Lab Assignment

Q1.

server.js

const express = require('express');

const bodyParser = require('body-parser');

const bcrypt = require('bcrypt');

const session = require('express-session');

const mongoose = require('mongoose');

const app = express();

// Middleware

app.use(bodyParser.urlencoded({ extended: true }));

app.use(session({

secret: 'your\_secret\_key',

resave: false,

saveUninitialized: true

}));

// MongoDB connection

mongoose.connect('mongodb://localhost:27017/test\_db', { useNewUrlParser: true, useUnifiedTopology: true })

.then(() => console.log('MongoDB connected'))

.catch(err => console.error('Error connecting to MongoDB:', err));

// User schema and model

const userSchema = new mongoose.Schema({

username: String,

password: String

});

const User = mongoose.model('User', userSchema);

// Authentication middleware

const authenticateUser = (req, res, next) => {

if (req.session && req.session.user) {

next();

} else {

res.status(401).send('Unauthorized');

}

};

// Routes

app.get('/', (req, res) => {

res.sendFile(\_\_dirname + '/login.html');

});

app.post('/login', async (req, res) => {

const { username, password } = req.body;

try {

const user = await User.findOne({ username: username });

if (user && bcrypt.compareSync(password, user.password)) {

req.session.user = username;

res.redirect('/dashboard');

} else {

res.status(401).send('Invalid username or password');

}

} catch (error) {

console.error('Error finding user:', error);

res.status(500).send('Internal Server Error');

}

});

app.get('/dashboard', authenticateUser, (req, res) => {

res.send(`Welcome, ${req.session.user}!`);

});

app.get('/logout', (req, res) => {

req.session.destroy((err) => {

if (err) {

console.error(err);

} else {

res.redirect('/');

}

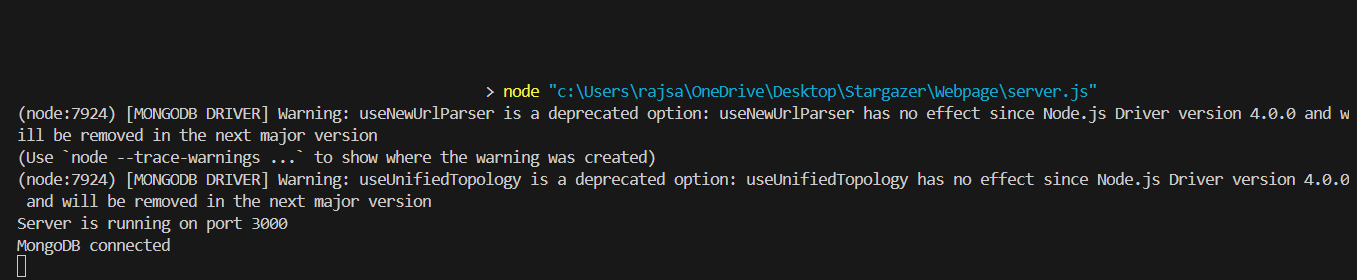
});

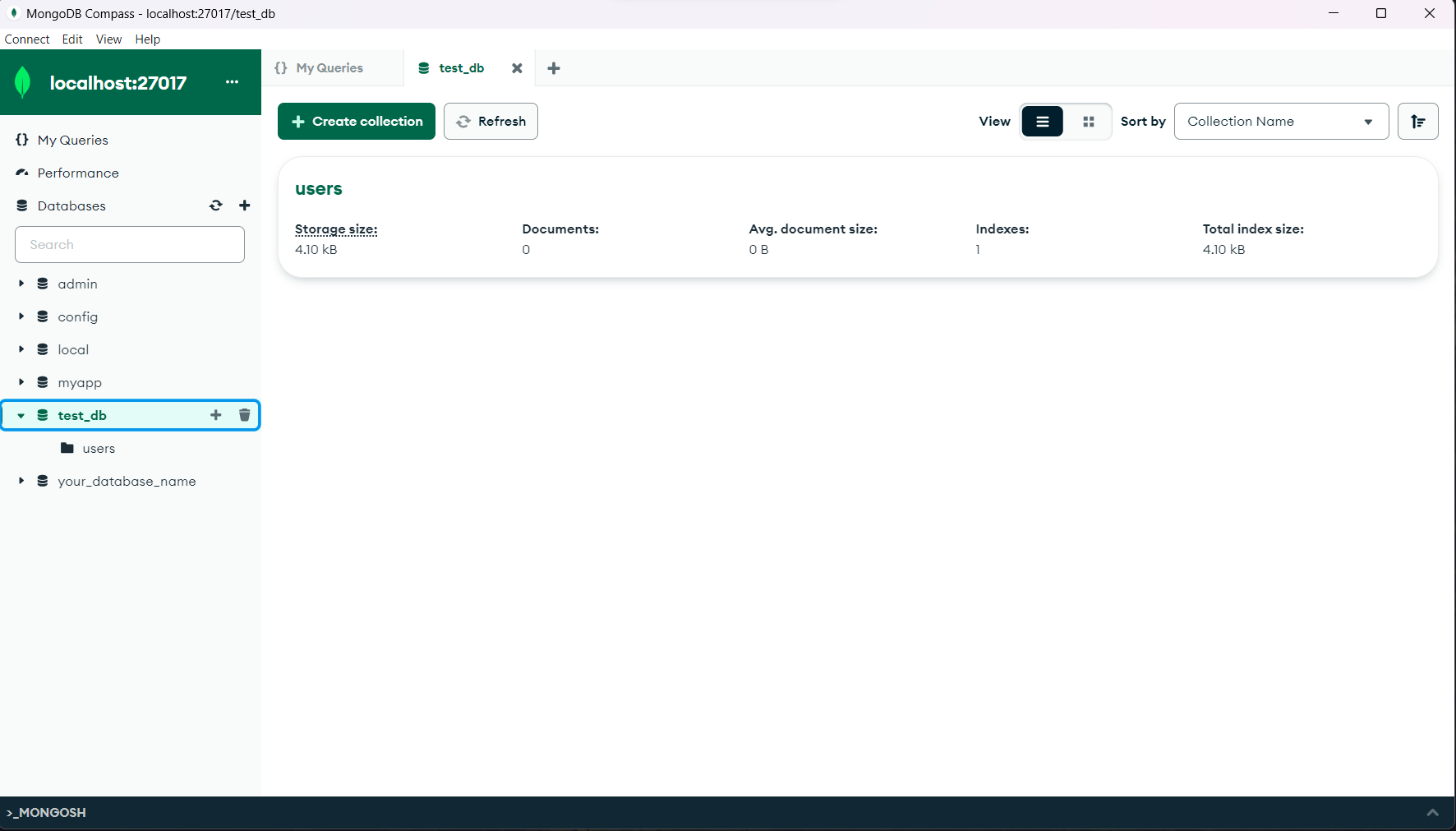
});

