Exercises: Spring Boot Introduction

Problems for exercises and homework for the “Java Web Development Basics” course @ SoftUni. Submit your solutions on the **course page** of the **current instance**.

Let’s exercise a little the knowledge we’ve accumulated during the lesson for Spring Boot. As you have seen only the basics, and theory, the problems won’t be that hard. But they will definitely be fun and productive.

We will implement a simple application, only with Controllers and Spring data, exploring the most interesting exploits of the Spring Platform.

Real Estate Agency

Real Estate Agency is an application in which you **registers** and **finds** a real estate offers based on a given criteria.

You will have to create a simple MVC application which has several pages and 1 object entity.

You will be given all resources like **styles, images** and **html** files.

1. Data

This is the data layer of the application. There is 1 data object for you to implement.

Offer

Create a **`Offer`** class, which holds the following properties:

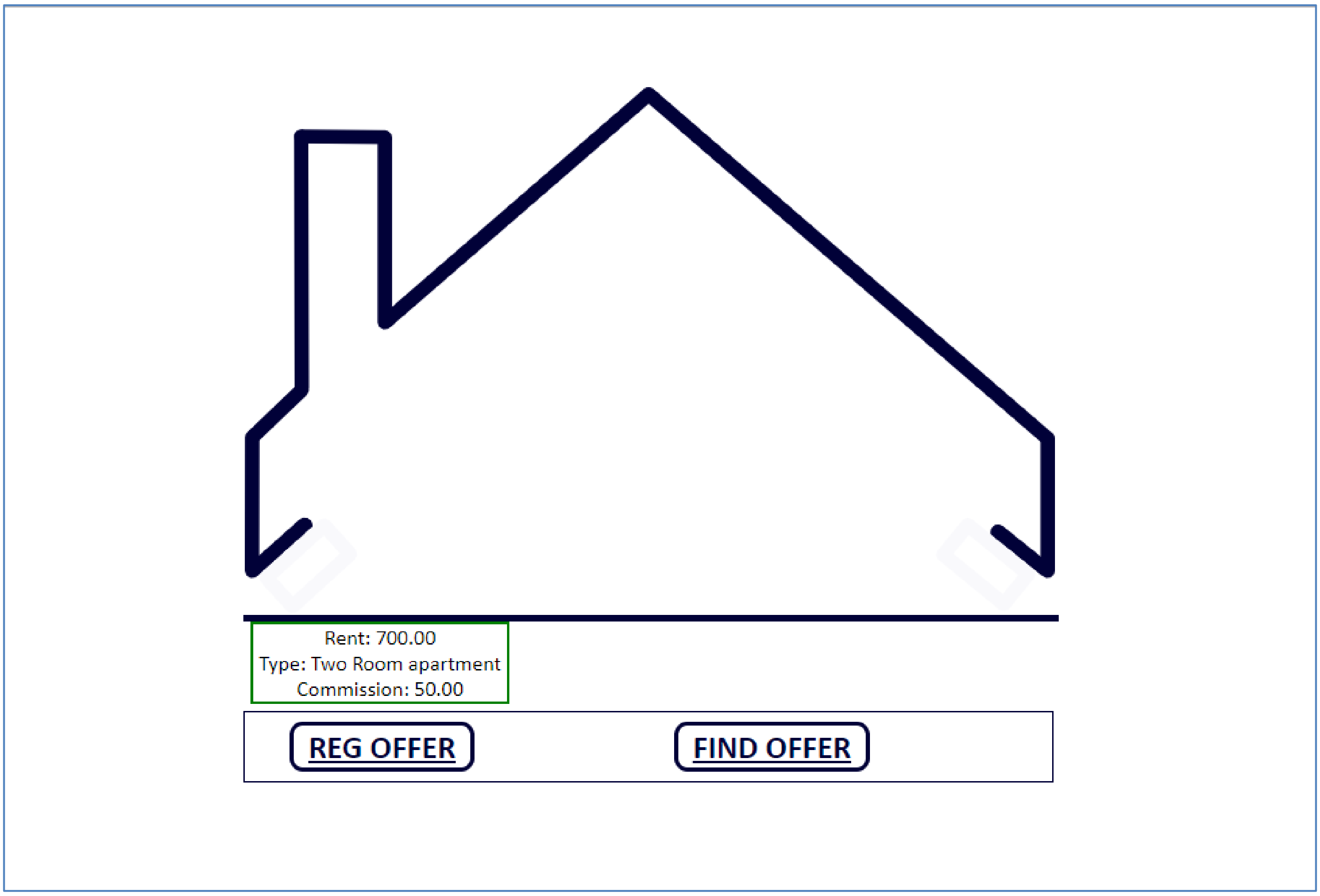
- `id` – a **String**.
- `apartmentRent` – a **BigDecimal**.
- `apartmentType` – a **String**.
- `agencyCommission` – a **BigDecimal**.

