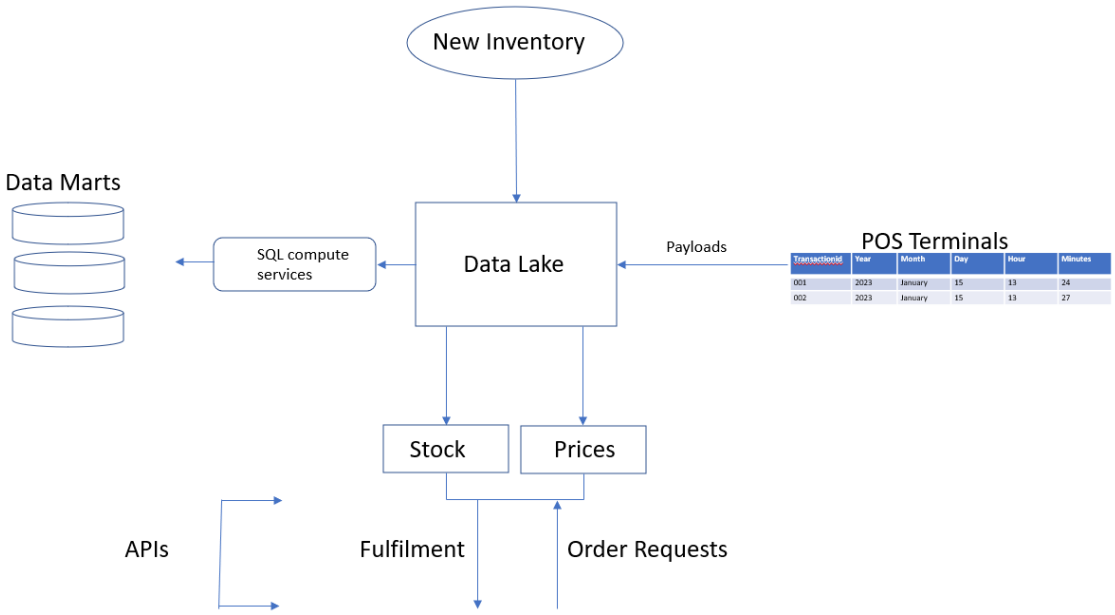
Touch Bistro Technical Challenge

Question 1: Case study

Process Model of data architecture:



Issues:

- Data duplication and concurrency issues from batch jobs taking longer than their allotted time to run . i.e. The same payload is being actioned by multiple sequential scripts at the same time

Workarounds

- Guardrails for jobs to timeout after 5 minutes
- Any incomplete jobs would be rolled back with a notification sent for manual execution

Possible Solutions

- Adopt a warehousing solution that can process data can parallel process data at a faster rate
    - Trade-offs:
        - Might be more expensive to implement and maintain
        - Might require more servers and increased cost of maintenance
- Opt for real time data streaming as opposed to batch processing.
    - Trade-offs:
        - Might be more expensive to implement and maintain
        - Harder to track issues in the pipeline

Assumptions

- The SQL data model is optimized
- There's FTE with the requisite skills and knowledge to work with novel software and tools
- The business case has been made and approved to invest in process improvement
- BAU processes won't be impacted while new infrastructure is introduced

Question 2: Covid-Revenue Analysis

Business objective

- To understand how the daily movement in covid cases during the pandemic impacted restaurant revenue

Data Sources

- Covid-19 API
- Sample Restaurant Revenue

Code Logic

- Please see *TouchBistro_Technical_Test_Ade Ogunlewe.py* file in zip folder

Outputs (Covid_19_Analysis_Ade Ogunlewe.xlsx)

None	week_start	week_end	region	Revenue	Revenue	Revenue	Revenue	Covid_Cases	Covid_Cases	Covid_Cases	Covid_Cases
None				mean	min	max	var	mean	min	max	var
10	2020-03-16 00:00:00	2020-03-22 00:00:00	ON	2000	1000	3000	625000	86	72	99	183
11	2020-03-23 00:00:00	2020-03-29 00:00:00	ON	3723.71	1102	7278	5.6329e+06	209.571	138	297	3472.95
12	2020-03-30 00:00:00	2020-04-05 00:00:00	ON	6384.33	763	9859	1.03948e+07	419.714	349	465	1987.9
13	2020-04-27 00:00:00	2020-05-03 00:00:00	ON	4630	2143	7117	1.23703e+07	410.571	333	488	3622.95
14	2020-05-04 00:00:00	2020-05-10 00:00:00	ON	4595.86	798	9505	1.06927e+07	329.571	294	409	1704.62
15	2020-05-11 00:00:00	2020-05-17 00:00:00	ON	5367.29	1237	9937	1.14913e+07	335.857	216	421	5282.14
16	2020-05-18 00:00:00	2020-05-24 00:00:00	ON	7233.14	2864	9823	7.75702e+06	362.571	329	400	747.952
17	2020-05-25 00:00:00	2020-05-31 00:00:00	ON	5973.43	369	9979	1.59278e+07	383.286	273	606	11896.6
18	2020-06-01 00:00:00	2020-06-07 00:00:00	ON	9527	9527	9527	nan	299.429	207	396	5255.29
19	2020-08-10 00:00:00	2020-08-16 00:00:00	ON	5879.14	2263	8930	5.36419e+06	88.5714	69	100	97.2857
20	2020-08-17 00:00:00	2020-08-23 00:00:00	ON	3529.43	137	8804	1.01233e+07	104.286	82	133	229.238

Conclusion

- **Revenues peaked for this restaurant in June and September of 2020.**
- **This might be explained as people relying more on takeout, alcohol etc during covid.**
- **But a closer look suggests this pattern is confined to the summer months, so seasonality also looks to be a strong factor.**

Question 3: SQL Query

Note: Ran into some issues using SQL-Fiddle, so use PostgreSQL on my local machine

Business Objective

- Get the total salary incurred per employee for a given period

Given Assumptions

1. An employee will always have a clock-in time.

2. An employee could have a blank clock-out time, either due to error, or because they are still mid-shift.

3. If the shift does not have a clock-out time, but it has started at most 12 hours prior to your window's start time, it should be considered in your set.

4. When calculating the time spent working, please only consider the time within the window given in the input, regardless of when the employee clocked in or out.

Added Assumptions

- A shift lasts no more than 8 hours.

Code Logic

- Please see *TouchBistro_TechnicalTest2_SQL_Ade Ogunlewe.sql* file in zip folder

Outputs

employee_id	first_name	last_name	role	total_accrued_hourly	total_hours_work	total_pay
12345678910	John	Wilkinson	manager	90.00	45 seconds 13 minutes 16 hours	1440
12345678911	Bean	Borealis	associate	40.00	20 seconds 30 minutes 14 hours	560
12345678912	Jean	French	senior associate	25.00	17 seconds 48 minutes 10 hours	250
12345678913	Albert	Munk	director	40.00	31 seconds 8 minutes 10 hours	400

Conclusion

- John earns the most pay during this period
- Albert earns the least during this period
- There's a fairly strong correlation between hours worked and total pay.