

# Vacuuming Postgres Databases: Takeaways



by Dataquest Labs, Inc. - All rights reserved © 2021

## Syntax

- Deleting rows from table:

```
conn = psycopg2.connect(dbname="dq", user="hud_admin", password="abc123")
cur = conn.cursor()
cur.execute("DELETE FROM homeless_by_coc;")
```

- Removing dead rows from a table:

```
# Vacuum a single table
VACUUM homeless_by_coc;
```

OR

```
# Vacuum each user created table
VACUUM;
```

- Committing changes immediately:

```
conn = psycopg2.connect(dbname="dq", user="hud_admin", password="abc123")
conn.autocommit = True
```

- Updating table statistics:

```
VACUUM ANALYZE table_name;
```

- Reclaiming lost space for a table:

```
VACUUM FULL homeless_by_coc;
```

- Checking if autovacuum is on:

```
SELECT setting FROM pg_settings
WHERE name = 'autovacuum';
```

- Finding your last vacuum and autovacuum times:

```
SELECT last_vacuum, last_autovacuum FROM pg_stat_user_tables
WHERE relname = 'table_name';
```

## Concepts

- When running a `DELETE` query on a table, Postgres marks rows as dead, which means they will eventually be removed as opposed to removing them entirely.
- Postgres transactions follow a set of properties called ACID. ACID stands for:
  - Atomicity: If one thing fails in the transaction the whole transaction fails.
  - Consistency: A transaction will move the database from one valid set to another.
  - Isolation: concurrent effects to the database will be followed through as sequential changes.
  - Durability: Once the transaction is committed, it will stay that way regardless of crash, power outage, or some other catastrophic event.

- Postgres uses multi-version control that a user keeps a consistent version of her expected database state during the transaction.
- Vacuuming a table will remove the marked dead rows and reclaim the space they took from the table.
- No insert, update, or delete queries can be issued against the table during the vacuum duration with the `FULL` option. Select queries on the table are considerably slowed down to the point where they are unusable.
- Postgres offers a feature called autovacuum and it runs periodically on tables to ensure the dead rows are removed and your statistics are up-to-date.

## Resources

- [Postgres vacuum](#)
- [Autovacuum](#)