app.listen(3001, () => {

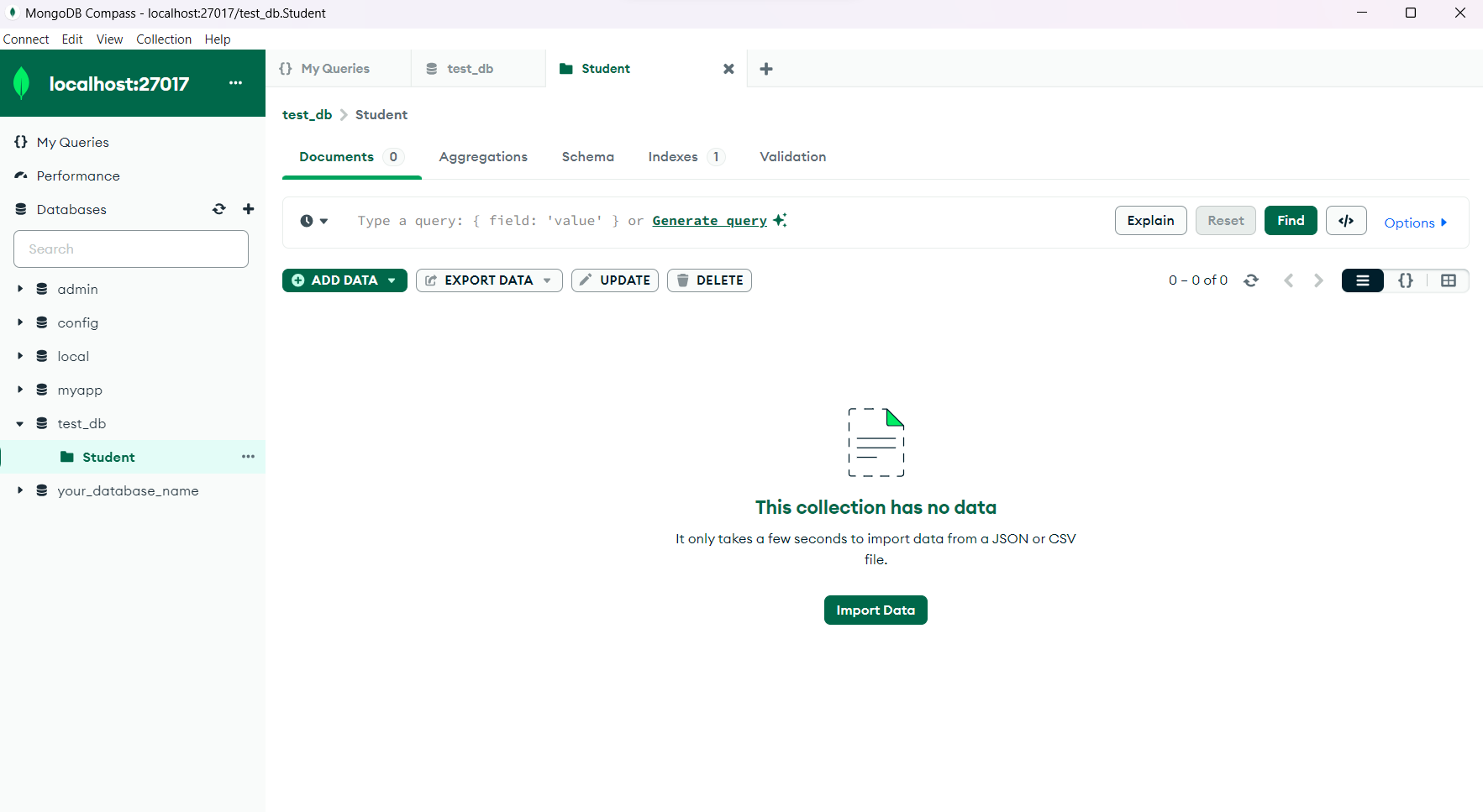
console.log('Server is running on port 3000');

});





Q3



Q4.

[

{ "name": "Saksham", "age": 25, "email": "john@example.com" },

{ "name": "Sarthak", "age": 22, "email": "jane@example.com" },

{ "name": "Michael Kuntal", "age": 30, "email": "michael@example.com" },

{ "name": "Saurabh Pandey", "age": 28, "email": "emily@example.com" },

{ "name": "Shikhar Wilson", "age": 26, "email": "david@example.com" },

{ "name": "Shivanshu Kahyap", "age": 24, "email": "sarah@example.com" },

{ "name": "Matthew Urade", "age": 29, "email": "matthew@example.com" },

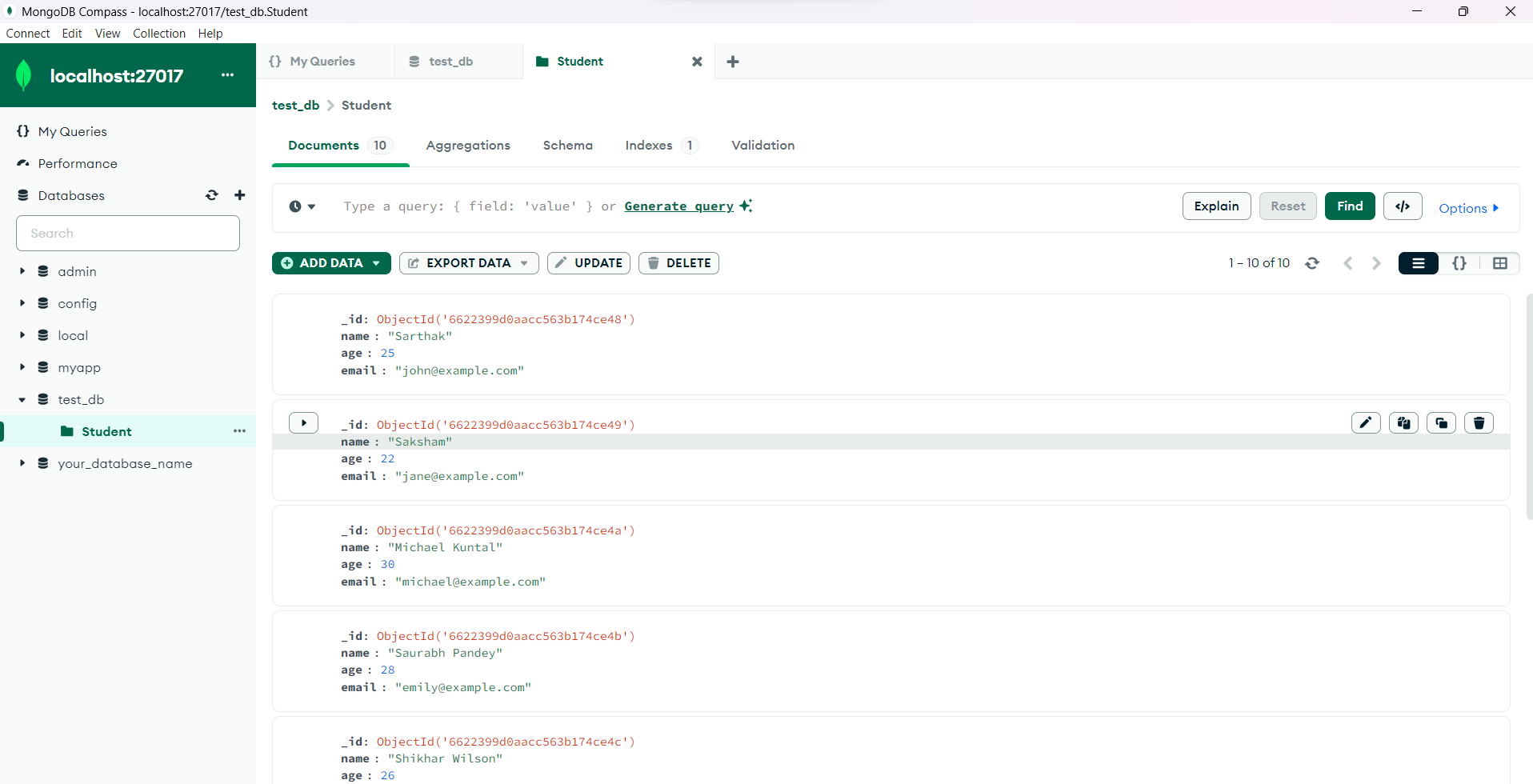
{ "name": "Shubham", "age": 23, "email": "emma@example.com" },

