

**CATEGORY Database**

### **PostgreSQL Admin**

**Course Snapshot**

* **Course:** PostgreSQL Admin
* **Duration:** 3 days
* **Skill-level**: Basic level database skills for the skilled team members.
* **Targeted Audience**: This course is designed for Database Administrators -DBAs, Data Scientist and Data Analyst.
* **Hands-on Learning:** In this course, we will go over the basics of PostgreSQL. We will cover topics ranging from installations, to writing basic queries and retrieving data from tables. We will also explore the logic of joining tables to retrieve data and much more.
* **Delivery Format:** This course is available for onsite private classroom presentation.
* **Customizable:** This course may be tailored to target your specific training skills objectives, tools of choice and learning goals.

###### **What You Will Learn:**

* Install PostgreSQL Server
* Load sample database
* Create a database
* Create a table
* Insert data into tables
* Update existing records inside a table
* Delete Records in a table
* Remove duplicate records
* Query data from a table
* Create a subquery
* Get data from multiple tables
* Create and manage roles
* Create a view
* Create tablespace
* Backup and restore database
* Filter and sort data
* Use various operators

**Topics Covered**: This is a high-level list of topics covered in this course. Please see the detailed Agenda below

* First Steps
* Exploring the Database
* Configuration
* Server Control
* Tables and Data
* Security
* Database Administration
* Monitoring and Diagnosis
* Regular Maintenance
* Performance and Concurrency
* Backup and Recovery
* Replication and Upgrades

**Audience & Prerequisites**

This course will equip you with the skills and confidence to work with PostgreSQL .

**Pre-Requisites:** Students should have familiar with

* Basic understanding of database concept would help
* Basic understanding of SQL would help

**Course Agenda / Topics**

1. **First Steps**

* Getting PostgreSQL
* Connecting to PostgreSQL server
* Enabling access for network/remote users
* Using graphical administration tools
* Using psql query and scripting tool
* Changing your password securely
* Avoiding hard coding your password
* Using a connection service file
* Troubleshooting a failed connection

1. **Exploring the Database**

* What version is the server?
* What is the server uptime?
* Locate the database server files
* Locate the database server message log
* How many tables in a database?
* How much disk space does a database use?
* How much disk space does a table use?
* Which are my biggest tables?
* How many rows in a table?
* Understanding object dependencies

1. **Configuration**

* Reading the fine manual
* Planning a new database
* Changing parameters in your programs
* Finding the current configuration settings
* Updating the parameter file
* Setting parameters for particular groups of users
* The basic server configuration checklist
* Adding an external module to PostgreSQL
* Using an installed module
* Managing installed extensions

1. **Server Control**

* Starting the database server manually
* Stopping the server safely and quickly
* Reloading the server configuration files
* Restarting the server quickly
* Preventing new connections
* Pushing users off the system
* Deciding on a design for multi tenancy
* Using multiple schemas
* Giving users their own private database
* Running multiple servers on one system
* Setting up a connection pool
* Accessing multiple servers using the same host and port

1. **Tables and Data**

* Choosing good names for database objects
* Handling objects with quoted names
* Enforcing the same name and definition for columns
* Identifying and removing duplicates
* Preventing duplicate rows
* Finding a unique key for a set of data
* Generating test data
* Randomly sampling data
* Loading data from a spreadsheet
* Loading data from flat files

1. **Security**

* The PostgreSQL superuser
* Revoking user access to a table
* Granting user access to a table
* Granting user access to specific columns and rows
* Creating a new user
* Temporarily preventing a user from connecting
* Removing a user without dropping their data
* Giving limited superuser powers to specific users
* Auditing database access
* Integrating with LDAP
* Connecting using SSL
* Using SSL certificates to authenticate
* Encrypting sensitive data

1. **Database Administration**

* Writing a script that either succeeds entirely or fails entirely
* Using psql variables
* Placing query output into psql variables
* Writing a conditional psql script
* Investigating a psql error
* Performing actions on many tables
* Adding/removing columns on a table
* Changing the data type of a column
* Adding/removing schemas
* Moving objects between schemas
* Moving objects between tablespaces
* Accessing objects in other PostgreSQL databases
* Accessing objects in other foreign databases
* Updatable views
* Using materialized views

1. **Monitoring and Diagnosis**

* Providing PostgreSQL information to monitoring tools
* Real-time viewing using pgAdmin or OmniDB
* Checking whether a user is connected
* Checking whether a computer is connected
* Repeatedly executing a query in psql
* Checking which queries are active or blocked
* Knowing who is blocking a query
* Knowing whether anybody is using a specific table
* Knowing when a table was last used
* Usage of disk space by temporary data
* Understanding why queries slow down
* Investigating and reporting a bug

1. **Regular Maintenance**

* Controlling automatic database maintenance
* Avoiding auto-freezing and page corruptions
* Removing issues that cause bloat
* Removing old prepared transactions
* Identifying and fixing bloated tables and indexes
* Monitoring and tuning vacuum
* Maintaining indexes
* Adding a constraint without checking existing rows
* Finding unused indexes
* Planning maintenance

1. **Performance and Concurrency**

* Finding slow SQL statements
* Finding out what makes SQL slow
* Reducing the number of rows returned
* Simplifying complex SQL queries
* Discovering why a query is not using an index
* Forcing a query to use an index
* Using parallel query
* Using optimistic locking

1. **Backup and Recovery**

* Understanding and controlling crash recovery
* Planning backups
* Hot logical backups of one database
* Backups of database object definitions
* Standalone hot physical database backup
* Hot physical backup and continuous archiving
* Recovery of all databases
* Recovery of a dropped/damaged database
* Improving performance of backup/recovery
* Incremental/differential backup and restore
* Hot physical backups with Barman
* Recovery with Barman

1. **Replication and Upgrades**

* Replication concepts
* Setting up file-based replication – deprecated
* Setting up streaming replication
* Hot Standby and read scalability
* Managing streaming replication
* Using repmgr
* Using replication slots
* Monitoring replication
* Delaying, pausing, and synchronizing replication
* Logical replication
* Bi-directional replication
* Archiving transaction log data
* Upgrading minor releases

**Student Materials:** Each student will receive a **Student Guide** with course notes, code samples, software tutorials, diagrams and related reference materials and links (as applicable). Our courses also include step by step hands-on lab instructions and solutions, clearly illustrated for users to complete hands-on work in class, and to revisit to review or refresh skills at any time. Students will also receive the project files (or code, if applicable) and solutions required for the hands-on work.