

Universidad ORT Uruguay
Facultad de Ingeniería
Escuela de Tecnología
OBLIGATORIO BASES DE DATOS 2
DOCUMENTO DE ANÁLISIS



Gastón Jaurena – 342304



Matias Pietrafesa - 318802

Docente: Alberto Villar

Analista en Tecnologías de la Información

Fecha de entrega del documento (23-06-2025)

Contents

Evidencia de ejecución 3

Código fuente 10

 DDL..... 10

 DML 11

 Consultas SQL..... 12

 Procedure 14

 Función 15

 Triggers 16

 Vista..... 16

 MongoDB 17

Evidencia de ejecución

DDL.sql - HPPIETRAFESA\SQLEXPRESS01.DeviceService (HPPIETRAFESA\pietr (52)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

DDL.sql - HPPIETRAFESA\SQLEXPRESS01.DeviceService (HPPIETRAFESA\pietr (52)) ConsultasEjercicio3...TRAFESA\pietr (61)) DML.sql - HPPIETRAFESA\pietr (57))

USE DeviceService

GO

CREATE DATABASE DeviceService

GO

USE DeviceService

GO

CREATE TABLE Empleado (IdEmp int identity not null, NonEmp varchar(30) not null, FchNacEmp date not null, SueldoEmp money, TipoEmp char(1) not null, constraint pk_Empleado primary key (IdEmp), constraint ck_TipoEmp check (TipoEmp in ('T','C')))

GO

CREATE TABLE Producto (IdProd int identity not null, DscProd varchar(30) not null, StkProd int, CostoProd money, constraint pk_Producto primary key (IdProd))

GO

89 %

Messages

Commands completed successfully.

Completion time: 2020-06-23T08:04:05.5526567-03:00

Query executed successfully.

HPPIETRAFESA\SQLEXPRESS01 (HPPIETRAFESA\pietr (52)) DeviceService 00:00:00 0 rows

Item(s) Saved Ln 58 Col 3 INS

DML.sql - HPPIETRAFESA\SQLEXPRESS01.DeviceService (HPPIETRAFESA\pietr (57)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

DML.sql - HPPIETRAFESA\SQLEXPRESS01.DeviceService (HPPIETRAFESA\pietr (57)) ConsultasEjercicio3...TRAFESA\pietr (61)) DML.sql - HPPIETRAFESA\pietr (57))

USE DeviceService

GO

-- Insertar Empleados

INSERT INTO Empleado (NonEmp, FchNacEmp, SueldoEmp, TipoEmp) VALUES

((Juan Pérez', '1985-02-15', 34000, 'T'),

((María González', '1990-07-22', 55000, 'T'),

((Diego Fernández', '1975-11-05', 72000, 'C'),

((Laura Rodríguez', '1988-03-30', 25000, 'C'),

((Martín López', '1992-12-12', 65000, 'T'),

((Sofía Martínez', '1983-08-08', 50000, 'C'),

((Sebastián Gómez', '1970-01-20', 90000, 'T'),

((Valentina D'Angelo', '1995-09-17', 40000, 'T'),

((Camille Silva', '1993-04-10', 20000, 'C'))

GO

-- Insertar Productos

INSERT INTO Producto (DscProd, StkProd, CostoProd) VALUES

((Producto A', 100, 10000),

((Producto B', 200, 20000),

((Producto C', 300, 30000),

((Producto D', 400, 40000),

((Producto E', 500, 50000),

((Producto F', 600, 60000),

((Producto G', 700, 70000),

((Producto H', 800, 80000),

((Producto I', 900, 90000),

((Producto J', 1000, 100000)

GO

89 %

Messages

(9 rows affected)

(7 rows affected)

(10 rows affected)

(19 rows affected)

Completion time: 2020-06-23T08:04:22.5666393-03:00

Query executed successfully.

HPPIETRAFESA\SQLEXPRESS01 (HPPIETRAFESA\pietr (57)) DeviceService 00:00:00 0 rows

Item(s) Saved Ln 82 Col 1 Ch 1 INS

ConsultasEjercicio3.sql - HPPIETRAFESA\SQLEXPRESS01.DeviceService (HPPIETRAFESA\pietr (61)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

DeviceService

DDL.sql - HPPIETRA...RAFESA\pietr (52)* ConsultasEjercicio3...RAFESA\pietr (61) DML.sql - HPPIETRA...RAFESA\pietr (57)

```
--Consulta a.
Select
p.IdProd as [Codigo Producto],
p.DscProd as [Descripción],
SELECT COUNT(*)
FROM Unidad u, Repara r
WHERE u.NumSerie = r.NumSerie and u.IdProd = r.IdProd and u.IdProd = p.IdProd AND r.IdEmpQA IS NOT NULL as [Reparaciones con control de calidad],
SELECT COUNT(*)
FROM Unidad u, Repara r
WHERE u.NumSerie = r.NumSerie and u.IdProd = r.IdProd and u.IdProd = p.IdProd AND r.IdEmpQA IS NULL as [Reparaciones sin control de calidad],
SELECT COUNT(*)
FROM Unidad u, Repara r
WHERE u.NumSerie = r.NumSerie and u.IdProd = r.IdProd and u.IdProd = p.IdProd AND r.CostoRepara > 100 as [Reparaciones > $100]
From
Producto p;
```

89 %

	Codigo Producto	Descripción	Reparaciones con control de calidad	Reparaciones sin control de calidad	Reparaciones > \$100
1	1	Hiperpropulsor X	2	1	2
2	2	Motor de Plasma	2	2	2
3	3	Escudo Cuántico	2	1	1
4	4	Sensor Gravitacional	2	1	2
5	5	Drone Autónomo Estelar	1	1	2
6	6	Nave Interceptor Mk II	1	1	2
7	7	Crstal de Energía Zeta	1	1	1

Query executed successfully.