2. Home Controller

Implement a **controller**, which listens on the **index route** (“**/**”).

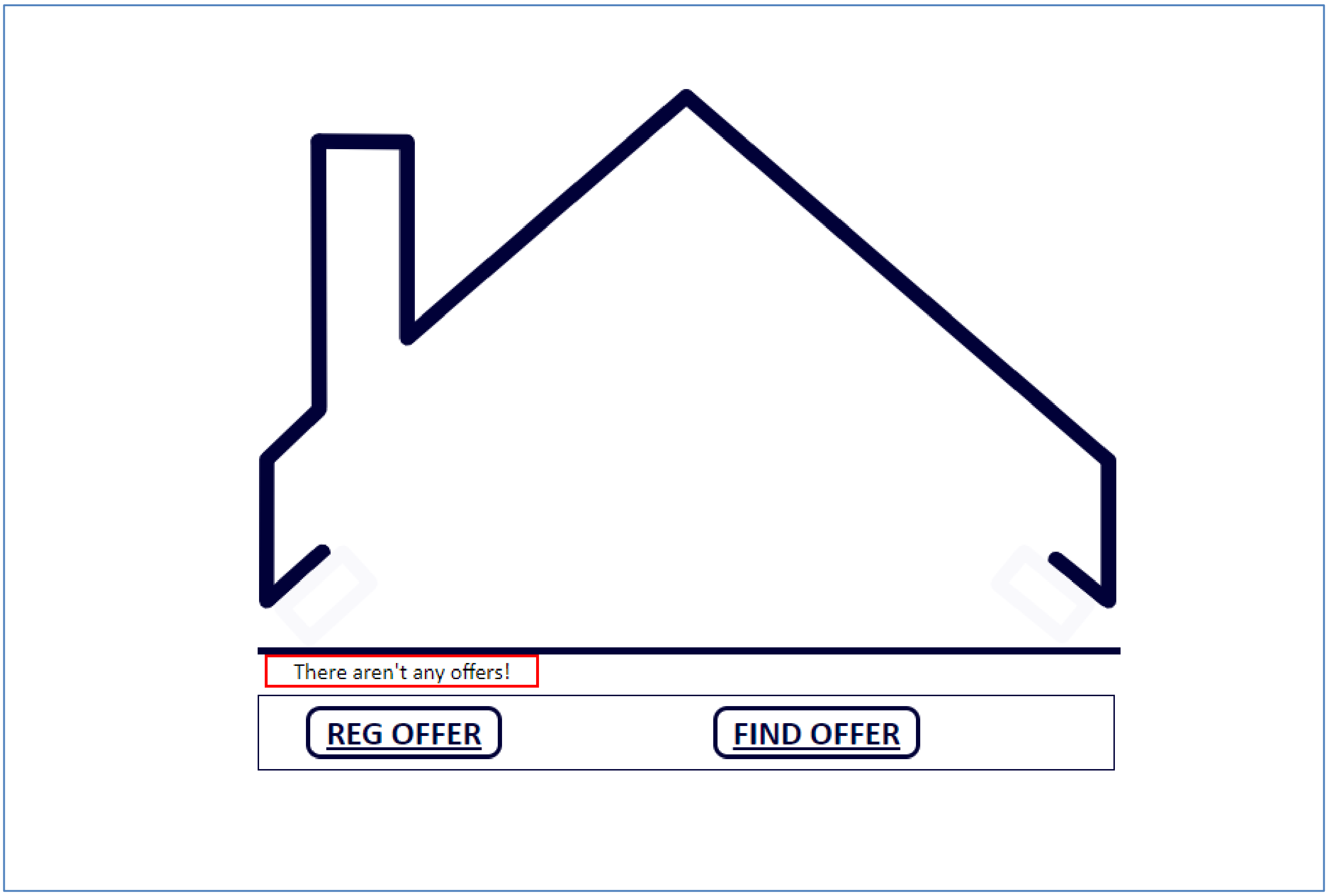
It should support only a **GET** request.

It should return the following HTML page, upon a **GET** request.



Rent: 700.00
Type: Two Room apartment
Commission: 50.00

On the home page you must list all **available** offers. If there aren't any a message must be displayed.



