



Base de datos con SQL SERVER

Proyecto Final:

**Gestión de Indicadores Diarios Sistema de
Microfinanzas**

Autor:

Aníbal Junior Unda Tejada

Jhenss Malkyn Chavez Cajigas

Docente:

Ing. Kevin Rivera Vergaray

Perú

2025

I. Iniciación: Definición de los objetivos y el alcance inicial de la base de datos.

Título del Problema: Gestión de Indicadores Diarios Sistema de Microfinanzas

Descripción del Problema:

Las entidades de microfinanzas cuentan con sistemas tecnológicos que le permiten tener información de sus clientes y créditos otorgados a una vista mensual; no obstante, la información cambia en el día a día; por lo que se necesita información organizada, robusta y consistente en el tiempo que tengan datos de cada crédito a nivel nacional organizado y jerarquizado según las áreas de la entidad microfinanciera..

En cada crédito se detallan las principales características del mismo, el código que lo identifica, el analista que originó el crédito y su categoría, el analista que es responsable del crédito y su categoría, el titular del crédito, el saldo otorgado (desembolsado), saldo actual, estado del crédito; etc.

Se necesita un sistema para organizar y administrar la gestión de los indicadores que se pueden obtener del estado de estos créditos de forma diaria, distinguiendo a los responsables y agencias a las que pertenece, el sistema debe centralizar la información y optimizar la gestión de los recursos.

Requerimientos Funcionales:

1. Registrar información básica del crédito(titular, saldo otorgado, saldo actual, estado).
2. Gestionar a los responsables:
 - ❖ Analista origen (que se encargó de evaluar y desembolsar el crédito) y categoría del analista.
 - ❖ Analista actual (puede ser el mismo que origen o puede reemplazar a un analista origen) y categoría del analista
 - ❖ Gerente de la agencia donde se encuentra el crédito y su categoría.
3. Información de la agencia donde se gestionó el crédito(nombre de la agencia, distrito, provincia, región, estado de la agencia).
4. Gestionar créditos (almacenar los créditos, a sus responsables y relacionar sus agencias).
5. Almacenar indicadores diarios (registrar el indicador y la fecha donde registra información).
6. Registrar indicadores (según el crédito, unidad de medida, nombre del indicador)..
7. Generar reportes de indicadores diarios (valor meta valor real, fecha en la que se reporta, código del indicador; etc).

Entidades Identificadas:

1. **Créditos:** Información básica de cada crédito.
2. **Analista Responsable:** Información sobre los responsables del crédito.
3. **Agencia:** Información sobre las agencias donde opera la entidad.
4. **Gerente Agencia:** Personal responsable de la administración de las agencias.
5. **Indicador:** Información de los indicadores a tener en cuenta..
6. **Fecha:** Información de las fechas donde se registran datos.
7. **Indicador Diario SF:** Información sobre los indicadores de la agencia, sus valores reales y valores meta.
8. **Desviaciones Indicador:** Diferencias del indicador diario entre el valor real y el valor meta.
9. **Sistema Fuente:** Sistema donde se recopila toda la información de cada crédito.
10. **Indicador Diario:** Información sobre el indicador y la fecha reportante..

Tablas y Campos Relevantes:

1. Tabla Créditos: Contiene la información básica de todos los créditos de la entidad microfinanciera.

- Id
- Titular
- Agencia Id
- Saldo Capital
- Estado Credito
- Saldo Otorgado

2. Tabla Agencia: Contiene la información de las agencias de la entidad a nivel nacional.

- Id
- Cod Agencia (código)
- Nombre
- Departamento
- Región
- Estado

3. Tabla Gerente Agencia: Contiene la información del personal responsable de administrar cada agencia.

- Id
- Agencia Id
- Nombre Gerente

- Categoría Gerente
- Fecha Inicio
- Fecha Fin

4. Tabla Sistema Fuente: Almacena la información de cada crédito, vincula información entre el crédito, el analista responsable y la agencia.

- Id
- Créditos Id
- Analista Responsable Id
- Agencia Id

5. Tabla Analista Responsable: Registra la información de los analistas responsables de los créditos.

- Id
- Cod Analista Origen (Código)
- Cod Analista Actual (Código)
- Categoría Analista Origen
- Categoría Analista Actual
- Nombre Analista Actual
- Nombre Analista Origen

6. Tabla Indicador: Registra el nombre del indicador, su unidad de medida y lo relaciona al sistema fuente.

- Id
- Sistema Fuente Id
- Nombre Indicador
- Unidad de Medida

7. Indicador Diario: Relaciona al indicador con la fecha de registro

- Id
- Indicador Id
- Fecha Id

8. Fecha: Fecha en la que se registra el indicador

- Id
- Agencia Id
- Fecha Reporte

9. Indicador Diario SF: Se registra el valor de los indicadores así como la meta que tienen, se relaciona con la fecha de registro, la agencia y el indicador que le corresponde.

- Id
- Indicador Id

- Agencia Id
- Valor Meta
- Valor Real
- Fecha Id

10. Desviaciones Indicador: Registra las desviaciones que pueden existir entre el valor real y el valor meta del indicador diario.