HPPIETRAFESA\SQLEXPRESS01 (... HPPIETRAFESA\pietr (61) DeviceService 00:00:00 7 rows

ConsultasEjercicio3.sql - HPPIETRAFESA\SQLEXPRESS01.DeviceService (HPPIETRAFESA\pietr (61)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

DeviceService

DDL.sql - HPPIETRA...RAFESA\pietr (52)* ConsultasEjercicio3...RAFESA\pietr (61) DML.sql - HPPIETRA...RAFESA\pietr (57)

```
FROM Unidad u, Repara r
WHERE u.NumSerie = r.NumSerie and u.IdProd = r.IdProd and u.IdProd = p.IdProd AND r.CostoRepara > 100 as [Reparaciones > $100]
From
Producto p;

--Consulta b.
Select top 1
e.*
FROM (r.IdRepara) as 'Reparaciones Realizadas'
From Empleado e, Repara r
where e.IdEmp = r.IdEmp
Group by e.IdEmp, e.NomEmp, e.FchNacEmp, e.SueIdoEmp, e.TipoEmp
Order by count(r.IdRepara) desc;

--Consulta c.
```

89 %

	IdEmp	NomEmp	FchNacEmp	SueIdoEmp	TipoEmp	Reparaciones Realizadas
1	1	Juan Pérez	1985-02-15	70000.00	T	8

Query executed successfully.

HPPIETRAFESA\SQLEXPRESS01 (... HPPIETRAFESA\pietr (61) DeviceService 00:00:00 1 rows

ConsultasEjercicio3.sql - HPPIETRAFESA\SQLEXPRESS01.DeviceService (HPPIETRAFESA\pietr (61)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

DeviceService

DDL.sql - HPPIETRA...RAFESA\pietr (52)* ConsultasEjercicio3...RAFESA\pietr (61) DML.sql - HPPIETRA...RAFESA\pietr (57)

```
Group by e.IdEmp, e.NomEmp, e.FchNacEmp, e.SueIdoEmp, e.TipoEmp
Order by count(r.IdRepara) desc;

--Consulta c.
Select
p.IdProd as [Id Producto],
p.DscProd as [Descripción],
p.StkProd as [Stock],
p.CostoProd as [Costo],
sum(r.CostoRepara) as [Costo total de reparaciones]
FROM Producto p, Repara r, Unidad u
where p.IdProd = u.IdProd and r.NumSerie = u.NumSerie and u.IdProd = r.IdProd
group by p.IdProd, p.DscProd, p.StkProd, p.CostoProd
having sum(r.CostoRepara) > 200;
```

89 %

	Id Producto	Descripción	Stock	Costo	Costo total de reparaciones
1	1	Hiperpropulsor X	10	1500.00	280.00
2	2	Motor de Plasma	5	2500.00	570.00
3	3	Escudo Cuántico	8	1200.00	350.00
4	4	Sensor Gravitacional	15	800.00	370.00
5	5	Drone Autónomo Estelar	3	5000.00	1200.00
6	6	Nave Interceptor Mk II	2	20000.00	900.00
7	7	Crstal de Energía Zeta	0	300.00	240.00

Query executed successfully.

HPPIETRAFESA\SQLEXPRESS01 (... HPPIETRAFESA\pietr (61) DeviceService 00:00:00 7 rows

ConsultasEjercicio3.sql - HPIETRAFESA\SQLEXPRESS01.DeviceService (HPIETRAFESA\pietr (61)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

DDL.sql - HPIETRA...RAFESA\pietr (52) ConsultasEjercicio3...RAFESA\pietr (61) DML.sql - HPIETRA...RAFESA\pietr (57)

```
--Consulta d
select top 1
    p.IdProd as [Id Producto],
    p.DescProd as [Descripcion],
    p.StkProd as [Stock],
    p.CostoProd as [Costo],
    count(r.IdRepara) as [Numero de reparaciones]
from Producto p, Repara r, Unidad u
where p.IdProd = u.IdProd and u.NumSerie = r.NumSerie and u.IdProd = r.IdProd
group by p.IdProd, p.DescProd, p.StkProd, p.CostoProd
order by count(r.IdRepara) desc;
```

Results Messages

ID Producto	Descripcion	Stock	Costo	Numero de reparaciones
1	Motor de Plasma	5	2500.00	4

Query executed successfully. HPIETRAFESA\SQLEXPRESS01 (... HPIETRAFESA\pietr (61) | DeviceService 00:00:00 1 rows

ConsultasEjercicio3.sql - HPIETRAFESA\SQLEXPRESS01.DeviceService (HPIETRAFESA\pietr (61)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

DDL.sql - HPIETRA...RAFESA\pietr (52) ConsultasEjercicio3...RAFESA\pietr (61) DML.sql - HPIETRA...RAFESA\pietr (57)

```
--Clasificación de salario (Alto > 66000, Medio 40000-66000, Bajo < 40000)
select
    e.IdEmp as [ID Empleado],
    e.NonEmp as [Nombre],
    e.TipoEmp as [Tipo],
    e.SueldoEmp as [Salario],
    [Nivel Salarial] = case
        when e.SueldoEmp > 66000 then 'Alto'
        when e.SueldoEmp BETWEEN 40000 AND 66000 then 'Medio'
        else 'Bajo'
    end,
    [Categoría] = case
        when e.TipoEmp = 'T' and e.SueldoEmp > 66000 then 'Senior (Tiempo Completo)'
        when e.TipoEmp = 'T' and e.SueldoEmp between 40000 and 66000 then 'Experimentado (Tiempo Completo)'
        when e.TipoEmp = 'J' then 'Junior (Tiempo Completo)'
        when e.TipoEmp = 'C' and e.SueldoEmp > 66000 then 'Senior (Contratado)'
        when e.TipoEmp = 'C' and e.SueldoEmp between 40000 and 66000 then 'Experimentado (Contratado)'
        when e.TipoEmp = 'C' and e.SueldoEmp < 40000 then 'Junior (Contratado)'
    end
from Empleado e
```

Results Messages

ID Empleado	Nombre	Tipo	Salario	Nivel Salarial	Categoría	Reparaciones Realizadas
7	Sebastián Gómez	T	90000.00	Alto	Senior (Tiempo Completo)	1
3	Diego Fernández	C	72000.00	Alto	Senior (Contratado)	2
5	Martín López	T	65000.00	Medio	Experimentado (Tiempo Completo)	2
2	Maria González	T	55000.00	Medio	Experimentado (Tiempo Completo)	3
6	Sofía Martínez	C	50000.00	Medio	Experimentado (Contratado)	1
8	Valentina D'Ángelo	T	40000.00	Medio	Experimentado (Tiempo Completo)	1
1	Juan Pérez	T	34000.00	Bajo	Junior (Tiempo Completo)	8
4	Laura Rodríguez	C	25000.00	Bajo	Junior (Contratado)	1
9	Camila Silva	C	20000.00	Bajo	Junior (Contratado)	0

