SELECT, JOIN, GROUP BY, ORDER BY, los cuales son temas vistos en clase.</p>

Results Messages				
	Anio	Mes	MUNICIPIO_RES	Total_Defunciones
1	2020	7	159	1
2	2020	7	162	1
3	2020	7	165	1
4	2020	7	168	1
5	2020	7	170	1
6	2020	7	195	1
7	2020	7	242	1
8	2020	7	243	1
9	2020	7	265	1
10	2020	7	272	1
11	2020	7	284	1
12	2020	7	285	1

Query executed successfully.

ROBERT_21\SQLEXPRESS (16.0 ... | ROBERT_21\OSBET (52) | Covid_Historico | 00:00:12 | 111 rows

Numero de consulta:	Número 9. Listar el top 3 de municipios con menos casos recuperados en el año 2021.
Requisitos:	Crear una consulta que identifique los tres municipios con menor cantidad de casos recuperados en el año 2021, utilizando una vista previa que agrupe los casos por municipio y año.
Significado de los valores de los catálogos:	CLASIFICACION_FINAL IN (1,2,3): Filtra los casos confirmados de COVID-19 según el catálogo <i>cat_clasificacion_final</i> . ENTIDAD_RES: Clave de la entidad de residencia del paciente, relacionada con <i>cat_entidades</i> . MUNICIPIO_RES: Clave del municipio de residencia del paciente, relacionada con <i>cat_municipios</i> . FECHA_INGRESO: Fecha de ingreso del caso. total_casos: Total de casos confirmados en cada municipio y año.
Responsable de la consulta:	Robert Roa Karla Guadalupe
Comentarios:	Se implementaron instrucciones vistas en clase, como JOIN para unir tablas y ORDER BY para ordenar los resultados. Además, se utilizó la nueva instrucción IN (año, año) para filtrar los datos de años específicos, facilitando el análisis de los casos negativos por estado.

Results Messages		
	Municipio	TotalCasos
1	187	1
2	389	1
3	182	1
AROBERT\karla (53) CovidHistorico_ 00:00:11 3 rows		

Numero de consulta:	Número 10. Listar el porcentaje de casos confirmado por género en los años 2020 y 2021.
Requisitos:	Crear una consulta que muestre el porcentaje de casos de COVID-19 confirmados (clasificación final 1, 2 o 3) por año (2020 y 2021) y sexo, respecto al total general de casos en esos dos años.
Significado de los valores de los catálogos:	CLASIFICACION_FINAL IN ('1', '2', '3'): Filtra los casos confirmados de COVID-19 según el catálogo cat_clasificacion_final. FECHA_INGRESO: Fecha de ingreso del caso. SEXO: Sexo del paciente. año: Año de ingreso del caso. porcentaje: Porcentaje de casos por año y sexo respecto al total general.
Responsable de la consulta:	Sánchez Villagrana Osmar Roberto
Comentarios:	Se utilizó una subconsulta con WITH para obtener el total de casos generales en los años 2020 y 2021. Calculamos el porcentaje de casos por año y sexo utilizando una división y CAST para formatear el resultado con 15 decimales. En la consulta se utilizaron temas antes visto como Select, Join, Order By, Group by. Y no hubo la necesidad de implementar hasta el momento una mejora para poder optimizar la consulta

Results Messages				
	año	sexo	total_casos	porcentaje
1	2020	1	754215	18.735495929296000
2	2020	2	768663	19.094399484895000
3	2021	1	1272723	31.615781422567000
4	2021	2	1229993	30.554323163240000

Query executed successfully. ROBERT_21\SQLEXPRESS (16.0 ... ROBERT_21\OSBET (58) Covid_Historico 00:00:38 4 rows

Numero de consulta:	Número 11. Listar el porcentaje de casos hospitalizados por estado en el año 2020.
Requisitos:	Crear una consulta que calcule el porcentaje de pacientes hospitalizados por estado en el año 2020, considerando únicamente aquellos registros con ingreso en dicho año.
Significado de los valores de los catálogos:	TIPO_PACIENTE = 2: Indica casos hospitalizados según el catálogo <i>cat_tipo_paciente</i> . ENTIDAD_UM: Clave del estado donde se registró el caso, relacionada con <i>cat_entidades</i> . FECHA_INGRESO: Fecha de ingreso del paciente a la unidad médica. porcentaje_hospitalizados: Porcentaje de hospitalizados respecto al total de casos por estado.
Responsable de la consulta:	Robert Roa Karla Guadalupe
Comentarios:	Se utilizó la función CASE dentro de COUNT para contar únicamente los casos hospitalizados (donde tipo_paciente = 2). Además, se aplicó ROUND para redondear el porcentaje de hospitalizados a dos decimales y CAST para mostrar el resultado con un formato decimal específico.