{ "name": "Dhote", "age": 27, "email": "christopher@example.com" },

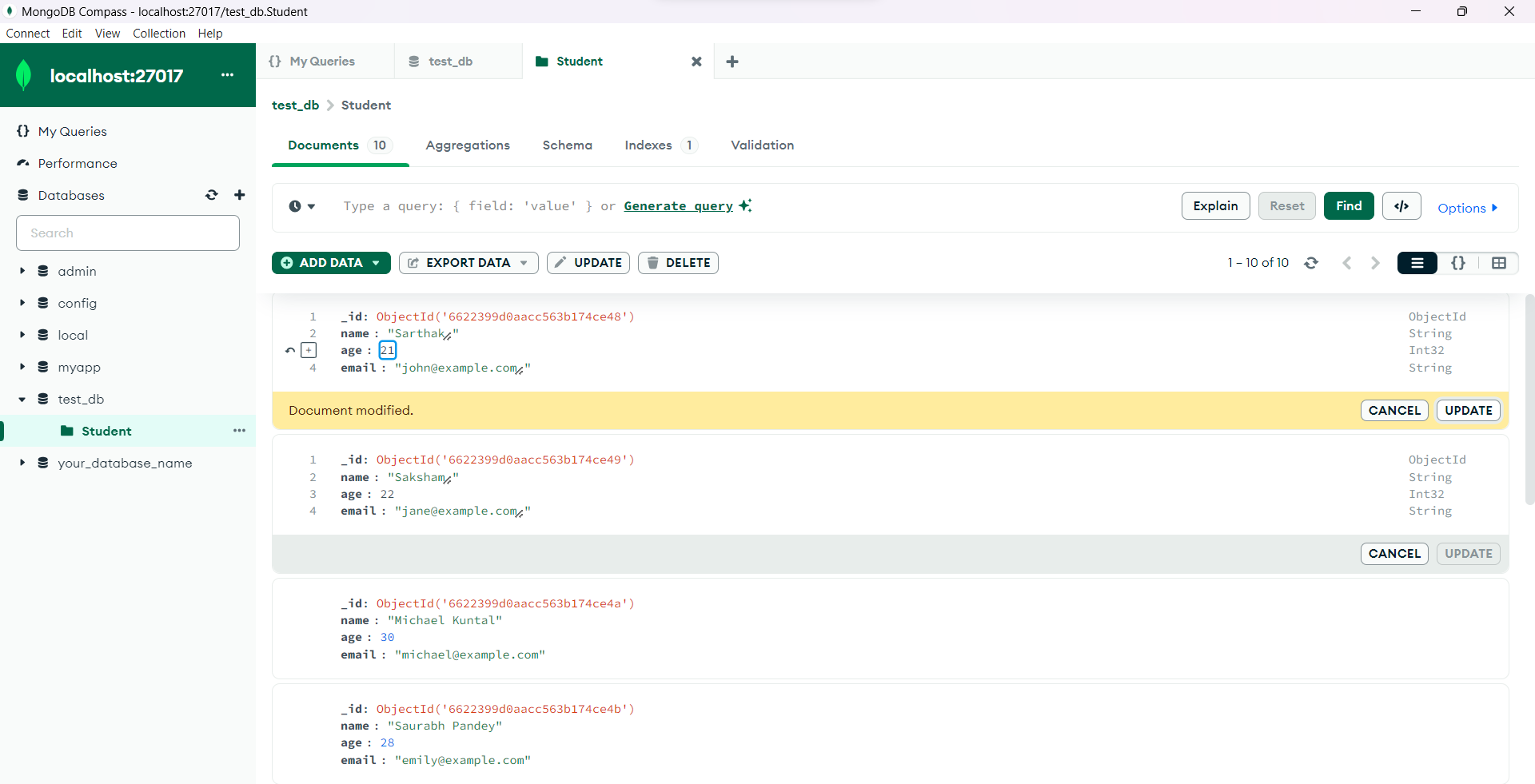
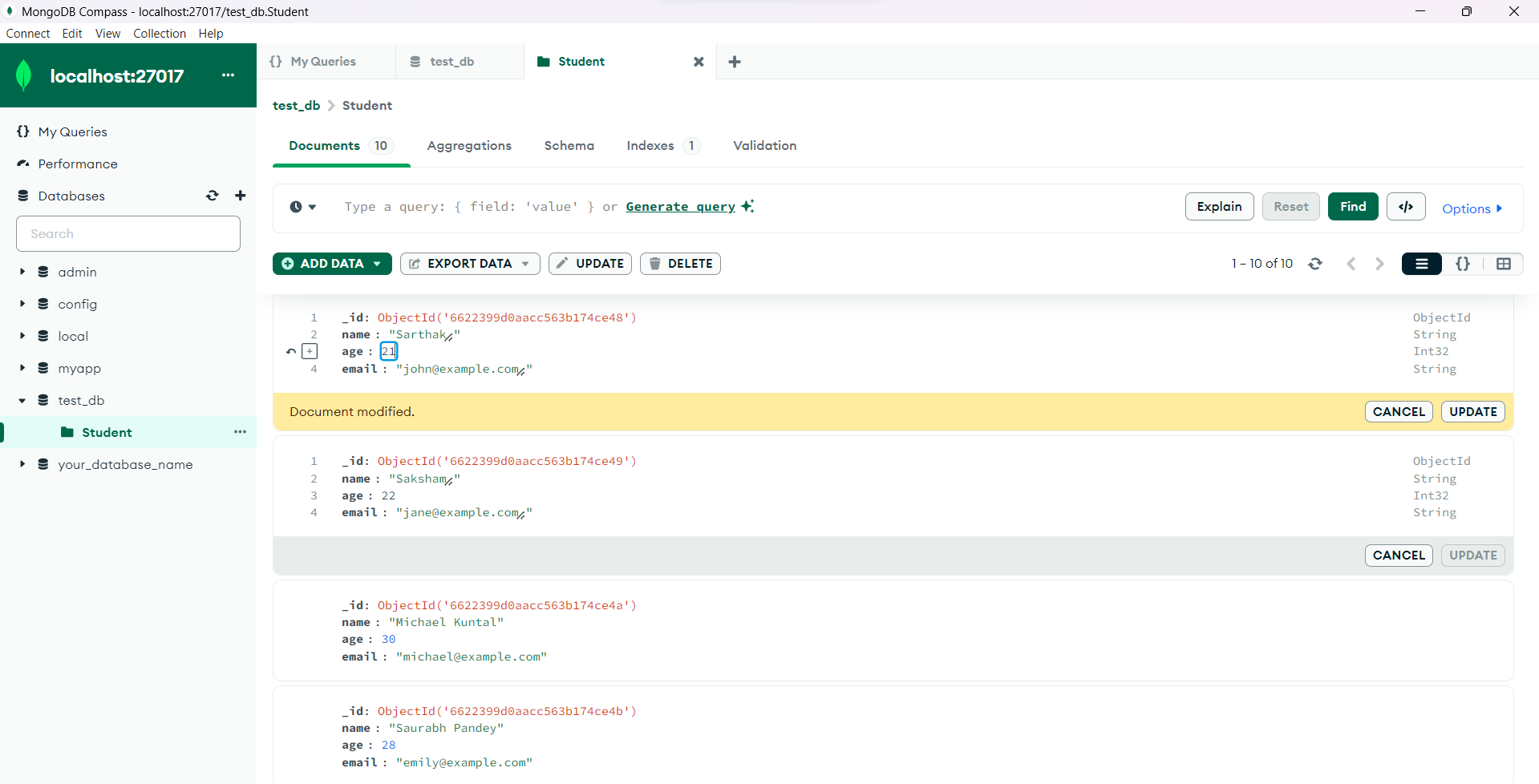
