Advanced Weather

We will continue upgrading the protocol, and at the same time take advantage of it by adding more client-side logic

Instructions

1. Rename server/client to `advancedweather_server.py` and `advancedweather_client.py`.
2. Make the server able to handle multiple requests per connection, until the client disconnects. 2.1. Again, keep it simple: Expect every message fits in its entirety into one recv()
3. Add a new client->server message of the format: `{"request": "get_cities"}`. 3.1. Server will respond to this with: `{"response": "success", "cities": ["London", "Paris", "New York", "Tokyo"]}`
4. Change the client: Instead of asking the command-line user to input a city name, ask for a minimum temperature. The client will proceed to display weather information for all cities that have their temperature over or as requested.
   Client output for example (notice user input is "22"):

```
Enter a minimum temperature: 22
== Weather information for Paris ==
Temperature: 22°C
Humidity: 55%
Description: Sunny
== Weather information for Tokyo ==
Temperature: 25°C
Humidity: 80%
Description: Rainy
```

To submit

Submit files `advancedweather_server.py` and `advancedweather_client.py`.