There aren't any offers!

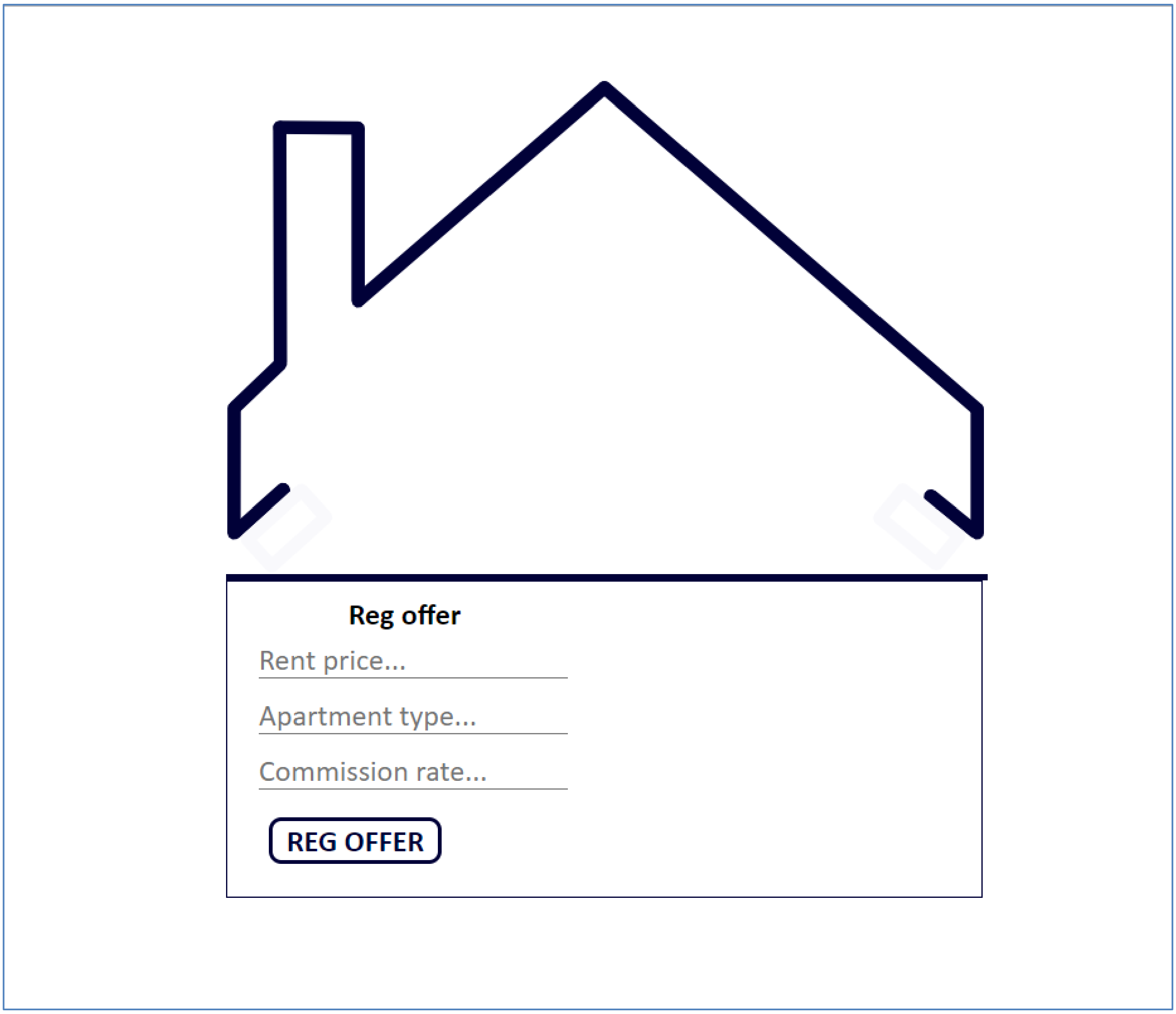
3. Offer Controller

Implement a **controller**, which listens on the following **routes:**

Register (“/register”)

It should support **GET** & **POST** requests.

It should return the following HTML page, upon a **GET** request.



Rent price and commission rate should be **numbers**. Rent price must be **positive number** (greater than zero (0)). Commission rate should be greater or equal to zero (0) and lower or equal to one hundred (100). Apartment type should be **non-empty string**.

