

REPORTE DE PRÁCTICA NO 2.1

FRAGMENTOS

ALUMNO: Esperilla Mendoza Luis Erick
Dr. Eduardo Cornejo-Velázquez



1. Introducción

En esta práctica, se llevó a cabo el diseño e implementación de fragmentación horizontal y vertical en una base de datos MySQL utilizada para la gestión de flotillas de autos.

La fragmentación es una técnica de distribución de datos que permite mejorar el rendimiento, la disponibilidad y la administración de la información en sistemas distribuidos.

Para lograrlo, primero se restauró la base de datos original a partir de un archivo de respaldo. Posteriormente, se implementaron cinco fragmentos horizontales y cinco fragmentos verticales utilizando vistas dentro de MySQL, asegurando que cada fragmento cumpliera con los criterios de particionamiento definidos.

Como parte de la evidencia de la actividad, se alojaron en un repositorio de GitHub tanto el reporte de la práctica como el archivo de respaldo de la base de datos. Finalmente, se registró la URL del repositorio en la Plataforma Garza para su evaluación.

2. Marco teórico

Fragmentación de Bases de Datos

La fragmentación es una técnica utilizada en bases de datos distribuidas para dividir una tabla en partes más pequeñas llamadas fragmentos, que pueden almacenarse en diferentes ubicaciones. Existen dos tipos principales de fragmentación:

1. Fragmentación Horizontal

La fragmentación horizontal consiste en dividir una tabla en subconjuntos de filas, basándose en una condición específica. Cada fragmento contiene un subconjunto de los registros de la tabla original, manteniendo la misma estructura de columnas.

2. Fragmentación Vertical

La fragmentación vertical divide una tabla en subconjuntos de columnas, asegurando que cada fragmento contenga la clave primaria para poder reconstruir la tabla original. Se utiliza para mejorar el rendimiento de consultas específicas y optimizar el acceso a datos.

Vistas en MySQL

Las vistas en MySQL son consultas almacenadas que permiten representar subconjuntos de datos sin modificar la tabla original. Son útiles para la fragmentación, ya que permiten simular particiones lógicas de los datos.

3. Herramientas empleadas

1. DataGrip.

DataGrip es un entorno de desarrollo integrado (IDE) para bases de datos desarrollado por JetBrains. Permite gestionar bases de datos de manera eficiente con una interfaz avanzada para la escritura y ejecución de consultas SQL.

Uso en la práctica:

- Conexión y gestión de la base de datos MySQL.
- Creación y edición de vistas para la fragmentación.
- Validación de los fragmentos generados mediante consultas SQL.

2. Github.

GitHub es una plataforma de control de versiones que facilita la colaboración y el almacenamiento de archivos de proyectos. Se utilizó para alojar la evidencia de la actividad.

Uso en la práctica:

- Subida del archivo de respaldo de la base de datos.
- Almacenamiento del reporte de la práctica.
- Registro de la URL del repositorio en la Plataforma Garza.

4. Desarrollo

Fragmentos Horizontales

– 1. Fragmento de conductores activos

Listing 1: Crear la vista para mostrar a los conductores que estan activos

```
1 CREATE VIEW fragmento_conductores_activos AS
2 SELECT * FROM conductor WHERE estado = 'Activo';
```

	conductorId	nombre	numeroLicencia	vencimientoLicencia	estado
1	1	Pedro Díaz	LIC-000001	2026-01-09	Activo
2	2	Patricia Vázquez	LIC-000002	2025-03-02	Activo
3	3	Carlos García	LIC-000003	2025-10-15	Activo
4	4	Andrés Díaz	LIC-000004	2025-08-14	Activo
5	5	Fernanda Díaz	LIC-000005	2025-12-22	Activo
6	6	Carmen López	LIC-000006	2025-06-21	Activo
7	7	Francisco Martínez	LIC-000007	2025-10-30	Activo
8	8	Patricia Flores	LIC-000008	2025-11-28	Activo
9	9	Laura Sánchez	LIC-000009	2025-05-06	Activo
10	10	Pedro Hernández	LIC-000010	2026-02-16	Activo
11	11	Andrés López	LIC-000011	2025-04-04	Activo
12	12	Carmen Castillo	LIC-000012	2025-08-14	Activo

