Khi sử dụng Getter & Setter thì bên trong function mình truy cập data bằng method

this.getDataValue(‘field\_name’)

Vì trong getter nếu gọi thẳng field\_name 🡪 bị loop đến chết

Đồng thời ở bên ngoài nếu muốn truy cập vào value gốc (k thông qua get & set) 🡪 vẫn sử dụng

model.getDataValue(‘field\_name’)

## Getters

const User = sequelize.define('user', {

username: {

type: DataTypes.STRING,

get() {

const rawValue = this.getDataValue(username);

return rawValue ? rawValue.toUpperCase() : null;

}

}

});

const user = User.build({ username: 'SuperUser123' });

console.log(user.username); // 'SUPERUSER123'

console.log(user.getDataValue(username)); // 'SuperUser123'

## Setters

const User = sequelize.define('user', {

username: DataTypes.STRING,

password: {

type: DataTypes.STRING,

set(value) {

this.setDataValue('password', hash(value));

}

}

});

const user = User.build({ username: 'someone', password: 'NotSo§tr0ngP4$SW0RD!' });

console.log(user.password); // '7cfc84b8ea898bb72462e78b4643cfccd77e9f05678ec2ce78754147ba947acc'

console.log(user.getDataValue(password)); // '7cfc84b8ea898bb72462e78b4643cfccd77e9f05678ec2ce78754147ba947acc'

## Virtual Fields

Là field sẽ không được lưu vào DB (không tạo column luôn)

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

const User = sequelize.define('user', {

firstName: DataTypes.TEXT,

lastName: DataTypes.TEXT,

fullName: {

type: DataTypes.VIRTUAL,

get() {

return `${this.firstName} ${this.lastName}`;

},

set(value) {

throw new Error('Do not try to set the `fullName` value!');

}

}

});

## getterMethods and setterMethods

giống như Vuejs có thể sử dụng getter & setter riêng không cần tạo field

const User = sequelize.define('user', {

firstName: DataTypes.STRING,

lastName: DataTypes.STRING

}, {

getterMethods: {

fullName() {

return this.firstName + ' ' + this.lastName;

}

},

setterMethods: {

fullName(value) {

const names = value.split(' ');

const firstName = names[0];

const lastName = names.slice(1).join(' ');

this.setDataValue('firstName', firstName);

this.setDataValue('lastName', lastName);

}

}

});