Results Messages		
	Estado	Porcentaje_Hospitalizados
1	SINALOA	26.94
2	VERACRUZ DE IGNACIO DE LA LLAVE	26.41
3	MÉXICO	26.34
4	HIDALGO	24.12
5	CHIHUAHUA	23.57
6	MORELOS	22.48
7	NAYARIT	20.95
8	JALISCO	20.85
9	QUINTANA ROO	19.97
10	OAXACA	19.10
11	COLIMA	18.67
12	BAJA CALIFORNIA	18.66
13	GUERRERO	17.23
14	PUEBLA	17.08
15	TAMAULIPAS	15.76
16	SONORA	15.26

Que... | LAPKARLAROBERT\SQLEXPRESS (... | LAPKARLAROBERT\karla (53) | CovidHistorico_ | 00:00:13 | 32 rows

Numero de consulta:	Número 12. Listar total de casos negativos por estado en los años 2020 y 2021.
Requisitos:	Crear una vista que resuma el total de casos negativos por estado y año. Simplificando las consultas enfocándose en casos negativos.
Significado de los valores de los catálogos:	CLASIFICACION_FINAL = 7: Indica casos negativos según el catálogo cat_clasificacion_final. ENTIDAD_UM: Clave del estado donde se registró el caso, relacionada con cat_entidades. año: Año de ingreso del caso, extraído de FECHA_INGRESO. total negativos: Número de casos negativos agrupados por estado y año.
Responsable de la consulta:	Robert Roa Karla Guadalupe
Comentarios:	Se agregó la instrucción IN para filtrar los datos de los años 2020 y 2021, mientras que las demás instrucciones utilizadas, como JOIN y ORDER BY, ya se habían visto previamente en clase.

Results		Messages	
	Año	Estado	Total_Negativos
1	2020	AGUASCALIENTES	31260
2	2021	AGUASCALIENTES	64971
3	2020	BAJA CALIFORNIA	22900
4	2021	BAJA CALIFORNIA	124795
5	2020	BAJA CALIFORNIA SUR	32874
6	2021	BAJA CALIFORNIA SUR	97422
7	2020	CAMPECHE	12474
8	2021	CAMPECHE	34887
9	2020	CHIAPAS	8310
10	2021	CHIAPAS	44079
11	2020	CHIHUAHUA	24955
12	2021	CHIHUAHUA	64276
13	2020	CIUDAD DE MÉXICO	798204

\\LROBERT\SQLEXPRESS (... | LAPKARLAROBERT\karla (53) | CovidHistorico_ | 00:00:30 | 64 rows

Numero de consulta:	Número 13. Listar porcentajes de casos confirmados por género en el rango de edades de 20 a 30 años, de 31 a 40 años, de 41 a 50 años, de 51 a 60 años y mayores a 60 años a nivel nacional.
Requisitos:	Crear una consulta que muestre el número total de casos de COVID-19 confirmados (clasificación final 3) por sexo y rango de edad, junto con el porcentaje que representan respecto al total general de casos. Utilizar una vista previa (casos_genero_edad) que agrupe los casos por año, sexo y rango de edad.
Significado de los valores de los catálogos:	CLASIFICACION_FINAL = CLASIFICACION_FINAL IN (1, 2, 3): Filtra los casos confirmados de COVID-19 según el catálogo cat_clasificacion_final. FECHA_INGRESO: Fecha de ingreso del caso. SEXO: Sexo del paciente. EDAD: Edad del paciente. rango_edad: Rango de edad del paciente (20-30, 31-40, 41-50, 51-60, 60+). total_casos: Total de casos confirmados por sexo y rango de edad. Porcentaje: Porcentaje de casos por sexo y rango de edad respecto al total general.
Responsable de la consulta:	Sánchez Villagrana Osmar Roberto
Comentarios:	Se creo una vista casos_genero_edad para agrupar los casos por año, sexo y rango de edad, utilizando la expresión CASE para definir los rangos de edad Se realiza una consulta sobre esta vista para calcular el total de casos y el porcentaje por sexo y rango de edad Se utiliza una función de ventana SUM OVER para calcular el total general de casos y calcular el porcentaje. En la consulta se utilizaron temas antes visto como Select, Join, Order By, Group by. Y no hubo la necesidad de implementar hasta el momento una mejora para poder optimizar la consulta.

108 %

Results Messages

	sexo	rango_edad	total_casos	Porcentaje
1	1	20-30	694483	12.46
2	1	31-40	660480	11.85
3	1	41-50	577196	10.35
4	1	51-60	401082	7.19
5	1	60+	564839	10.13
6	2	20-30	634886	11.39
7	2	31-40	594364	10.66
8	2	41-50	498442	8.94
9	2	51-60	363255	6.52
10	2	60+	586581	10.52

Query executed successfully. ROBERT_21\SQLXPRESS (16.0 ... ROBERT_21\OSBET (81) Covid_Historico 00:00:17 10 rows

