# Spring Data Retake Exam – 18 December 2022

# Job Finder

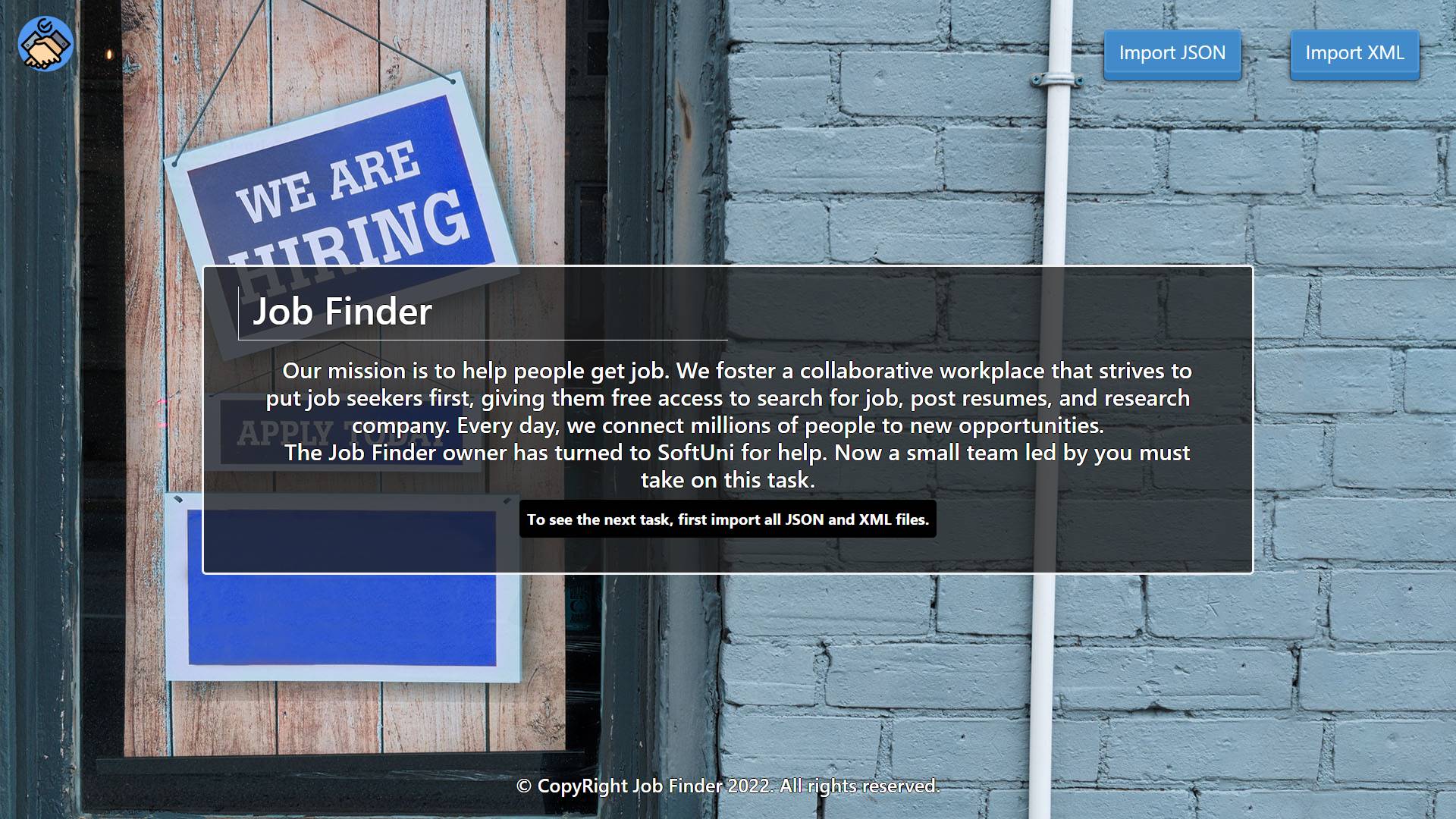
Our mission is to help people get job. We foster a collaborative workplace that strives to put job seekers first, giving them free access to search for job, post resumes, and research company. Every day, we connect millions of people to new opportunities. The Job Finder owner has turned to SoftUni for help. Now a small team led by you must take on this task.

## Functionality Overview

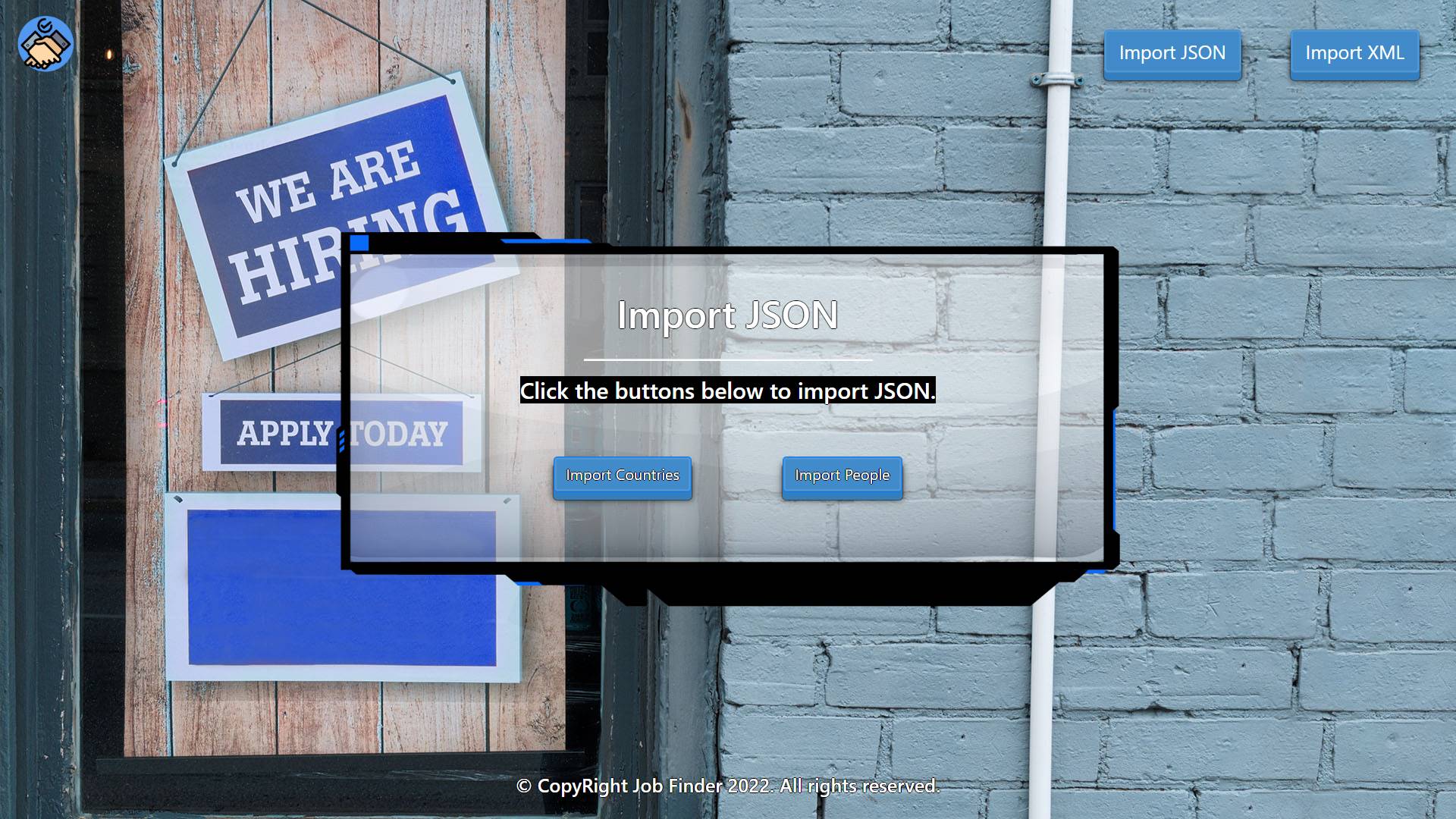
The application should be able to easily **import** hard-formatted data and **support functionalities** for also **exporting** the imported data. The application is called – **Job Finder**.

