Managing Partitions - Range

Let us understand how to manage partitions for the table `users_range_part`.

- All users data created in a specific year should go to the respective partition created.
- For example, all users data created in the year of 2016 should go to `users_range_part_2016`.
- We can add partition to existing partitioned table using `CREATE TABLE partition_name PARTITION OF table_name`.
- We can have a partition for default values so that all the data that does not satisfy the partition condition can be added to it.
- We can have a partition for specific range of values using `FOR VALUES FROM (from_value) TO (to_value)` as part of `CREATE TABLE partition_name PARTITION OF table_name`.
- Once partitions are added, we can insert data into the partitioned table.

Note

Here is how we can create partition for default values for a range partitioned table **users_range_part**.

```
%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

CREATE TABLE users_range_part_default
PARTITION OF users_range_part DEFAULT
```

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

```
[]
```

```
%%sql

CREATE TABLE users_range_part_2016
PARTITION OF users_range_part
FOR VALUES FROM ('2016-01-01') TO ('2016-12-31')
```

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

```
[]
```

Error

As there is a overlap between the previous partition and below one, command to create partition for data ranging from 2016-01-01 till 2017-12-31 will fail.

```
%%sql

CREATE TABLE users_range_part_2017
PARTITION OF users_range_part
FOR VALUES FROM ('2016-01-01') TO ('2017-12-31')
```

```
 * postgresql://itversity_sms_user:***@localhost:5432/itversity_sms_db
(psycopg2.errors.InvalidObjectDefinition) partition "users_range_part_2017" would overlap
partition "users_range_part_2016"

[SQL: CREATE TABLE users_range_part_2017 PARTITION OF users_range_part
FOR VALUES FROM ('2016-01-01') TO ('2017-12-31')]
(Background on this error at: http://sqlalche.me/e/13/f405)
```

Note

This is how we can create partitions for the years **2017**, **2018**, **2019** etc

```
%%sql

CREATE TABLE users_range_part_2017
PARTITION OF users_range_part
FOR VALUES FROM ('2017-01-01') TO ('2017-12-31')
```

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

```
[]
```

```
%%sql

CREATE TABLE users_range_part_2018
PARTITION OF users_range_part
FOR VALUES FROM ('2018-01-01') TO ('2018-12-31')
```

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

```
[]
```

```
%%sql

CREATE TABLE users_range_part_2019
PARTITION OF users_range_part
FOR VALUES FROM ('2019-01-01') TO ('2019-12-31')
```

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

```
[]
```

```
%%sql

CREATE TABLE users_range_part_2020
PARTITION OF users_range_part
FOR VALUES FROM ('2020-01-01') TO ('2020-12-31')
```

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

```
[]
```

```
%%sql

INSERT INTO users_range_part
    (user_first_name, user_last_name, user_email_id, created_dt)
VALUES
    ('Scott', 'Tiger', 'scott@tiger.com', '2018-10-01'),
    ('Donald', 'Duck', 'donald@duck.com', '2019-02-10'),
    ('Mickey', 'Mouse', 'mickey@mouse.com', '2017-06-22')
```

```
 * postgresql://itversity_sms_user:***@localhost:5432/itversity_sms_db
3 rows affected.
```

```
[]
```

```
%%sql

SELECT user_first_name, user_last_name, user_email_id, created_dt
FROM users_range_part_default
```

```
 * postgresql://itversity_sms_user:***@localhost:5432/itversity_sms_db
0 rows affected.
```

user_first_name	user_last_name	user_email_id	created_dt

```
%%sql

SELECT user_first_name, user_last_name, user_email_id, created_dt
FROM users_range_part_2017
```

```
 * postgresql://itversity_sms_user:***@localhost:5432/itversity_sms_db
1 rows affected.
```

user_first_name	user_last_name	user_email_id	created_dt
Mickey	Mouse	mickey@mouse.com	2017-06-22

```
%%sql

SELECT user_first_name, user_last_name, user_email_id, created_dt
FROM users_range_part_2018
```

```
 * postgresql://itversity_sms_user:***@localhost:5432/itversity_sms_db
1 rows affected.
```

user_first_name	user_last_name	user_email_id	created_dt
Scott	Tiger	scott@tiger.com	2018-10-01

```
%%sql

SELECT user_first_name, user_last_name, user_email_id, created_dt
FROM users_range_part_2019
```

```
 * postgresql://itversity_sms_user:***@localhost:5432/itversity_sms_db
1 rows affected.
```

user_first_name	user_last_name	user_email_id	created_dt
Donald	Duck	donald@duck.com	2019-02-10

```
%%sql

SELECT user_first_name, user_last_name, user_email_id, created_dt
FROM users_range_part_2020
```

```
 * postgresql://itversity_sms_user:***@localhost:5432/itversity_sms_db
0 rows affected.
```

user_first_name	user_last_name	user_email_id	created_dt