- Id
- Indicador Diario SF Id
- Diferencia Absoluta
- Diferencia Porcentual

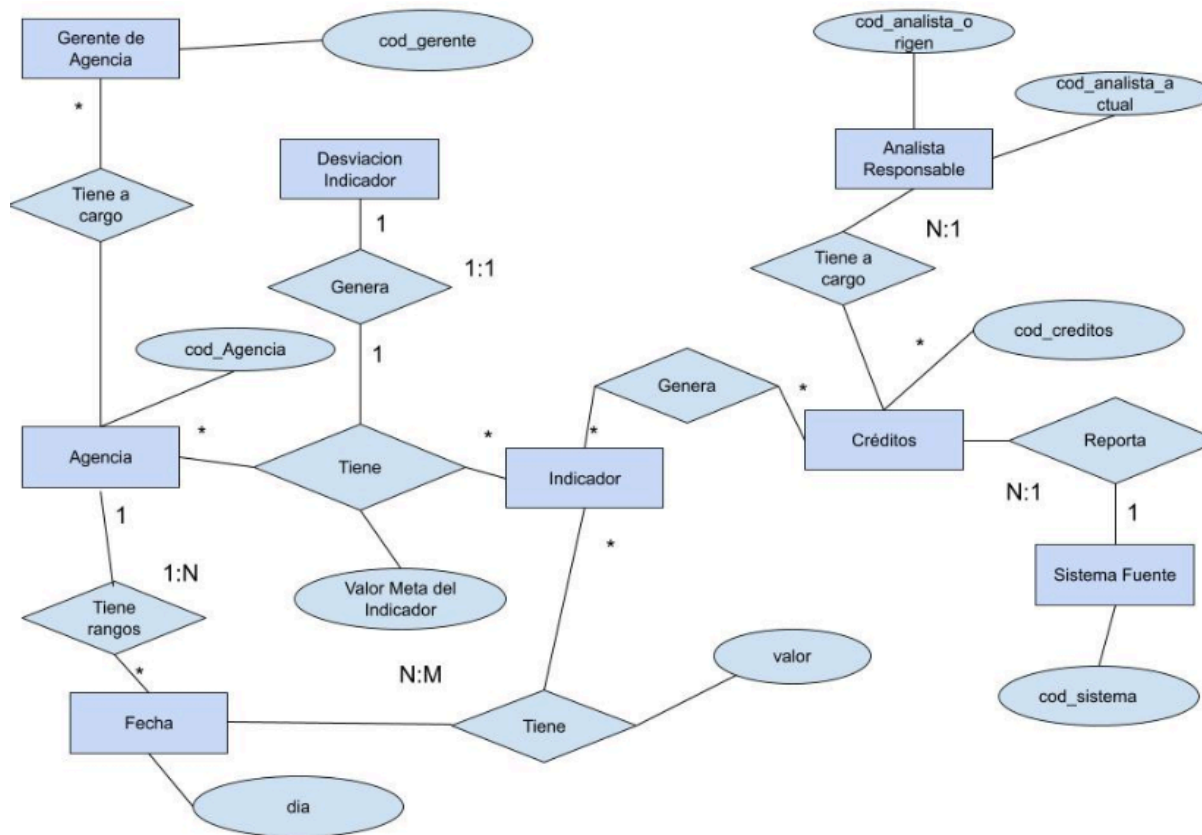
Desafíos:

- **Rotación de personal responsable:** La empresa necesita asignar personal responsable a cada crédito por la alta rotación en el puesto, así como de vacaciones debido a que los créditos no se pueden quedar sin un analista responsable
- **Seguimiento de los créditos diarios:** Es crucial que se pueda monitorear el estado actual de los créditos para garantizar mejores resultados a cierre de cada mes.
- **Optimización de recursos:** Es crucial poder tomar decisiones estratégicas en tiempo real para una mejor planificación estratégica del negocio.

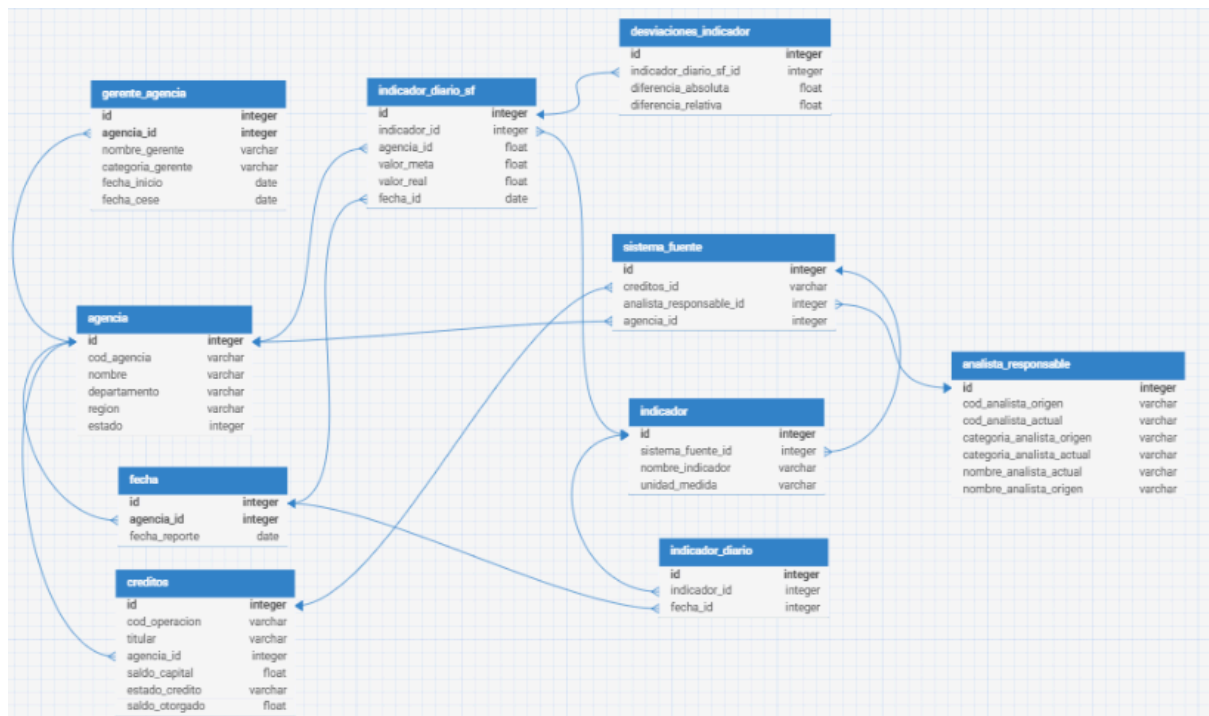
Beneficios Esperados:

- Asegurar la eficiencia operativa en la gestión de créditos por agencia y entidad microfinanciera.
- Facilitar la toma de decisiones estratégicas basadas en el comportamiento diario de las agencias..
- Mejor control y seguimiento de los créditos..

II. Análisis: Diseño Conceptual de la base de datos.



III. Diseño: Definición del Modelo Lógico de la Base de Datos.



IV. Construcción: Desarrollo de las sentencias SQL que permiten la construcción de las tablas y demás estructuras de la Base de Datos en SQL SERVER.

Creación de la **base de datos**

```
-- Creación de la base de datos

create database bd_proyecto_indicadores_financieros;
go

-- Seleccionar la base de datos
use bd_proyecto_indicadores_financieros;
go
```

Creación de la tabla **Créditos**:

```
-- Creamos la tabal credits
create table credits (
  id int identity (1,1) primary key,
  titular varchar (100) not null,
  agencia_id int not null,
  saldo_capital decimal (9,2) not null,
  estado_credito varchar (100) not null,
  saldo_otorgado decimal (9,2) not null,
  foreign key (agencia_id) references agencia(id)
);
```

Creación de la tabla **Agencia**:

```
-- Creamos la tabla agencia
create table agencia(
  id int identity(1,1) primary key,
  cod_agencia char(5) unique not null,
  nombre varchar(100) not null,
  departamento varchar(100) not null,
  region varchar(100) not null,
  estado varchar(25) not null
);
```

Creación de la tabla **Gerente Agencia**:

```
-- Creamos la tabla gerente_agencia
create table gerente_agencia (
  id int identity (1,1) primary key,
  agencia_id int not null,
  nombre_gerente varchar (100) not null,
  categoria_gerente varchar(55),
  fecha_inicio date not null,
  fecha_cese date null
  foreign key (agencia_id) references agencia(id)
);
```

Creación de la tabla **Sistema Fuente**:

```
-- Creamos la tabla sistema_fuente
create table sistema_fuente(
  id int identity (1,1) primary key,
  creditos_id int not null,
  analista_responsable_id int not null,
  agencia_id int not null,
  foreign key (creditos_id) references creditos(id),
  foreign key (analista_responsable_id) references analista_responsable(id),
  foreign key (agencia_id) references agencia(id)
);
```

Creación de la tabla **Analista Responsable**:

```
-- Creamos la tabla analista_responsable
create table analista_responsable (
  id int identity(1,1) primary key,
  cod_analista_origen char(8) unique not null,
  cod_analista_actual char(8) not null,
  categoria_analista_origen varchar(100) not null,
  categoria_analista_actual varchar(100) not null,
  nombre_analista_actual varchar(100) not null,
  nombre_analista_origen varchar(100) not null
);
```

Creación de la tabla **Indicador**

```
-- Creamos la tabla indicador
create table indicador(
  id int identity (1,1) primary key,
  sistema_fuente_id int not null,
  nombre_indicador varchar (100) not null,
  unidad_medida varchar (100) not null,
  foreign key (sistema_fuente_id) references sistema_fuente(id)
);
```


Creación de la tabla **Indicador Diario**

```
-- Creamos la tabla indicador_diario
create table indicador_diario(
  id int identity (1,1) primary key,
  indicador_id int not null,
  fecha_id int not null,
  foreign key (indicador_id) references indicador(id),
  foreign key (fecha_id) references fecha(id)
);
```

Creación de la tabla **Fecha**

```
-- Creamos la tabla fecha
create table fecha (
  id int identity (1,1) primary key,
  agencia_id int not null,
  fecha_reporte datetime default getdate() not null,
  foreign key (agencia_id) references agencia(id)
);
```