Figure 1: Ejecución de código. 87 filas

– 2. Fragmento de conductores con licencia vigente y estado activo

Listing 2: Crear la vista para mostrar a los conductores que tienen licencia vigente y estan activos

```
1 CREATE VIEW fragmento_conductores_vigentes AS
2 SELECT * FROM conductor WHERE estado = 'Activo' AND
   vencimientoLicencia > CURDATE();
```

	conductorId	nombre	numeroLicencia	vencimientoLicencia	estado
1	1	Pedro Díaz	LIC-000001	2026-01-09	Activo
2	3	Carlos García	LIC-000003	2025-10-15	Activo
3	4	Andrés Díaz	LIC-000004	2025-08-14	Activo
4	5	Fernanda Díaz	LIC-000005	2025-12-22	Activo
5	6	Carmen López	LIC-000006	2025-06-21	Activo
6	7	Francisco Martínez	LIC-000007	2025-10-30	Activo
7	8	Patricia Flores	LIC-000008	2025-11-28	Activo
8	9	Laura Sánchez	LIC-000009	2025-05-06	Activo
9	10	Pedro Hernández	LIC-000010	2026-02-16	Activo
10	11	Andrés López	LIC-000011	2025-04-04	Activo
11	12	Carmen Castillo	LIC-000012	2025-08-14	Activo
12	13	Miguel Vázquez	LIC-000013	2025-08-12	Activo

Figure 2: Ejecución de código. 84 filas

– 3. Fragmento de vehículos activos con verificación reciente (en el último año)

Listing 3: Crear la vista para mostrar a los vehiculos que estan activos y tienen verificacion vigente y estan activos

```
1 CREATE VIEW fragmento_vehiculos_activos_verificados AS
2 SELECT * FROM vehiculo WHERE estado = 'Activo' AND fechaVerificacion >=
   DATE_SUB(CURDATE(), INTERVAL 1 YEAR);
```

	vehiculoId	flotillaId	tipo	modelo	marca	anio	estado	fechaVerificacion
1	6	95	Pickup	Ford F-150	Ram	2023	Activo	2025-03-10
2	7	23	Pickup	Volvo VNL	Ram	2017	Activo	2025-02-14

Figure 3: Ejecución de código. 2 filas

- 4. Fragmento de mantenimientos costosos (mayores a \$500) y pendientes

Listing 4: Crear la vista para mostrar a los mantenimientos que superan los \$500) y estan pendientes

```
1 CREATE VIEW fragmento_mantenimientos_costosos_pendientes AS
2 SELECT * FROM mantenimiento WHERE estado <> 'Completado' AND costo > 500;
```

	mantenimientoId	vehiculoId	fechaServicio	tipoServicio	descripcion	costo	estado
1	2	88	2021-10-14	Alineación	Servicio realizado en taller Querétaro	526.93	En Proceso
2	3	44	2023-01-28	Rotación de llantas	Servicio realizado en taller Querétaro	3244.78	Pendiente
3	4	57	2021-01-20	Balanceo	Servicio realizado en taller Querétaro	2259.34	En Proceso
4	5	56	2021-10-10	Revisión de frenos	Servicio realizado en taller Monterrey	2294.09	En Proceso
5	6	23	2023-12-05	Reparación motor	Servicio realizado en taller Monterrey	2250.66	Pendiente
6	7	72	2022-01-30	Rotación de llantas	Servicio realizado en taller Guadalajara	1287.90	Pendiente
7	8	79	2020-10-14	Balanceo	Servicio realizado en taller Pachuca	1301.27	Pendiente
8	11	19	2022-07-29	Alineación	Servicio realizado en taller Pachuca	2538.17	Pendiente
9	12	44	2022-02-19	Rotación de llantas	Servicio realizado en taller CDMX	2593.43	Pendiente
10	13	3	2022-11-03	Rotación de llantas	Servicio realizado en taller Querétaro	590.03	En Proceso
11	17	57	2022-01-30	Cambio de filtros	Servicio realizado en taller Pachuca	809.96	En Pro
12	18	74	2021-12-04	Sistema eléctrico	Servicio realizado en taller Guadalajara	1667.66	Pendiente