Query executed successfully. HPIETRAFESA\SQLEXPRESS01 (... HPIETRAFESA\pietr (61) | DeviceService 00:00:00 9 rows

ConsultasEjercicio3.sql - HPIETRAFESA\SQLEXPRESS01.DeviceService (HPIETRAFESA\pietr (61)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

DDL.sql - HPIETRA...RAFESA\pietr (52) ConsultasEjercicio3...RAFESA\pietr (61) DML.sql - HPIETRA...RAFESA\pietr (57)

```
order by
    e.SueldoEmp desc,
    count(r.IdRepara) desc;

--Consulta f.
select
    [ISNULL(CAST(e.IdEmp AS VARCHAR(20)), 'TOTAL GENERAL')] AS [ID Empleado],
    sum(r.CostoRepara) as [Costo Total Reparaciones]
from Empleado e
left join Repara r ON e.IdEmp = r.IdEmp
group by e.IdEmp with rollup --El ROLLUP muestra los registros detallados, genera filas de subtotales y muestra el total general

--Consulta g
select e.IdEmp, e.NonEmp, e.FchMacEmp, e.SueldoEmp, e.TipoEmp
from Empleado e, Repara r, Unidad u
```

Results Messages

ID Empleado	Costo Total Reparaciones
1	2280.00
2	210.00
3	150.00
4	200.00
5	650.00
6	50.00
7	250.00
8	120.00
9	NULL
10	TOTAL GENERAL 3910.00

Query executed successfully. HPIETRAFESA\SQLEXPRESS01 (... HPIETRAFESA\pietr (61) | DeviceService 00:00:00 10 rows

ConsultasEjercicio3.sql - HPIETRAFESA\SQLEXPRESS01.DeviceService (HPIETRAFESA\pietr (61)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

DDL.sql - HPIETRA...RAFESA\pietr (52) ConsultasEjercicio3...RAFESA\pietr (61)* DML.sql - HPIETRA...RAFESA\pietr (57)

```

SELECT
    ISNULL(CAST(e.IdEmp AS VARCHAR(20)), 'TOTAL GENERAL') AS [ID Empleado],
    sum(r.CostoRepara) as [Costo Total Reparaciones]
from Empleado e
left join Repara r ON e.IdEmp = r.IdEmp
group by e.IdEmp with rollup --El ROLLUP muestra los registros detallados, genera filas de subtotales y muestra el total general

--Consulta g
select e.IdEmp, e.NomEmp, e.FchMacEmp, e.SueldoEmp, e.TipoEmp
from Empleado e, Repara r, Unidad u
where e.IdEmp = r.IdEmp and r.NumSerie = u.NumSerie and r.IdProd = u.IdProd and e.TipoEmp = 'T'
group by e.IdEmp, e.NomEmp, e.FchMacEmp, e.SueldoEmp, e.TipoEmp
having count(distinct u.IdProd) = (select count(*) from Producto)

```

89 %

IdEmp	NomEmp	FchMacEmp	SueldoEmp	TipoEmp
1	Juan Pérez	1985-02-15	34000.00	T

Query executed successfully.

HPPIETRAFESA\SQLEXPRESS01 (... HPIETRAFESA\pietr (61) DeviceService 00:00:00 1 rows

ConsultasEjercicio3.sql - HPIETRAFESA\SQLEXPRESS01.DeviceService (HPIETRAFESA\pietr (61)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

DDL.sql - HPIETRA...RAFESA\pietr (52) ConsultasEjercicio3...RAFESA\pietr (61)* DML.sql - HPIETRA...RAFESA\pietr (57)

```

having count(distinct u.IdProd) = (select count(*) from Producto)

--4.a
create procedure spu_RegistrarReparacion
@NumSerie char(10),
@IdProd int,
@IdEmp int,
@CostoRepara money
as
begin
    declare @mensaje varchar(300);
    declare @fecha datetime = getdate();
    --Insertar aquí la lógica de la consulta

```

89 %

Messages

Command completed successfully.

Completion time: 2025-06-20T09:50:29.2175860-03:00

Query executed successfully.

HPPIETRAFESA\SQLEXPRESS01 (... HPIETRAFESA\pietr (61) DeviceService 00:00:00 0 rows

test@localhost - NoSQLBooster for MongoDB

File Edit Options View Favorites Tools Window Help

Connect Import Export Monitoring Tasks DataGen Schema

testordenes@localhost x test@localhost x

reparaciones

```

64 })
65
66 db.reparaciones.insertOne({
67     "idRepara": "REP-234",
68     "NumSerie": "234",
69     "idEmp": "EMP-234",
70     "idEmpQA": "EMPQA-234",
71     "costoRepara": 5000,
72     "stsRepara": "En proceso",
73     "idProd": "PROD-234",
74     "fchRepara": "24-05-2025".

```

0.136 s

Key	Value	Type
acknowledged	true	Bool
insertedid	6859448d88e3202bdf3cbe5	Objectid

