

# Sequelize Association one-to-many many-to-many

1. Relasi antara model Subject dan Teacher adalah *one Subject has many Teacher*. Asosiasikan model Subject dan Teacher dengan menambahkan field pada salah satu model untuk menampung referensi dari model lainnya. Lakukan migrasi dan sesuaikan model dengan migrasi yang telah dilakukan
2. Buatlah fitur CRUD untuk tabel Teachers. Ikuti bentuk CRUD untuk tabel Students dan samakan pola routing pada /students/\*.
3. Pada laman yang menampilkan semua data Teacher, munculkan Subject yang ditentukan untuk Teacher tersebut. Bila belum ditentukan, munculkan tulisan 'unassigned'

## All Teachers Data

[Add Teachers Data](#)

Id	First Name	Last Name	Email	Subject	Options
1	Bambang	Suprpto	bambangsuprpto@sekolah.id	unassigned	<a href="#">Edit</a>   <a href="#">Delete</a>
2	Rukmana	Fatmawati	rukmanafatmawati@sekolah.id	unassigned	<a href="#">Edit</a>   <a href="#">Delete</a>
3	Butet	Naiborhu	butetnaiborhu@sekolah.id	unassigned	<a href="#">Edit</a>   <a href="#">Delete</a>
4	Yulius	Prawiranegara	yuliusprawiranegara@sekolah.id	unassigned	<a href="#">Edit</a>   <a href="#">Delete</a>

4. Pada route /subjects , tampilkan semua Subject dengan Teacher yang telah di-assign untuk subject tersebut.
  - a. Pada laman edit Teacher, bila Teacher sudah ditentukan Subject-nya, pastikan Subject tersebut yang muncul pada dropdown pilihan Subject

## All Teachers Data

[Add Teachers Data](#)

Id	First Name	Last Name	Email	Subject	Options
1	Bambang	Suprpto	bambangsuprpto@sekolah.id	Ekonomi	<a href="#">Edit</a>   <a href="#">Delete</a>
2	Rukmana	Fatmawati	rukmanafatmawati@sekolah.id	Ekonomi	<a href="#">Edit</a>   <a href="#">Delete</a>
3	Butet	Naiborhu	butetnaiborhu@sekolah.id	Fisika	<a href="#">Edit</a>   <a href="#">Delete</a>
4	Yulius	Prawiranegara	yuliusprawiranegara@sekolah.id	Fisika	<a href="#">Edit</a>   <a href="#">Delete</a>

## All Subjects Data

Id	Subject Name	Teacher
1	Fisika	1. Yulius Prawiranegara 2. Butet Naiborhu
2	Ekonomi	1. Bambang Suprpto 2. Rukmana Fatmawati

- Asosiasikan model Student dan model Subject. Relasi antara kedua model tersebut adalah *many-to-many*, dimana satu Student dapat mengambil lebih dari satu Subject dan satu Subject memiliki banyak Student. Lakukan pembuatan model dan migrasi yang perlu dilakukan. Tambahkan field 'score : integer' pada tabel conjunction.
- Pada laman yang menampilkan data semua Student, tambahkan fitur 'add subject' dimana masing-masing Student dapat memilih Subject untuk diambil. Gunakan routing
  - GET /students/:id/addsubject (menampilkan data student dan pilihan subject)
  - POST /students/:id/addsubject (meng-handle input form)
- Pada /subjects, tambahkan fitur 'enrolled students' untuk menampilkan semua siswa yang mengambil Subject tersebut. Gunakan routing /subjects/:id/enrolledstudents
  - Gambar
- Pada halaman /subjects/:id/showstudents tambahkan fitur 'give score' untuk memberikan score Student yang mengambil Subject tersebut.
  - Gambar
- Pada routing /student tambahkan fitur 'academic report' sehingga setiap Student dapat data diri student tersebut serta melihat semua Subject yang diambil beserta score yang diberikan. Bila ada Subject yang tidak memiliki score, tuliskan 'empty'
- Perbaiki kerapian data di tabel, tambahkan kolom 'No' di paling kiri tabel dan berikan yang berurutan. Urutkan data berdasarkan alfabet (data Teacher dan Student berdasarkan first\_name)