

01) DDL – criar tabela, constraints e índices

-- Limpeza opcional para reexecução

```
DROP TABLE IF EXISTS credito CASCADE;
```

```
CREATE TABLE credito (  
    id BIGINT GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,  
    numero_credito VARCHAR(50) NOT NULL,  
    numero_nfse VARCHAR(50) NOT NULL,  
    data_constituicao DATE NOT NULL,  
    valor_issqn NUMERIC(15,2) NOT NULL,  
    tipo_credito VARCHAR(50) NOT NULL,  
    simples_nacional BOOLEAN NOT NULL,  
    aliquota NUMERIC(5,2) NOT NULL,  
    valor_faturado NUMERIC(15,2) NOT NULL,  
    valor_deducacao NUMERIC(15,2) NOT NULL,  
    base_calculo NUMERIC(15,2) NOT NULL  
);
```

-- numero_credito único (ajuste se no seu domínio puder repetir)

```
ALTER TABLE credito  
    ADD CONSTRAINT uq_credito_numero UNIQUE (numero_credito);
```

-- Regras de domínio

```
ALTER TABLE credito  
    ADD CONSTRAINT chk_aliquota_range CHECK (aliquota >= 0 AND aliquota <= 100);
```

```
ALTER TABLE credito  
    ADD CONSTRAINT chk_valores_nao_negativos CHECK (  
        valor_issqn >= 0  
        AND valor_faturado >= 0  
        AND valor_deducacao >= 0  
        AND base_calculo >= 0  
    );
```

-- Coerência: base_calculo = valor_faturado - valor_deducacao (com arredondamento a 2 casas)

```
ALTER TABLE credito  
    ADD CONSTRAINT chk_base_calculo_coerente  
    CHECK ( base_calculo = round(valor_faturado - valor_deducacao, 2) );
```

-- Índices para buscas

```
CREATE INDEX idx_credito_nfse ON credito (numero_nfse);  
CREATE INDEX idx_credito_numero ON credito (numero_credito);  
CREATE INDEX idx_credito_data ON credito (data_constituicao DESC);
```

02) SEED – inserir dados de exemplo

```
TRUNCATE TABLE credito RESTART IDENTITY;
```

```
INSERT INTO credito  
    (numero_credito, numero_nfse, data_constituicao, valor_issqn,  
     tipo_credito, simples_nacional, aliquota, valor_faturado, valor_deducacao,  
     base_calculo)  
VALUES  
    ('123456', '7891011', DATE '2024-02-25', 1500.75, 'ISSQN', TRUE, 5.00,
```

```

30000.00, 5000.00, 25000.00),
('789012', '7891011', DATE '2024-02-26', 1200.50, 'ISSQN', FALSE, 4.50,
25000.00, 4000.00, 21000.00),
('654321', '1122334', DATE '2024-01-15', 800.50, 'Outros', TRUE, 3.50,
20000.00, 3000.00, 17000.00),
('900001', '5550001', DATE '2024-03-10', 750.00, 'ISSQN', TRUE, 3.00,
25000.00, 0000.00, 25000.00),
('900002', '5550001', DATE '2024-03-11', 900.00, 'ISSQN', TRUE, 3.60,
26000.00, 1000.00, 25000.00),
('900003', '5550002', DATE '2024-04-05', 1100.00, 'ISSQN', FALSE, 4.40,
27500.00, 1500.00, 26000.00);

```

03) “Endpoints” via SQL – funções que retornam JSON

A) GET /api/creditos/{numeroNfse} → lista (JSON array)

DROP FUNCTION IF EXISTS api_creditos_por_nfse(text);

```

CREATE OR REPLACE FUNCTION api_creditos_por_nfse(p_numero_nfse text)
RETURNS jsonb
LANGUAGE sql
STABLE
AS $$
SELECT COALESCE(
(
SELECT jsonb_agg(obj ORDER BY data_constituicao DESC)
FROM (
SELECT
jsonb_build_object(
'numeroCredito', c.numero_credito,
'numeroNfse', c.numero_nfse,
'dataConstituicao', to_char(c.data_constituicao, 'YYYY-MM-DD'),
'valorIssqn', c.valor_issqn,
'tipoCredito', c.tipo_credito,
'simplesNacional', CASE WHEN c.simples_nacional THEN 'Sim' ELSE 'Não'
END,
'aliquota', c.aliquota,
'valorFaturado', c.valor_faturado,
'valorDeducacao', c.valor_deducacao,
'baseCalculo', c.base_calculo
) AS obj,
c.data_constituicao
FROM credito c
WHERE c.numero_nfse = p_numero_nfse
) t
),
'[]'::jsonb
);
$$;

```

Teste:

```
SELECT api_creditos_por_nfse('7891011');
```

B) GET /api/creditos/credito/{numeroCredito} → objeto (JSON)

```

DROP FUNCTION IF EXISTS api_credito_por_numero(text);

CREATE OR REPLACE FUNCTION api_credito_por_numero(p_numero_credito text)
RETURNS jsonb
LANGUAGE sql
STABLE
AS $$
SELECT COALESCE(
  (
    SELECT jsonb_build_object(
      'numeroCredito',    c.numero_credito,
      'numeroNfse',       c.numero_nfse,
      'dataConstituicao',  to_char(c.data_constituicao, 'YYYY-MM-DD'),
      'valorIssqn',       c.valor_issqn,
      'tipoCredito',      c.tipo_credito,
      'simplesNacional',  CASE WHEN c.simples_nacional THEN 'Sim' ELSE 'Não'
    END,
      'aliquota',          c.aliquota,
      'valorFaturado',     c.valor_faturado,
      'valorDeducacao',    c.valor_deducacao,
      'baseCalculo',       c.base_calculo
    )
    FROM credito c
    WHERE c.numero_credito = p_numero_credito
    LIMIT 1
  ),
  '{}'::jsonb
);
$$;

```

Teste:

```
SELECT api_credito_por_numero('123456');
```