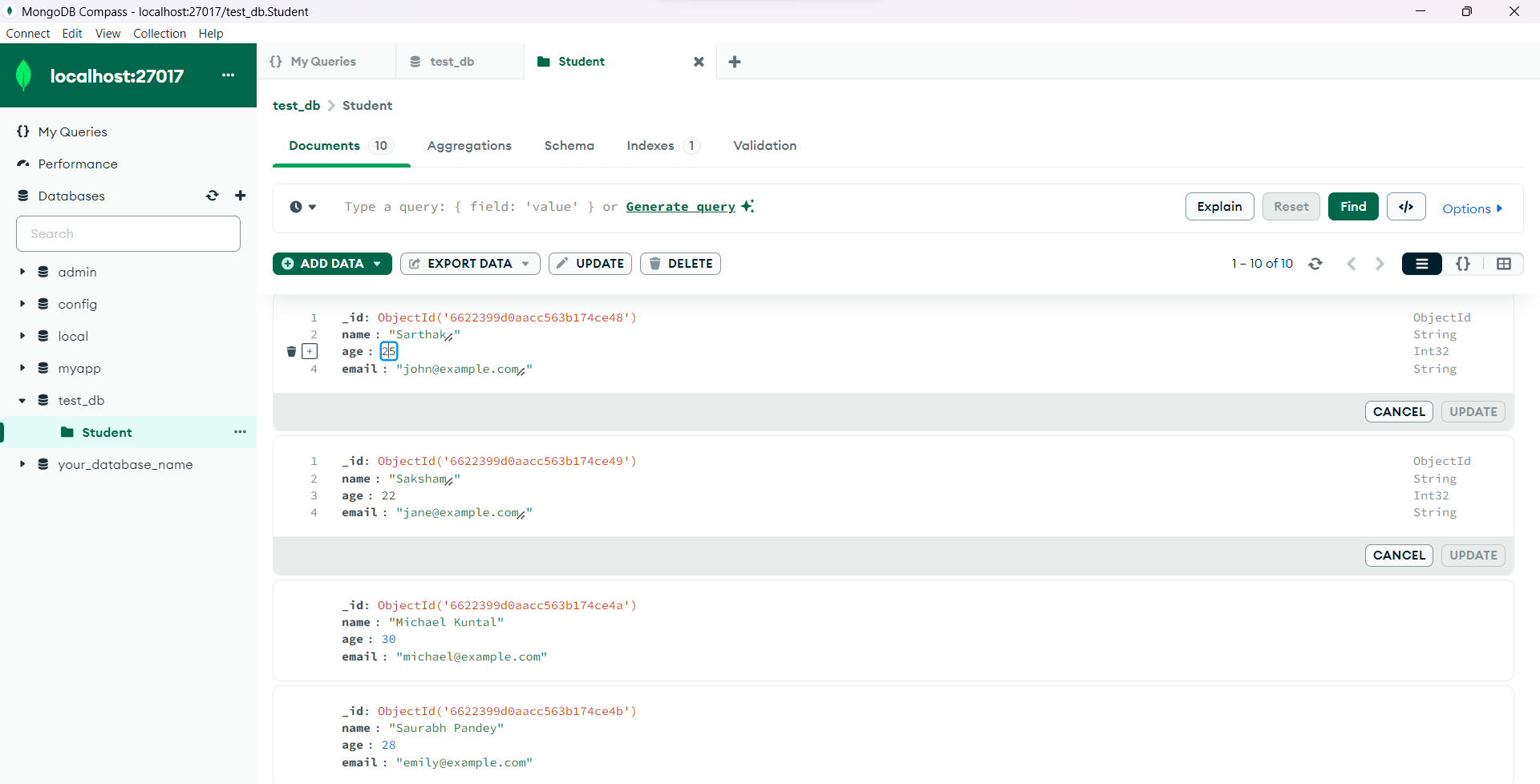
{ "name": "RohiT Sharma", "age": 31, "email": "olivia@example.com" }

]