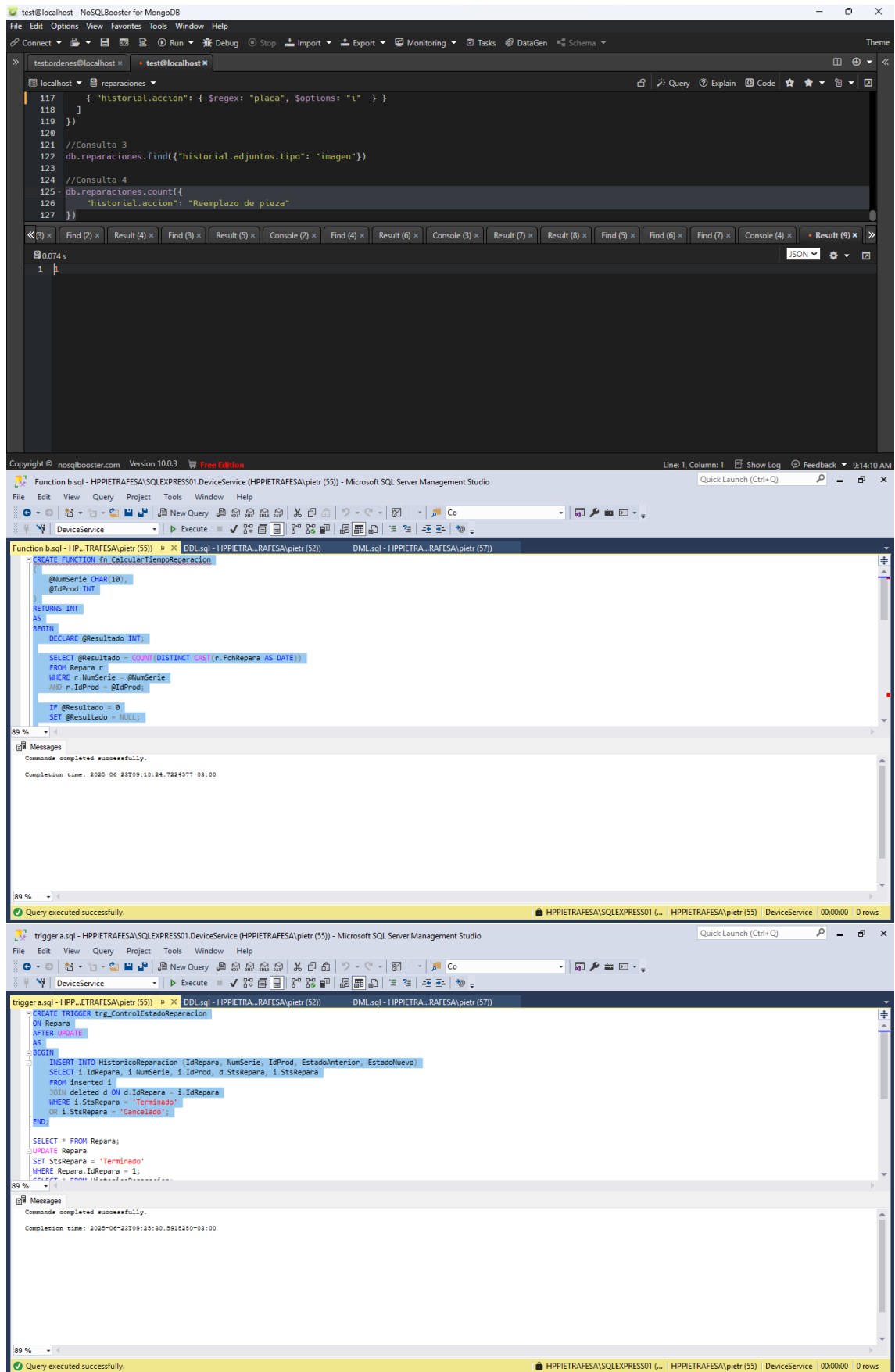
Copyright © nosqlbooster.com Version 10.0.3 Free Edition

Line: 103, Column: 1 (2838 selected) Show Log Feedback 9:12:06 AM

```
test@localhost - NoSQLBooster for MongoDB
File Edit Options View Favorites Tools Window Help
Connect Import Export Monitoring Tasks DataGen Schema
testordenes@localhost test@localhost
localhost reparaciones
101 }
102 }}
103
104
105
106
107
108
109 //Consulta 1
110 db.reparaciones.find({"idRepara": "REP-123"}, {"historial": 1})
111
Result (1) x Find (1) x Result (2) x Find (1) x Result (3) x Find (2) x Result (4) x Find (3) x Result (5) x Console (2) x Find (4) x Result (6) x Console (3) x Result (7) x Result (8) x Find (5) x
reparaciones 0.084 s 1 Doc
1- {
2   "_id" : ObjectId("685944d8d8e3202bfd3c6be4"),
3   "historial" : [
4     {
5       "idEvento" : "EVE-123",
6       "accion" : "Se identifico el problema",
7       "descripcion" : "Al equipo no le anda la pantalla debido a descuido",
8       "notas" : [ "Se identifica el problema de la pantalla y se procede a enviar a reparacion" ],
9       "fecha" : ISODate("2025-05-14T11:00:00.000-03:00"),
10      "adjuntos" : [
11        {
12          "tipo" : "imagen",
13          "descripcion" : "Foto de pantalla descuidada",
14          "url" : "imagenesAdjuntas/pantalla-03.jpg"
15        }
16      ]
17    }
18  ]
19 }
```

```
test@localhost - NoSQLBooster for MongoDB
File Edit Options View Favorites Tools Window Help
Connect Import Export Monitoring Tasks DataGen Schema
testordenes@localhost test@localhost
localhost reparaciones
110 db.reparaciones.find({"idRepara": "REP-123"}, {"historial": 1})
111
112 //Consulta 2
113 db.reparaciones.find(
114   $or: [
115     { "historial.descripcion": { $regex: "placa", $options: "i" } },
116     { "historial.notas": { $regex: "placa", $options: "i" } },
117     { "historial.accion": { $regex: "placa", $options: "i" } }
118   ]
119 )
120
Find x Result (2) x Find (1) x Result (3) x Find (2) x Result (4) x Find (3) x Result (5) x Console (2) x Find (4) x Result (6) x Console (3) x Result (7) x Result (8) x Find (5) x Find (6) x
reparaciones 0.065 s 1 Doc
1- {
2   "_id" : ObjectId("685944d8d8e3202bfd3c6be5"),
3   "idRepara" : "REP-234",
4   "NumSerie" : "234",
5   "idEmp" : "EMP-234",
6   "idEmpQA" : "EMPQA-234",
7   "costoRepara" : 5000,
8   "stsRepara" : "En proceso",
9   "idProd" : "PROD-234",
10  "fchRepara" : "24-05-2025",
11  "historial" : [
12    {
13      "idEvento" : "EVE-234",
14      "accion" : "Se identifico el problema",
15      "descripcion" : "El equipo tiene sucia la placa",
16      "notas" : [ "Se identifica el problema de la placa y se procede a enviar a reparacion" ],
17    }
18  ]
19 }
```

```
test@localhost - NoSQLBooster for MongoDB
File Edit Options View Favorites Tools Window Help
Connect Import Export Monitoring Tasks DataGen Schema
testordenes@localhost test@localhost
localhost reparaciones
113 { "historial.descripcion": { $regex: "placa", $options: "i" } },
116 { "historial.notas": { $regex: "placa", $options: "i" } },
117 { "historial.accion": { $regex: "placa", $options: "i" } }
118 ]
119 }
120
121 //Consulta 3
122 db.reparaciones.find({"historial.adjuntos.tipo": "imagen"})
123
124 //Consulta 4
125 db.reparaciones.count(
126   "historial.accion": "Se identifico el problema"
127 )
128
Alt (2) x Find (1) x Result (3) x Find (2) x Result (4) x Find (3) x Result (5) x Console (2) x Find (4) x Result (6) x Console (3) x Result (7) x Result (8) x Find (5) x Find (6) x Find (7) x
reparaciones 0.090 s 1 Doc
1- {
2   "_id" : ObjectId("685944d8d8e3202bfd3c6be4"),
3   "idRepara" : "REP-123",
4   "NumSerie" : "123",
5   "idEmp" : "EMP-123",
6   "idEmpQA" : "EMPQA-123",
7   "costoRepara" : 2000,
8   "stsRepara" : "Inticado",
9   "idProd" : "PROD-123",
10  "fchRepara" : "12-06-2025",
11  "historial" : [
12    {
13      "idEvento" : "EVE-123",
14      "accion" : "Se identifico el problema",
15      "descripcion" : "Al equipo no le anda la pantalla debido a descuido",
16      "notas" : [ "Se identifica el problema de la pantalla y se procede a enviar a reparacion" ],
17    }
18  ]
19 }
```



