

-- Crear la base de datos

CREATE DATABASE FyaCreditManagement;

GO

USE FyaCreditManagement;

GO

-- =====

-- TABLA: Comerciales/Ejecutivos

-- =====

CREATE TABLE Comerciales (

 ComercialId INT IDENTITY(1,1) PRIMARY KEY,

 Nombre NVARCHAR(150) NOT NULL,

 Email NVARCHAR(100) UNIQUE NOT NULL,

 Telefono NVARCHAR(20),

 Activo BIT NOT NULL DEFAULT 1,

 FechaCreacion DATETIME2 NOT NULL DEFAULT GETDATE(),

 FechaModificacion DATETIME2 NOT NULL DEFAULT GETDATE()

);

-- =====

-- TABLA: Clientes

-- =====

CREATE TABLE Clientes (

 ClienteId INT IDENTITY(1,1) PRIMARY KEY,

 Nombre NVARCHAR(200) NOT NULL,

 NumeroIdentificacion NVARCHAR(50) UNIQUE NOT NULL,

 TipoIdentificacion NVARCHAR(20) NOT NULL DEFAULT 'CC', -- CC, NIT, CE, PP

 Email NVARCHAR(100),

```

Telefono NVARCHAR(20),
Direccion NVARCHAR(300),
Ciudad NVARCHAR(100),
Activo BIT NOT NULL DEFAULT 1,
FechaCreacion DATETIME2 NOT NULL DEFAULT GETDATE(),
FechaModificacion DATETIME2 NOT NULL DEFAULT GETDATE(),

-- Índices para optimizar búsquedas
INDEX IX_Clientes_NumeroIdentificacion NONCLUSTERED (NumeroIdentificacion),
INDEX IX_Clientes_Nombre NONCLUSTERED (Nombre)
);

-- =====
-- TABLA: Estados de Crédito
-- =====

CREATE TABLE EstadosCredito (
    EstadoId INT IDENTITY(1,1) PRIMARY KEY,
   Codigo NVARCHAR(20) UNIQUE NOT NULL,
   Descripcion NVARCHAR(100) NOT NULL,
   Activo BIT NOT NULL DEFAULT 1
);

-- =====
-- TABLA: Créditos (Tabla principal)
-- =====

CREATE TABLE Creditos (
    CreditId INT IDENTITY(1,1) PRIMARY KEY,
    ClienteId INT NOT NULL,
    ComercialId INT NOT NULL,

```

```

EstadoId INT NOT NULL,

-- Datos del crédito
ValorCredito DECIMAL(18,2) NOT NULL,
TasaInteres DECIMAL(5,4) NOT NULL, -- Hasta 99.9999%
PlazoMeses INT NOT NULL,

-- Cálculos automáticos
ValorCuota AS (ValorCredito * (TasaInteres/100/12) * POWER(1 + TasaInteres/100/12,
PlazoMeses)) / (POWER(1 + TasaInteres/100/12, PlazoMeses) - 1) PERSISTED,
ValorTotal AS (ValorCredito * (1 + (TasaInteres/100 * PlazoMeses/12))) PERSISTED,

-- Fechas
FechaRegistro DATETIME2 NOT NULL DEFAULT GETDATE(),
FechaAprobacion DATETIME2 NULL,
FechaVencimiento AS DATEADD(MONTH, PlazoMeses, FechaAprobacion),

-- Auditoría
FechaCreacion DATETIME2 NOT NULL DEFAULT GETDATE(),
FechaModificacion DATETIME2 NOT NULL DEFAULT GETDATE(),
UsuarioCreacion NVARCHAR(100) NOT NULL DEFAULT SYSTEM_USER,
UsuarioModificacion NVARCHAR(100) NOT NULL DEFAULT SYSTEM_USER,

-- Constraints
CONSTRAINT FK_Creditos_Cliente FOREIGN KEY (ClientId) REFERENCES
Clientes(ClientId),

CONSTRAINT FK_Creditos_Comercial FOREIGN KEY (ComercialId) REFERENCES
Comerciales(ComercialId),

CONSTRAINT FK_Creditos_Estado FOREIGN KEY (EstadoId) REFERENCES
EstadosCredito(EstadoId),