Look at the pictures below to see what must happen:

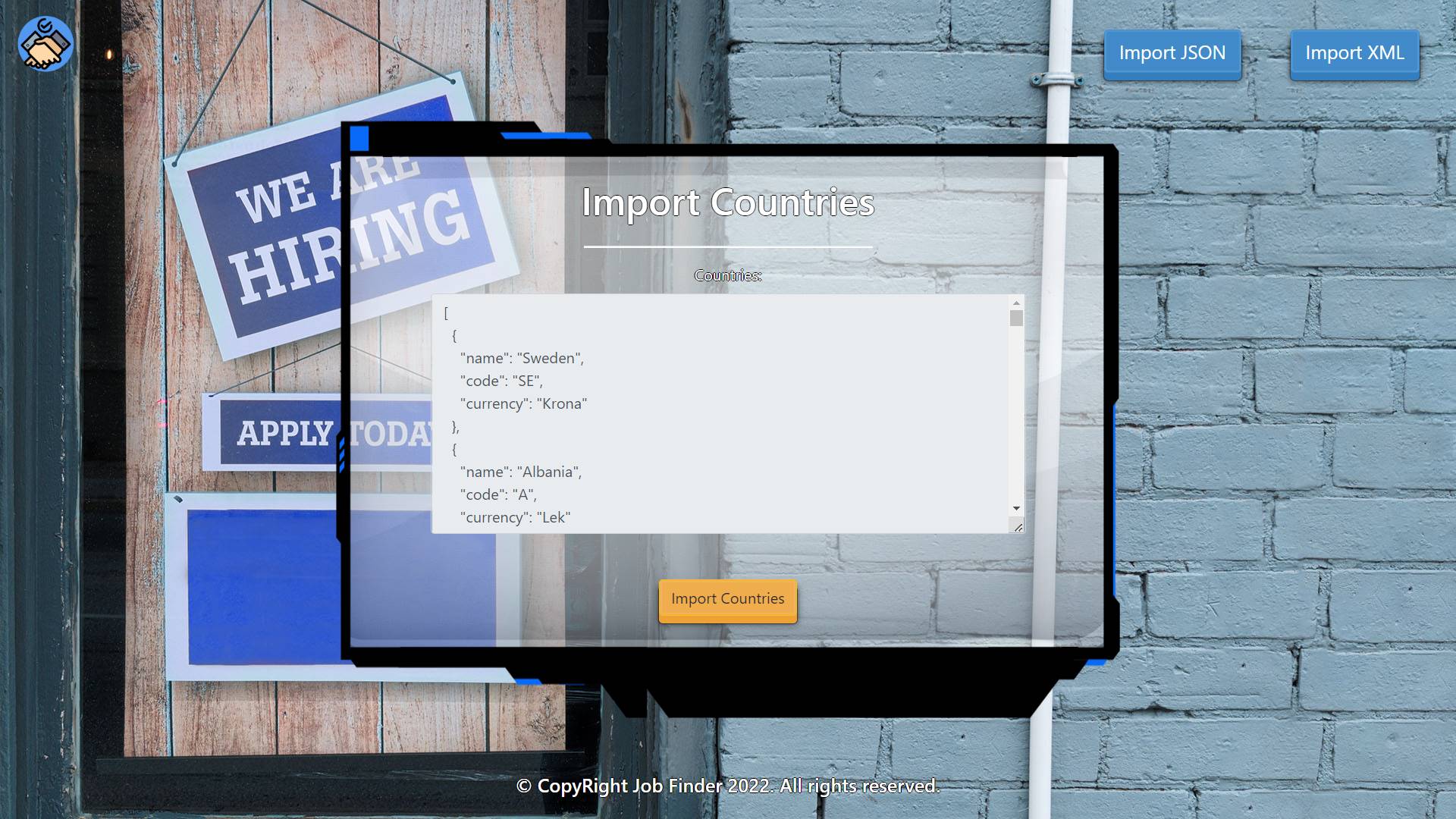
* The home page before importing anything:



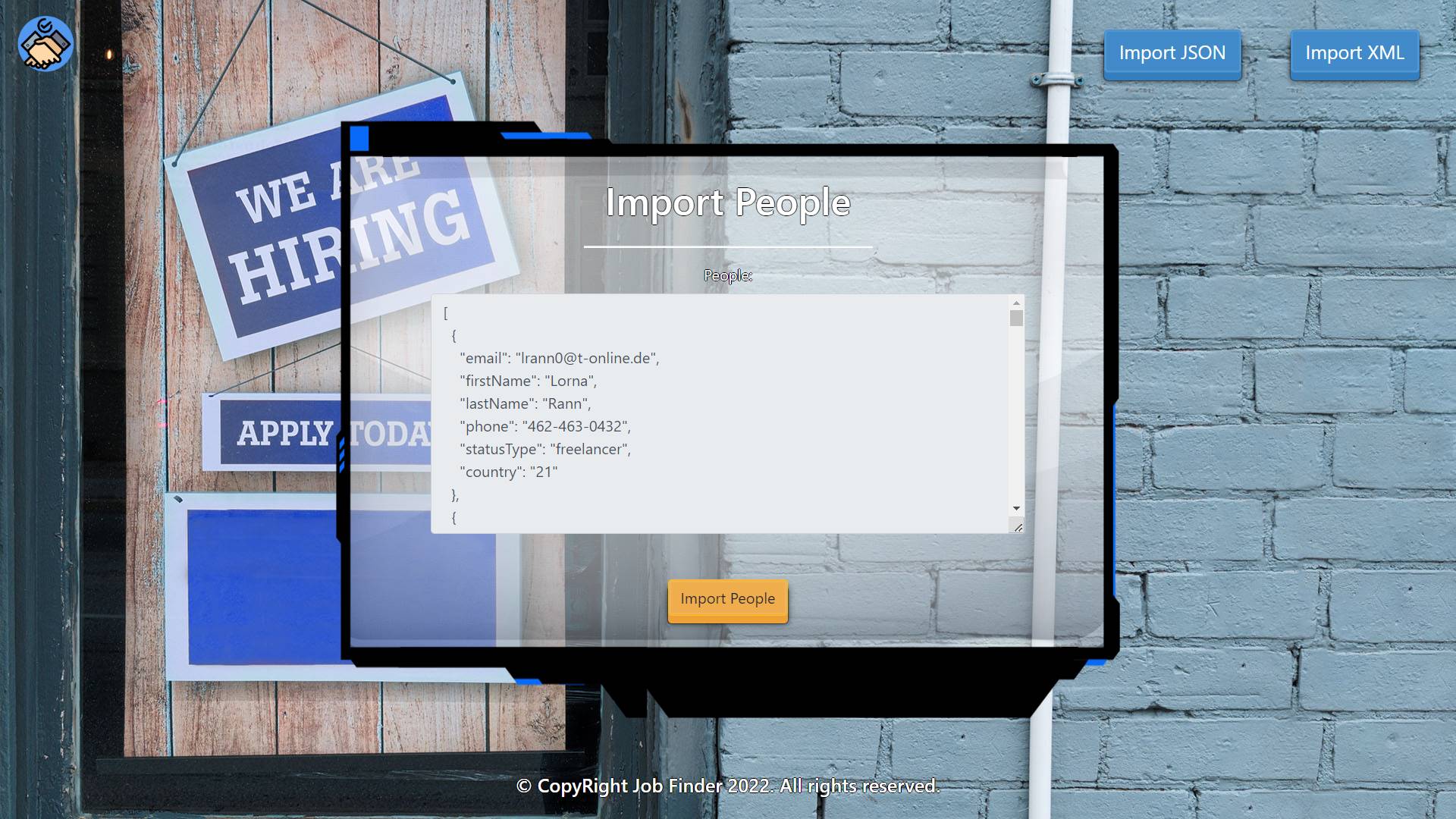
* The import JSON page before importing anything:



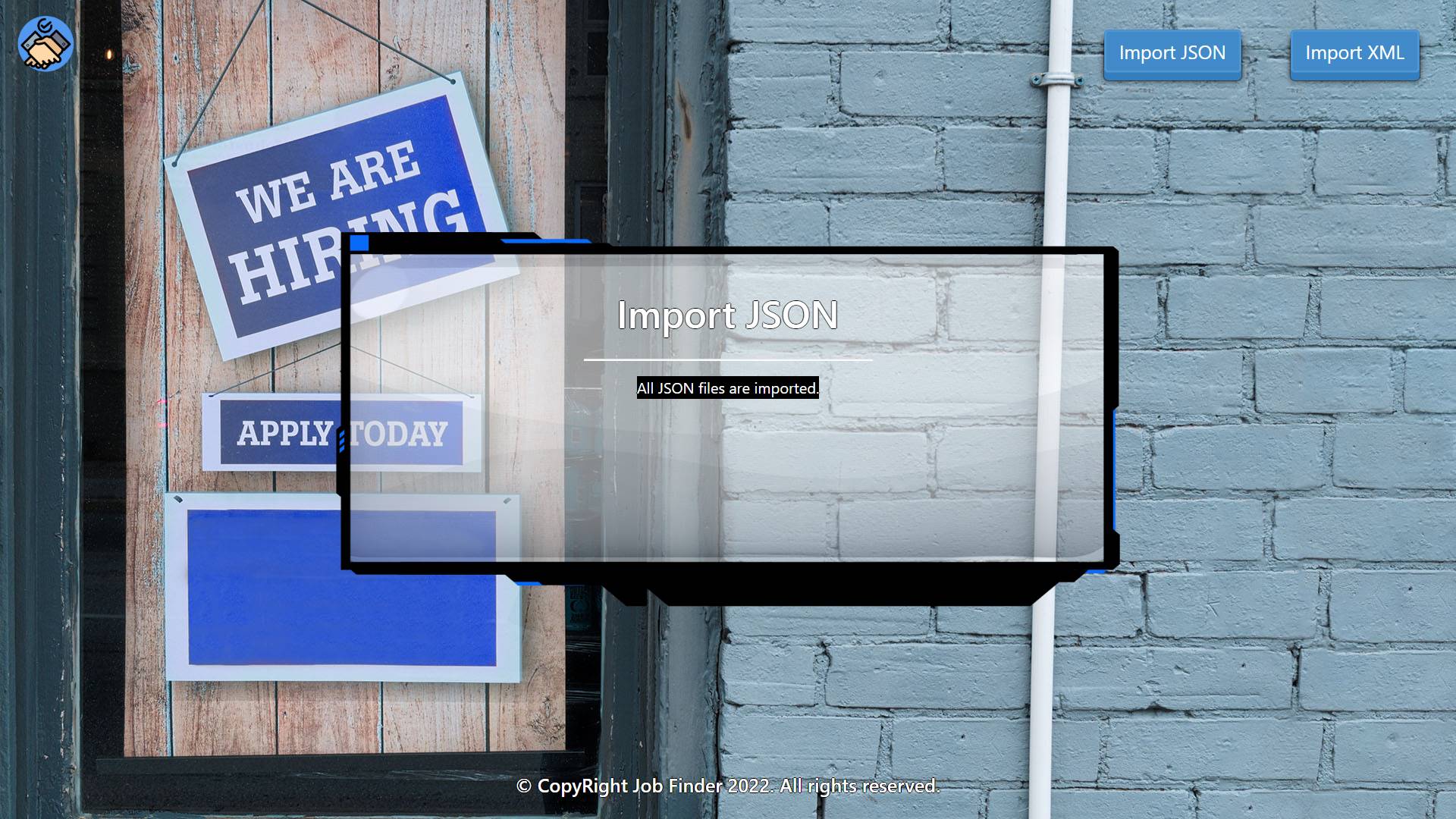
* Import the countries first:



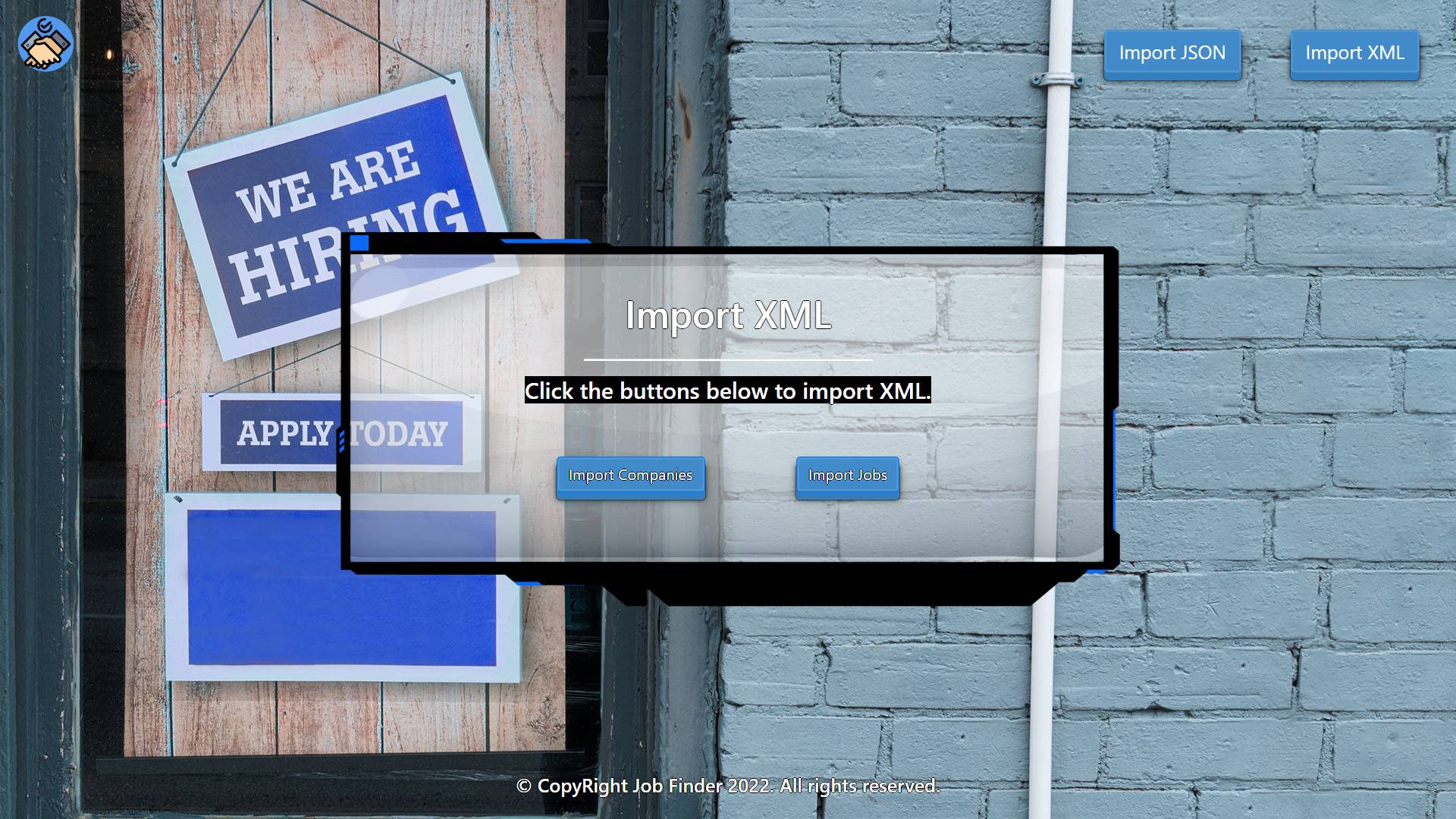
* Import the people second:



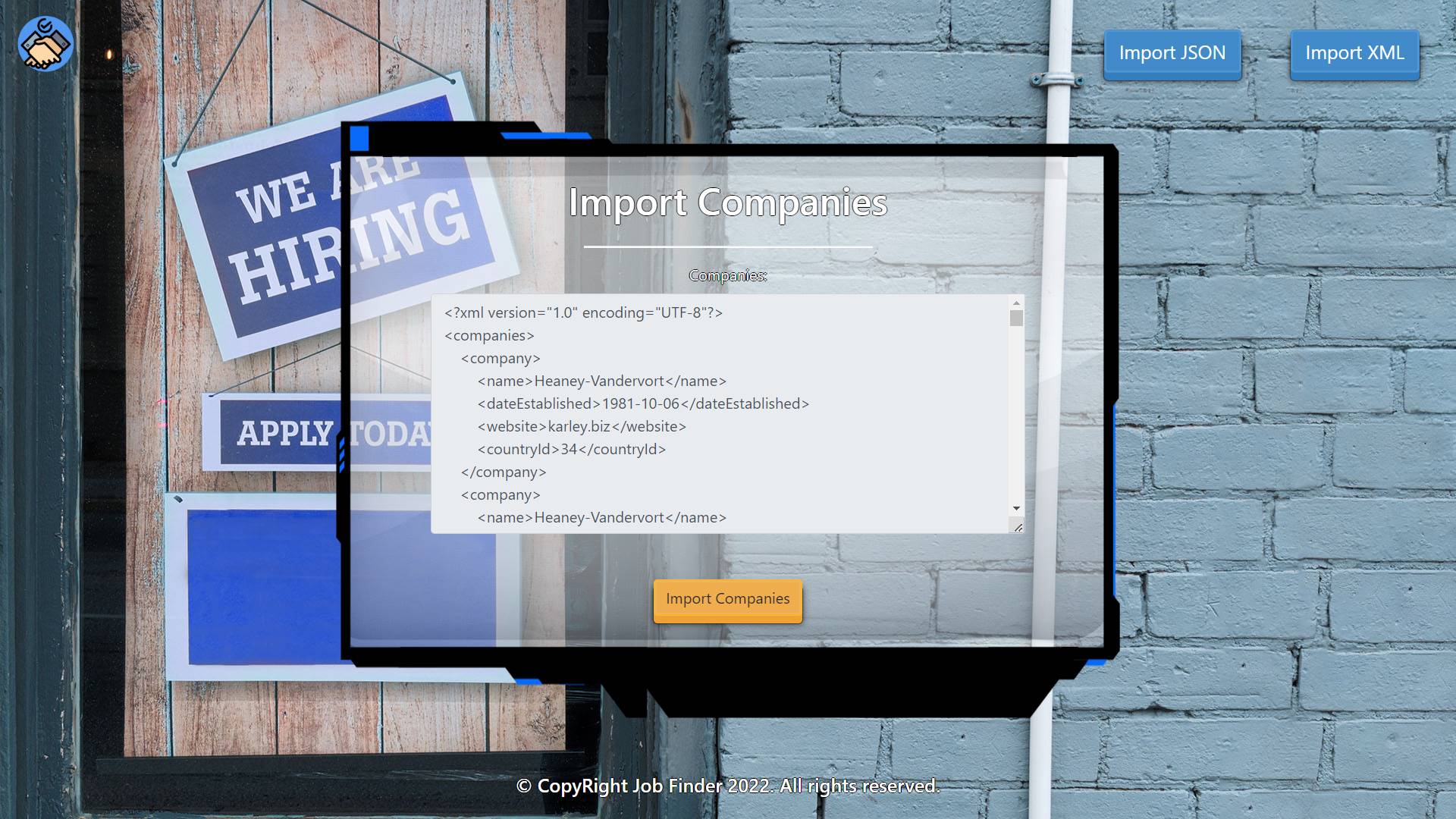
* The import JSON page after importing both files:



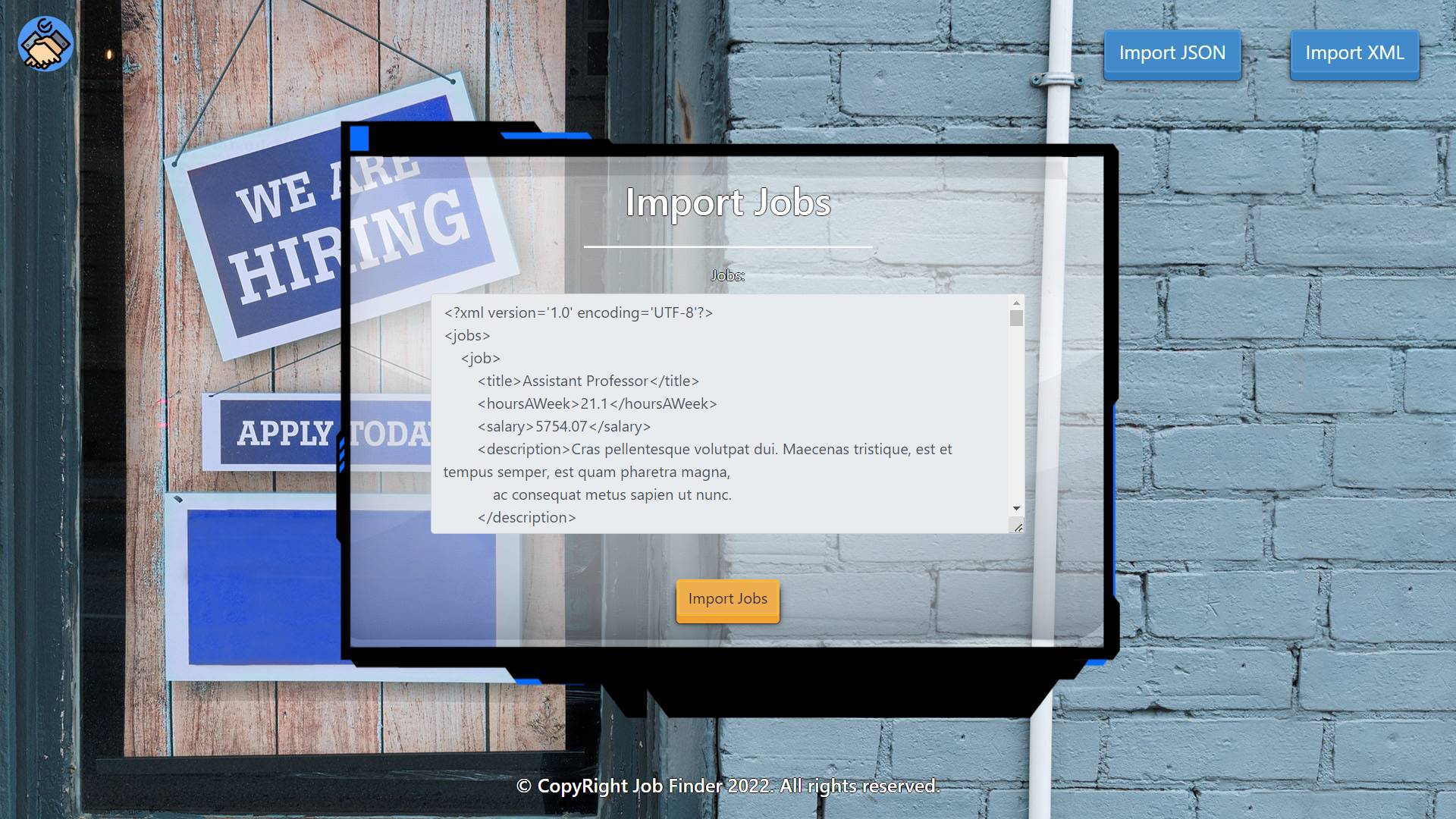
* The import XML page before importing the given data:



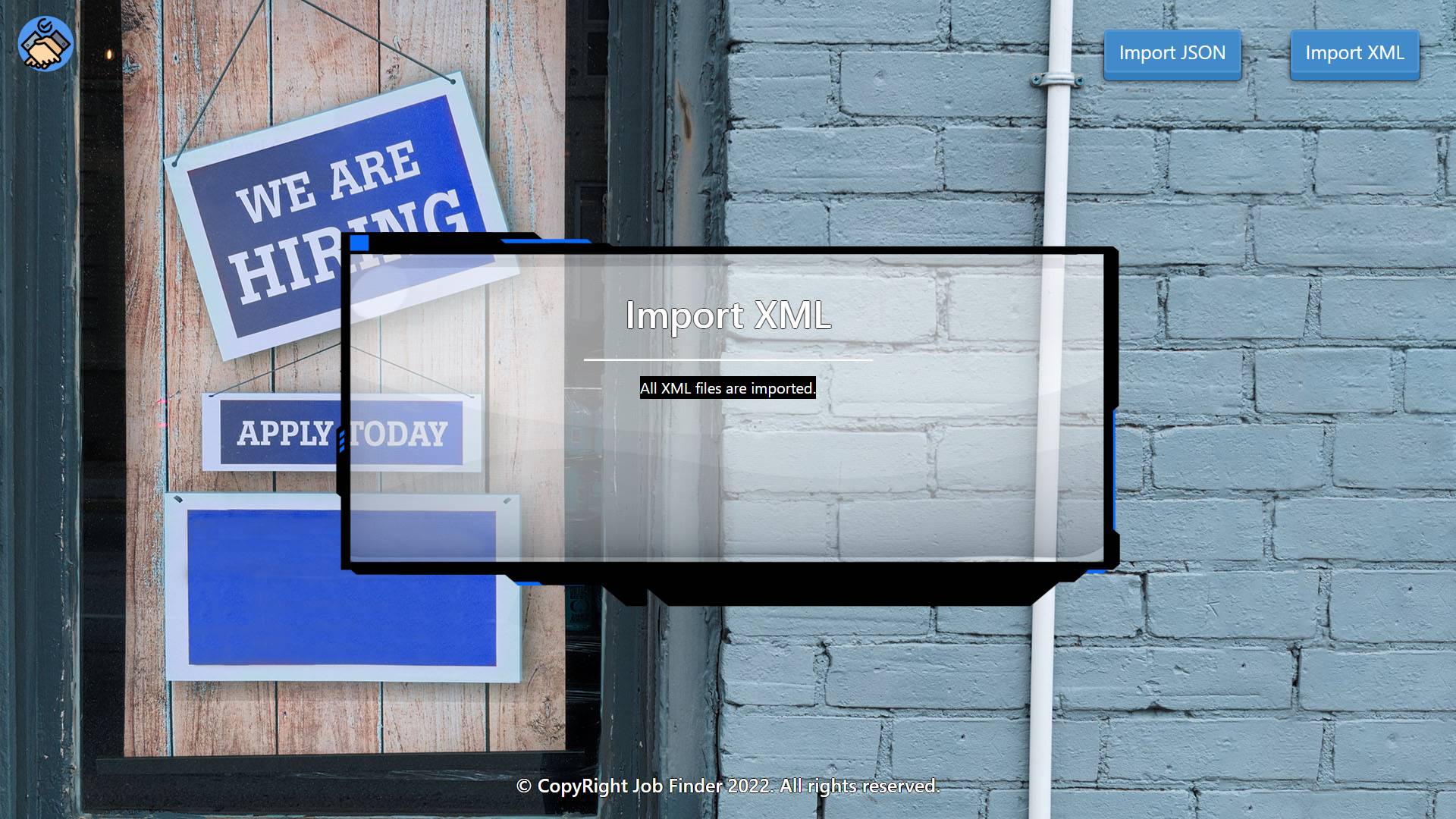
* Import the companies data:



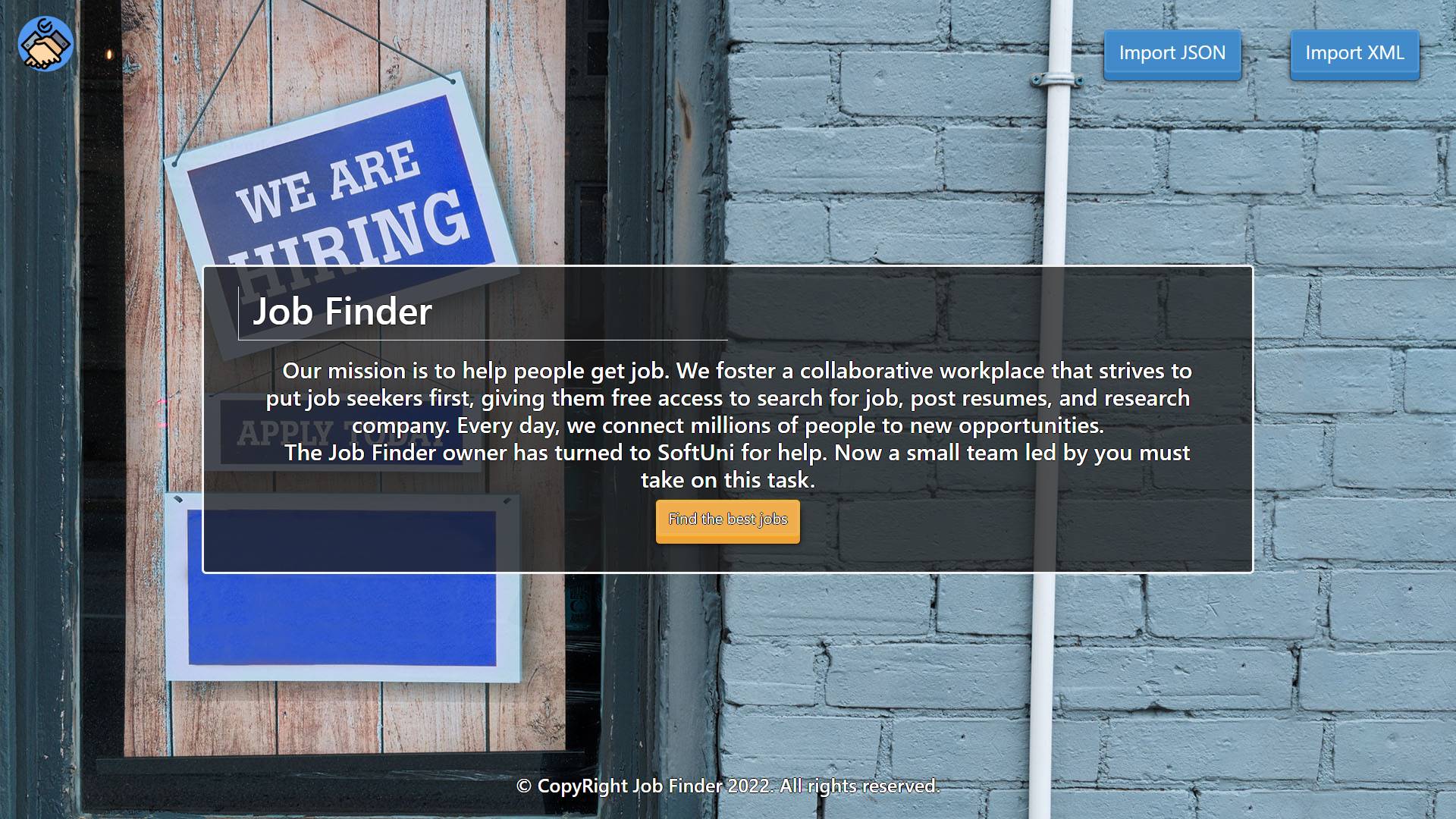
* Import the jobs data:



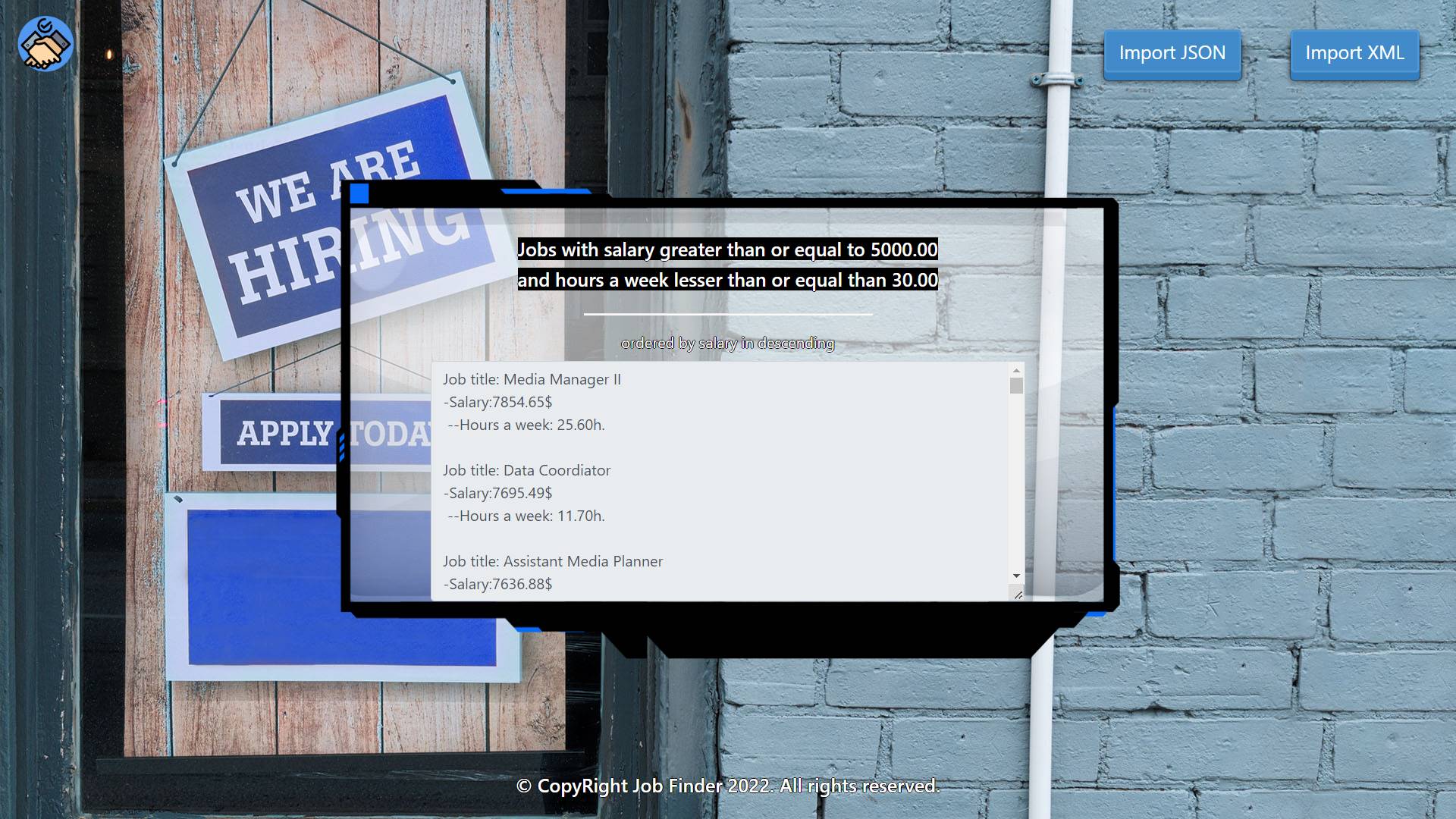
* The import XML page after importing the data:



* The home page after the data is imported:



* Export the best jobs:



## Project Skeleton Overview

You will be given a **skeleton**, containing a **certain architecture (MVC)** with **several classes**, some of which are completely empty. The **Skeleton** will include the **files** with which you will **seed** the **database**.

## Model Definition

There are 4 main models that the **Job Finder database** application should contain in its functionality.

Design them in the **most appropriate** way, considering the following **data constraints**:

### Country

* id – accepts **integer** values, a **primary identification field, an auto incremented field**.
* name – accepts **char sequence** (between **2** to **30** inclusive). The values are **unique in the database**.
* code - accepts **char sequence** (between **2** to **19** inclusive). The values are **unique in the database**.
* currency – accepts **char sequence** (between **2** to **19** inclusive).

### Person

* id – accepts **integer** values, a **primary identification field, an auto incremented field**.
* first name - accepts **char** **sequence** (between **2** to **30** inclusive). The values are **unique in the database**.
* last name - accepts **char** **sequence** (between **2** to **30** inclusive).
* email – an **email** (must contains ‘@’ and ‘.’ – dot). The email of a person is **unique**.
* phone – accepts **char** **sequence** (between **2** to **13** inclusive). Can be **nullable.** The phone of a person is **unique**.
* status type – the employment status of the person. Ordinal enumeration, one of the following – **unemployed, employed, freelancer**
* Constraint: The people table has a relation with the countries table.

### Company

* id – accepts **integer** values, a **primary identification field, an auto incremented field**.
* name – accepts **char** **sequence** (between **2** to **40** inclusive). The values are **unique in the database**.
* website – accepts **char** **sequence** (between **2** to **30** inclusive).
* date established – a date in the "**yyyy-MM-dd**" format.
* jobs – list with all jobs that the company is offering.
* Constraint: The companies table has a relation with jobs table
* Constraint: The companies table has a relation with countries table

### Job

* id – accepts **integer** values, a **primary identification field, an auto incremented field**.
* title – accepts **char** **sequence** (between **2** to **40** inclusive).
* **salary** – accepts number values that are more than or equal to **300.00**.
* **hours a week** – accepts number values that are more than or equal to **10.00**.
* **description** - a long and detailed description of needed skill for this job with a character length value higher than or equal to **5**.
* **Constraint**: The jobs table has a relation with the companies table.

#### Relationships

Your partners gave you a little hint about the more complex relationships in the database, so that you can implement it correctly.

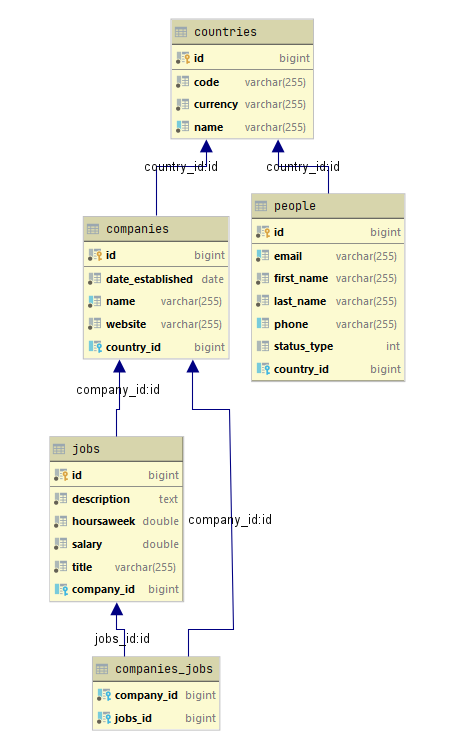
One **Job** may have only one **Company**, but one **Company** may have many **Jobs**.

One **Company** may have only one **Country**, but one **Country** can have many **Companies**.

One **Country** can have many **People**, but one **Person** may have only one **Country**.

**Constraint**:

* Name the entities and their class members **exactly** in the **format stated** above.
* All fields are **NOT NULL** unless explicitly stated to be nullable.



## Data Import

Use the provided files to populate the database with data. Import all the information from those files into the database.

**You are not allowed to modify the provided files.**

**ANY INCORRECT** data should be **ignored** and a message:

"**Invalid {country / person / company / job}** should be printed.