trigger b.sql - HPIETRAFESA\SQLEXPRESS01.DeviceService (HPIETRAFESA\pietr (55)) - Microsoft SQL Server Management Studio

```
CREATE TRIGGER trg_PrevenirEliminacionReparaciones
ON Reparacion
INSTEAD OF DELETE
AS
BEGIN
    DELETE r FROM Reparacion r
    JOIN deleted d ON r.IdReparacion = d.IdReparacion
    WHERE r.StsReparacion = 'Iniciado'
    OR r.StsReparacion = 'Cancelado';

    INSERT INTO HistoricoEliminacionReparaciones (IdReparacion, NumSerie, IdProd, StsReparacion)
    SELECT d.IdReparacion, d.NumSerie, d.IdProd, d.StsReparacion
    FROM deleted d
    WHERE d.StsReparacion = 'Iniciado'
    OR d.StsReparacion = 'Cancelado';

    SELECT d.IdReparacion, d.NumSerie, d.IdProd, d.StsReparacion, 'No se permite eliminar' AS Mensaje
    FROM deleted d
    WHERE d.StsReparacion = 'Iniciado'
    OR d.StsReparacion = 'Cancelado';
END
```

Messages

Command completed successfully.

Completion time: 2025-06-20T09:21:42.6970210-03:00

89 %

Query executed successfully. HPIETRAFESA\SQLEXPRESS01 (... HPIETRAFESA\pietr (55)) DeviceService 00:00:00 0 rows

View.sql - HPIETRAFESA\SQLEXPRESS01.DeviceService (HPIETRAFESA\pietr (51)) - Microsoft SQL Server Management Studio

```
CREATE VIEW vw_ReparacionesActivas
AS
SELECT r.IdReparacion, r.NumSerie, r.IdProd, p.DscProd, e.NomEmp, r.FchReparacion, r.StsReparacion
FROM Reparacion r
JOIN Producto p ON r.IdProd = p.IdProd
JOIN Empleado e ON e.IdEmp = r.IdEmp
WHERE r.StsReparacion = 'Iniciado'
OR r.StsReparacion = 'En testing';
```

Messages

Command completed successfully.

Completion time: 2025-06-20T09:25:54.1107957-03:00

89 %

Query executed successfully. HPIETRAFESA\SQLEXPRESS01 (... HPIETRAFESA\pietr (51)) DeviceService 00:00:00 0 rows

Código fuente

DDL

```
CREATE DATABASE DeviceService
GO
USE DeviceService
GO
CREATE TABLE Empleado(IdEmp int identity not null,
                        NomEmp varchar(30) not null,
                        FchNacEmp date not null,
                        SueldoEmp money,
                        TipoEmp char(1) not null,
                        constraint pk_Empleado primary key(IdEmp),
                        constraint ck_TipoEmp check(TipoEmp in
('T','C')))
GO
CREATE TABLE Producto(IdProd int identity not null,
                        DscProd varchar(30) not null,
                        StkProd int,
                        CostoProd money,
                        constraint pk_Producto primary key(IdProd))
GO
CREATE TABLE Unidad(NumSerie character(10) not null,
                      IdProd int not null,
                      FchFab date,
                      FchVto date,
                      constraint pk_Unidad primary
key(NumSerie,IdProd),
                      constraint fk_ProdUnidad foreign key(IdProd)
references Producto(IdProd))
GO
CREATE TABLE Repara(IdRepara int identity not null,
                      NumSerie character(10) not null,
                      IdProd int not null,
                      IdEmp int not null,
                      FchRepara datetime not null,
                      CostoRepara money,
                      StsRepara varchar(20) default 'Iniciado',
                      IdEmpQA int,
                      constraint pk_Repara primary key(IdRepara),
                      constraint uk_Repara
unique(NumSerie,IdProd,IdEmp,FchRepara),
                      constraint ck_StsRepara check(StsRepara in
('Iniciado','En testing','Terminado','Cancelado')),
                      constraint fk_UnidadRepara foreign
key(NumSerie,IdProd) references Unidad(NumSerie,IdProd),
                      constraint fk_EmpRepara foreign key(IdEmp)
references Empleado(IdEmp),
                      constraint fk_EmpQA foreign key(IdEmpQA)
references Empleado(IdEmp))
GO
CREATE TABLE HistoricoReparacion (
    IdHist INT IDENTITY PRIMARY KEY,
    IdRepara INT NOT NULL,
    NumSerie CHAR(10) NOT NULL,
    IdProd INT NOT NULL,
    EstadoAnterior VARCHAR(20) NOT NULL,
    EstadoNuevo VARCHAR(20) NOT NULL,
    FchCambio DATETIME DEFAULT GETDATE());
GO
CREATE TABLE HistoricoEliminacionReparaciones (
```

```

        IdHist INT IDENTITY PRIMARY KEY,
        IdRepara INT NOT NULL,
        NumSerie CHAR(10) NOT NULL,
        IdProd INT NOT NULL,
        StsRepara VARCHAR(20) NOT NULL,
        FchEliminacion DATETIME DEFAULT GETDATE()
    );
GO

