DE Technical Challenge

Python Exercise

Below is a list of database models:

Employee	Company	Position	Job
• **guid** • **status** • **state**	• guid • name	• guid • name	• guid • company_guid • position_guid • employee_guid

For our data needs, we'll be getting extra data from all of our models, the following information will be appended to each model:

- action (INSERT, UPDATE, or DELETE)
- timestamp

Your Challenge

You're tasked with writing a python script. Your python script must do the following:

- Read in a file that contains multiple lines
- Each line will be a JSON formatted record (which may be from any of the models above)
- You'll be able to differentiate these apart by an extra column inside each record called *source_table*
- Print out the final state of each record (after all *actions* have been applied)
- This can be found using *source_table*, *guid*, *action,* and *timestamp*
- Note: You are free to add additional columns to your models as you see fit for the purpose of testing your application

A **sample input file** is provided for you. When processing this file, your code should print an output similar to this:

```
Company
==========
{'guid': '1c898066-858e-406c-a15d-36146c9642de', 'name': 'Paylocity',
'status': '2'}
{'guid': '0090d7b0-b07a-47cd-b295-ff798a6c0613', 'name': 'Burrito Shack',
'status': '2'}

Job
==========
{'guid': '5ab54bb5-b72d-40f8-9a49-e0d2d004d7a9', 'company_guid': '0090d7b0-
b07a-47cd-b295-ff798a6c0613', 'employee_guid': '259d5154-5f76-481b-b0f9-
53e24c3b570e'}
```

```
{'guid': 'f73a2796-4579-4779-8345-f0dfcf7dd533', 'company_guid': '0090d7b0-
b07a-47cd-b295-ff798a6c0613', 'employee_guid': '4e0c8c17-b031-4a72-b73d-
f0a85570826d'}
{'guid': '58291fe5-4e4c-41da-87a5-e1fccb8aac25', 'company_guid': '1c898066-
858e-406c-a15d-36146c9642de', 'employee_guid': 'e086115c-0ca1-480c-8fa8-
5e1773558b9f'}

Position
==========
{'guid': '40a36493-f450-4331-874c-5ef01aabe1d5', 'name': 'Software Engineer',
'status': '1'}
{'guid': 'f9b3ee71-7fb2-4dd5-9c13-b4c10d11fde7', 'name': 'Data Engineer',
'status': '1'}

Employee
==========
{'guid': '4e0c8c17-b031-4a72-b73d-f0a85570826d', 'state': 'MI', 'status':
'3'}
{'guid': 'd4926109-b6c9-4447-a53c-b787684e10f1', 'state': 'IL', 'status':
'3'}
{'guid': 'e086115c-0ca1-480c-8fa8-5e1773558b9f', 'state': 'FL', 'status':
'2'}
{'guid': '259d5154-5f76-481b-b0f9-53e24c3b570e', 'state': 'NY', 'status':
'1'}
```

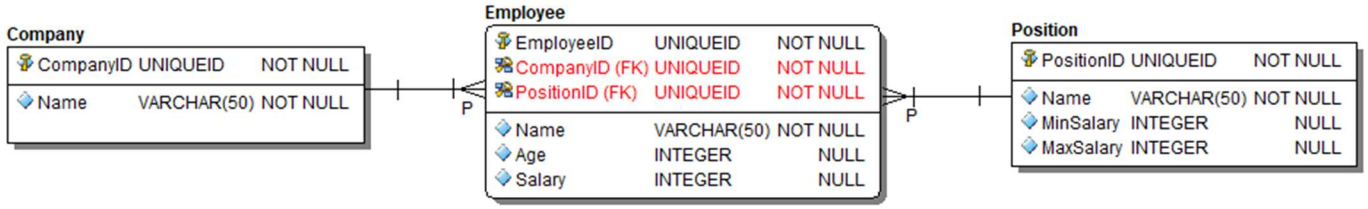

What we'll be evaluating and asking questions around:

- Your algorithm
- Your print management
- Code abstraction
- Documentation
- Testing

This is your chance to show us what you can do! Have fun with the exercise!

SQL Exercise

Given three tables:



Do the following:

1. Write a SQL report that gives us the average salary for all employees
2. Update previous query to give us the average salary for each company
3. Let's assume your previous query was long running how would you go about debugging it and finding the root causes of its sluggishness.