Figure 4: Ejecución de código. 60 filas

- 5. Fragmento de rutas en curso con distancia mayor a 50 km

Listing 5: Crear la vista para mostrar a los las rutas con distancia mayor a 50km

```
1 CREATE VIEW fragmento_rutas_largas_en_curso AS
2 SELECT * FROM ruta WHERE estado = 'Pendiente' AND distancia > 50;
```

	rutaId	vehiculoId	conductorId	horaInicio	horaFin	distancia	ubicacionInicio	ubicacionFin
1	1	54	30	2023-01-30 12:00:00	2023-01-18 18:00:00	997.62	Guadalajara, Jalisco	Hermosillo, Son
2	10	37	2	2023-02-02 06:00:00	2023-01-27 23:00:00	170.50	Monterrey, NL	Cancún, QR
3	11	9	93	2023-01-12 23:00:00	2023-01-02 00:00:00	106.64	Monterrey, NL	Tijuana, BC
4	17	1	58	2023-01-29 05:00:00	2023-01-18 03:00:00	67.12	Puebla, Pue	Mérida, Yuc
5	18	48	59	2023-01-18 01:00:00	2023-01-27 21:00:00	510.07	Pachuca, Hidalgo	Puebla, Pue
6	19	87	56	2023-01-06 11:00:00	2023-01-06 12:00:00	357.50	Ciudad de México	Cancún, QR
7	23	40	92	2023-01-14 01:00:00	2023-01-08 03:00:00	887.63	Mérida, Yuc	Monterrey, NL
8	28	42	86	2023-01-02 18:00:00	2023-01-24 05:00:00	348.17	Querétaro, Qro	Querétaro, Qro
9	29	45	27	2023-02-02 16:00:00	2023-01-05 02:00:00	678.03	Hermosillo, Son	Mérida, Yuc
10	30	8	79	2023-01-25 09:00:00	2023-01-10 19:00:00	301.12	Querétaro, Qro	Querétaro, Qro
11	33	64	24	2023-01-10 10:00:00	2023-01-24 11:00:00	680.46	Monterrey, NL	Pachuca, Hidalgo
12	40	36	61	2023-01-02 01:00:00	2023-02-01 08:00:00	416.93	Ciudad de México	Pachuca, Hidalgo

Figure 5: Ejecución de código. 31 filas

Fragmentos Verticales

- 1. Fragmento de conductores y sus transacciones de combustible

Listing 6: Crear la vista para seleccionar a los conductores junto con las transacciones de combustible

```
1 CREATE VIEW fragmento_conductores_transacciones AS
2 SELECT c.conductorId, c.nombre, c.numeroLicencia, t.transaccionId, t.monto,
3        t.cantidad, t.tipoCombustible, t.fechaTransaccion
4 FROM conductor c
JOIN transaccioncombustible t ON c.conductorId = t.conductorId;
```

	conductorId	nombre	numeroLicencia	transaccionId	monto	cantidad	tipoCombustible
1	71	Roberto Ramírez	LIC-000071		83.22	60.34	Diesel
2	78	Patricia Morales	LIC-000078		148.78	106.31	Diesel
3	78	Patricia Morales	LIC-000078		374.93	31.10	Premium
4	20	Roberto Ortiz	LIC-000020		337.47	69.58	Diesel
5	37	María Sánchez	LIC-000037		173.23	54.43	Diesel
6	20	Roberto Ortiz	LIC-000020		224.91	56.79	Diesel
7	47	Verónica Hernández	LIC-000047		241.13	90.40	Premium
8	77	Juan López	LIC-000077		281.24	41.60	Diesel
9	72	Luis Díaz	LIC-000072		83.55	57.57	Premium
10	26	Carmen Pérez	LIC-000026		276.21	90.72	Magna
11	46	Luis Morales	LIC-000046		214.24	68.71	Premium
12	14	Carlos Hernández	LIC-000014		332.41	83.09	Premium