```

```
CONSTRAINT CHK_Creditos_ValorCredito CHECK (ValorCredito > 0),  
CONSTRAINT CHK_Creditos_TasaInteres CHECK (TasaInteres >= 0 AND TasaInteres <= 50),  
CONSTRAINT CHK_Creditos_PlazoMeses CHECK (PlazoMeses > 0 AND PlazoMeses <=  
360),
```

```
-- Índices para optimizar consultas
```

```
INDEX IX_Creditos_FechaRegistro NONCLUSTERED (FechaRegistro DESC),
```

```
INDEX IX_Creditos_Cliente NONCLUSTERED (Clienteld),
```

```
INDEX IX_Creditos_Comercial NONCLUSTERED (Comercialld),
```

```
INDEX IX_Creditos_Estado NONCLUSTERED (Estadold),
```

```
INDEX IX_Creditos_ValorCredito NONCLUSTERED (ValorCredito DESC)
```

```
);
```

```
-- =====
```

```
-- TABLA: Log de Envío de Correos (RPA)
```

```
-- =====
```

```
CREATE TABLE LogEnvioCorreos (
```

```
    LogId INT IDENTITY(1,1) PRIMARY KEY,
```

```
    Creditold INT NOT NULL,
```

```
    DestinatarioEmail NVARCHAR(100) NOT NULL,
```

```
    AsuntoCorreo NVARCHAR(200) NOT NULL,
```

```
    CuerpoCorreo NVARCHAR(MAX) NOT NULL,
```

```
    EstadoEnvio NVARCHAR(20) NOT NULL DEFAULT 'PENDIENTE', -- PENDIENTE, ENVIADO,  
    ERROR
```

```
    MensajeError NVARCHAR(500) NULL,
```

```
    FechaIntento DATETIME2 NOT NULL DEFAULT GETDATE(),
```

```
    FechaEnvio DATETIME2 NULL,
```

```
    CONSTRAINT FK_LogEnvioCorreos_Credito FOREIGN KEY (Creditold) REFERENCES  
    Creditos(Creditold),
```

```
INDEX IX_LogEnvioCorreos_FechaIntento NONCLUSTERED (FechaIntento DESC),  
INDEX IX_LogEnvioCorreos_EstadoEnvio NONCLUSTERED (EstadoEnvio)  
);
```

```
-- =====
```

```
-- TABLA: Auditoría de Cambios
```

```
-- =====
```

```
CREATE TABLE AuditoriaCreditos (  
    Auditoriald INT IDENTITY(1,1) PRIMARY KEY,  
    Creditold INT NOT NULL,  
    Accion NVARCHAR(20) NOT NULL, -- INSERT, UPDATE, DELETE  
    ValoresAnteriores NVARCHAR(MAX) NULL, -- JSON con valores anteriores  
    ValoresNuevos NVARCHAR(MAX) NOT NULL, -- JSON con valores nuevos  
    Usuario NVARCHAR(100) NOT NULL,  
    FechaAccion DATETIME2 NOT NULL DEFAULT GETDATE(),  
    DireccionIP NVARCHAR(45) NULL,  
  
    CONSTRAINT FK_AuditoriaCreditos_Credito FOREIGN KEY (Creditold) REFERENCES  
Creditos(Creditold),  
    INDEX IX_AuditoriaCreditos_FechaAccion NONCLUSTERED (FechaAccion DESC),  
    INDEX IX_AuditoriaCreditos_Creditold NONCLUSTERED (Creditold)  
);
```

```
CREATE TRIGGER tr_AuditoriaCreditos  
ON Creditos  
AFTER INSERT, UPDATE, DELETE  
AS  
BEGIN  
    SET NOCOUNT ON;
```

-- Para INSERT

IF EXISTS(SELECT * FROM inserted) AND NOT EXISTS(SELECT * FROM deleted)

BEGIN

INSERT INTO AuditoriaCreditos (Creditold, Accion, ValoresNuevos, Usuario)

SELECT

Creditold,

'INSERT',

(SELECT * FROM inserted i WHERE i.Creditold = inserted.Creditold FOR JSON AUTO),

SYSTEM_USER

FROM inserted;

END

-- Para UPDATE

IF EXISTS(SELECT * FROM inserted) AND EXISTS(SELECT * FROM deleted)

BEGIN

INSERT INTO AuditoriaCreditos (Creditold, Accion, ValoresAnteriores, ValoresNuevos, Usuario)

SELECT

i.Creditold,

'UPDATE',

(SELECT * FROM deleted d WHERE d.Creditold = i.Creditold FOR JSON AUTO),

(SELECT * FROM inserted ins WHERE ins.Creditold = i.Creditold FOR JSON AUTO),

SYSTEM_USER

FROM inserted i;

END

-- Para DELETE

IF EXISTS(SELECT * FROM deleted) AND NOT EXISTS(SELECT * FROM inserted)

```
BEGIN

  INSERT INTO AuditoriaCreditos (Creditold, Accion, ValoresAnteriores, Usuario)

  SELECT

    Creditold,

    'DELETE',

    (SELECT * FROM deleted d WHERE d.Creditold = deleted.Creditold FOR JSON AUTO),

    SYSTEM_USER

  FROM deleted;

END

END;
```