**View do Curso**

CREATE VIEW ViewCurso AS

SELECT

c.id,

c.nome,

c.id\_instituicao,

i.nome AS nome\_instituicao

FROM Curso c

JOIN Instituicao i ON c.id\_instituicao = i.id;

**Function Insert**

CREATE OR REPLACE FUNCTION InsertViewCurso()

RETURNS trigger AS $$

BEGIN

INSERT INTO Curso (id, nome, id\_instituicao)

VALUES (NEW.id, NEW.nome, NEW.id\_instituicao);

RETURN NEW;

END;

$$ LANGUAGE plpgsql;

**Trigger**

CREATE TRIGGER TriggerInsertViewCurso

INSTEAD OF INSERT ON ViewCurso

FOR EACH ROW

EXECUTE FUNCTION InsertViewCurso();

**ViewCurso.js (Model)**

const { Sequelize, DataTypes } = require('sequelize');

const postgresDb = new Sequelize('pmdb', 'postgres', 'A12345678a', {

host: 'localhost',

dialect: 'postgres'

});

module.exports = (sequelize, DataTypes) => {

const ViewCurso = sequelize.define('ViewCurso', {

id: {

type: DataTypes.INTEGER,

primaryKey: true

},

nome: {

type: DataTypes.STRING,

allowNull: false

},

id\_instituicao: {

type: DataTypes.INTEGER,

allowNull: false

},

nome\_instituicao: {

type: DataTypes.STRING,

allowNull: true

}

}, {

tableName: 'viewcurso',

timestamps: false,

freezeTableName: true

});

return ViewCurso;

};**viewCursoController.js (Controller)**

import ViewCurso form '../models/ViewCurso.js');

export const inserirCurso = async (req, res) => {

try {

const { id, nome, id\_instituicao } = req.body;

const novoCurso = await ViewCurso.create({

id,

nome,

id\_instituicao

});

console.log('Curso inserido com sucesso:', novoCurso.toJSON());

res.status(200).json({'message':'Curso inserido com sucesso'});

} catch (err) {

console.error('Erro ao inserir curso:', err);

res.status(505).json({'message':'Curso não inserido'});

}

}

export const listarCursos = async (req, res) => {

try {

const cursos = await ViewCurso.findAll();

console.table(cursos.map(c => c.toJSON()));

} catch (err) {

console.error('Erro ao listar cursos:', err);

res.status(505).json({'message':'Erro ao listar cursos'});

}

}

**viewCursoRoute.js (route)**

const express = require('express');

const router = express.Router();

const { VwCurso } = require('../models'); // ajuste o caminho se necessário

// Listar todos os cursos

router.get('/', async (req, res) => {

try {

const cursos = await VwCurso.findAll();

res.json(cursos);

} catch (err) {

console.error('Erro ao listar cursos:', err);

res.status(500).json({ error: 'Erro ao buscar cursos' });

}

});

// Inserir um novo curso na view

router.post('/', async (req, res) => {

const { id, nome, id\_instituicao } = req.body;

if (!id || !nome || !id\_instituicao) {

return res.status(400).json({ error: 'Campos id, nome e id\_instituicao são obrigatórios' });

}

try {

const novoCurso = await VwCurso.create({ id, nome, id\_instituicao });

res.status(201).json(novoCurso);

} catch (err) {

console.error('Erro ao inserir curso:', err);

res.status(500).json({ error: 'Erro ao inserir curso' });

}

});

module.exports = router;