Figure 6: Ejecución de código. 100 filas

– 2. Fragmento de vehículos y su historial de mantenimiento

Listing 7: Crear la vista para seleccionar a los vehiculos con sus mantenimientos

```

1 CREATE VIEW fragmento_vehiculos_mantenimientos AS
2     SELECT v.vehiculoId, v.tipo, v.modelo, v.marca, m.mantenimientoId, m.
           fechaServicio, m.tipoServicio, m.costo, m.estado
3     FROM vehiculo v
4     JOIN mantenimiento m ON v.vehiculoId = m.vehiculoId;

```

	vehiculoId	tipo	modelo	marca	mantenimientoId	fechaServicio	tipoServicio
1	92	SUV	Kenworth T680	Ford	1	2020-07-11	Reparación motor
2	88	SUV	Chevrolet Silverado	Volvo	2	2021-10-14	Alineación
3	44	Sedán	Chevrolet Silverado	Ram	3	2023-01-28	Rotación de llantas
4	57	Sedán	Hino 338	Freightliner	4	2021-01-20	Balanceo
5	56	SUV	Chevrolet Silverado	Mercedes-Benz	5	2021-10-10	Revisión de frenos
6	23	SUV	Kenworth T680	Kenworth	6	2023-12-05	Reparación motor
7	72	Pickup	Ford F-150	Peterbilt	7	2022-01-30	Rotación de llantas
8	79	Tráiler	Kenworth T680	Mercedes-Benz	8	2020-10-14	Balanceo
9	19	Camión	Peterbilt 579	Ram	9	2022-04-14	Cambio de aceite
10	83	Sedán	Chevrolet Silverado	Freightliner	10	2021-07-02	Rotación de llantas

Figure 7: Ejecución de código. 100 filas

– 3. Fragmento de conductores con transacciones de combustible en una ubicación específica (Estación Mobil - Guadalajara)

Listing 8: Crear la vista para seleccionar a los conductores que tienen transacciones de combustible en (Estación Mobil - Guadalajara)

```

1 CREATE VIEW fragmento_conductores_transacciones_ciudad AS
2     SELECT c.conductorId, c.nombre, t.transaccionId, t.monto, t.tipoCombustible,
           t.fechaTransaccion, t.ubicacion
3     FROM conductor c
4     JOIN transaccioncombustible t ON c.conductorId = t.conductorId
5     WHERE t.ubicacion = 'Estación Mobil - Guadalajara';

```

	conductorId	nombre	transaccionId	monto	tipoCombustible	fechaTransaccion	ubicacion
1	20	Roberto Ortiz	4	337.47	Diesel	2023-01-20 10:00:00	Estación Mobil - Guadalajara
2	27	Lucía Díaz	17	108.60	Premium	2023-01-10 16:00:00	Estación Mobil - Guadalajara
3	84	Fernanda Morales	96	460.41	Premium	2023-01-23 09:00:00	Estación Mobil - Guadalajara

Figure 8: Ejecución de código. 3 filas

– 4. Fragmento de rutas completadas por conductores con licencias vencidas

Listing 9: Crear la vista para seleccionar las rutas completadas por conductores con licencias vencidas

```

1      CREATE VIEW fragmento_rutas_conductores_licencia_vencida AS
2      SELECT r.rutaId, r.vehiculoId, r.horaInicio, r.horaFin, c.conductorId, c.
           nombre, c.vencimientoLicencia
3      FROM ruta r
4      JOIN conductor c ON r.conductorId = c.conductorId
5      WHERE r.estado = 'Completada' AND c.vencimientoLicencia < CURDATE();

