Week 1 – Project Data model i

Data model

Actor
actor_id: BIGINT NOT NULL [PK]
actor_login: VARCHAR(50) NOT NULL
actor_display_login: VARCHAR(50) NOT NULL
actor_gravatar_id: VARCHAR(50)
actor_url: VARCHAR(100) NOT NULL
actor_avatar_url: VARCHAR(100) NOT NULL

Org
org_id: BIGINT NOT NULL [PK]
org_login: VARCHAR(50) NOT NULL
org_gravatar_id: VARCHAR(50)
org_url: VARCHAR(100) NOT NULL
org_avatar_url: VARCHAR(100) NOT NULL

Event
event_id: VARCHAR(20) NOT NULL [PK]
event_type: VARCHAR(50) NOT NULL
event_public: BOOLEAN NOT NULL
event_created_at: TIMESTAMP NOT NULL
actor_id: BIGINT NOT NULL [FK]
repo_id: BIGINT NOT NULL [FK]
org_id: BIGINT [FK]
payload_push_id: BIGINT [FK]

Repo
repo_id: BIGINT NOT NULL [PK]
repo_name: VARCHAR(100) NOT NULL
repo_url: VARCHAR(150) NOT NULL

Payload
payload_push_id: BIGINT NOT NULL [PK]
payload_size: BIGINT NOT NULL
payload_ref: VARCHAR(200) NOT NULL
payload_commit_sha: VARCHAR(100) [FK]

Commit
commit_sha: VARCHAR(100) NOT NULL [PK]
commit_email: VARCHAR(100) NOT NULL
commit_name: VARCHAR(100) NOT NULL
commit_url: VARCHAR(200) NOT NULL

Tables

Fact table:

- Event

Dimension table:

- Actor
- Repo
- Org
- Payload : collect only payload appeared the push_id
- Commit

Relations

Event – Actor:

    each event must have an actor / each actor can belong to many events

Event – Repo:

    each event must have a repo / each repo can belong to many events

Event – Org:

    each event can appear or disappear an org / each org can belong to many events

Event – Payload:

    each event can appear or disappear a payload / each payload belong to single event

Payload – Commit:

    each payload can appear or disappear a commit / each commit belong to single payload

Project implementation instruction

1. Reach the project repository 'swu-ds525/01-data-modeling-I'

   : *$ cd 01-data-modeling-i*

   ```
   ● (base) JC@Napchins-MacBook-Air swu-ds525 % cd 01-data-modeling-i
   ○ (base) JC@Napchins-MacBook-Air 01-data-modeling-i %
   ```

2. Create visual environment for the project's resources, named 'ENV'
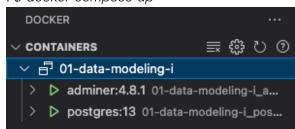
   : *$ python -m venv ENV*

   ```
   ∨ SWU-DS525
      ∨ 01-data-modeling-i
         > ENV
   ```

3. Activate visual environment 'ENV' to be used

   : *$ source ENV/bin/activate*

   ```
   ● (base) JC@Napchins-MacBook-Air 01-data-modeling-i % source ENV/bin/activate
   ○ (ENV) (base) JC@Napchins-MacBook-Air 01-data-modeling-i %
   ```
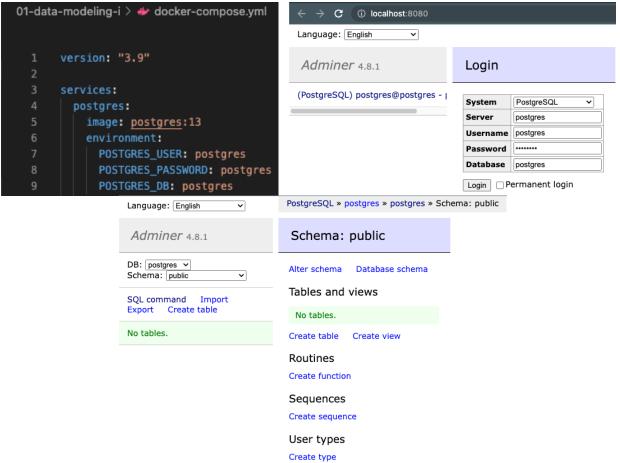
4. Install mandatory libraries from configuration file, named 'requirements.txt'

   ```
   1    numpy==1.23.2
   2    psycopg2==2.9.3
   3    python-dateutil==2.8.2
   4    pytz==2022.2.1
   5    six==1.16.0
   6    psycopg2-binary
   7
   ```
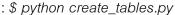
   : *$ pip install -r requirements.txt*

5. Start Docker with services 'Postgres' and 'Adminer' from YMAL file, named 'docker-compose.yml'

   : *$ docker-compose up*

   ```
   DOCKER                              ⋯

   ∨ CONTAINERS             ≡× ⚙ ↻ ⊘
      ∨ 🗗 01-data-modeling-i
         >  ▷ adminer:4.8.1 01-data-modeling-i_a...
         >  ▷ postgres:13 01-data-modeling-i_pos...
   ```
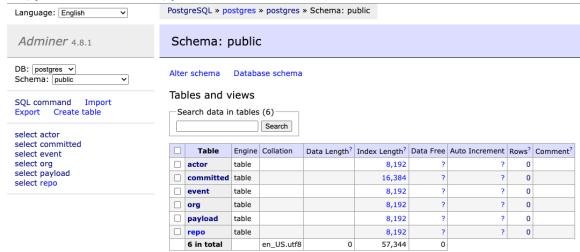
6. Connect Postgres thru Adminer
   o Open web browser 'http://localhost:8080/'
   o Login with the credential setup in 'docker-compose.yml'

7. Create tables thru python script, named 'create_tables.py'

   : *$ python create_tables.py*

8. Insert data thru python script, named 'etl.py'

    : *$ python etl.py*

Select: event

Select data   Show structure   Alter table   New item

Select   Search   Sort   Limit   Text length   Action
                         50   100   Select

SELECT * FROM "event" LIMIT 50 (0.001 s) Edit

☐ Modify	event_id	event_type	event_public	event_created_at	event_repo_id	event_actor_id	event_org_id	event_payload_push_id
☐ edit	23488014775	PushEvent	1	2022-08-17 15:55:05	520238237	100631798	78902940	10755001913
☐ edit	23488014835	PushEvent	1	2022-08-17 15:55:05	183910394	50076953	*NULL*	10755001961
☐ edit	23488014776	PushEvent	1	2022-08-17 15:55:05	525809133	266306	13507209	10755001931
☐ edit	23488014811	PushEvent	1	2022-08-17 15:55:05	494995936	533180	*NULL*	10755001954
☐ edit	23488014823	PublicEvent	1	2022-08-17 15:55:05	463636623	45897686	*NULL*	*NULL*
☐ edit	23488014790	CreateEvent	1	2022-08-17 15:55:05	525861687	111345762	*NULL*	*NULL*
☐ edit	23488014819	WatchEvent	1	2022-08-17 15:55:05	81810486	34191203	19199542	*NULL*
☐ edit	23488014824	DeleteEvent	1	2022-08-17 15:55:05	317845893	41286756	79963507	*NULL*
☐ edit	23488014774	CreateEvent	1	2022-08-17 15:55:05	525861731	104261999	*NULL*	*NULL*
☐ edit	23488014739	IssueCommentEvent	1	2022-08-17 15:55:05	177839194	105309205	6154722	*NULL*

9. [optional] Shutdown the environment

    a. Stop 'Postgres' and 'Adminer' services by shutdown Docker

        : *$ docker-compose down*

    b. Deactivate the visual environment 'ENV'

        : *$ deactivate*