Creación de la tabla **Indicador Diario Sf**

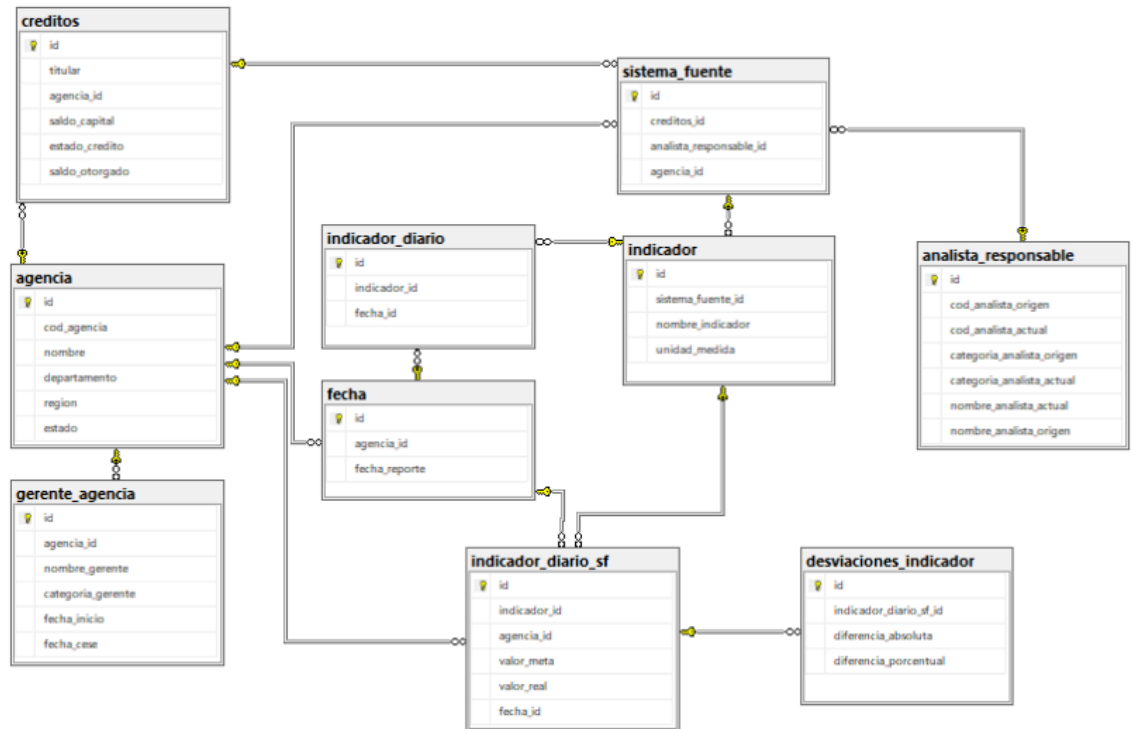
```
-- Creamos la tabla indicador_diario_sf
create table indicador_diario_sf(
  id int identity(1,1) primary key,
  indicador_id int not null,
  agencia_id int not null,
  valor_meta VARCHAR (255) NOT NULL,
  valor_real decimal (9,2) NOT NULL,
  fecha_id int not null,
  foreign key (indicador_id) references indicador(id),
  foreign key (agencia_id) references agencia(id),
  foreign key (fecha_id) references fecha(id)
);
```

Creación de la tabla **Desviaciones Indicador**

```
-- Creamos la tabla desviaciones_indicador
create table desviaciones_indicador(
  id INT IDENTITY(1,1) PRIMARY KEY,
  indicador_diario_sf_id INT NOT NULL,
  diferencia_absoluta DECIMAL(9,2) NOT NULL,
  diferencia_porcentual DECIMAL(9,2) NOT NULL,
  FOREIGN KEY (indicador_diario_sf_id) REFERENCES indicador_diario_sf(id),
);
```

V. Producto: Diseño Físico, Script de la Base de datos con inserción de datos (Mínimo 5 registros por tablas independientes o no transaccionales y mínimo 15 registros por cada tabla transaccional).

DISEÑO FÍSICO



Link del SCRIPT

Se adjunta link de mi repositorio en **GITHUB** donde está incluido el Script:

- <https://github.com/AnibalUnda/SQL-BDA7.git>

INSERCIÓN DE DATOS

Tabla Créditos

```
-- Insertamos datos de la tabla 'créditos'
insert into creditos (titular, agencia_id, saldo_capital, estado_credito, saldo_otorgado)
values
('Juan Pérez', 1, 10000.00, 'Activo', 15000.00),
('Lucía Ramos', 2, 5000.00, 'Activo', 7000.00),
('Carlos Díaz', 3, 8000.00, 'Moroso', 10000.00),
('Marta Salas', 4, 6000.00, 'Activo', 6000.00),
('Pedro Yupanqui', 5, 2000.00, 'Cancelado', 2500.00),
('Laura Ríos', 6, 3000.00, 'Activo', 4000.00),
('Miguel Torres', 7, 9000.00, 'Activo', 11000.00),
('Andrea Vega', 8, 4000.00, 'Activo', 4500.00),
('Luis Méndez', 9, 7000.00, 'Moroso', 9000.00),
('Ana Flores', 10, 5500.00, 'Activo', 6000.00),
('Eduardo Reyes', 11, 6200.00, 'Activo', 8000.00),
('Tatiana León', 12, 1000.00, 'Cancelado', 1200.00),
('Fernando Ramos', 13, 15000.00, 'Activo', 16000.00),
('Elena García', 14, 4700.00, 'Activo', 5000.00),
('Ramón Torres', 15, 20000.00, 'Moroso', 22000.00);
```

