Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

Выполнили:

cтуденты группы 16ВВ2  
Боровов М.С.  
Белов В.А.

Приняли:

к.т.н. Дубравин А.В.  
Кармышева Н.С.

## Отчёт

## по дисциплине «Web-программирование»

## на тему «Реализация REST API»

Лабораторная работа № 6

Пенза 2020

**Цель работы**

Реализовать REST API.

**Задание**

Необходимо доработать лабораторную работу №5 реализовав в ней поддержку REST API и реализовать два обработчика:

1. Получает запрос с названием животного и дополнительными параметрами и подбирает по этому запросу подходящее изображение из числа имеющихся;

2. По имени и параметрам изображения возвращает картинку.

|  |  |
| --- | --- |
| **Вариант** | **Животные** |
| 4 | Гиена, газель, геккон |

**Листинг**

**Form.html**

<html>

    <head>

       <meta http-equiv="Content-Type"content="text/html;charset=UTF-8"/>

    </head>

    <body>

        <b>Какое животное вы ищете?</b>

        <br>

        <br>

        <a>Имя</a>

        <br>

        <input id="name" rows="1" cols="10"></input>

        <p></p>

        <a>Травоядный или хищник</a>

        <br>

        <input id="food" rows="1" cols="10"></input>

        <p></p>

        <a>Класс (млекопитающее или рептилия)</a>

        <br>

        <input id="class" rows="1" cols="10"></input>

        <p></p>

        <a>ID</a>

        <br>

        <input id="id" rows="1" cols="10"></input>

        <p></p>

        <a>Номер</a>

        <br>

        <input id="number" rows="1" cols="10"></input>

        <p></p>

        <inputtype="submit"value="Найти изображение"onclick="findImg()"/>

    </body>

    <script>

        async function findImg() {

            var imgname = null;

            var imgfood = null;

            var imgclass = null;

            var imgid = null;

            var imgnumber = null;

            var imgimage = null;

            imgname = document.getElementById("name").value;

            imgfood = document.getElementById("food").value;

            imgclass = document.getElementById("class").value;

            imgid = document.getElementById("id").value;

            imgnumber = document.getElementById("number").value;

            imgimage = String(imgname) + String(imgnumber) + ".jpg";

            let beastsJSON = {

                name: imgname,

                food: imgfood,

                class: imgclass,

                id: imgid,

                image: imgimage,

                value: imgnumber

            };

            let responce = await fetch('./api/beast',{

                method: 'POST',

                headers :{

                    'Content-Type': 'application/json'

                },

                body: JSON.stringify(beastsJSON)

            });

            const json = await responce.json();

            console.log(json);

            // Ошибка

            if (json.error != null)

            {

                console.log(json.error);

                var pjson = JSON.stringify(json);

                document.write(pjson);

            }

            else

            {

                document.write('<img id="reqImg" src ="">');

                console.log(json.image);

                document.getElementById("reqImg").src="img/"+json.image;

                delete json.\_id;

                delete json.\_\_v;

                var pjson = JSON.stringify(json);

                document.write(pjson);

            }

        }

    </script>

</html>

**index.js**

const express = require("express");

const mongoose = require("mongoose");

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

const app = express();

mongoose.connect("mongodb://localhost/beast-db",

{

    useNewUrlParser: true,

    useUnifiedTopology: true

});

app.use('/img', express.static('img'));

app.use(bodyParser.json());

app.use("/api",require("./api"));

app.listen(8888, ()=>{

    console.log("Server started");

});

**api.js**

const express = require("express");

var fs = require("fs");

const router = express.Router();

const Beast = require("./beast");

let jsonbody = null;

router.get("", (req, res) => {

    var page = fs.readFileSync("Form.html");

    res.writeHead(200, {"Content-Type": "text/html"});

    res.write(page);

    res.end();

})

router.get("/beast", (req, res)=>{

    console.log(jsonbody);

    res.send(jsonbody);

});

router.post("/beast", (req, res)=>{

    console.log(req.body);

    var name = req.body.name;

    var food = req.body.food;

    var bclass = req.body.class;

    var id = req.body.id;

    var number = req.body.number;

    let finderror = null;

    if (food == "") {

        food = "herbivorous";

    }

    if (bclass == ""){

        bclass = "mammal";

    }

    if (id == ""){

        id = "1";

    }

    Beast.findOne({name: name, number: number})

    .then(beast => {

        if (beast != null)

        {

            Beast.findOne({food: food, class: bclass})

            .then(beast => {

                if(beast != null)