```

DML

```

USE DeviceService
GO

```

```

-- Insertar Empleados
INSERT INTO Empleado (NomEmp, FchNacEmp, SueldoEmp, TipoEmp) VALUES
    ('Juan Pérez', '1985-02-15', 34000, 'T'),
    ('María González', '1990-07-22', 55000, 'T'),
    ('Diego Fernández', '1975-11-05', 72000, 'C'),
    ('Laura Rodríguez', '1988-03-30', 25000, 'C'),
    ('Martín López', '1992-12-12', 65000, 'T'),
    ('Sofía Martínez', '1983-08-08', 50000, 'C'),
    ('Sebastián Gómez', '1970-01-20', 90000, 'T'),
    ('Valentina D'Angelo', '1995-09-17', 40000, 'T'),
    ('Camila Silva', '1993-04-10', 20000, 'C');
GO

```

```

-- Insertar Productos
INSERT INTO Producto (DscProd, StkProd, CostoProd) VALUES
    ('Hiperpropulsor X', 10, 1500),
    ('Motor de Plasma', 5, 2500),
    ('Escudo Cuántico', 8, 1200),
    ('Sensor Gravitacional', 15, 800),
    ('Drone Autónomo Estelar', 3, 5000),
    ('Nave Interceptor Mk II', 2, 20000),
    ('Cristal de Energía Zeta', 0, 300);
GO

```

```

-- Insertar Unidades
INSERT INTO Unidad (NumSerie, IdProd, FchFab, FchVto) VALUES
    ('HPX0000001', 1, '2023-05-20', '2025-05-20'),
    ('HPX0000002', 1, '2024-12-01', NULL),
    ('MPL0000001', 2, '2022-11-11', '2025-11-11'),
    ('MPL0000002', 2, '2023-07-07', '2024-07-07'),
    ('ESQ0000001', 3, '2024-03-03', '2026-03-03'),
    ('SNG0000001', 4, '2025-01-01', '2027-01-01'),
    ('SNG0000002', 4, '2023-02-02', '2024-02-02'),
    ('DAS0000001', 5, '2025-06-01', NULL),
    ('NIM0000001', 6, '2024-08-08', '2026-08-08'),
    ('CEZ0000001', 7, '2023-03-03', '2025-03-03');
GO

```

```

-- Insertar Reparaciones
INSERT INTO Repara (NumSerie, IdProd, IdEmp, FchRepara, CostoRepara,
    StsRepara, IdEmpQA) VALUES
    -- Producto 4, unidad SNG0000001, Juan Pérez (IdEmp=1)
    ('SNG0000001', 4, 1, '2025-01-05 08:00:00', 150, 'En testing', 7),
    -- Producto 1, unidad HPX0000001, Sofía Martínez (IdEmp=6)
    ('HPX0000001', 1, 6, '2025-02-01 12:00:00', 50, 'Iniciado', NULL),
    -- Producto 3, unidad ESQ0000001, Juan Pérez (dos reparaciones el mismo día)
    ('ESQ0000001', 3, 1, '2025-02-10 11:00:00', 300, 'Terminado', 2),
    ('ESQ0000001', 3, 1, '2025-02-10 15:00:00', 20, 'Iniciado', NULL),
    -- Producto 3, unidad ESQ0000001, Diego Fernández (IdEmp=2 en la FK de QA)

```

```

('ESQ0000001', 3, 2, '2025-03-01 10:00:00', 30, 'Terminado', 5),
-- Producto 4, unidad SNG0000002, Diego Fernández
('SNG0000002', 4, 3, '2025-03-05 14:00:00', 100, 'Cancelado', NULL),
-- Producto 2, unidad MPL0000001, Juan Pérez
('MPL0000001', 2, 1, '2025-03-20 14:30:00', 200, 'Terminado', 5),
-- Producto 2, unidad MPL0000001, Diego Fernández
('MPL0000001', 2, 3, '2025-03-20 16:00:00', 50, 'Cancelado', NULL),
-- Producto 1, unidad HPX0000001, Juan Pérez
('HPX0000001', 1, 1, '2025-04-01 09:00:00', 120, 'Terminado', 2),
-- Producto 2, unidad MPL0000001, Sebastián Gómez (IdEmp=7)
('MPL0000001', 2, 7, '2025-04-01 14:00:00', 250, 'Terminado', 2),
-- Producto 1, unidad HPX0000002, María González (IdEmp=2)
('HPX0000002', 1, 2, '2025-04-02 10:00:00', 110, 'Terminado', 1),
-- Producto 4, unidad SNG0000001, Valentina D'Angelo (IdEmp=8)
('SNG0000001', 4, 8, '2025-04-10 10:00:00', 120, 'En testing', 1),
-- Producto 6, unidad NIM0000001, Juan Pérez
('NIM0000001', 6, 1, '2025-04-20 09:30:00', 400, 'Terminado', 2),
-- Producto 2, unidad MPL0000002, María González
('MPL0000002', 2, 2, '2025-05-01 09:00:00', 70, 'Iniciado', NULL),
-- Producto 7, unidad CEZ0000001, Juan Pérez
('CEZ0000001', 7, 1, '2025-05-10 13:00:00', 90, 'Cancelado', NULL),
-- Producto 5, unidad DAS0000001, Laura Rodríguez (IdEmp=4)
('DAS0000001', 5, 4, '2025-05-20 16:00:00', 200, 'Terminado', 2),
-- Producto 5, unidad DAS0000001, Juan Pérez
('DAS0000001', 5, 1, '2025-06-01 12:00:00', 1000, 'Iniciado', NULL),
-- Producto 6, unidad NIM0000001, Martín López (IdEmp=5)
('NIM0000001', 6, 5, '2025-06-02 08:00:00', 500, 'Iniciado', NULL),
-- Producto 7, unidad CEZ0000001, Martín López
('CEZ0000001', 7, 5, '2025-06-03 09:00:00', 150, 'En testing', 3);
GO

```

Consultas SQL

--1 Indices que consideramos utiles para las consultas:

--Tabla Producto:

```
create index idx_producto_DscProd on Producto(DscProd);
```

--Tabla Empleado:

```
create index idx_empleado_SueldoEmp on Empleado(SueldoEmp)
```

```
create index idx_empleado_TipoEmp on Empleado(TipoEmp)
```

--Tabla Unidad

```
create index idx_unidad_IdProd on Unidad(IdProd)
```

```
create index idx_unidad_NumSerie on Unidad(NumSerie)
```

--Tabla Repara

```
create index idx_Repara_IdEmp on Repara(IdEmp);
```

```
create index idx_Repara_CostoRepara on Repara(CostoRepara);
```

```
create index idx_Repara_StsRepara on Repara(StsRepara);
```

```
create index idx_Repara_FchRepara on Repara(FchRepara);
```

