Hash Partitioning

Let us understand how we can take care of Hash partitioning of tables.

- It is primarily used to create partitions based up on modulus and reminder.
- Here are the steps involved in creating table using hash partitioning strategy.
    - Create table using `PARTITION BY HASH`
    - Add default and remainder specific partitions based up on modulus.
    - Validate by inserting data into the table
- We can detach as well as drop the partitions from the table.
- Hash partitioning is typically done on sparse columns such as `user_id`.
- If we want to use hash partitioning on more than one tables with common key, we typically partition all the tables using same key.

Create Partitioned Table

Let us create partitioned table with name `users_hash_part`.

- It contains same columns as `users`.
- We will partition the table based up on `user_id` field.
- We will create one partition for each reminder with modulus 8.

```
%load_ext sql
```

```
The sql extension is already loaded. To reload it, use:
  %reload_ext sql
```

```
%env DATABASE_URL=postgresql://itversity_sms_user:sms_password@localhost:5432/itversity_sms_db
```

```
env:
DATABASE_URL=postgresql://itversity_sms_user:sms_password@localhost:5432/itversity_sms_db
```

```
%sql DROP TABLE IF EXISTS users_hash_part
```

```
 * postgresql://itversity_sms_user:***@localhost:5432/itversity_sms_db
Done.
```

```
[]
```

```
%%sql

CREATE TABLE users_hash_part (
    user_id SERIAL,
    user_first_name VARCHAR(30) NOT NULL,
    user_last_name VARCHAR(30) NOT NULL,
    user_email_id VARCHAR(50) NOT NULL,
    user_email_validated BOOLEAN DEFAULT FALSE,
    user_password VARCHAR(200),
    user_role VARCHAR(1) NOT NULL DEFAULT 'U', --U and A
    is_active BOOLEAN DEFAULT FALSE,
    created_dt DATE DEFAULT CURRENT_DATE,
    last_updated_ts TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    PRIMARY KEY (user_id)
) PARTITION BY HASH(user_id)
```

```
 * postgresql://itversity_sms_user:***@localhost:5432/itversity_sms_db
Done.
```

```
[]
```

> **ℹ Note**
>
> We will not be able to insert the data until we add at least one partition.