

# Data2Bots Data Engineering Technical Assessment

 **This is a data2bots take-home assignment. For the Data Engineer role at Data2Bots!**

## Overview:

This guide details an assessment, which serves as the technical assessment phase associated with the **Data Engineering** role at Data2Bots. Upon completion of this assessment, your submission will be graded, and based on this you will receive further correspondence from our HR Team.

This guide details the following:

- [The goal of the assessment](#)
- [The assessment preamble](#)
- [Submission requirements](#)
- [Important Guidelines](#)

## Goal

The purpose of this exercise is for you to demonstrate how you would solve a real-world Data Engineering problem in the absence of the pressure of a live coding exercise.

 **This assessment especially tests your thought process and approach.**

## Assessment Preamble

As the sole data engineer of ABC Inc, a business stakeholder has come to you with the following requirements;

- Hey, we have data coming into our central data lake (i.e. a file system) every day.

 **Our central data lake is an Amazon S3 Bucket:**

**Bucket Name: d2b-internal-assessment-bucket**

**Data Locations: s3://d2b-internal-assessment-bucket/orders\_data/\***

- This directory contains the following files:
  - `orders.csv`: This data is a fact table about orders gotten on our website ABC.com
  - `reviews.csv`: This data is a fact table on reviews given for a particular delivered product
  - `shipments_deliveries.csv`: This is a fact table on shipments and their delivery dates

**YOU CAN USE THIS SNIPPET OF CODE TO ACCESS THE S3 BUCKET:**

```
import boto3
from botocore import UNSIGNED
from botocore.client import Config

s3 = boto3.client('s3', config=Config(signature_version=UNSIGNED))

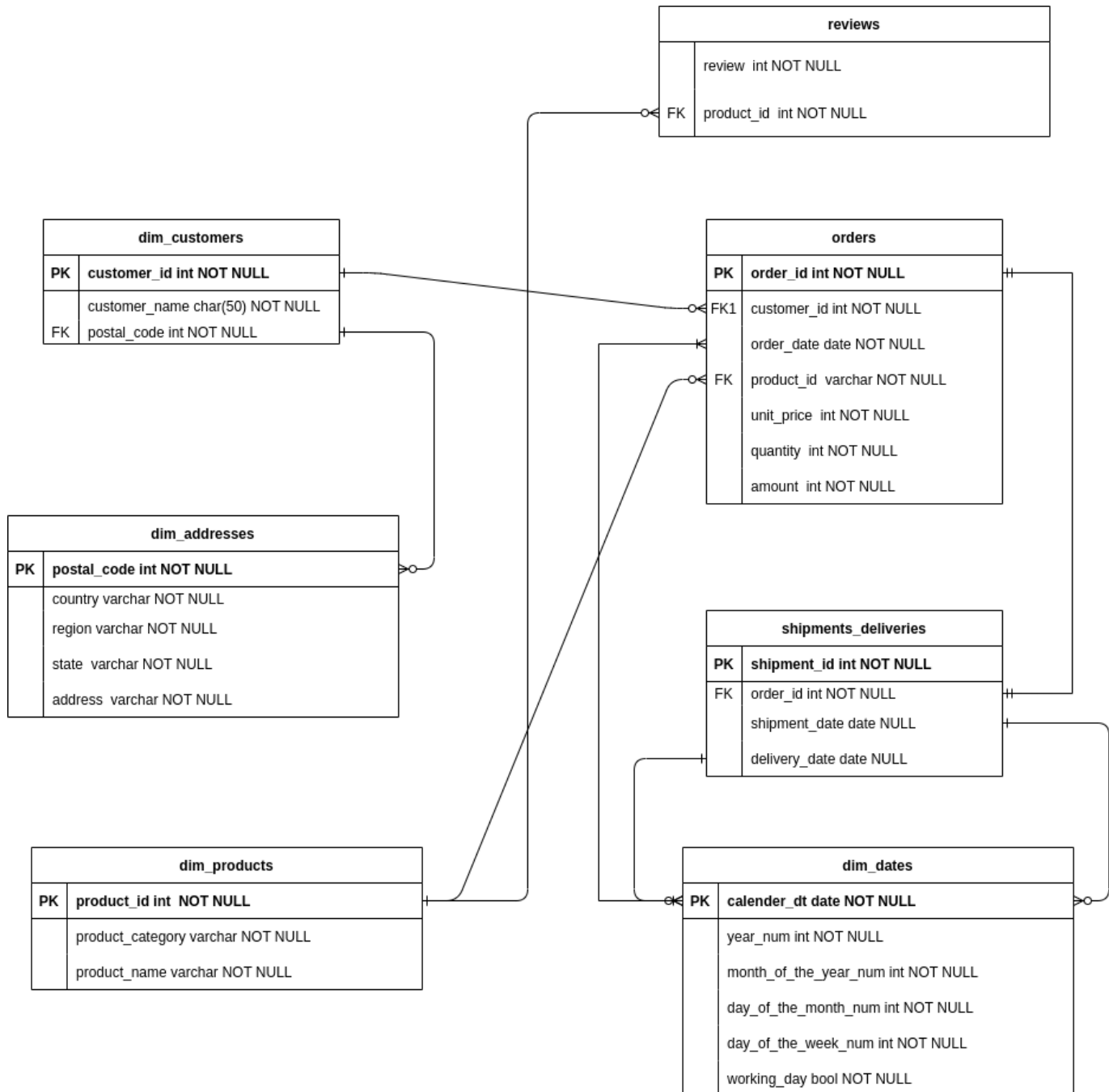
bucket_name = "d2b-internal-assessment-bucket"
response = s3.list_objects(Bucket=bucket_name, Prefix="orders_data")

# for example to download the orders.csv
s3.download_file(bucket_name, "orders_data/orders.csv", "orders.csv")
```

