# **Data Modelling with Postgre Documentation**

# **Introduction**

# The goal of this project is to build a PostgreSQL database utilizing the data on users activity and songs metadata. Building the database helps us do complex analytics regarding users activity as well as song play analysis.

# **Project Dataset**

* Song Dataset: files are partitioned by the first three letters of each song's track ID e.g. */data/song\_data.*.json. Sample:

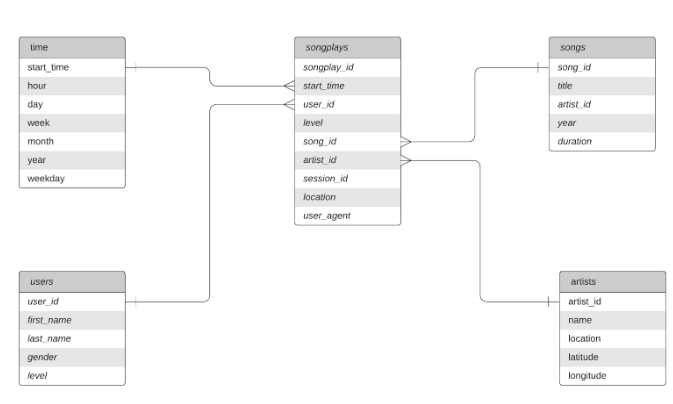
{"artist\_id": "ARD7TVE1187B99BFB1", "artist\_latitude": null, "artist\_location": "California - LA", "artist\_longitude": null, "artist\_name": "Casual", "duration": 218.93179, "num\_songs": 1, "song\_id": "SOMZWCG12A8C13C480", "title": "I Didn't Mean To", "year": 0}

* Log Dataset: files in the dataset you'll be working with are partitioned by year and month e.g. \*/data/log\_data.\*json. Sample:

{"artist": "Stephen Lynch", "auth": "Logged In", "firstName": "Jayden", "gender": "M", "itemInSession": 0, "lastName": "Bell", "length": 182.85669, "level": "free", "location": "Dallas-Fort Worth-Arlington", "method": "TX PUT", "page": "NextSong", "registration": 1.540992.., "sessionId": "829", "song":"Jim Henson's Dead", "status": 200, "ts": 1543537327796, "userAgent": "Mozilla/5.0 (compatible; MSIE 10.0; Windows NT...", "userId": 91}

**Data Modeling**

We will use the Star Schema: one fact table consisting of the measures associated with each event *songplays*, and referencing four dimensional tables *songs*, *artists*, *users* and *time*, each with a primary key that is being referenced from the fact table.



## **Project template**

The data files, the project includes five files:

1. *create\_tables.py*: drops and creates your tables. You run this file to reset your tables before each time you run your ETL scripts.
2. *etl.ipynb*: reads and processes a single file from *song\_data* and *log\_data* and loads the data into your tables. This notebook contains detailed instructions on the ETL process for each of the tables.
3. *etl.py*: reads and processes files from *song\_data* and *log\_data* and loads them into your tables. You can fill this out based on your work in the ETL notebook.
4. *sql\_queries.py*: contains all your sql queries, and is imported into the last three files above.
5. *test.ipynb*: displays the first few rows of each table to let us check on the database.

## **How to Run**

1. Run *create\_tables.py* to create the database and tables.
2. Run *etl.py* to process for loading, extracting and inserting the data.
3. Run *test.ipynb* to confirm the creation of database and columns.