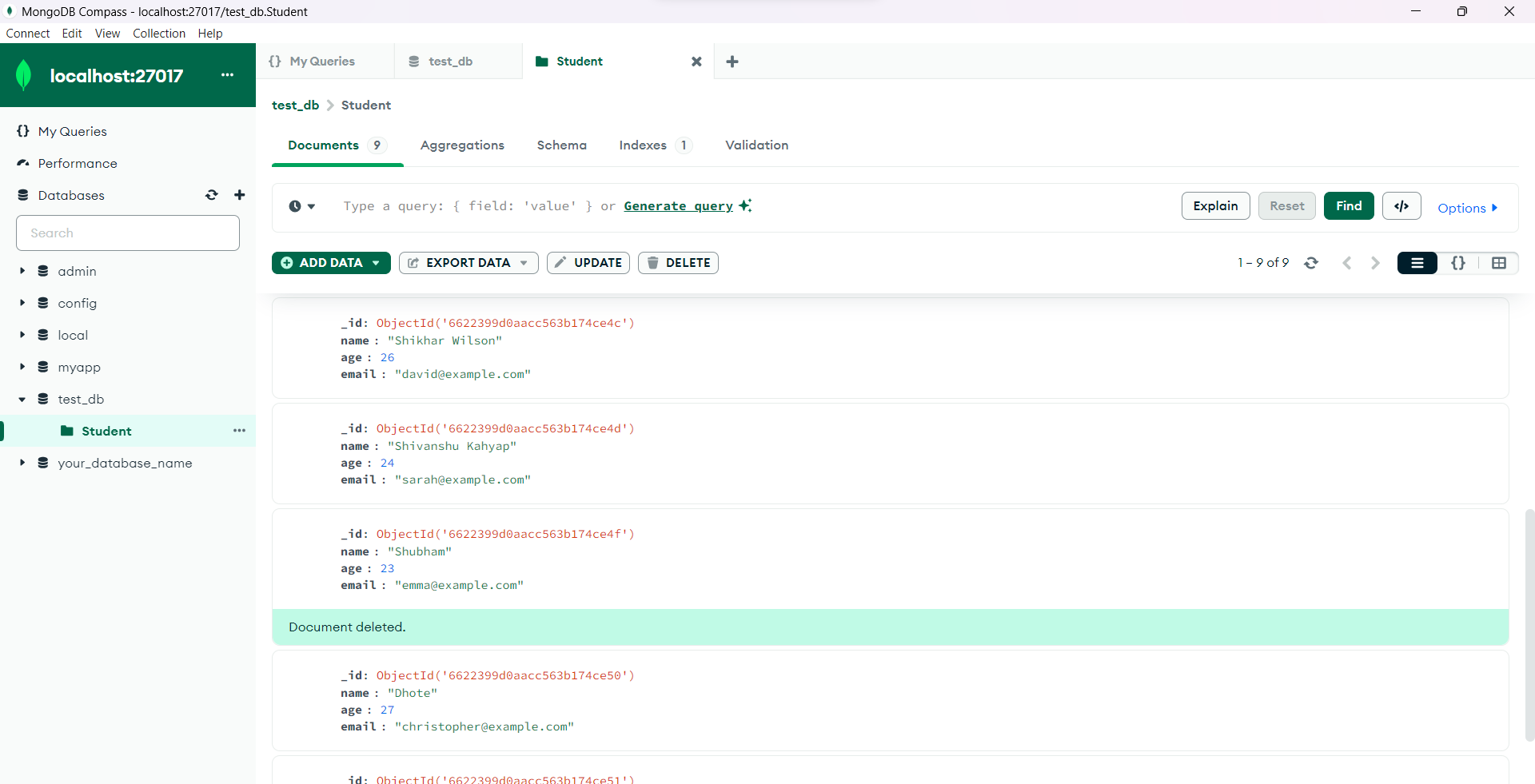
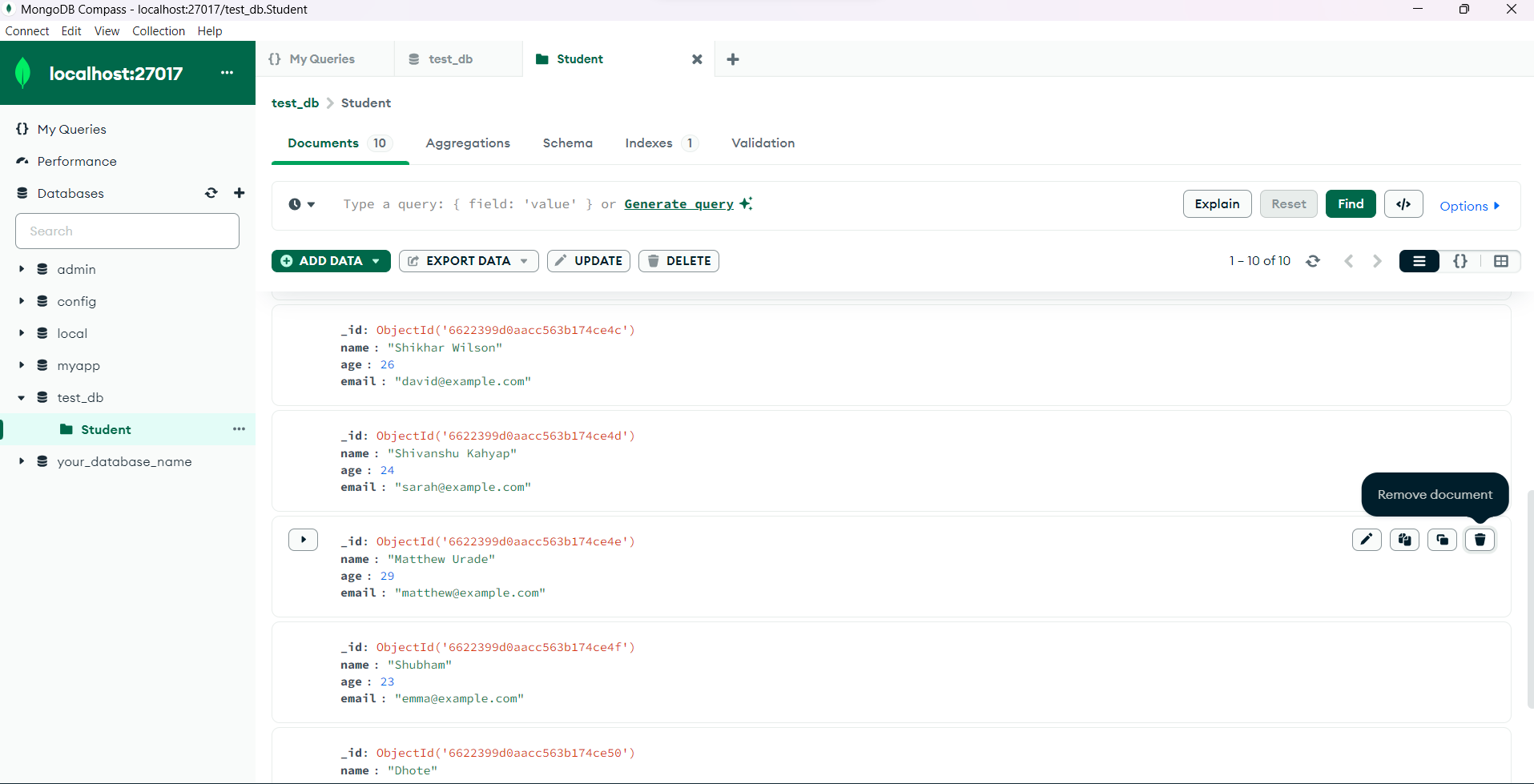
Question 5



