sequelize model:generate --name User --attributes email:string, password:string

sequelize model:generate --name Post --attributes title:string, content:string, userId:integer

## Model

"use strict";

const { Model } = require("sequelize");

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

    class User extends Model {

        static associate(models) {

            User.hasMany(models.Post);

        }

    }

    User.init(

        {

            email: DataTypes.STRING,

            password: DataTypes.STRING,

        },

        {

            sequelize,

            modelName: "User",

        }

    );

    return User;

};

"use strict";

const { Model } = require("sequelize");

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

    class Post extends Model {

        static associate(models) {

            Post.belongsTo(models.User, {

                foreignKey: "userId",

                allowNull: false,

                onDelete: "CASCADE",

            });

        }

    }

    Post.init(

        {

            title: DataTypes.STRING,

            content: DataTypes.STRING,

            userId: DataTypes.INTEGER,

        },

        {

            sequelize,

            modelName: "Post",

        }

    );

    return Post;

};

## Table

'use strict';

module.exports = {

  up: async (queryInterface, Sequelize) => {

    await queryInterface.createTable('Users', {

      id: {

        allowNull: false,

        autoIncrement: true,

        primaryKey: true,

        type: Sequelize.INTEGER

      },

      email: {

        type: Sequelize.STRING

      },

      password: {

        type: Sequelize.STRING

      },

      createdAt: {

        allowNull: false,

        type: Sequelize.DATE

      },

      updatedAt: {

        allowNull: false,

        type: Sequelize.DATE

      }

    });

  },

  down: async (queryInterface, Sequelize) => {

    await queryInterface.dropTable('Users');

  }

};

"use strict";

module.exports = {

    up: async (queryInterface, Sequelize) => {

        await queryInterface.createTable("Posts", {

            id: {

                allowNull: false,

                autoIncrement: true,

                primaryKey: true,

                type: Sequelize.INTEGER,

            },

            title: {

                type: Sequelize.STRING,

            },

            content: {

                type: Sequelize.STRING,

            },

            userId: {

                type: Sequelize.INTEGER,

                references: {

                    model: {

                        tableName: "users",

                    },

                    key: "id",

                },

                onDelete: "CASCADE",

            },

            createdAt: {

                allowNull: false,

                type: Sequelize.DATE,

            },

            updatedAt: {

                allowNull: false,

                type: Sequelize.DATE,

            },

        });

    },

    down: async (queryInterface, Sequelize) => {

        await queryInterface.dropTable("Posts");

    },

};

Seeder

"use strict";

module.exports = {

    up: async (queryInterface, Sequelize) => {

        await queryInterface.bulkInsert("users", [

            {

                email: "test1@gmail.com",

                password: "123456",

                createdAt: new Date(),

                updatedAt: new Date(),

            },

            {

                email: "test2@gmail.com",

                password: "123456",

                createdAt: new Date(),

                updatedAt: new Date(),

            },

        ]);

    },

    down: async (queryInterface, Sequelize) => {},

};

## CRUD

const router = require("express").Router();

const { User, Post } = require("../models");

router.get("/", async (req, res) => {

    const users = await User.findAll();

    res.render("user-list", {

        users,

    });

});

router.get("/:id", async (req, res) => {

    const id = req.params.id;

    const user = await User.findByPk(id, {

        include: {

            model: Post,

        },

    });

    res.render("user-detail", {

        user,

        posts: user.Posts,

    });

});

router.get("/create", (req, res) => {

    res.render("user-form");

});

router.get("/update/:id", async (req, res) => {

    const id = req.params.id;

    const user = await User.findByPk(id);

    res.render("user-form", {

        user,

    });

});

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

    const form = req.body;

    if (!form.id) {

        // CREATE

        console.log("Creating");

        await User.create({

            email: form.email,

            password: form.password,

        });

    } else {

        // UPDATE

        console.log("Updating");

        await User.update(

            {

                email: form.email,

                password: form.password,

            },

            { where: { id: form.id } }

        );

    }

    res.redirect("/");

});

router.post("/delete/:id", async (req, res) => {

    const id = req.params.id;

    await User.destroy({

        where: {

            id: id,

        },

    });

    res.redirect("/");

});

module.exports = router;

lưu ý chỉ có 1 cách để import Model vào thôi, đó chính là từ index.js