Exercises - Session 20

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!

<u>Forms with Sinatra</u>

As you might have already realized last time, the last few exercises involved more of your own thinking and researching skills. Since using new gems and frameworks requires anyone to read a bit of documentation and experimenting, we think it's necessary that you get into the habit of exploring gems by yourself. It's also more fun if you find things out by yourself, right? Let's get to the exercise!

1. Set up sinatra like you did last time and start with a simple welcoming page at the root path ("/"). Check on rubygems.org if sinatra version 1.4.6 is still the latest. If not, update it in your Gemfile.

2. You want to organize a party and now since you started with Sinatra why not build a nice little web page to hype it for the weeks to come? If people visit /party, they should see all the people that already signed up. For this, let's write a CSV file with guests that we know already signed up. Our columns are: Name, Number of friends they bring, Time they plan to arrive.

```
1    Marion, 3, 21:00
2    Ferdinand, 4, 20:00
3    Hana, 2, 19:30
4    Thomas, 3, 20:30
5    Mario, 77, 19:58
```

3. Alright, now let's display all those guests on our party page (see also the screenshot). We want to show their name, how many friends they plan to bring and when they plan to arrive.

4. Sweet, so now we have all the guests. But new guests should be able to join. Build a form where people can sign up with their name, the number of friends they plan to bring and when they'll arrive.
   Bonus points: find out how you can display a grey example text in the input fields.

5. Now that you've built the form you can submit it. Boom! It failed, and that's to be expected. Check the development tools network tab. You should find a POST request and within that, all your form data with their correct values. If you can find that, you're almost there.

6. You now built your form correctly, but your server does not yet accept the form data. Create a new resource for that in your app.rb with points to /party/guests, where your form will be POSTed to.
   In your new resource, make sure to print all the form data the console and verify that they are printed correctly to the console.

7. Now that we know the form data POSTed correctly to our server, make sure to create a new Guest and save it to the CSV file.
   As a last step in your resource, you can call redirect "/party" so people land on your party page again after they signed up.