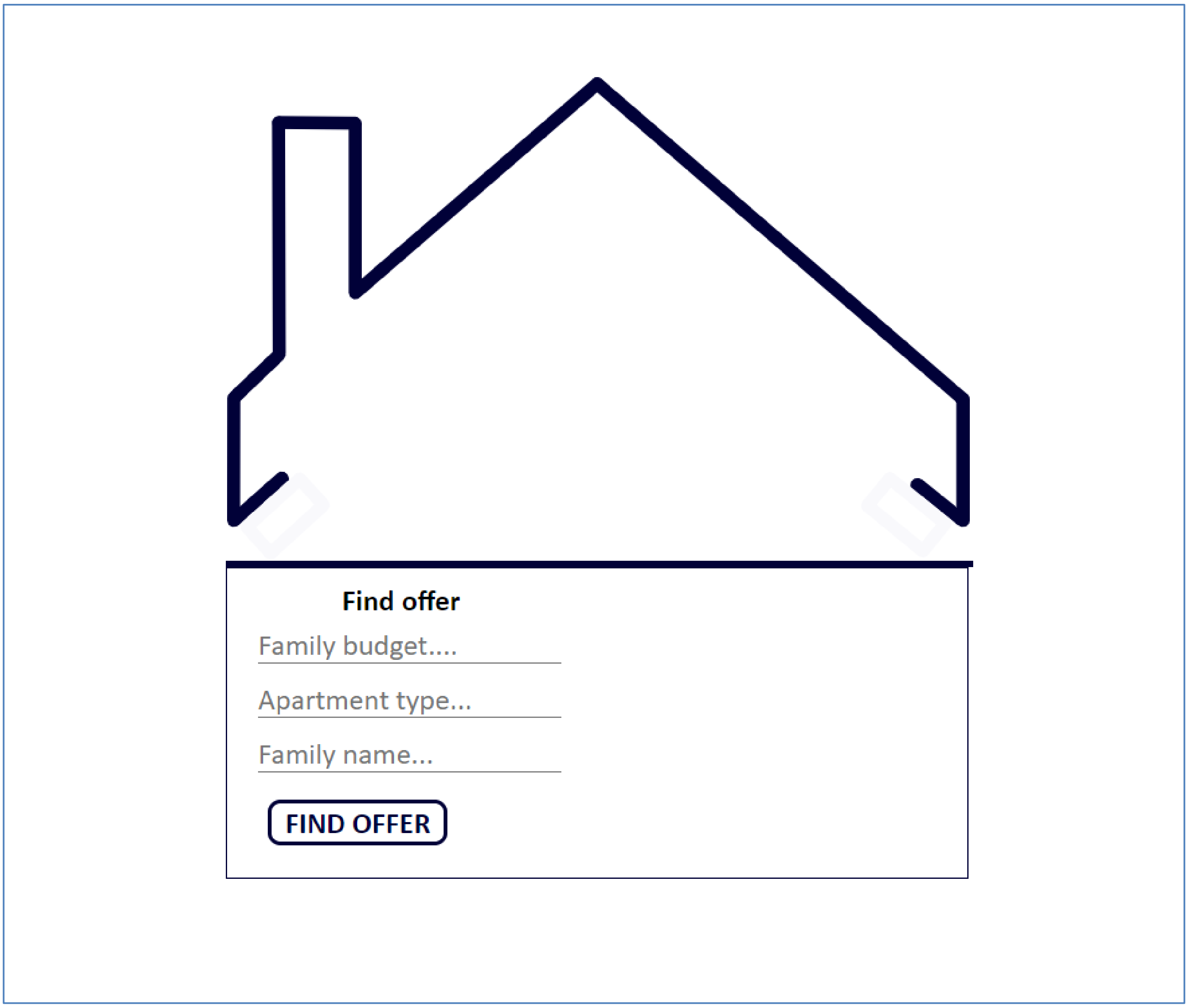
If some of these conditions or all of them **are not met**, you should redirect to register page.

If **all is good** you should 'post' the new offer on the agency and redirect to home page.

Find ("/find")

It should support **GET** & **POST** requests.

It should return the following HTML page, upon a **GET** request.



Find offer

Family budget....

Apartment type...

Family name...

FIND OFFER

Family budget must be **positive number (greater than zero (0)).** Apartment type and Family name should be **non-empty strings**.

If some of these conditions or all of them **are not met**, you should redirect to find page.

If **all is good** you should start searching through all registered offers into the agency. The first offer that meets the requirements, namely:

- The type of apartment you are looking for is **absolutely the same** as the registered one.

- The family budget needs to be **enough** to take the current apartment: rent + commission in percent of the current rental of the apartment.

Is leased! That means you should remove the offer from the database and you should redirect to home page.