```

rutaId	vehiculoId	horaInicio	horaFin	conductorId	nombre	vencimientoLicencia
1	3	17 2023-01-20 01:00:00	2023-01-28 11:00:00	2	Patricia Vázquez	2025-03-02

Figure 9: Ejecución de código. 1 filas

– 5. Fragmento de vehículos con mantenimientos en un intervalo de dos años y costos superiores a \$1000

Listing 10: Crear la vista para seleccionar a los vehiculos con sus mantenimientos que se realizaron en un intervalo de 2 años y superiores a \$1000

```

1      CREATE VIEW fragmento_vehiculos_mantenimientos_costosos AS
2      SELECT v.vehiculoId, v.tipo, v.modelo, m.mantenimientoId, m.fechaServicio, m.
           tipoServicio, m.costo
3      FROM vehiculo v
4      JOIN mantenimiento m ON v.vehiculoId = m.vehiculoId
5      WHERE m.fechaServicio >= DATE_SUB(CURDATE(), INTERVAL 2 YEAR) AND m.costo >
           1000;

```

vehiculoId	tipo	modelo	mantenimientoId	fechaServicio	tipoServicio	costo
1	23 SUV	Kenworth T680	6	2023-12-05	Reparación motor	2250.66
2	4 SUV	Chevrolet Silverado	14	2023-12-01	Revisión de frenos	2712.99
3	69 Pickup	Ford Transit	15	2023-10-16	Revisión de frenos	1473.41
4	2 SUV	Isuzu NPR	23	2023-12-15	Alineación	2870.10
5	79 Tráiler	Kenworth T680	24	2023-08-08	Reparación motor	1770.83
6	24 SUV	Freightliner Cascadia	27	2023-10-27	Cambio de filtros	2336.52
7	30 Camión	International LT	44	2024-05-06	Cambio de aceite	3087.32
8	60 Tráiler	Kenworth T680	49	2023-12-31	Reparación motor	1236.28
9	74 Sedán	Ram 1500	50	2024-05-09	Reparación motor	3235.76
10	61 SUV	International LT	52	2023-07-28	Revisión de frenos	2329.08
11	20 Tráiler	Hino 338	53	2023-04-23	Rotación de llantas	

Figure 10: Ejecución de código. 21 filas

5. Conclusiones

En esta práctica, se logró implementar la fragmentación horizontal y vertical de una base de datos MySQL, utilizando herramientas como MySQL, DataGrip y GitHub. A través de la fragmentación, se mejoró el rendimiento y la organización de los datos, dividiendo la información en partes más manejables y específicas.

Se utilizaron vistas para simular los fragmentos, permitiendo una distribución lógica de los datos sin alterar la estructura original de la base de datos.

La restauración de la base de datos a partir de un archivo de respaldo fue un proceso sencillo, y las vistas creadas sirvieron para separar la información de manera eficiente según criterios de uso. Al finalizar, se subieron los archivos y el reporte al repositorio de GitHub y se registró la URL en la Plataforma Garza para su evaluación.

Referencias Bibliográficas

References

- [1] García, J.; Sánchez, P. (2018). Introducción a la fragmentación de bases de datos. *Revista de Investigación en Tecnología y Sistemas*, 12(2), 45-57.
- [2] López, R.; Pérez, M. (2020). Fragmentación vertical y horizontal en bases de datos distribuidas. *Revista de Ciencias de la Computación*, 18(3), 80-92.
- [3] Martínez, F. (2021). El uso de vistas para la optimización de consultas en bases de datos. *Tecnología y Desarrollo Informático*, 25(4), 115-128.
- [4] Rodríguez, C.; Martínez, A. (2019). Modelos de fragmentación en bases de datos distribuidas. *Sistemas de Información y Gestión*, 22(1), 32-40.
- [5] Sánchez, D. (2022). Vistas en MySQL: Un enfoque práctico para mejorar el rendimiento. *Journal of Database Systems*, 10(3), 74-85.