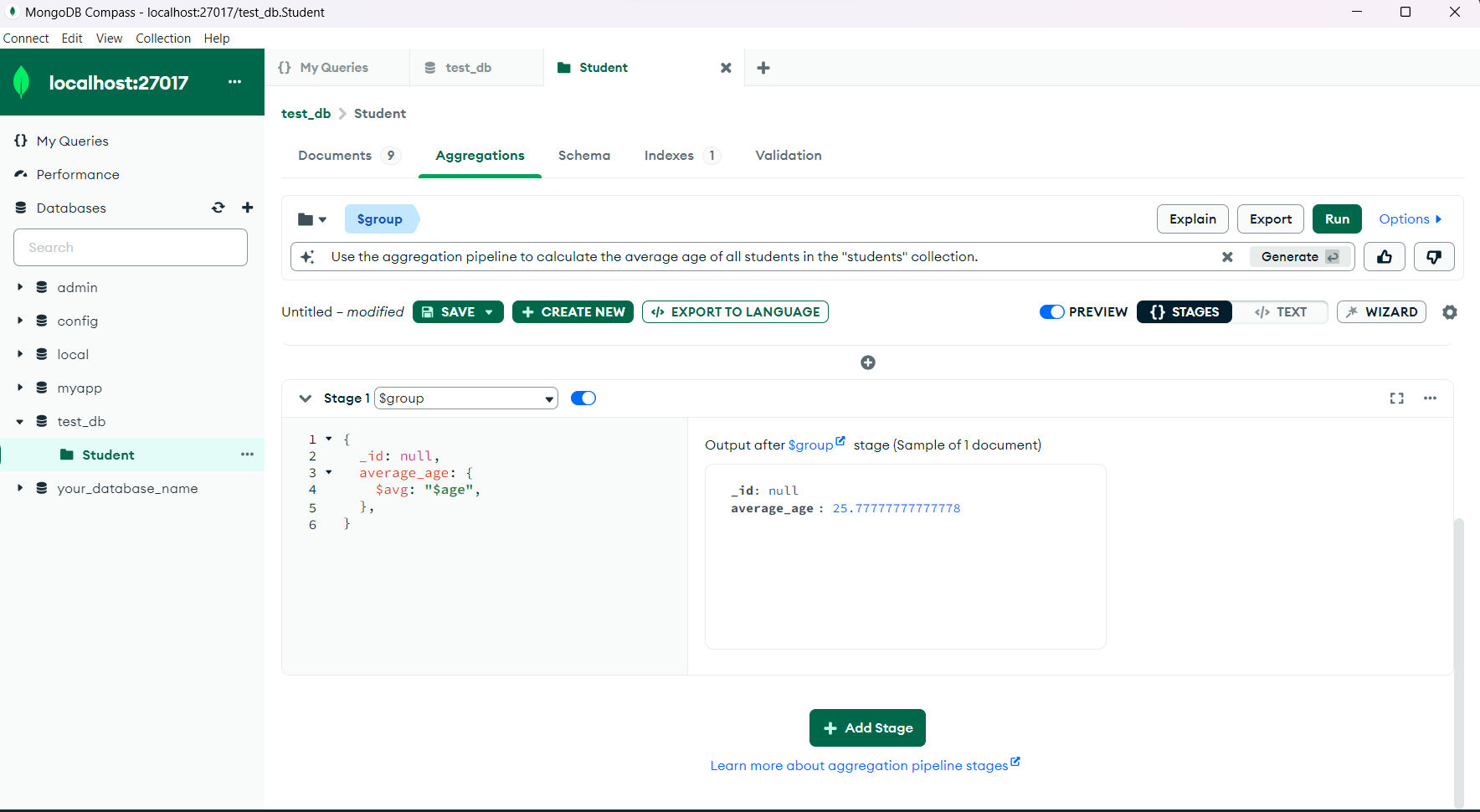
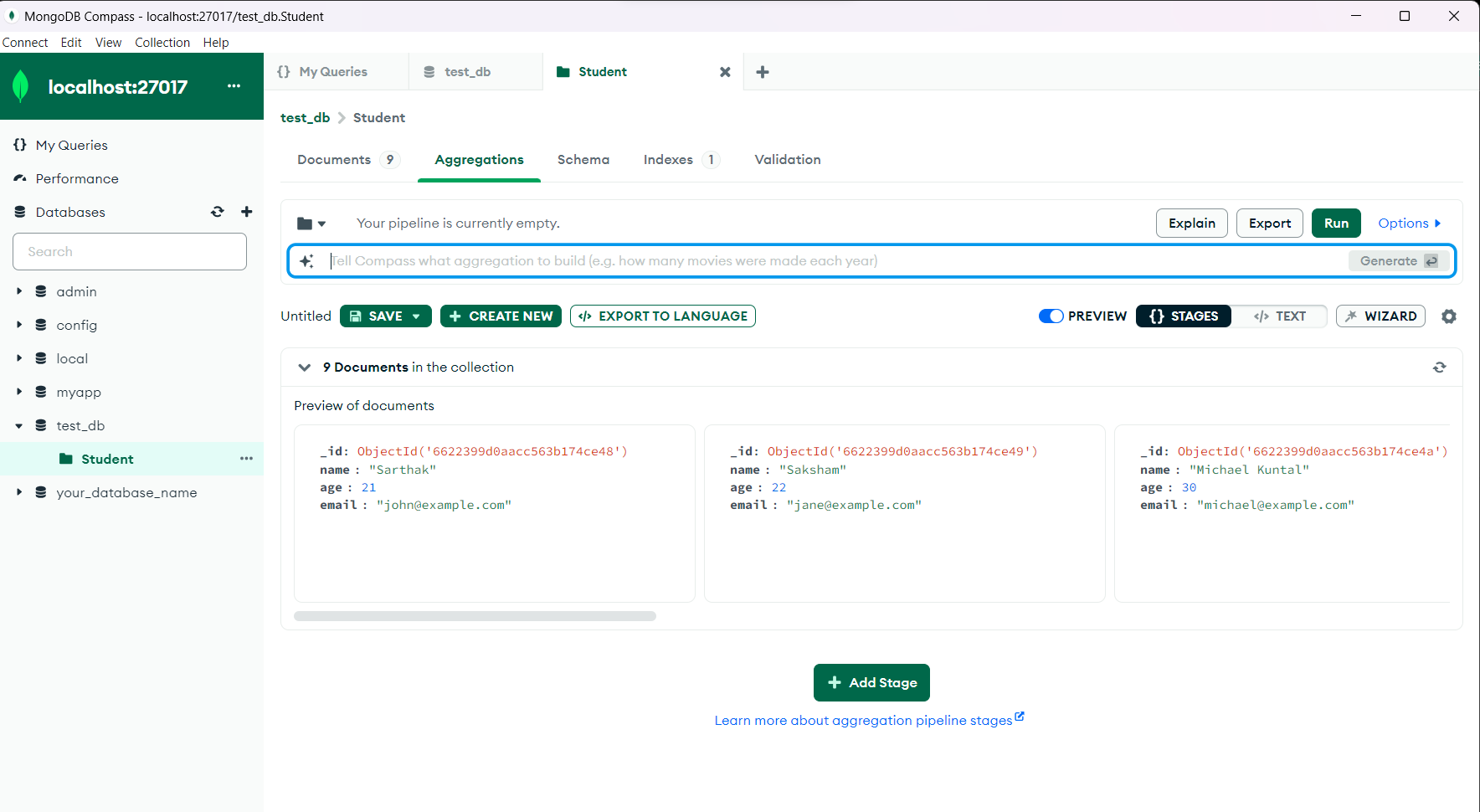
Q6