                {

                    Beast.findOne({id: id})

                    .then(beast => {

                        if(beast != null)

                        {

                            jsonbody = beast;

                            res.redirect("./beast");

                        }

                        else

                        {

                            finderror = {error: "ID Error", name: name, number: number };

                            res.send(finderror);

                        }

                    });

                }

                else

                {

                    finderror = {error: "Food or class error", name: name, number: number };

                    res.send(finderror);

                }

            });

        }

        else

        {

            finderror = {error: "Name Error", name: name, number: number };

            res.send(finderror);

        }

    });

});

module.exports = router;

**beast.js**

const mongoose = require("mongoose");

const Schema = mongoose.Schema;

const beastSchema = new Schema({

    name: String,

    food: String,

    class: String,

    ID: String,

    image: String,

    value: Number

});

const beast = mongoose.model("beast",beastSchema);

module.exports = beast;

**Результат работы**

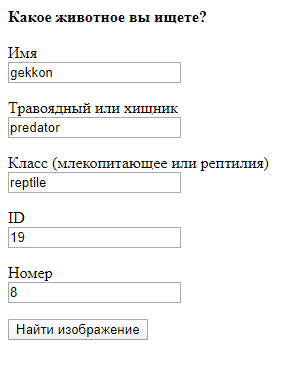


Рисунок 1 – API

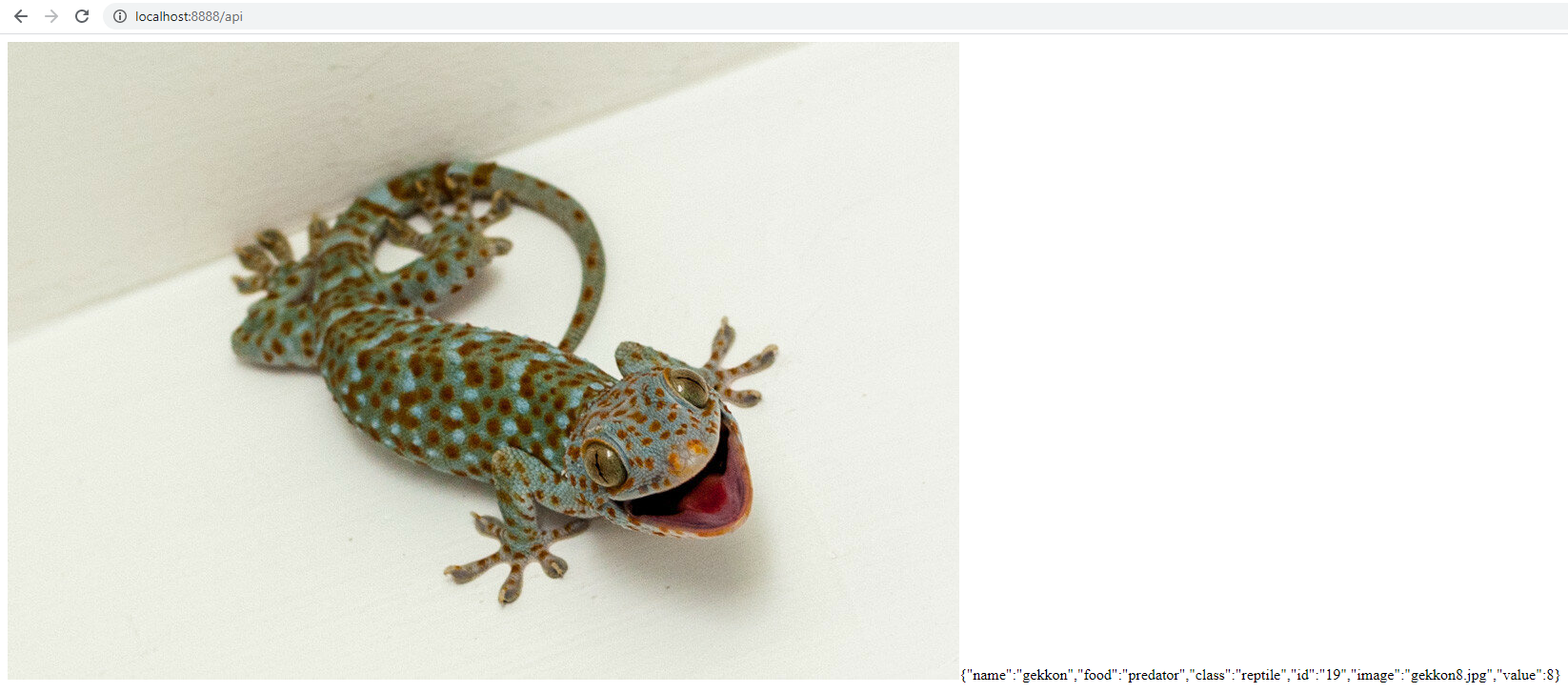


Рисунок 2 - Результат работы – удачный поиск

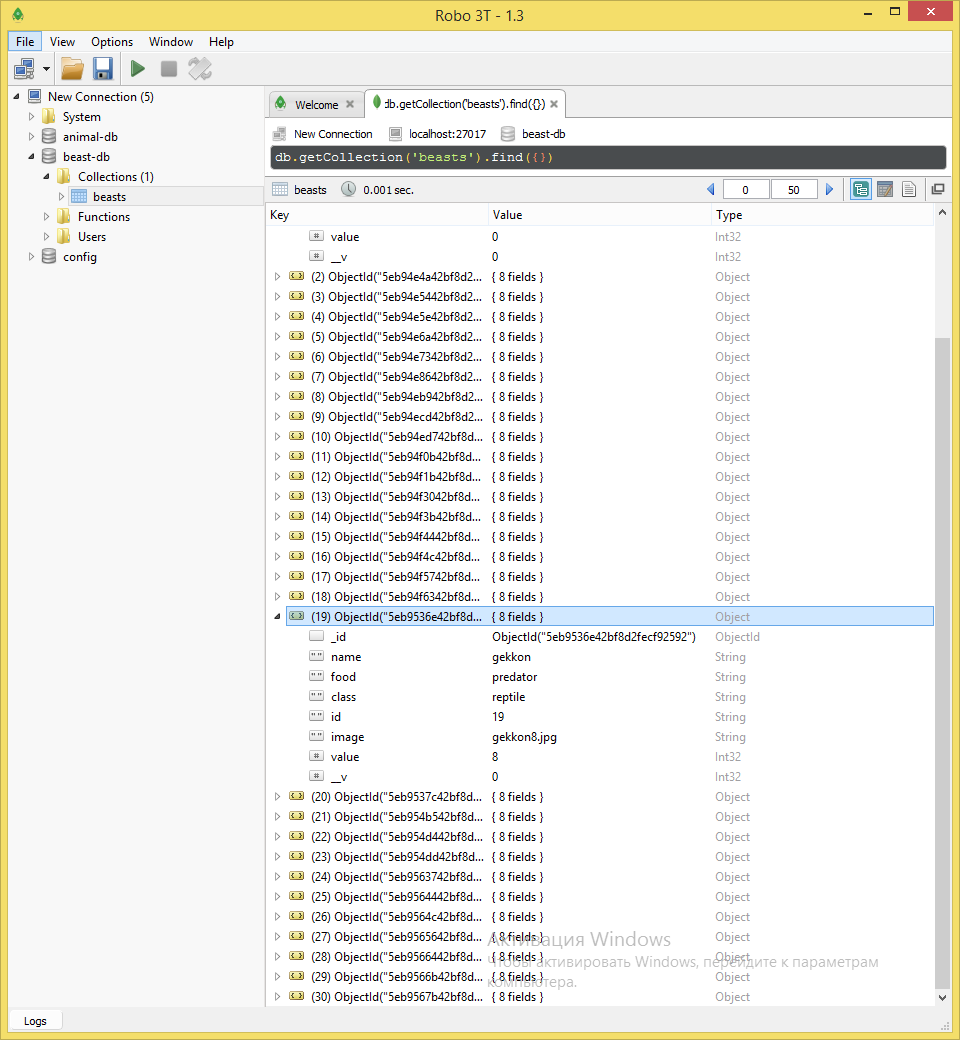


Рисунок 3 – Поля искомой записи в базе данных

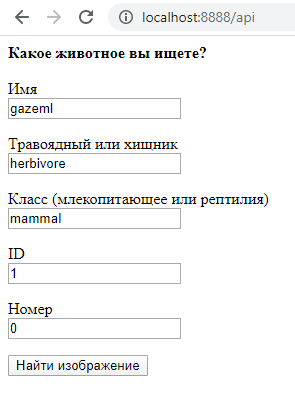


Рисунок 4 - ввод данных с ошибкой

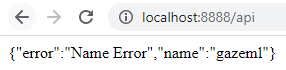


Рисунок 5 - результат работы

**Вывод**

В ходе выполнения лабораторной работы научились реализовывать REST API.