Below is the ER shows the data model of our data warehouse;

**Please note all dim\_\* tables are loaded into the if\_common schema.**

For example, you can access the `dim_customers` table with `if_common.dim_customers`  
**You have only select access on these tables.**



- We would like to load the raw data files into a schema `{your_id}_staging` within our enterprise data warehouse which is a **Postgres** database.

Your schema which holds your tables are already created within the data warehouse.

And you can create and access tables in that schema using `{your_id}_staging.{table_name}`

- We want to know the following:
  - The total number of orders placed on a public holiday every month, for the past year.
    - A public holiday is a day with a `day_of_the_week` number in the range 1 - 5 and a `working_day` value of `false`.

- After your transformation, the derived table `agg_public_holiday` should be loaded into the `{your_id}_analytics` schema, using the below table schema.

agg_public_holiday	
PK	ingestion_date date NOT NULL
	tt_order_hol_jan int NOT NULL
	tt_order_hol_feb int NOT NULL
	tt_order_hol_mar int NOT NULL
	tt_order_hol_apr int NOT NULL
	tt_order_hol_may int NOT NULL
	tt_order_hol_jun int NOT NULL
	tt_order_hol_jul int NOT NULL
	tt_order_hol_aug int NOT NULL
	tt_order_hol_sep int NOT NULL
	tt_order_hol_oct int NOT NULL
	tt_order_hol_nov int NOT NULL
	tt_order_hol_dec int NOT NULL

- Total number of late shipments
  - A late shipment is one with `shipment_date` greater than or equal to 6 days after the `order_date` and `delivery_date` is NULL
- Total number of undelivered shipments
  - An undelivered shipment is one with `delivery_date` as NULL and `shipment_date` as NULL and the `current_date` 15 days after `order_date`.

**NB::** current\_date here refers to 2022-09-05

- Write the two transformations into the following table in your `{your_id}_analytics` schema:

agg_shipments	
PK	ingestion_date date NOT NULL
	tt_late_shipments int NOT NULL
	tt_undelivered_items int NOT NULL

- The product with the highest reviews, the day it was ordered the most, either that day was a public holiday, total review points, percentage distribution of the review points, and percentage distribution of early shipments to late shipments for that particular product.
  - Write this transformation into the following table in your `{your_id}_analytics` schema:

best_performing_product
-------------------------

PK	ingestion_date date NOT NULL
	product_name varchar NOT NULL
	most_ordered_day date NOT NULL
	is_public_holiday bool NOT NULL
	tt_review_points int NOT NULL
	pct_one_star_review float NOT NULL
	pct_two_star_review float NOT NULL
	pct_three_star_review float NOT NULL
	pct_four_star_review float NOT NULL
	pct_five_star_review float NOT NULL
	pct_early_shipments float NOT NULL
	pct_late_shipments float NOT NULL

All ingestion\_date are the current date the table was generated


- We would also like an export of these tables to be loaded into the analytics\_export folder on our data lake. As analytics\_export/{your\_id}/best\_performing\_product.csv

For example, say you have done the transformation of the **best\_performing\_product**, and your id is user1234 you will export the table to the following s3 location.

s3://d2b-internal-assessment-bucket/analytics\_export/user1234/best\_performing\_product.csv



## Submission Requirements

After reading and understanding the above preamble:






-  Create a private git repository and do your work in there as if this was a real project at work.

Please share with us the repo once you're done.

**IMPORTANT WARNING! Your submission WILL NOT be considered if your repository is NOT PRIVATE.**

-  Build an ELT pipeline (batch or streaming) that loads the business data in our data warehouse and performs the transformation.
-  Explain your work in a README file.

## Important Guidelines

-  We are not judging you based on submission time. If there are things you wanted to include, but had no time for them, explain them in the README.
-  Please write your extract and load code in python, your transformation in SQL, and if you plan to use any infrastructure as code 'framework' preferably use Terraform.
-  Feel free to be creative and come up with requirements to make the assignment feel more real. Please note these down for us in the README.
-  We don't have a concrete solution in mind. We are heavily interested in your thinking process, and the way you design and build a solution.
-  Although this is small dummy data and a toy exercise, try to aim for a production-grade solution. Think of scalability, maintainability, and reliability.