САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Дисциплина: Бэк-энд разработка

Практическая работа No2

Выполнил: Бурак Петр

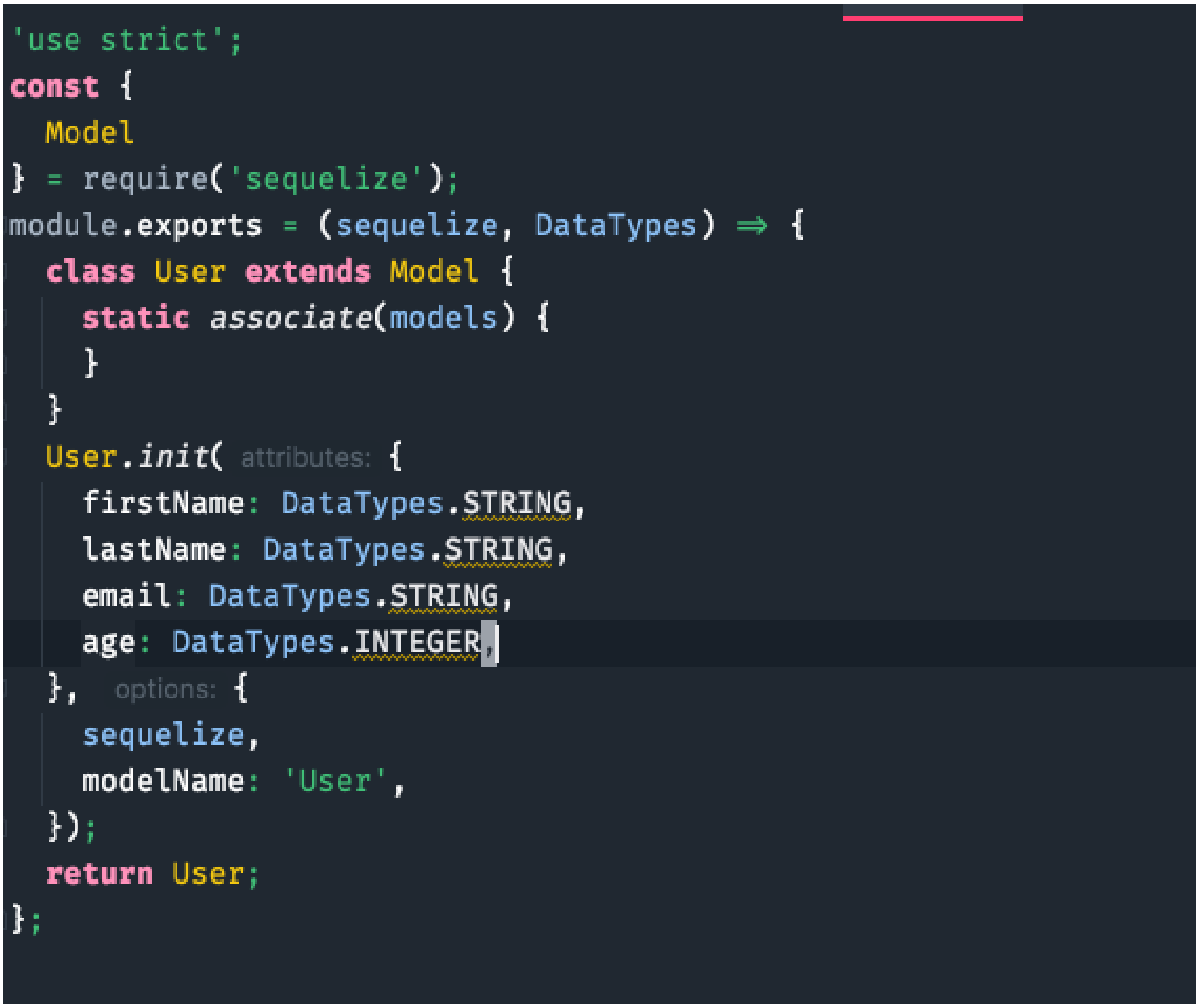
Группа: K33402

Проверил: Добряков Д. И.

