

```
express nome_do_projeto --ejs
npm install
```

instalar as dependências

```
npm install express mysql2 sequelize dotenv
npm install nodemon sequelize-cli -D
```

criar arquivos na raiz do projeto
.env e .gitignore

no arquivo .env colocar as variáveis com os valores de acordo com seu ambiente Dev

```
DB_USER=
DB_PASS=
DB_NAME=
DB_HOST=127.0.0.1
DB_PORT=3306
```

No arquivo .gitignore
ignorar os seguintes arquivos

```
/node_modules
.env
```

Criar as seguintes pastas na raiz do projeto

```
models
controllers
config
```

dentro da pasta config criar o arquivo database.js
colocar o código de conexão do banco

```
require('dotenv').config()
```

```
module.exports = {
  username:process.env.DB_USER,
  password:process.env.DB_PASS,
  database:process.env.DB_NAME,
  host:process.env.DB_HOST,
  dialect:"mysql"
}
```

dentro da pasta models criar um arquivo chamado index.js
colocar as configurações conforme abaixo:

Ou você pode usar o comando: npx sequelize init:models

Obs.: Se você utilizar o comando lembrar de corrigir as linhas "const config = require('../config/d
"sequelize = new Sequelize(process.env[config.use_env_variable], config);" Tirando o "[env]" e pass
caminho de sua conexão com banco de dados

```
'use strict';
```

```
const fs = require('fs');
const path = require('path');
const Sequelize = require('sequelize');
const basename = path.basename(__filename);
```

```

const env = process.env.NODE_ENV || 'development';
const config = require('../config/database');
const db = {};

let sequelize;
if (config.use_env_variable) {
  sequelize = new Sequelize(process.env[config.use_env_variable], config);
} else {
  sequelize = new Sequelize(config.database, config.username, config.password, config);
}

fs
  .readdirSync(__dirname)
  .filter(file => {
    return (file.indexOf('.') !== 0) && (file !== basename) && (file.slice(-3) === '.js');
  })
  .forEach(file => {
    const model = sequelize['import'](path.join(__dirname, file));
    db[model.name] = model;
  });

Object.keys(db).forEach(modelName => {
  if (db[modelName].associate) {
    db[modelName].associate(db);
  }
});

db.sequelize = sequelize;
db.Sequelize = Sequelize;

module.exports = db;

```

na Raiz do projeto criar o arquivo .sequelizerc e colocar o seguinte código

```

const path = require("path");
module.exports = {
  config: path.resolve("config", "database.js"),
  models: path.resolve("models"),
}

```

dentro de models criar um arquivo js
Em nosso exemplo vamos criar um arquivo de Usuario.js
Dentro de Usuario.js usar o código abaixo como modelo
Lembrando que devemos ter essa tabela em nosso banco de dados

```

const Usuario = (sequelize, DataTypes) => {
  let usuario = sequelize.define(
    'Usuario',
    {
      id_usuario: {
        type: DataTypes.INTEGER,
        primaryKey: true,
        autoIncrement: true,

```

```

        allowNull: true
      },
      nome: {
        type: DataTypes.STRING,
        allowNull: true
      },
      email: {
        type: DataTypes.STRING,
        allowNull: true
      },
      senha: {
        type: DataTypes.STRING,
        allowNull: true
      }
    },
    {
      tableName: "usuario",
      timestamps: false
    }
  )

  usuario.associate = (models) => {
    //usuario.hasMany(models.Endereco, { foreignKey: 'fk_usuarios', as: 'enderecos' });
    //usuario.hasMany(models.Pedido, { foreignKey: 'fk_usuarios', as: 'pedidos' });
  };

  return usuario;
}

module.exports = Usuario;

```

Para rodar o teste

Criar uma pasta na raiz do projeto chamada testes;

Dentro desta criar um arquivo chamado usuarioTeste.js e colocar o seguinte código

```

const { sequelize, Usuario } = require('../models');

```

```

Usuario.findAll().then(
  data => {
    console.log(data.map( u => u.toJSON()));
    sequelize.close();
  }
)

```

Para rodar o teste basta executar
 node testes/usuarioTeste.js

Referências de código padrão:

<https://github.com/ALESSANDROLMENEZES/sequelize/tree/73080f7b67e18711805f97ba005c702096d408be>