Tabla Agencia

```
-- Insertadmos datos de la tabla 'agencia'
insert into agencia (cod_agencia, nombre, departamento, region, estado)
values
('AG001', 'Agencia Central', 'Lima', 'Costa', 'Activa'),
('AG002', 'Agencia Norte', 'Piura', 'Costa', 'Activa'),
('AG003', 'Agencia Sur', 'Arequipa', 'Sierra', 'Activa'),
('AG004', 'Agencia Este', 'Iquitos', 'Selva', 'Activa'),
('AG005', 'Agencia Oeste', 'Callao', 'Costa', 'Activa'),
('AG006', 'Agencia Altiplano', 'Puno', 'Sierra', 'Activa'),
('AG007', 'Agencia Selva', 'Tarapoto', 'Selva', 'Activa'),
('AG008', 'Agencia Valle', 'Cusco', 'Sierra', 'Activa'),
('AG009', 'Agencia Mar', 'Chiclayo', 'Costa', 'Activa'),
('AG010', 'Agencia Centro', 'Huancayo', 'Sierra', 'Activa'),
('AG011', 'Agencia Noreste', 'Tumbes', 'Costa', 'Activa'),
('AG012', 'Agencia Frontera', 'Tacna', 'Sierra', 'Activa'),
('AG013', 'Agencia Metropolitana', 'Lima', 'Costa', 'Activa'),
('AG014', 'Agencia Central Sur', 'Ayacucho', 'Sierra', 'Activa'),
('AG015', 'Agencia Río', 'Ucayali', 'Selva', 'Activa');
```

Tabla Gerente Agencia

```
insert into gerente_agencia (agencia_id, nombre_gerente, categoria_gerente, fecha_inicio, fecha_cese)
values
(1, 'Javier Prado', 'A', '2022-01-01', null),
(2, 'Ana Martínez', 'B', '2021-05-10', null),
(3, 'Luis Ramos', 'A', '2020-03-15', '2023-12-31'),
(4, 'Sofía Delgado', 'C', '2023-01-20', null),
(5, 'Mario Lazo', 'B', '2021-11-30', null),
(6, 'Elena Dávila', 'A', '2022-07-01', null),
(7, 'Carlos Medina', 'B', '2020-09-01', null),
(8, 'Tatiana Quispe', 'A', '2021-01-01', '2023-08-10'),
(9, 'Daniel Pinto', 'C', '2023-03-05', null),
(10, 'Marta Vera', 'B', '2020-12-12', null),
(11, 'Rosa Paredes', 'A', '2022-04-04', null),
(12, 'Andrés Alfaro', 'B', '2021-10-10', null),
(13, 'Iván Tello', 'C', '2023-06-01', null),
(14, 'César Luján', 'A', '2022-08-08', null),
(15, 'Natalia Ruiz', 'B', '2023-02-14', null);

select * from gerente_agencia
```

Tabla Sistema Fuente

```
-- Insertamos datos en la tabla 'sistema_fuente'

insert into sistema_fuente (creditos_id, analista_responsable_id, agencia_id)
values
(1, 1, 1), (2, 2, 2), (3, 3, 3), (4, 4, 4), (5, 5, 5),
(6, 6, 6), (7, 7, 7), (8, 8, 8), (9, 9, 9), (10, 10, 10),
(11, 11, 11), (12, 12, 12), (13, 13, 13), (14, 14, 14), (15, 15, 15);
```

Tabla Analista Responsable

```
insert into analista_responsable (cod_analista_origen,
                                cod_analista_actual,
                                categoria_analista_origen,
                                categoria_analista_actual,
                                nombre_analista_actual,
                                nombre_analista_origen)
values
('A0000001', 'A0000101', 'Junior', 'Senior', 'Luis Pérez', 'Carlos Gómez'),
('A0000002', 'A0000102', 'Junior', 'Semi-Senior', 'Ana Torres', 'Juan León'),
('A0000003', 'A0000103', 'Semi-Senior', 'Senior', 'Marco Sánchez', 'Laura Ríos'),
('A0000004', 'A0000104', 'Junior', 'Senior', 'Diana Morales', 'Pedro Núñez'),
('A0000005', 'A0000105', 'Senior', 'Senior', 'Mario Herrera', 'Mario Herrera'),
('A0000006', 'A0000106', 'Junior', 'Semi-Senior', 'Claudia Rivas', 'Eduardo Díaz'),
('A0000007', 'A0000107', 'Senior', 'Senior', 'Elena Suárez', 'Elena Suárez'),
('A0000008', 'A0000108', 'Junior', 'Senior', 'Alfredo Lima', 'Gloria Ruiz'),
('A0000009', 'A0000109', 'Semi-Senior', 'Senior', 'Beatriz Soto', 'Fernando Rojas'),
('A0000010', 'A0000110', 'Junior', 'Junior', 'Andrea Lozano', 'Andrés Salas'),
('A0000011', 'A0000111', 'Junior', 'Senior', 'Esteban García', 'Rosa Vargas'),
('A0000012', 'A0000112', 'Semi-Senior', 'Senior', 'Lucas Aguirre', 'Marta Mendoza'),
('A0000013', 'A0000113', 'Junior', 'Semi-Senior', 'Tatiana Flores', 'Ramón Vera'),
('A0000014', 'A0000114', 'Senior', 'Senior', 'Ricardo Peña', 'Ricardo Peña'),
('A0000015', 'A0000115', 'Junior', 'Senior', 'Juliana López', 'Óscar Méndez');

select * from analista_responsable
```