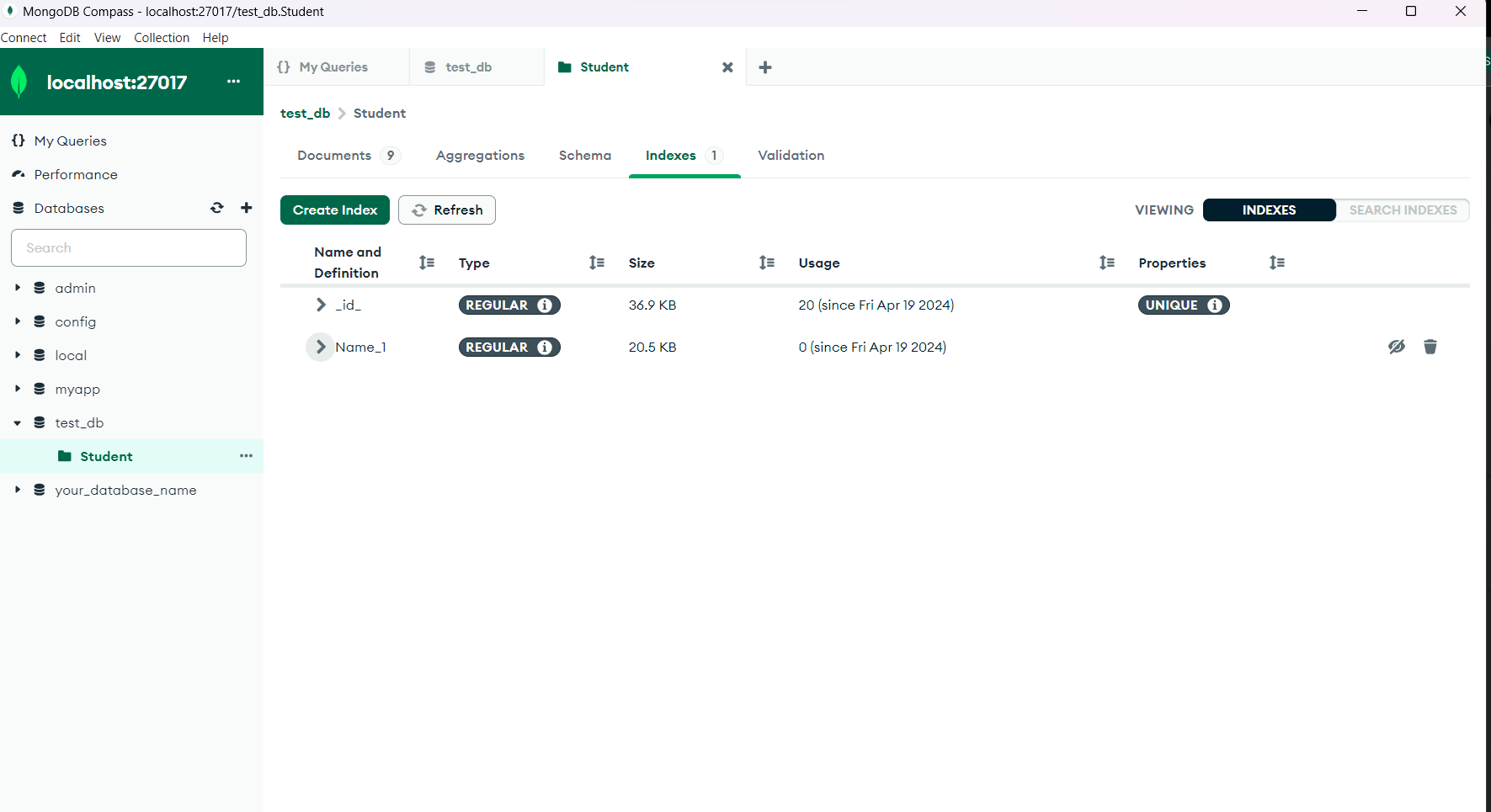
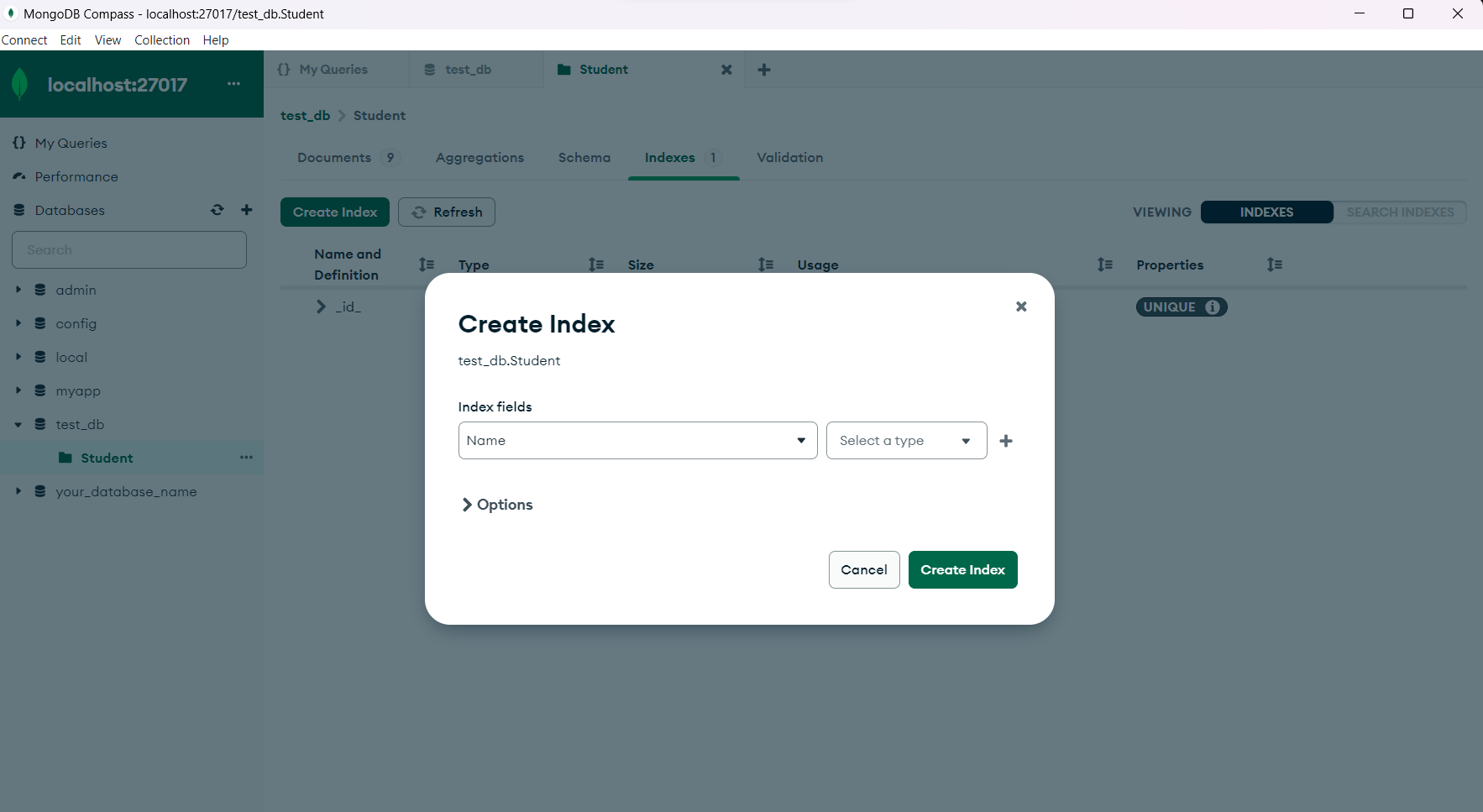
Q7



