# Task 1

You are given json dump of data from MongoDB database of the following collections:
- complex.json
- tower.json

create a data pipeline to RDBMS of your choice (MySQL/PostgreSQL/etc) to replicate existing dataset from MongoDB collections into RDBMS tables with the following requirements:
- these fields are mandatory and must be NOT NULL:

| **table/ collection** | **initial field name** | **new field name** | **type** | **note** |
|---|---|---|---|---|
| complex | status | IsActive | bool | 0 = inactive<br>1 = active |
| | category | Category | string | |
| | name | Name | string | |
| tower | status | IsActive | bool | 0 = inactive<br>1 = active |
| | category | Category | string | |
| | name | Name | string | |
| unit | status | IsActive | bool | -1 = inactive<br>0 = active |
| | category | Category | string | |
| | size | Size | numeric | |

- the rest of the fields are optional
- `numeric` means anything number (int, long, bigint, float, decimal, up to your decision)
- use Python/Go is a plus (we primarily use those two, but you are free to use any other language)

hints
- `_id` can be used as Primary Key (`id` is just string form of `_id`)
- modify the collections structure and data types as necessary

# Task 2

Based on existing data, write a query/program to solve one of the following tasks:
rank complexes based on the completeness of images using this point metric:

| **field name** | **0** | **1-2** | **3-4** | **5 or more** |
|---|---|---|---|---|
| images_developer | 0 | 5 pt | 7 pt | 10 pt |
| images_banner | 0 | 10 | 10 | 10 |
| images_brochure | 0 | 7 | 10 | 10 |
| images_interior | 0 | 5 | 7 | 10 |
| images_exterior | 0 | 3 | 5 | 10 |
| images_360 | 0 | 7 | 10 | 10 |
| video_link | 0 | 10 | 10 | 10 |

hints
- create intermediary view table

# Task 3

The management want to plan long term strategy that our company needs to adopt, but to do that, they require data-driven feedback from the team. All the necessary data all available at production database, but they are not easy to query, hence we may need to do some transformation before porting the data to analytic database.
Eventually, the data at analytic database needs to be:
- easy to query (aggregate, group, filter by time range)
- updated frequently

Based on the existing requirements, design data architecture to meet this data needs