Tabla Indicador

```
-- Insertamos datos en la tabla 'indicador'

insert into indicador (sistema_fuente_id, nombre_indicador, unidad_medida)
values
(1, 'Mora Crediticia', '%'),
(2, 'Rentabilidad Neta', '%'),
(3, 'Cartera Vigente', 'S/'),
(4, 'Clientes Nuevos', 'Unidades'),
(5, 'Clientes Recuperados', 'Unidades'),
(6, 'Cartera Vencida', 'S/'),
(7, 'Nuevos Créditos', 'S/'),
(8, 'Créditos Cancelados', 'S/'),
(9, 'Monto Otorgado', 'S/'),
(10, 'Saldo Capital Promedio', 'S/'),
(11, 'Ingresos Financieros', 'S/'),
(12, 'Créditos Refinanciados', 'Unidades'),
(13, 'Clientes Activos', 'Unidades'),
(14, 'Costos Operativos', 'S/'),
(15, 'Rotación de Cartera', '%');
```

Tabla Indicador Diario

```
-- Insertamos datos en la tabla 'indicador_diario'

insert into indicador_diario (indicador_id, fecha_id)
values
(1,1), (2,2), (3,3), (4,4), (5,5),
(6,6), (7,7), (8,8), (9,9), (10,10),
(11,11), (12,12), (13,13), (14,14), (15,15);
```

Tabla Fecha

```
-- Insertamos datos de la tabla 'fecha'

insert into fecha (agencia_id, fecha_reporte)
values
(1, '2024-01-01'), (2, '2024-01-02'), (3, '2024-01-03'), (4, '2024-01-04'), (5, '2024-01-05'),
(6, '2024-01-06'), (7, '2024-01-07'), (8, '2024-01-08'), (9, '2024-01-09'), (10, '2024-01-10'),
(11, '2024-01-11'), (12, '2024-01-12'), (13, '2024-01-13'), (14, '2024-01-14'), (15, '2024-01-15');
```

Tabla Indicador Diario Sf

```

insert into indicador_diario_sf (indicador_id, agencia_id, valor_meta, valor_real, fecha_id)
values
(1,1,5.00, 4.80, 1),
(2,2,10.00, 9.50, 2),
(3,3,100000.00, 95000.00, 3),
(4,4,150, 140, 4),
(5,5,50, 48, 5),
(6,6,80000.00, 79000.00, 6),
(7,7,120000.00, 125000.00, 7),
(8,8,40000.00, 42000.00, 8),
(9,9,130000.00, 129000.00, 9),
(10,10,50000.00, 48000.00, 10),
(11,11,60000.00, 61000.00, 11),
(12,12,30, 28, 12),
(13,13,300, 310, 13),
(14,14,35000.00, 34000.00, 14),
(15,15,7.5, 7.0, 15);

```

Tabla Desviaciones Indicador

```

-- Insertamos datos en la tabla 'desviaciones_indicador'

insert into desviaciones_indicador (indicador_diario_sf_id, diferencia_absoluta, diferencia_porcentual)
select
id as indicador_diario_sf_id,
abs(valor_real - valor_meta) as diferencia_absoluta,
case
when valor_meta = 0 then 0
else abs(valor_real - valor_meta) / valor_meta * 100
end as diferencia_porcentual
from indicador_diario_sf;

```

IV. Anexos: Evidencias de Consultas, funciones, procedimientos almacenados, vistas, etc. Realizados sobre la base de datos implementada. (5 ejercicios por integrante del grupo, Integrantes individuales (7 ejercicios))

1. Se desea visualizar los departamentos donde tenemos créditos morosos y la cantidad de los mismos.

```

-- Consultar créditos morosos por departamento

SELECT
ag.departamento as departamento,
count(*) as cantidad_morosos
FROM creditos c
LEFT JOIN agencia ag ON c.agencia_id = ag.id
WHERE
c.estado_credito = 'Moroso'
GROUP BY
ag.departamento

```

	departamento	cantidad_morosos
1	Arequipa	1
2	Chiclayo	1
3	Ucayali	1

- Se requiere identificar aquellos indicadores que registran una desviación menor al 5%

```
-- Consultar indicadores con desviación menor al 5%

SELECT
    di.id,
    i.nombre_indicador,
    idsf.valor_real,
    idsf.valor_meta,
    di.diferencia_porcentual
FROM desviaciones_indicador di
LEFT JOIN indicador_diario_sf idsf ON di.indicador_diario_sf_id = idsf.id
LEFT JOIN indicador i ON idsf.indicador_id = i.id
WHERE
    di.diferencia_porcentual < 5;
```

	id	nombre_indicador	valor_real	valor_meta	diferencia_porcentual
1	1	Mora Crediticia	4.80	5.00	4.00
2	5	Cientes Recuperados	48.00	50.00	4.00
3	6	Cartera Vencida	79000.00	80000.00	1.25
4	7	Nuevos Créditos	125000.00	120000.00	4.17
5	9	Monto Otorgado	129000.00	130000.00	0.77
6	10	Saldo Capital Promedio	48000.00	50000.00	4.00
7	11	Ingresos Financieros	61000.00	60000.00	1.67
8	13	Cientes Activos	310.00	300.00	3.33
9	14	Costos Operativos	34000.00	35000.00	2.86

- Se requiere conocer el total de créditos otorgados por región.

```
-- Consultar total de créditos otorgados por región

SELECT
    a.region,
    sum(c.saldo_otorgado) as Total_Saldo_Otorgado
FROM creditos c
LEFT JOIN agencia a on c.agencia_id = a.id
GROUP BY
    a.region;
```