Numero de consulta:	Número 14. Listar el rango de edad con más casos confirmados y que fallecieron en los años 2020 y 2021.
Requisitos:	Se utilizó la tabla principal de la base de datos denominada datoscovid , aplicando un CASE para agrupar las edades en rangos específicos, lo cual permitió analizar de forma detallada la distribución de los casos.
Significado de los valores de los catálogos:	CLASIFICACION_FINAL IN (1, 2, 3): Filtra solo casos confirmados de acuerdo con el catálogo cat_clasificacion_final. FECHA_DEF: Fecha de defunción, usada para determinar casos fallecidos. rangoEdad: Categoriza la edad en grupos específicos. casosConfirmados: Total de casos confirmados por rango de edad. casosFallecidos: Total de fallecidos por rango de edad.
Responsable de la consulta:	Robert Roa Karla Guadalupe
Comentarios:	Se utilizó CASE para organizar las edades en grupos específicos y facilitar el análisis. También se empleó TRY_CONVERT para asegurar que las fechas estuvieran en el formato correcto y DATEPART para extraer el año de esos registros. Además, con LEFT JOIN se combinó la información de las tablas y con TOP 1 se seleccionó el grupo de edad con más fallecimientos. Tanto LEFT JOIN como TOP ya se habían visto en clase.

Results		Messages	
	rangoEdad	casosConfirmados	casosFallecidos
1	60+	1151420	179837

APKARLAROBERT\karla (53) | CovidHistorico_ | 00:00:26 | 1 rows

CONCLUSIONES

En esta práctica, trabajamos con una base de datos relacionada con casos de COVID-19 y realizamos diversas consultas SQL para obtener información específica. Para interpretar correctamente los datos, fue importante el apoyarnos en catálogos y descriptores, los cuales nos ayudaron a entender los valores de la base de datos, como los códigos de estados, municipios y las clasificaciones de los pacientes. De cierta manera, esta fue nuestra primera estrategia como equipo, ya que, nuestro primer punto a reconocer fue el de la importancia de identificar y comprender adecuadamente los atributos antes de realizar consultas. Esto incluyó analizar las relaciones entre tablas y comprender los tipos de datos, lo que garantizó que nuestras consultas devolvieran resultados precisos y coherentes. Sin los catálogos y descriptores, habría sido complicado interpretar los datos correctamente y detectar patrones relevantes en la información.

Una vez teniendo en cuenta la distribución de nuestra base de datos y las relaciones dentro de ella, implementamos estrategias para optimizar y organizar la información, como la creación de vistas. Las vistas no solo facilitaron la reutilización de consultas complejas, sino que también mejoraron la claridad, lo cual nos resultó demasiado útil al trabajar en una base con grandes volúmenes de datos. Sin embargo, conforme íbamos avanzando con el desarrollo de la práctica, nos percatamos que esta estrategia no tenía mucho sentido de ser implementada en todas las consultas, ya que, muchas de ellas no requerían una consulta lo suficiente compleja como para la utilización de una vista en ella.

Por lo tanto, al no continuar con la estrategia previa decidimos continuar con algo más sencillo y más común en nuestra implementación con forme a lo visto en clase, optando por una nueva estrategia de hacer mayor uso de funciones de agregación y filtrado, como COUNT, SUM, AVG, junto con las cláusulas GROUP BY y WHERE para obtener estadísticas precisas, como el total de casos por región, edades promedio y tasas de recuperación, organizando los resultados de manera que facilitaron su análisis e interpretación al momento de realizar las consultas correspondientes.

Además, notamos que había consultas donde se necesitaba relacionar múltiples columnas de la tabla para generar el resultado deseado, por lo que, decidimos implementar subconsultas y combinaciones de tipo (JOIN), para conectar la información entre múltiples tablas, como condiciones preexistentes, ubicación geográfica y evolución de los pacientes. Así mismo, las subconsultas nos permitieron realizar consultas dentro de consultas, filtrando datos específicos antes de combinarlos. Por ejemplo, podríamos haber usado una subconsulta para obtener la lista de pacientes con diabetes y luego usar un JOIN para combinar esa lista con la información de ubicación de los pacientes.

Por otro lado, la cláusula WITH nos permitió crear tablas temporales dentro de nuestras consultas, lo que hizo que las consultas complejas fueran mucho más legibles y fáciles de mantener. Pensamos que si teníamos una consulta larga que realizaba varios cálculos intermedios, podríamos dividirla en partes más pequeñas y manejables mediante la cláusula WITH, así cada parte creaba una tabla temporal con un resultado intermedio, y luego podíamos combinar esas tablas temporales para obtener el resultado final.

Sin embargo, conforme íbamos planteando y opinando sobre los posibles siguientes pasos para dar solución a los demás ejercicios notamos que necesitábamos incluir otro tipo de funciones diferentes a las vistas en clase, optando por revisar estrategias avanzadas. Por lo tanto, decidimos hacer uso de funciones ventana como `RANK ()` y `ROW_NUMBER ()` las cuales empleamos para ordenar y clasificar datos de manera eficiente, mediante la asignación de números a las filas de una partición correspondiente.

En general, esta práctica no solo reforzó nuestras habilidades en SQL, sino que también nos ayudó a comprender la importancia de estructurar y documentar correctamente las bases de datos. Este enfoque estructurado es esencial para el análisis eficiente de grandes volúmenes de datos y para la toma de decisiones basadas en información precisa y organizada.