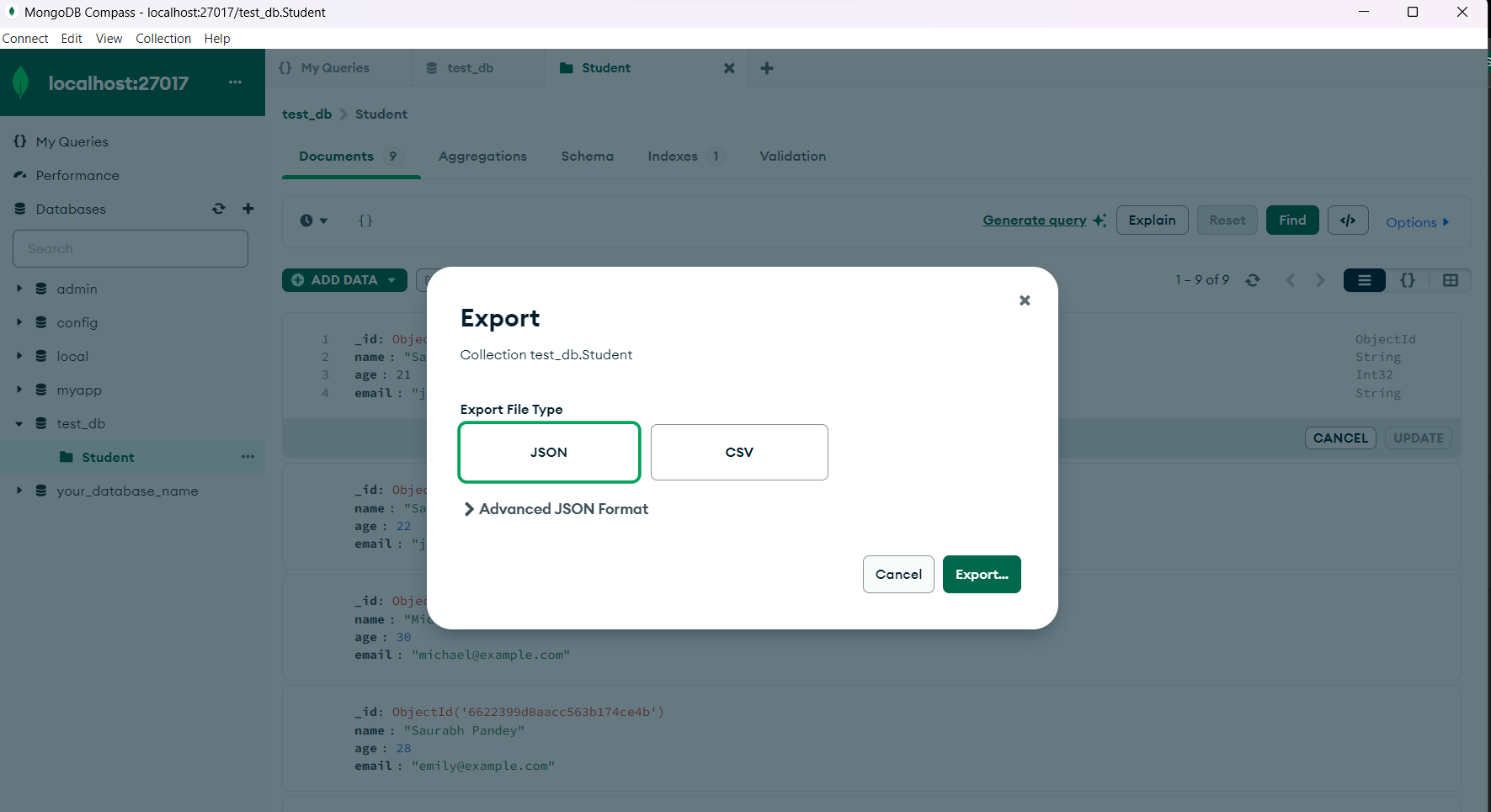
Question 8



Q9



Q10



Question 11

