// Defining

Post.hasMany(Reaction);

Reaction.belongsTo(Post);

async function makePostWithReactions(content, reactionTypes) {

const post = await Post.create({ content });

await Reaction.bulkCreate(

reactionTypes.map(type => ({ type, postId: post.id }))

);

return post;

}

await makePostWithReactions('Hello World', [

'Like', 'Angry', 'Laugh', 'Like', 'Like', 'Angry', 'Sad', 'Like'

]);

await makePostWithReactions('My Second Post', [

'Laugh', 'Laugh', 'Like', 'Laugh'

]);

Fetch

Post.findAll({

attributes: {

include: [

[

// Note the wrapping parentheses in the call below!

sequelize.literal(`(

SELECT COUNT(\*)

FROM reactions AS reaction

WHERE

reaction.postId = post.id

AND

reaction.type = "Laugh"

)`),

'laughReactionsCount'

]

]

},

order: [

[sequelize.literal('laughReactionsCount'), 'DESC']

]

});

// result

[

{

"id": 1,

"content": "Hello World",

"laughReactionsCount": 1

},

{

"id": 2,

"content": "My Second Post",

"laughReactionsCount": 3

}

]