Санкт-Петербург 2022 г.

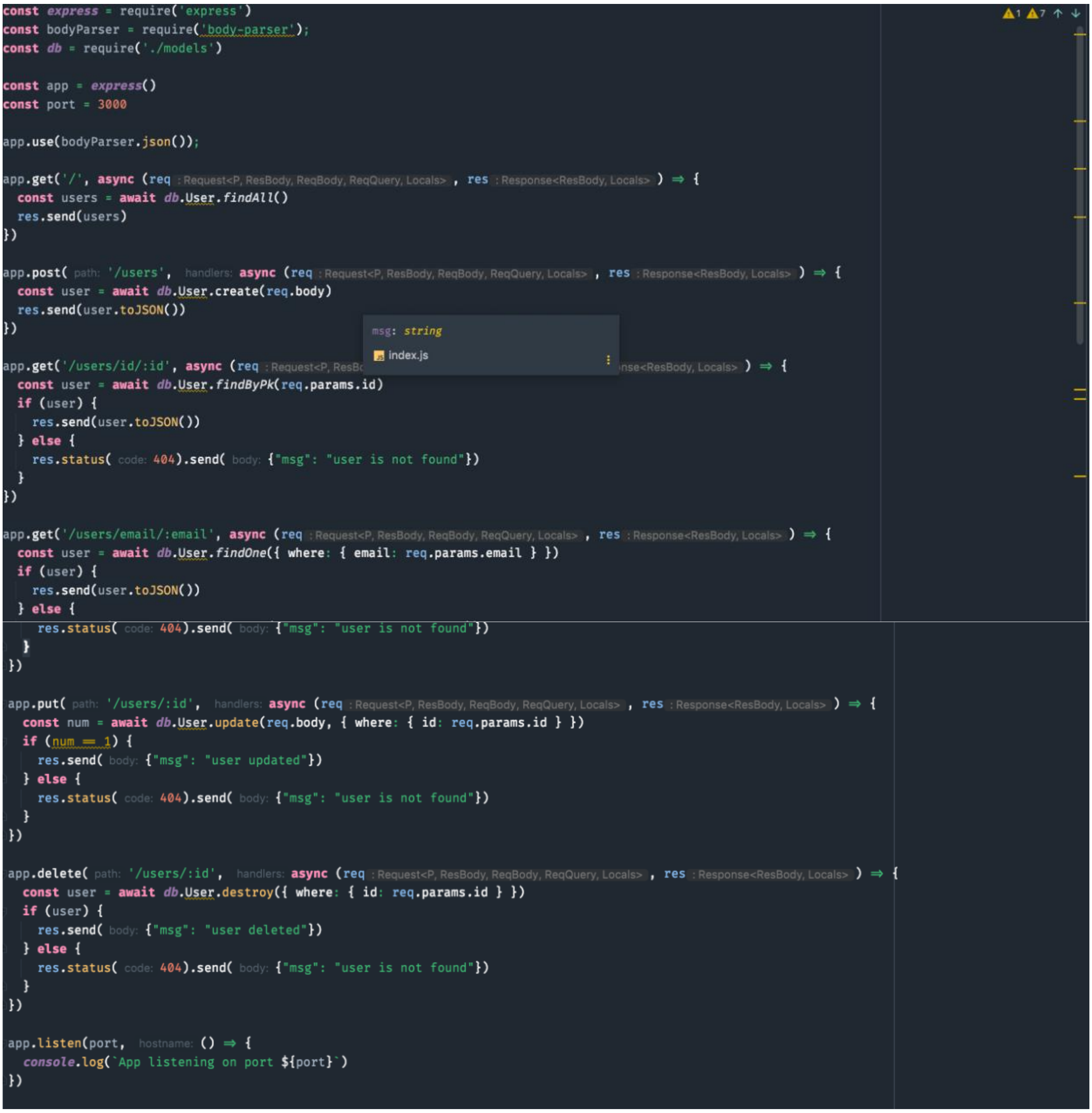
Ход работы

Созданная модель пользователя:



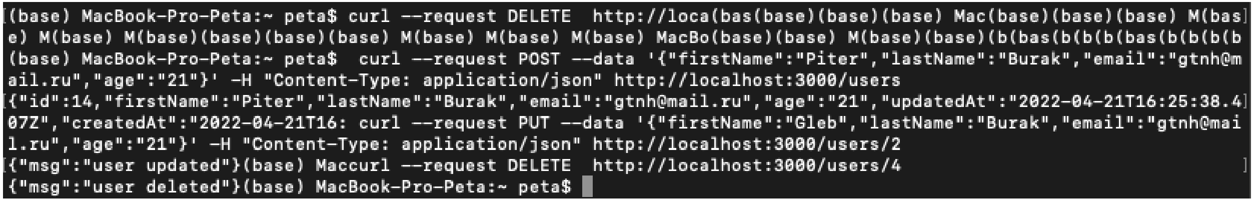
```
'use strict';
const {
  Model
} = require('sequelize');
module.exports = (sequelize, DataTypes) => {
  class User extends Model {
    static associate(models) {
    }
  }
  User.init( attributes: {
    firstName: DataTypes.STRING,
    lastName: DataTypes.STRING,
    email: DataTypes.STRING,
    age: DataTypes.INTEGER,
  },  options: {
    sequelize,
    modelName: 'User',
  });
  return User;
};
```

CRUD-методы для работы с пользователем:



```javascript
const express = require('express')
const bodyParser = require('body-parser');
const db = require('./models')

const app = express()
const port = 3000

app.use(bodyParser.json());

app.get('/', async (req, res) => {
  const users = await db.User.findAll()
  res.send(users)
})

app.post('/users', async (req, res) => {
  const user = await db.User.create(req.body)
  res.send(user.toJSON())
})

app.get('/users/id/:id', async (req, res) => {
  const user = await db.User.findByPk(req.params.id)
  if (user) {
    res.send(user.toJSON())
  } else {
    res.status(404).send({"msg": "user is not found"})
  }
})

app.get('/users/email/:email', async (req, res) => {
  const user = await db.User.findOne({ where: { email: req.params.email } })
  if (user) {
    res.send(user.toJSON())
  } else {
    res.status(404).send({"msg": "user is not found"})
  }
})

app.put('/users/:id', async (req, res) => {
  const num = await db.User.update(req.body, { where: { id: req.params.id } })
  if (num == 1) {
    res.send({"msg": "user updated"})
  } else {
    res.status(404).send({"msg": "user is not found"})
  }
})

app.delete('/users/:id', async (req, res) => {
  const user = await db.User.destroy({ where: { id: req.params.id } })
  if (user) {
    res.send({"msg": "user deleted"})
  } else {
    res.status(404).send({"msg": "user is not found"})
  }
})

app.listen(port, () => {
  console.log(`App listening on port ${port}`)
})
```

Пример работы



```
(base) MacBook-Pro-Peta:~ peta$ curl --request DELETE  http://loca(bas(base)(base)(base) Mac(base)(base)(base) M(bas
e) M(base) M(base)(base)(base)(base) M(base) M(base) M(base) MacBo(base)(base) M(base)(base)(b(bas(b(b(b(bas(b(b(b(b
(base) MacBook-Pro-Peta:~ peta$  curl --request POST --data '{"firstName":"Piter","lastName":"Burak","email":"gtnh@m
ail.ru","age":"21"}' -H "Content-Type: application/json" http://localhost:3000/users
{"id":14,"firstName":"Piter","lastName":"Burak","email":"gtnh@mail.ru","age":"21","updatedAt":"2022-04-21T16:25:38.4
07Z","createdAt":"2022-04-21T16: curl --request PUT --data '{"firstName":"Gleb","lastName":"Burak","email":"gtnh@mai
l.ru","age":"21"}' -H "Content-Type: application/json" http://localhost:3000/users/2
{"msg":"user updated"}(base) Maccurl --request DELETE  http://localhost:3000/users/4
{"msg":"user deleted"}(base) MacBook-Pro-Peta:~ peta$ 
```