Results			Messages
	region	Total_Saldo_Otorgado	
1	Costa	57500.00	
2	Selva	39000.00	
3	Sierra	30700.00	

- Se requiere crear una vista del desempeño de indicadores diarios con metas y desviaciones

```
-- Vista de desempeño de indicadores diarios con metas y desviaciones
CREATE VIEW vw_indicadores_desempeno AS
SELECT
    i.nombre_indicador,
    a.nombre AS agencia,
    f.fecha_reporte,
    idsf.valor_meta,
    idsf.valor_real,
    abs(idsf.valor_real - idsf.valor_meta) AS diferencia_absoluta,
    CASE
        WHEN idsf.valor_meta = 0 THEN 0
        ELSE abs(idsf.valor_real - idsf.valor_meta) / idsf.valor_meta * 100
    END AS diferencia_porcentual
FROM indicador_diario_sf idsf
LEFT JOIN agencia a ON idsf.agencia_id = a.id
LEFT JOIN fecha f ON idsf.fecha_id = f.id
LEFT JOIN indicador i ON idsf.indicador_id = i.id;
```

3 %

Messages

Commands completed successfully.

Completion time: 2025-06-10T21:03:03.8258426-05:00

```
select * from [dbo].[vw_indicadores_desempeno]
```

Results								Messages
	nombre_indicador	agencia	fecha_reporte	valor_meta	valor_real	diferencia_absoluta	diferencia_porcentual	
1	Mora Crediticia	Agencia Central	2024-01-01 00:00:00.000	5.00	4.80	0.20	4.000000000000	
2	Rentabilidad Neta	Agencia Norte	2024-01-02 00:00:00.000	10.00	9.50	0.50	5.000000000000	
3	Cartera Vigente	Agencia Sur	2024-01-03 00:00:00.000	100000.00	95000.00	5000.00	5.000000000000	
4	Cientes Nuevos	Agencia Este	2024-01-04 00:00:00.000	150.00	140.00	10.00	6.666666666600	
5	Cientes Recuperados	Agencia Oeste	2024-01-05 00:00:00.000	50.00	48.00	2.00	4.000000000000	
6	Cartera Vencida	Agencia Altiplano	2024-01-06 00:00:00.000	80000.00	79000.00	1000.00	1.250000000000	
7	Nuevos Créditos	Agencia Selva	2024-01-07 00:00:00.000	120000.00	125000.00	5000.00	4.166666666600	
8	Créditos Cancelados	Agencia Valle	2024-01-08 00:00:00.000	40000.00	42000.00	2000.00	5.000000000000	
9	Monto Otorgado	Agencia Mar	2024-01-09 00:00:00.000	130000.00	129000.00	1000.00	0.769230769200	
10	Saldo Capital Promedio	Agencia Centro	2024-01-10 00:00:00.000	50000.00	48000.00	2000.00	4.000000000000	
11	Ingresos Financieros	Agencia Noreste	2024-01-11 00:00:00.000	60000.00	61000.00	1000.00	1.666666666600	

- Se requiere crear una vista de los créditos activos con un analista responsable.


```
-- Vista de créditos activos con analistas responsables
CREATE VIEW vw_creditos_analista AS
SELECT
    c.titular,
    c.saldo_capital,
    ar.nombre_analista_actual,
    ar.categoria_analista_actual,
    ag.nombre AS agencia
FROM creditos c
LEFT JOIN sistema_fuente sf ON c.id = sf.creditos_id
LEFT JOIN analista_responsable ar ON sf.analista_responsable_id = ar.id
LEFT JOIN agencia ag ON c.agencia_id = ag.id
WHERE c.estado_credito = 'Activo';
```

83 %

Messages

Commands completed successfully.

Completion time: 2025-06-10T21:07:07.7862018-05:00

```
SELECT * FROM dbo.vw_creditos_analista
```

83 %

	titular	saldo_capital	nombre_analista_actual	categoria_analista_actual	agencia
1	Juan Pérez	10000.00	Luis Pérez	Senior	Agencia Central
2	Lucía Ramos	5000.00	Ana Torres	Semi-Senior	Agencia Norte
3	Marta Salas	6000.00	Diana Morales	Senior	Agencia Este
4	Laura Ríos	3000.00	Claudia Rivas	Semi-Senior	Agencia Altiplano
5	Miguel Torres	9000.00	Elena Suárez	Senior	Agencia Selva
6	Andrea Vega	4000.00	Alfredo Lima	Senior	Agencia Valle
7	Ana Flores	5500.00	Andrea Lozano	Junior	Agencia Centro
8	Eduardo Reyes	6200.00	Esteban García	Senior	Agencia Noreste
9	Fernando Ramos	15000.00	Tatiana Flores	Semi-Senior	Agencia Metropolitana
10	Elena García	4700.00	Ricardo Peña	Senior	Agencia Central Sur

6. Se requiere crear la vista de gerente de agencia activo y fecha de inicio

```
-- Vista de gerente de agencia activo y fecha de inicio
CREATE VIEW vw_gerentes_activos AS
SELECT
    a.nombre AS agencia,
    g.nombre_gerente,
    g.fecha_inicio
FROM gerente_agencia g
LEFT JOIN agencia a ON g.agencia_id = a.id
WHERE g.fecha_cese IS NULL;
```

3 %

Messages

Commands completed successfully.

