Exercises - Session 26

In case you get stuck anywhere, don't be afraid to ask the coaches! They are here to help and will gladly explain everything to you! Take notes during the exercises. Even if you never look at them again, they will help you memorise things!

Pictures, layouts and in-app links

We will build a small web page for our 3 fictional roommates: Shirley, Ines and Peter. They own a house outside of Zurich which was most probably built by an English architect.

In this exercise, the skeleton of the app is already built, but there are pieces missing! First, get the app from: https://github.com/rubymonstas-zurich/party-guests/tree/session25

1. We now have a layout which is reused for every page we build, so we don't have have to write all the HTML for every sub-page again. If you look at the screenshot on the right, you'll see that the blue part is where we put the content of the rendered page. The white part outside will always be rendered and comes from `/views/layout.erb`. Take a look at that file first before you continue. In there, you find a method call to `yield`, which is where our content is rendered.

   Welcome to our webpage!

   - People in the flat
     - Ines
     - Shirley
     - Peter
   - Our house

   Welcome to our home page!

   We are Shirley, Ines and Peter!

   (C) 2015 RubyMonstas Productions Ltd.

   a. As a first step, add a point to the navigation that says "Our house" and links to "/house". Make sure that you have that route in your app.rb and that it renders the template `house.erb`.

   b. In the `house.erb` template, tell our audience in your own words that this is the house they live in. Make sure to also show a picture of the house (`/images/house.jpg`)

2. We also want to show all the people that live in the house. There's already a file with all the profiles: `profiles.csv`. In there, you'll find rows for Shirley, Ines and Peter.

   a. Now, take a look at `ProfileRepository` and its two methods `all_profiles` and `find_profile`. What happens in these methods? What are the input types and the output types?

   b. Start up your irb and check your answers:

   ```
   $ irb
   irb(main):001:0> require './profile_repository''
   => true
   irb(main):002:0>
   ```

   Check what these methods do. Were you right?

   c. `Profile` is actually not implemented. Go ahead and implement it according to what `ProfileRepository` expects. Now check your implementation again with irb.

d. Now go ahead with the actual profile page. We want to link to the 3 profile pages according to their nickname, so we link to:
   i. /profiles/ines
   ii. /profiles/shirley
   iii. /profiles/peter

   You will find a route for this in `app.rb` already. Make sure that you load the correct profile according to the nickname that is passed to you and render the template `profile.erb`.

e. We're almost done now. The pages load correctly, but the image is too big! Make sure to set the width to 200 pixels so it looks nicer.

Optional

1. Each profile shows the list of friends, but just as text. Make sure to link to the other profiles.

2. Ines moves out and her twin sister (phew, you can keep the image, noone will notice!) Julie moves in. Make sure that this is corrected on the web page. By the way, Julie likes gardening and hiking and apparently has no friends. How does your web page code react to profiles without friends?

3. Spend some time improving your web page design. It sure lacks colors! Take some inspiration from http://www.wonder-tonic.com/geocitiesizer/ and see how great www.rubymonstas.ch could have looked like if we had this course in 1996!