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 (5)
- 2. Buatlah fitur CRUD untuk tabel Teachers. Ikuti bentuk CRUD untuk tabel Students dan samakan pola routing pada /students/*. (20, each 5)
- 3. Pada laman yang menampilkan semua data Teacher, munculkan Subject yang ditentukan untuk Teacher tersebut. Bila belum ditentukan, munculkan tulisan 'unassigned' (15)

All Teachers Data

Add Teachers Data

ld	First Name	Last Name	Email	Subject	Options
1	Bambang	Suprapto	bambangsuprapto@sekolah.id	unassigned	Edit Delete
2	Rukmana	Fatmawati	rukmanafatmawati@sekolah.id	unassigned	Edit Delete
3	Butet	Naiborhu	butetnaiborhu@sekolah.id	unassigned	Edit Delete
4	Yulius	Prawiranegara	yuliusprawiranegara@sekolah.id	unassigned	Edit Delete

- 4. Pada route /subjects , tampilkan semua Subject dengan Teacher yang telah di-assign untuk subject tersebut. (10)
 - 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

ld	First Name	Last Name	Email	Subject	Options
1	Bambang	Suprapto	bambangsuprapto@sekolah.id	Ekonomi	Edit Delete
2	Rukmana	Fatmawati	rukmanafatmawati@sekolah.id	Ekonomi	Edit Delete
3	Butet	Naiborhu	butetnaiborhu@sekolah.id	Fisika	Edit Delete
4	Yulius	Prawiranegara	yuliusprawiranegara@sekolah.id	Fisika	Edit Delete

All Subjects Data

ld	Subject Name	Teacher	
1	Fisika	Yulius Prawiranegara	
		2. Butet Naiborhu	
2	Ekonomi	Bambang Suprapto	
		2. Rukmana Fatmawati	

- 5. 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. (5)
- 6. Pada laman yang menampikan data semua Student, tambahkan fitur 'add subject' dimana masing-masing Student dapat memilih Subject untuk diambil. Gunakan routing (10)
 - a. GET /students/:id/addsubject (menampilkan data student dan pilihan subject)

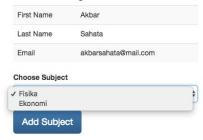
Students Data

add student

ld	First Name	Last Name	Email	Actions	Options
1	Akbar	Sahata	akbarsahata@mail.com	Add Subject	Edit Delete
17	Agindo	Rahmat	agindorahmat@mail.com	Add Subject	Edit Delete
18	Dul	Sumbang	dulsumbang@gmail.com	Add Subject	Edit Delete

b. POST /students/:id/addsubject (meng-handle input form)

Add Subject to Student



7. Pada /subjects, tambahkan fitur 'enrolled students' untuk menampilkan semua siswa yang mengambil Subject tersebut beserta score-nya. Gunakan routing /subjects/:id/enrolledstudents (10)

All Subjects Data

ld	Subject Name	Teacher	Actions
1	Fisika	Yulius Prawiranegara	Enrolled Students
		2. Butet Naiborhu	
2	Ekonomi	Bambang Suprapto	Enrolled Students
		Rukmana Fatmawati	

- 8. Pada halaman /subjects/:id/enrolledstudents tambahkan fitur 'give score' untuk memberikan score Student yang mengambil Subject tapi masih belum memiliki score. (20)
 - a. GET /subjects/:id/givescore (menampilkan form untuk memberi nilai)
 - b. POST /subjects/:id/givescore (meng-handle input data)

Fisika

return to subject list

Name	Score	
Agindo Rahmat	Give Score	
Akbar Sahata	90	

- 9. Perbaiki kerapihan data di tabel, tambahkan kolom `No` di paling kiri tabel dan berikan yang berurutan. Urutkan data berdasarkan alfabet (5)
 - a. Data Teacher dan Student berdasarkan first_name
 - b. Data enrolled students berdasarkan Student.first_name