Completion time: 2025-06-10T21:09:44.0352586-05:00

```
SELECT * FROM dbo.vw_gerentes_activos
```

83 %

Results Messages

	agencia	nombre_gerente	fecha_inicio
1	Agencia Central	Javier Prado	2022-01-01
2	Agencia Norte	Ana Martínez	2021-05-10
3	Agencia Este	Sofía Delgado	2023-01-20
4	Agencia Oeste	Mario Lazo	2021-11-30
5	Agencia Altiplano	Elena Dávila	2022-07-01
6	Agencia Selva	Carlos Medina	2020-09-01
7	Agencia Mar	Daniel Pinto	2023-03-05
8	Agencia Centro	Marta Vera	2020-12-12
9	Agencia Noreste	Rosa Paredes	2022-04-04
10	Agencia Frontera	Andrés Alfaro	2021-10-10
11	Agencia Metropolitana	Iván Tello	2023-06-01

7. Crear un procedimiento almacenado de insertar un nuevo crédito

```
-- Procedimiento almacenado de insertar un nuevo crédito
CREATE PROCEDURE sp_insertar_credito
    @titular VARCHAR(100),
    @agencia_id INT,
    @saldo_capital DECIMAL(9,2),
    @estado_credito VARCHAR(100),
    @saldo_otorgado DECIMAL(9,2)
AS
BEGIN
    INSERT INTO creditos (titular, agencia_id, saldo_capital, estado_credito, saldo_otorgado)
    VALUES (@titular, @agencia_id, @saldo_capital, @estado_credito, @saldo_otorgado);
END;
```

83 %

Messages

Commands completed successfully.

Completion time: 2025-06-10T21:14:02.2758542-05:00

```
EXEC dbo.sp_insertar_credito 'Jhenss Chavez',3,1650,'Moroso',20000
```

83 %

Messages

(1 row affected)

Completion time: 2025-06-10T21:16:48.9567900-05:00

8. Crear un procedimiento almacenado para obtener las desviaciones de un indicador específico.

```
-- Procedimiento almacenado para obtener desviaciones por indicador
CREATE PROCEDURE sp_desviaciones_por_indicador
    @nombre_indicador VARCHAR(100)
AS
BEGIN
    SELECT
        i.nombre_indicador,
        a.nombre AS agencia,
        idsf.valor_meta,
        idsf.valor_real,
        di.diferencia_porcentual
    FROM desviaciones_indicador di
    LEFT JOIN indicador_diario_sf idsf ON di.indicador_diario_sf_id = idsf.id
    LEFT JOIN indicador i ON idsf.indicador_id = i.id
    LEFT JOIN agencia a ON idsf.agencia_id = a.id
    WHERE i.nombre_indicador = @nombre_indicador;
END;
```

83 %

Messages

Commands completed successfully.

Completion time: 2025-06-10T21:19:08.2618537-05:00

```
EXEC dbo.sp_desviaciones_por_indicador 'Mora Crediticia'
```

	nombre_indicador	agencia	valor_meta	valor_real	diferencia_porcentual
1	Mora Crediticia	Agencia Central	5.00	4.80	4.00

9. Crear una función escalar para devolver el porcentaje de desviación de acuerdo del valor meta y valor real consignado

```
CREATE FUNCTION fn_porcentaje_desviacion (
    @valor_meta DECIMAL(9,2),
    @valor_real DECIMAL(9,2)
)
RETURNS DECIMAL(9,2)
AS
BEGIN
    DECLARE @resultado DECIMAL(9,2);
    IF @valor_meta = 0
        SET @resultado = 0;
    ELSE
        SET @resultado = abs(@valor_real - @valor_meta) / @valor_meta * 100;
    RETURN @resultado;
END;
```

Commands completed successfully.

Completion time: 2025-06-10T21:24:09.0489625-05:00

```
select dbo.fn_porcentaje_desviacion (50,55)
```

	(No column name)
1	10.00

10. Consulta avanzada para unir gerente, agencia y promedio de saldo por agencia

```
-- Consulta para ver el gerente de agencia, la agencia y el promedio de saldo por agencia
SELECT
    a.nombre AS agencia,
    g.nombre_gerente,
    AVG(c.saldo_capital) AS saldo_promedio
FROM agencia a
LEFT JOIN gerente_agencia g ON a.id = g.agencia_id
LEFT JOIN creditos c ON a.id = c.agencia_id
GROUP BY a.nombre, g.nombre_gerente;
```

83 %

Results Messages

	agencia	nombre_gerente	saldo_promedio
1	Agencia Norte	Ana Martínez	5000.000000
2	Agencia Frontera	Andrés Alfaro	1000.000000
3	Agencia Selva	Carlos Medina	9000.000000
4	Agencia Central Sur	César Luján	4700.000000
5	Agencia Mar	Daniel Pinto	7000.000000
6	Agencia Altiplano	Elena Dávila	3000.000000
7	Agencia Metropolitana	Iván Tello	15000.000000
8	Agencia Central	Javier Prado	10000.000000
9	Agencia Sur	Luis Ramos	4825.000000
10	Agencia Oeste	Mario Lazo	2000.000000
11	Agencia Centro	Marta Vera	5500.000000

LINK DE REPOSITORIO EN GITHUB:

- <https://github.com/AnibalUnda/SQL-BDA7.git>