# Implement backend

## Cách 1 : hiển thị theo parent 🡪 child

[

    {

        id: 1,

        title: "NodeJS",

        slug: "nodejs",

        description: "Môi trường danh cho dev web, sử dụng Chrome V8 engine",

        categoryId: null,

        createdAt: "2020-12-31T07:18:36.000Z",

        updatedAt: "2020-12-31T07:18:36.000Z",

        childCategories: [

            {

                id: 3,

                title: "Sequelize",

                slug: "sequelize",

                description: "ORM của SQL Database, dành cho NodeJS",

                categoryId: 1,

                createdAt: "2020-12-31T07:18:36.000Z",

                updatedAt: "2020-12-31T07:18:36.000Z",

            },

            {

                id: 4,

                title: "Express",

                slug: "express",

                description: "Framework của MySQL, Hỗ trợ phát triển web, apis vô cùng nhanh chóng",

                categoryId: 1,

                createdAt: "2020-12-31T07:18:36.000Z",

                updatedAt: "2020-12-31T07:18:36.000Z",

            },

        ],

    },

    {

        id: 2,

        title: "Java",

        slug: "java",

        description: "Ngôn ngữ mạnh mẽ, viết 1 lần chạy mọi nơi",

        categoryId: null,

        createdAt: "2020-12-31T07:18:36.000Z",

        updatedAt: "2020-12-31T07:18:36.000Z",

        childCategories: [

            {

                id: 5,

                title: "Hibernate",

                slug: "hibernate",

                description: "ORM dành cho Java, thường hay đi kèm với Spring Boot",

                categoryId: 2,

                createdAt: "2020-12-31T07:18:36.000Z",

                updatedAt: "2020-12-31T07:18:36.000Z",

            },

        ],

    },

];

Associate

Category.hasMany(models.Category, {

                foreignKey: "categoryId",

                as: "childCategories",

            });

Query

const categories = await Category.findAll({

        include: {

            model: Category,

            as: "childCategories",

        },

        where: {

            categoryId: {

                [Op.is]: null,

            },

        },

    });

    console.log(JSON.stringify(categories, null, 4));

## Cách 2 : hiển thị theo whose parent

[

    {

        "id": 1,

        "title": "NodeJS",

        "slug": "nodejs",

        "description": "Môi trường danh cho dev web, sử dụng Chrome V8 engine",

        "categoryId": null,

        "createdAt": "2020-12-31T07:18:36.000Z",

        "updatedAt": "2020-12-31T07:18:36.000Z",

        "parent": null

    },

    {

        "id": 2,

        "title": "Java",

        "slug": "java",

        "description": "Ngôn ngữ mạnh mẽ, viết 1 lần chạy mọi nơi",

        "categoryId": null,

        "createdAt": "2020-12-31T07:18:36.000Z",

        "updatedAt": "2020-12-31T07:18:36.000Z",

        "parent": null

    },

    {

        "id": 3,

        "title": "Sequelize",

        "slug": "sequelize",

        "description": "ORM của SQL Database, dành cho NodeJS",

        "categoryId": 1,

        "createdAt": "2020-12-31T07:18:36.000Z",

        "updatedAt": "2020-12-31T07:18:36.000Z",

        "parent": {

            "id": 1

        }

    },

    {

        "id": 4,

        "title": "Express",

        "slug": "express",

        "description": "Framework của MySQL, Hỗ trợ phát triển web, apis vô cùng nhanh chóng",

        "categoryId": 1,

        "createdAt": "2020-12-31T07:18:36.000Z",

        "updatedAt": "2020-12-31T07:18:36.000Z",

        "parent": {

            "id": 1

        }

    },

    {

        "id": 5,

        "title": "Hibernate",

        "slug": "hibernate",

        "description": "ORM dành cho Java, thường hay đi kèm với Spring Boot",        "categoryId": 2,

        "createdAt": "2020-12-31T07:18:36.000Z",

        "updatedAt": "2020-12-31T07:18:36.000Z",

        "parent": {

            "id": 2

        }

    }

]

Cách này thì dễ render ra view hơn

Associate

Category.belongsTo(models.Category, {

                foreignKey: "categoryId",

                allowNull: true,

                onDelete: "CASCADE",

                as: "parent",

            });

Query

const categories = await Category.findAll({

        include: {

            model: Category,

            attributes: ["id"],

            as: "parent",

        },

    });

    console.log(JSON.stringify(categories, null, 4));