--3 Implementar las consultas SQL

--Consulta a.

```
Select
```

```
p.IdProd as [Codigo Producto],
```

```
p.DscProd as [Descripcion],
```

```
(SELECT COUNT(*)
```

```
FROM Unidad u, Repara r
```

```

        WHERE u.NumSerie = r.NumSerie and u.IdProd = r.IdProd and u.IdProd =
p.IdProd AND r.IdEmpQA IS NOT NULL) as [Reparaciones con control de calidad],
(SELECT COUNT(*)
FROM Unidad u, Repara r
WHERE u.NumSerie = r.NumSerie and u.IdProd = r.IdProd and u.IdProd =
p.IdProd AND r.IdEmpQA IS NULL) as [Reparaciones sin control de calidad],
(SELECT COUNT(*)
FROM Unidad u, Repara r
WHERE u.NumSerie = r.NumSerie and u.IdProd = r.IdProd and u.IdProd =
p.IdProd AND r.CostoRepara > 100) as [Reparaciones > $100]
From
Producto p;

```

--Consulta b.

```

Select top 1
e.*,
count(r.IdRepara) as 'Reparaciones Realizadas'
From Empleado e, Repara r
where e.IdEmp = r.IdEmp
Group by e.IdEmp, e.NomEmp, e.FchNacEmp, e.SueldoEmp, e.TipoEmp
Order by count(r.IdRepara) desc;

```

--Consulta c.

```

Select
p.IdProd as [Id Producto],
p.DscProd as [Descripcion],
p.StkProd as [Stock],
p.CostoProd as [Costo],
sum(r.CostoRepara) as [Costo total de reparaciones]
FROM Producto p, Repara r, Unidad u
where p.IdProd = u.IdProd and r.NumSerie = u.NumSerie and u.IdProd = r.IdProd
group by p.IdProd, p.DscProd, p.StkProd, p.CostoProd
having sum(r.CostoRepara) > 200;

```

--Consulta d

```

Select top 1
p.IdProd as [Id Producto],
p.DscProd as [Descripcion],
p.StkProd as [Stock],
p.CostoProd as [Costo],
count(r.IdRepara) as [Numero de reparaciones]
From Producto p, Repara r, Unidad u
where p.IdProd = u.IdProd and u.NumSerie = r.NumSerie and u.IdProd = r.IdProd
group by p.IdProd, p.DscProd, p.StkProd, p.CostoProd
order by count(r.IdRepara) desc;

```

--Consulta e.

-- Clasificación de salario (Alto > 66000, Medio 40000-66000, Bajo < 40000)

```

select
e.IdEmp as [ID Empleado],
e.NomEmp as [Nombre],
e.TipoEmp as [Tipo],
e.SueldoEmp as [Salario],
[Nivel Salarial] = case
when e.SueldoEmp > 66000 then 'Alto'
when e.SueldoEmp BETWEEN 40000 AND 66000 then 'Medio'

```

```

        else 'Bajo'
    end,
    [Categoria] = case
        when e.TipoEmp = 'T' and e.SueldoEmp > 66000 then 'Senior (Tiempo
Completo)'
        when e.TipoEmp = 'T' and e.SueldoEmp between 40000 and 66000 THEN
'Experimentado (Tiempo Completo)'
        when e.TipoEmp = 'T' then 'Junior (Tiempo Completo)'
        when e.TipoEmp = 'C' and e.SueldoEmp > 66000 then 'Senior
(Contratado)'
        when e.TipoEmp = 'C' and e.SueldoEmp between 40000 AND 66000 then
'Experimentado (Contratado)'
        else 'Junior (Contratado)'
    end,
    count(r.IdRepara) [Reparaciones Realizadas]
from
    Empleado e
    LEFT JOIN Repara r ON e.IdEmp = r.IdEmp
group by
    e.IdEmp, e.NomEmp, e.TipoEmp, e.SueldoEmp
order by
    e.SueldoEmp desc,
    count(r.IdRepara) desc;

--Consulta f.

SELECT
    ISNULL(CAST(e.IdEmp AS VARCHAR(20)), 'TOTAL GENERAL') AS [ID Empleado],
    sum(r.CostoRepara) as [Costo Total Reparaciones]
from Empleado e
left join Repara r ON e.IdEmp = r.IdEmp
group by e.IdEmp with rollup --El ROLLUP muestra los registros detallados, genera
filas de subtotales y muestra el total general

--Consulta g

select e.IdEmp, e.NomEmp, e.FchNacEmp, e.SueldoEmp, e.TipoEmp
from Empleado e, Repara r, Unidad u
where e.IdEmp = r.IdEmp and r.NumSerie = u.NumSerie and r.IdProd = u.IdProd and
e.TipoEmp = 'T'
group by e.IdEmp, e.NomEmp, e.FchNacEmp, e.SueldoEmp, e.TipoEmp
having count(distinct u.IdProd) = (select count(*) from Producto)

```

Procedure

```

create procedure spu_RegistrarReparacion
@NumSerie char(10),
@IdProd int,
@IdEmp int,
@CostoRepara money
as
begin

declare @mensaje varchar(300);
declare @fecha datetime = getdate();

begin try
-----

```

```

if not exists (select 1 from Unidad u where u.NumSerie = @NumSerie and u.IdProd =
@IdProd)
begin
    set @mensaje = 'La Unidad no existe en la tabla unidad'
    select @mensaje;
    return;
end
if not exists (select 1 from Empleado e where e.IdEmp= @IdEmp)
begin
    set @mensaje = 'El empleado no existe en la tabla empleado'
    select @mensaje;
    return;
end

if exists (select r.IdRepara
from Repara r
where r.NumSerie = @NumSerie
and r.IdEmp = @IdEmp
and r.IdProd = @IdProd
and CONVERT(DATE, FchRepara) = CONVERT(DATE, @fecha))
begin
    set @mensaje = 'El empleado ya tiene una reparación para la misma
unidad hoy';
    select @mensaje;
    return;
end

if(0 > @CostoRepara)
begin
    set @mensaje = 'El costo de reparación no puede ser negativo.'
    select @mensaje;
    return;
end

--Si no pasa nada de lo anterior
INSERT INTO Repara (NumSerie,IdProd,IdEmp,FchRepara,CostoRepara,
StsRepara,IdEmpQA)
VALUES ( @NumSerie, @IdProd,@IdEmp,@fecha,@CostoRepara,'Iniciado',NULL);
    set @mensaje = 'Reparacion insertada.'
    select @mensaje;
    return;

-----
end try
begin catch
    set @mensaje = 'Error';
    select @mensaje;
end catch
end;

