# Documentation

*Note: feel free to modify/add your suggestions*

## Log

| **Updated At** | **Updated By** | **Detail** |
| --- | --- | --- |
| 2023-04-06 | Tue Nguyen | Created the documentation |
| 2023-04-06 | Tue Nguyen | Added the general outline, including Log, Coding style convention, Data, and Approaches sections |

## Coding style convention

| **Guideline** | **Example** |
| --- | --- |
| When naming, use all lowercase | - Good: accounts  - Bad: Accounts, ACCOUNTS |
| When a name has more than 2 words, separate words with underscores | - Good: emp\_hist  - Bad: EmpHist, emphist |
| Use short but meaningful names | - Good: emp\_hist  - Bad: employment\_history (too long), eh (too short) |
| Put a space on each side of an operator to increase readability | - Good: z = x + y - Bad: z=x+y |
| Add comments to code when necessary |  |

## Data

### Data dictionary

Note:

* For each CSV file, we will give it a short, descriptive name and use that name as the variable name when loading the data into a Pandas data frame. The proposed name for each file is listed in the Dataset column of the table below.
* Locations of the data
  + The original data should be located at data/original/<original-file-name>.csv
  + The cleaned data should be located at data/cleaned/<dataset-name>.csv

| **Dataset** | **N rows** | **N cols** | **Original File Name** | **Description** |
| --- | --- | --- | --- | --- |
| emp\_hist | 157.6K | 14 | AVTRRT\_\_Employment\_History\_\_Anonymized.csv | Data on the employment history of candidates |
| interviews | 5.1K | 37 | AVTRRT\_\_Interview\_\_Anonymized.csv | Data on interviews |
| jobs | 4.6K | 91 | AVTRRT\_\_Job\_Anonymized.csv | Data on job advertisements |
| applicants | 75.2K | 82 | AVTRRT\_\_Job\_Applicant\_\_Anonymized.csv | Data on job applicants |
| placement | 701 | 49 | AVTRRT\_\_Placement\_Anonymized.csv | Data on job placement |
| references | 7.8K | 16 | AVTRRT\_\_References\_\_Anonymized.csv | Data on references the candidate provides |
| accounts | 4.7K | 44 | Account\_Anonymized.csv | Data on companies |
| contacts | 56.2K | 92 | Contact\_Anonymized.csv | Contact point for each company (account) |
| records | 9 | 10 | RecordType\_Anonymized.csv | Too little data, probably ignore this table |

### Entity relationship diagram

To be done.

## Approaches

### Step 1: Explore the raw, original data

Goals

* To have a general understanding of the data
* Identify
  + What data are available
  + What tables and columns are helpful
  + What columns should be ignored (too many missing values, …)

Tasks

* Explore each CSV file in a separate notebook first
* Take notes of any interesting findings

### Step 2: Clean and subset the data

Goals

* Narrow the scope to a few tables we want to work with
* For each of these tables, keep only part of the original data that are relevant for further analyses
* Do some preprocessing/cleaning to ensure data quality

Tasks

* Based on Step 1, we might have a sense of what tables contain rich enough information for meaningful analyses. Then we will start with them first. We can extend to include more tables along the way.
* Also based on Step 1, we will write a function to clean and subset data for each CSV file
* The function will read the data from data/original and write the cleaned data into data/cleaned
  + Example input: Account\_Anonymized.csv
  + Example output: accounts.csv
* The function name follows the convention: clean\_<dataset\_name>. Example

def clean\_accounts(src, dest):

"""

Reads data from source (src), cleans it, and writes results

to destination (dest)

"""

# Load data

df = pd.read\_csv(f"data/original/{src}.csv")

# Clean data

...

# Write cleaned data

df.to\_csv(f"data/cleaned/{dest}.csv", index=False)

* Lastly, we will have a master function that calls all cleaning functions at once to perform data cleaning for all files. Thus, whenever you need to change the logic, we only need to call this master function

### Step 3: Further exploratory data analysis on cleaned data

Goals

* Get a deeper understanding of the overall data, not just a single table

Tasks

* Create a new notebook for this new analysis
* Analyze each table separately or jointly
* Take notes of any interesting finding

### Step 4: Narrow the scope of the analysis

Goals

* Define more clearly what problem we want to solve given the available data and results from previous analysis

Tasks

* Try to identify 2-3 problems we want to solve (and can solve with the data)
* Example: predict y=salary given x=[education, experience, matching level between candidate skills and job description, …]
* Note: read the Lab Description.pdf file might help

### Step 4: Solve the first problem

Goals

* Solve the problem we find most interesting

Tasks

* Start with the most interesting problem
* Build a simple, baseline model
* Iterate to improve the model until satisfied

### Step 5: Solve the remaining problems (optional)

Goals

* If time allows, we might want to solve the remaining problems

Tasks

* Repeat Step 4 for the remaining problems

### Step 5: Wrap up

Goals

* Prepare for the final presentation

Tasks

* Prepare the slide
* Allocate the presentation section to each team member

Problematic columns - by Aldo’s email:

* in the "interview" table, in the column AVTRRT\_\_Special\_Instructions\_to\_Candidates\_\_c > delete / info not relevant
* in the "job applicant" table, in the column AVTRRT\_resume\_c > need to be anonymised, depending on the type of analysis chosen
* in the "employment history" table, in the column AVTRRT\_description\_c > needs to be anonymised, depending on the type of analysis chosen
* in the "references" table, column AVTRRT\_notes > delete / info not relevant
* in the "placement" table, column note\_c > delete / info not relevant

Interesting References:

* <https://youtu.be/OY4eQrekQvs?t=1013>

Giordano Vitale:

* Employment history
* accounts

Enrico Darra:

* Placement
* Contacts (working on it)

Emile Rahal:

* References (working on it)
* Interview (working on it)

Tue Nguyen:

* Job
* Job applicants