



Sequelize



Ayudantes



¿Qué es?

Es un Framework ORM de Node.js para soluciones como MySQL o Postgres.

Los ORM son una parte muy importante del desarrollo de aplicaciones ya que nos permiten crear una abstracción de la base de datos en nuestra aplicación y, evitan el uso de sentencias SQL que si bien pueden funcionar en un momento, si después deseamos cambiar el motor de base de datos por uno más potente, entonces nos puede causar problemas.

Ahora... ¿Qué hago?

Pueden crear los modelos utilizados por Sequelize a partir de su base de datos.

- Instalar Sequelize-auto.
 - `npm install -g sequelize-auto`
- Crear los modelos con el siguiente comando.
 - `sequelize-auto -o "[CarpetaModelos]" -d [BaseDeDatos] -h [host] -u [Usuario] -p [Puerto BD] -x [Password] -e mysql`
 - #Ejemplo
 - `sequelize-auto -o "./models" -d test -h localhost -u root -p 3306 -x admin -e mysql`

Configurando la conexión

```
var fs = require("fs");
var path = require("path");
const router = express.Router();
var Sequelize = require("sequelize");
var sequelize = new Sequelize([bd_name],[user],[pass]{
  host: 'localhost',
  dialect: 'mysql'|'postgresql',
});

//Testing conection
sequelize
  .authenticate()
  .then(() => {
    console.log('Connection has been established successfully.');
```

Importando

Puedes usar un archivo para cada modelo, de ser así, debes agregar lo que está en la página 6 al archivo .js.

Luego definir los modelos como se muestra en la página 7.

Puedes definir cada modelo a continuación de la conexión, de ser así, saltar la página 6 y continuar el archivo .js con lo que se encuentra en la página 7:

```
const User = sequelize.define('user', {});
```

Definición de modelos.

Solo si los modelos están por separado

```
var db = {};  
  
fs.readdirSync(__dirname + '/../models').filter(function(file) {  
  return (file.indexOf(".") !== 0);  
}).forEach(function(file) {  
  var model = sequelize['import'](path.join(__dirname + '/../models/', file));  
  db[model.name] = model;  
});  
  
Object.keys(db).forEach(function(modelName) {  
  if (db[modelName].associate) {  
    db[modelName].associate(db);  
  }  
});  
  
db.sequelize = sequelize;  
db.Sequelize = Sequelize;  
  
module.exports = db;
```



Recuerden cambiar
estas rutas

Definición de modelos

```
const User = sequelize.define('user', {
  id: {
    type: Sequelize.INTEGER,
    autoIncrement: true,
    primaryKey: true
  },
  firstName: {
    type: Sequelize.STRING,
    allowNull: false
  },
  dateofbirth: {
    type: Sequelize.DATE
  }
});
```

```
User.sync({force: true}).then(() => {
  return User.create({
    firstName: 'John',
    Dateofbirth: Date.UTC(2000, 1, 1)
  });
});
```

Si no existe, se
crea la tabla

Se crea un
usuario

Crear y mostrar

```
var nuevo_user=User.sync().then(function () {  
    return User.create({  
        firstName: 'gema',  
        dateofbirth: Date.UTC(1997, 9, 1)  
    });  
});
```

```
User.findAll().then(function (users) {  
    users.forEach(function(elemento) {  
        console.log(elemento.dataValues);  
    });  
});
```


Queries...

```
const Op = Sequelize.Op;
```

```
User.findAll({  
  where: {  
    id: 2  
  }  
});
```


```
SELECT * FROM user WHERE id = 2
```

```
User.findAll({  
  where: {  
    id: 12,  
    status: 'active'  
  }  
});
```

```
SELECT * FROM user WHERE id = 12 AND status  
= 'active';
```

```
User.update({
  dateofbirth: Date.UTC(2000, 10, 1),
}, {
  where: {
    id: {
      [Op.ne]: 13
    }
  }
});
```

// UPDATE user SET dateofbirth = 01/10/2000
WHERE id NOT 13;



```
User.destroy({
  where: {
    status: 'inactive'
  }
});
```

// DELETE FROM user WHERE status = 'inactive';

Trabajo personal

- Existen variados tipos de datos. Ej string, date, text, integer, etc.
- Se pueden utilizar validaciones dentro de las definiciones de los modelos. Ej que el campo esté entre tales valores.
- Se pueden utilizar todo tipo de operadores. Ej mayor que, menor que, entre, etc.

Links de utilidad

- <http://docs.sequelizejs.com/>
- <https://scotch.io/tutorials/getting-started-with-node-express-and-postgres-using-sequelize>