```

Función

```

CREATE FUNCTION fn_CalcularTiempoReparacion
(
    @NumSerie CHAR(10),
    @IdProd INT
)
RETURNS INT
AS
BEGIN

```

```

DECLARE @Resultado INT;

SELECT @Resultado = COUNT(DISTINCT CAST(r.FchRepara AS DATE))
FROM Repara r
WHERE r.NumSerie = @NumSerie
AND r.IdProd = @IdProd;

IF @Resultado = 0
SET @Resultado = NULL;

RETURN @Resultado;
END;

```

Triggers

```

CREATE TRIGGER trg_ControlEstadoReparacion
ON Repara
AFTER UPDATE
AS
BEGIN
    INSERT INTO HistoricoReparacion (IdRepara, NumSerie, IdProd,
EstadoAnterior, EstadoNuevo)
    SELECT i.IdRepara, i.NumSerie, i.IdProd, d.StsRepara, i.StsRepara
    FROM inserted i
    JOIN deleted d ON d.IdRepara = i.IdRepara
    WHERE i.StsRepara = 'Terminado'
    OR i.StsRepara = 'Cancelado';
END;

CREATE TRIGGER trg_PrevenirEliminacionReparaciones
ON Repara
INSTEAD OF DELETE
AS
BEGIN
    DELETE r FROM Repara r
    JOIN deleted d ON r.IdRepara = d.IdRepara
    WHERE r.StsRepara = 'Iniciado'
    OR r.StsRepara = 'Cancelado';

    INSERT INTO HistoricoEliminacionReparaciones (IdRepara, NumSerie, IdProd,
StsRepara)
    SELECT d.IdRepara, d.NumSerie, d.IdProd, d.StsRepara
    FROM deleted d
    WHERE d.StsRepara = 'Iniciado'
    OR d.StsRepara = 'Cancelado';

    SELECT d.IdRepara, d.NumSerie, d.IdProd, d.StsRepara, 'No se permite
eliminar' AS Mensaje
    FROM deleted d
    WHERE d.StsRepara = 'En testing'
    OR d.StsRepara = 'Terminado';
END;

```

Vista

```

CREATE VIEW vw_ReparacionesActivas
AS
SELECT r.IdRepara, r.NumSerie, r.IdProd, p.DscProd, e.NomEmp, r.FchRepara,
r.StsRepara
FROM Repara r

```



```
JOIN Producto p ON r.IdProd = p.IdProd
JOIN Empleado e ON e.IdEmp = r.IdEmp
WHERE r.StsRepara = 'Iniciado'
OR r.StsRepara = 'En testing';
```

MongoDB

show dbs

use reparaciones

show collections

db.reparaciones.find()

db.reparaciones.insertOne({

 "idRepara": "REP-123",

 "NumSerie": "123",

 "idEmp": "EMP-123",

 "idEmpQA": "EMPQA-123",

 "costoRepara": 2000,

 "stsRepara": "Iniciado",

 "idProd": "PROD-123",

 "fchRepara": "12-06-2025",

 "historial": [

 {

 "idEvento": "EVE-123",

 "accion": "Se identifico el problema",

 "descripcion": "Al equipo no le anda la pantalla debido a descuido",

 "notas": [

 "Se identifica el problema de la pantalla y se procede a enviar a reparacion"

],

 "fecha": ISODate("2025-05-14T14:00:00Z"),

 "adjuntos": [

```
{
  "tipo": "imagen",
  "descripcion": "Foto de pantalla descuidada",
  "url": "imagenesAdjuntas/pantalla-03.jpg"
}
],
{
  "idEvento": "EVE-456",
  "accion": "Reemplazo de pieza",
  "descripcion": "Se le cambia la pieza",
  "notas": [
    "Se le cambio la pantalla"
  ],
  "fecha": ISODate("2025-05-16T14:15:00Z"),
  "adjuntos": [
    {
      "tipo": "imagen",
      "descripcion": "Foto de pantalla cambiada",
      "url": "imagenesAdjuntas/pantalla-03.jpg"
    },
    {
      "tipo": "documento",
      "descripcion": "Guia de cuidado de pantallas",
      "url": "documentosAdjuntos/pantalla-guia.pdf"
    }
  ]
}
]
```

```
}}
```

```
db.reparaciones.insertOne({
  "idRepara": "REP-234",
  "NumSerie": "234",
  "idEmp": "EMP-234",
  "idEmpQA": "EMPQA-234",
  "costoRepara": 5000,
  "stsRepara": "En proceso",
  "idProd": "PROD-234",
  "fchRepara": "24-05-2025",
  "historial": [
    {
      "idEvento": "EVE-234",
      "accion": "Se identifico el problema",
      "descripcion": "El equipo tiene sucia la placa",
      "notas": [
        "Se identifica el problema de la placa y se procede a enviar a reparacion"
      ],
      "fecha": ISODate("2025-05-25T14:00:00Z"),
      "adjuntos": [
        {
          "tipo": "documento",
          "descripcion": "Archivo de presentacion de limpieza de placa sucia",
          "url": "documentosAdjuntos/placa-05.pdf"
        }
      ]
    }
  ],
},
{
  "idEvento": "EVE-345",
  "accion": "Limpieza",
```

```
    "descripcion": "Se le limpia la placa al dispositivo",
    "notas": [
        "Se limpio satisfactoriamente"
    ],
    "fecha": ISODate("2025-05-25T14:00:00Z")
  }
]
})
```

//Consulta 1

```
db.reparaciones.find({"idRepara": "REP-123"}, {"historial": 1})
```

//Consulta 2

```
db.reparaciones.find({
  $or: [
    { "historial.descripcion": { $regex: "placa", $options: "i" } },
    { "historial.notas": { $regex: "placa", $options: "i" } },
    { "historial.accion": { $regex: "placa", $options: "i" } }
  ]
})
```

//Consulta 3

```
db.reparaciones.find({"historial.adjuntos.tipo": "imagen"})
```

//Consulta 4

```
db.reparaciones.count({
```

"historial.accion": "Reemplazo de pieza"

})