You just got hired as the first and only data practitioner at a small business experiencing exponential growth. The company needs more structured processes, guidelines, and standards. Your first mission is to structure the human resources data. The data is currently scattered across teams and files and comes in various formats: Excel files, CSVs, JSON files...

You'll work with the following data in the `datasets` folder:

- **Office addresses** are currently saved in `office_addresses.csv`. If the value for office is `NaN`, then the employee is remote.
- **Employee addresses** are saved on the first tab of `employee_information.xlsx`.
- **Employee emergency contacts** are saved on the second tab of `employee_information.xlsx`; this tab is called `emergency_contacts`. However, this sheet was edited at some point, and the headers were removed! The HR manager let you know that they should be: `employee_id`, `last_name`, `first_name`, `emergency_contact`, `emergency_contact_number`, and `relationship`.
- **Employee roles, teams, and salaries** have been exported from the company's human resources management system into a JSON file titled `employee_roles.json`. Here are the first few lines of that file:

``` {"A2R5H9": { "title": "CEO", "monthly_salary": "$4500", "team": "Leadership" }, ... }
```

In [3]:
```
import pandas as pd

# Read in office_addresses.csv
offices = pd.read_csv("datasets/office_addresses.csv")

# Declare a list of columns to keep from addresses
addresses_cols = ["employee_id", "employee_country", "employee_city", "employee_street", "employee_street_nu

# Read in employee_information.xlsx
addresses = pd.read_excel("datasets/employee_information.xlsx",
                          usecols=addresses_cols)

# Declare a list of new column names
emergency_contacts_header = ["employee_id", "last_name", "first_name",
                             "emergency_contact", "emergency_contact_number", "relationship"]

# Read in employee_information.xlsx
emergency_contacts = pd.read_excel("datasets/employee_information.xlsx",
                                   sheet_name="emergency_contacts",
                                   header=None,
                                   names=emergency_contacts_header)

# Read in employee_roles.json
roles = pd.read_json("datasets/employee_roles.json", orient="index")

# Merge addresses with offices
employees = addresses.merge(offices, left_on="employee_country", right_on="office_country", how="left")

# Merge employees with roles
```

```
employees = employees.merge(roles, left_on="employee_id", right_on=roles.index)

# Merge employees with emergency_contacts
employees = employees.merge(emergency_contacts, on="employee_id")

# Fill null values in office columns
for col in ["office", "office_country", "office_city", "office_street", "office_street_number"]:
    employees[col].fillna("Remote", inplace=True)

# Create final columns
final_columns = ["employee_id", "first_name", "last_name", "employee_country",
                 "employee_city", "employee_street", "employee_street_number",
                 "emergency_contact", "emergency_contact_number", "relationship",
                 "monthly_salary", "team", "title", "office", "office_country",
                 "office_city", "office_street", "office_street_number"]

# Subset for the required columns
employees_final = employees[final_columns]

# Set employee_id as the index
employees_final.set_index("employee_id", inplace=True)
```

In [4]:
```
employees_final
```

employee_id	first_name	last_name	employee_country	employee_city	employee_street	employee_street_number	emergency
A2R5H9	Jax	Hunman	BE	Leuven	Grote Markt	9	Opie Hurst
H8K0L6	Tara	Siff	GB	London	Baker Street	221	Wendy de Ma
G4R7V0	Gemma	Sagal	US	New-York	Perry Street	66	John Newma
M1Z7U9	Tig	Coates	FR	Paris	Rue de l'Université	7	Venus Noone