**When the import is finished:**

"**Successfully imported {country / person / company / job} {name – country code / first name last name / name – country / job title}**"

### JSON Import

Your new colleagues have prepared some JSON data for you to import.

|  |
| --- |
| **Country (countries.json)** |
| [  {  "name": "Sweden",  "code": "SE",  "currency": "Krona"  },  {  "name": "Albania",  "code": "A",  "currency": "Lek"  },  {  "name": "Canada",  "code": "CA",  "currency": "Dollar"  },  {  "name": "Greece",  "code": "GR",  "currency": "Euro"  },  {  "name": "Pakistan",  "code": "PK",  "currency": "Rupee"  },  ... |
| Successfully imported country Sweden - SE  Invalid country  Successfully imported country Canada - CA  Successfully imported country Greece - GR  Successfully imported country Pakistan - PK  Successfully imported country Nigeria – NG  … |

**Constraint:**

* **If a person with the same first name, email OR phone number already exists in the DB return "Invalid person".**

|  |
| --- |
| **Person (people.json)** |
| [  {  "email": "lrann0@t-online.de",  "firstName": "Lorna",  "lastName": "Rann",  "phone": "462-463-0432",  "statusType": "freelancer",  "country": "21"  },  {  "email": "ksterland1@hubpages.com",  "firstName": "Krystal",  "lastName": "Sterland",  "phone": "201-115-5358",  "statusType": "freelancer",  "country": "17"  },  {  "email": "gledeker2@amazon.com",  "firstName": "Gerladina",  "lastName": "Ledeker",  "phone": "534-162-5734",  "statusType": "employed",  "country": "28"  },  {  "email": "pburla@gmail.com",  "firstName": "Burla",  "lastName": "Desta",  "phone": "534-162-5734",  "statusType": "employed",  "country": "23"  },  {  "email": "dhebard4@histats.com",  "firstName": "Dulcy",  "lastName": "Hebard",  "phone": "654-577-5713",  "statusType": "freelancer",  "country": "29"  },  {  "email": "evlgmi@stats.com",  "firstName": "Don Fernando de Tomate ke Saragosa",  "lastName": "Burn",  "phone": "654-512-3713",  "statusType": "freelancer",  "country": "29"  },  . . . |
| Successfully imported person Lorna Rann  Successfully imported person Krystal Sterland  Successfully imported person Gerladina Ledeker  Invalid person  Successfully imported person Dulcy Hebard  Invalid person  Successfully imported person Jedd Bisley  … |

### XML Import

Your new colleagues have prepared some XML data for you to import.

**Company (companies.xml)**

**Constraint:**

* **If a company with the same name already exists in the DB return "Invalid company".**

|  |
| --- |
| **Company (companies.xml)** |
| *<?*xml version="1.0" encoding="UTF-8"*?>* <companies>  <company>  <name>Heaney-Vandervort</name>  <dateEstablished>1981-10-06</dateEstablished>  <website>karley.biz</website>  <countryId>34</countryId>  </company>  <company>  <name>Heaney-Vandervort</name>  <dateEstablished>1984-11-16</dateEstablished>  <website>blaze.iz</website>  <countryId>31</countryId>  </company>  <company>  <name>Kulas, Quigley and Walker</name>  <dateEstablished>1981-05-05</dateEstablished>  <website>rosalia.io</website>  <countryId>20</countryId>  </company>  <company>  <name>Greenfelder LLC</name>  <dateEstablished>2003-01-31</dateEstablished>  <website>cleveland.tv</website>  <countryId>38</countryId>  </company>  <company>  <name>Willy Wonkas Special Chocolate Factory</name>  <dateEstablished>2003-01-31</dateEstablished>  <website>Willy\_Wonkas\_Special\_Chocolate\_Factory@abv.bg</website>  <countryId>32</countryId>  </company>  <company>  <name>Cormier-Douglas</name>  <dateEstablished>2001-01-24</dateEstablished>  <website>hyman.biz</website>  <countryId>18</countryId>  </company>  <company>  <name>Feest Group</name>  <dateEstablished>2007-12-02</dateEstablished>  <website>lewis.us</website>  <countryId>34</countryId>  </company>  ... |
| Successfully imported company Heaney-Vandervort - 34  Invalid company  Successfully imported company Kulas, Quigley and Walker - 20  Successfully imported company Greenfelder LLC - 38  Invalid company  Successfully imported company Cormier-Douglas - 18  Successfully imported company Feest Group – 34  ... |

**Job (jobs.xml)**

|  |
| --- |
| **Job (jobs.xml)** |
| *<?*xml version='1.0' encoding='UTF-8'*?>* <jobs>  <job>  <title>Assistant Professor</title>  <hoursAWeek>21.1</hoursAWeek>  <salary>5754.07</salary>  <description>Cras pellentesque volutpat dui. Maecenas tristique, est et tempus semper, est quam pharetra magna,  ac consequat metus sapien ut nunc.  </description>  <companyId>38</companyId>  </job>  <job>  <title>Senior Financial Analyst</title>  <hoursAWeek>6.5</hoursAWeek>  <salary>6541.34</salary>  <description>Aliquam augue quam, sollicitudin vitae, consectetuer eget, rutrum at, lorem. Integer tincidunt ante  vel ipsum. Praesent blandit lacinia erat. Vestibulum sed magna at nunc commodo placerat.  </description>  <companyId>66</companyId>  </job>  <job>  <title>Account Representative II</title>  <hoursAWeek>53.1</hoursAWeek>  <salary>97.45</salary>  <description>Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Proin interdum mauris non ligula  pellentesque ultrices. Phasellus id sapien in sapien iaculis congue. Vivamus metus arcu, adipiscing  molestie, hendrerit at, vulputate vitae, nisl. Aenean lectus. Pellentesque eget nunc. Donec quis orci eget  orci vehicula condimentum.  </description>  <companyId>95</companyId>  </job>  <job>  <title>Research Nurse</title>  <hoursAWeek>66.5</hoursAWeek>  <salary>4373.33</salary>  <description>Donec posuere metus vitae ipsum. Aliquam non mauris. Morbi non lectus. Aliquam sit amet diam in  magna bibendum imperdiet. Nullam orci pede, venenatis non, sodales sed, tincidunt eu, felis. Fusce posuere  felis sed lacus.  </description>  <companyId>93</companyId>  </job>  ... |
| Successfully imported job Assistant Professor  Invalid job  Invalid job  Successfully imported job Research Nurse  Successfully imported job Editor  Successfully imported job Cost Accountant  Successfully imported job Software Engineer III  … |

## Data Export

Get ready to export the data you have imported in the previous task. Here you will have some complex database querying. Export the data in the formats specified below.

#### Export The Best Jobs from the Data Base

* Extract from the database, the **job title, job salary (to second digit after decimal point), and job's hours a week (to second digit after decimal point)**.
* **Filter only jobs where salary is greater than or equal to 5000.00 and hours a week are lesser than or equal 30.00 and order them by the salary in descending order.**
* Return the information in this format:
* **"Job title {jobTitle}**

**-Salary: {jobSalary}$­­**

**--Hours a week: {hoursAWeek}h.**

**. . ."** 