

## Quick Links

- [About](#)
- [Policies](#)
- [Feature Matrix](#)
- [Donate](#)
- [History](#)
- [Sponsors](#)
  - [Servers](#)
- [Latest News](#)
- [Upcoming Events](#)
  - [Past events](#)
- [Press](#)
- [Licence](#)

## About

### What is PostgreSQL?

PostgreSQL is a powerful, open source object-relational database system that uses and extends the SQL language combined with many features that safely store and scale the most complicated data workloads. The origins of PostgreSQL date back to 1986 as part of the [POSTGRES](#) project at the University of California at Berkeley and has more than 30 years of active development on the core platform.

PostgreSQL has earned a strong reputation for its proven architecture, reliability, data integrity, robust feature set, extensibility, and the dedication of the open source community behind the software to consistently deliver performant and innovative solutions. PostgreSQL runs on [all major operating systems](#), has been [ACID](#)-compliant since 2001, and has powerful add-ons such as the popular [PostGIS](#) geospatial database extender. It is no surprise that PostgreSQL has become the open source relational database of choice for many people and organisations.

[Getting started](#) with using PostgreSQL has never been easier - pick a project you want to build, and let PostgreSQL safely and robustly store your data.

### Why use PostgreSQL?

PostgreSQL comes with [many features](#) aimed to help developers build applications, administrators to protect data integrity and build fault-tolerant environments, and help you manage your data no matter how big or small the dataset. In addition to being [free and open source](#), PostgreSQL is highly extensible. For example, you can define your own data types, build out custom functions, even write code from [different programming languages](#) without recompiling your database!

PostgreSQL tries to conform with the [SQL standard](#) where such conformance does not contradict traditional features or could lead to poor architectural decisions. Many of the features required by the SQL standard are supported, though sometimes with slightly differing syntax or function. Further moves towards conformance can be expected over time. As of the version 14 release in September 2021, PostgreSQL conforms to at least 170 of the 179 mandatory features for SQL:2016 Core conformance. As of this writing, no relational database meets full conformance with this standard.

Below is an inexhaustive list of various features found in PostgreSQL, with more being added in every [major release](#):

- **Data Types**
  - Primitives: Integer, Numeric, String, Boolean
  - Structured: Date/Time, Array, Range / Multirange, UUID
  - Document: JSON/JSONB, XML, Key-value (Hstore)
  - Geometry: Point, Line, Circle, Polygon
  - Customizations: Composite, Custom Types
- **Data Integrity**
  - UNIQUE, NOT NULL
  - Primary Keys
  - Foreign Keys
  - Exclusion Constraints
  - Explicit Locks, Advisory Locks
- **Concurrency, Performance**
  - Indexing: B-tree, Multicolumn, Expressions, Partial
  - Advanced Indexing: GiST, SP-GiST, KNN-GiST, GIN, BRIN, Covering indexes, Bloom filters

- Advanced indexing: GIST, SP-GIST, RINN GIST, GIN, BRIN, COVERING INDEXES, BLOOM FILTERS
- Sophisticated query planner / optimizer, index-only scans, multicolumn statistics
- Transactions, Nested Transactions (via savepoints)
- Multi-Version concurrency Control (MVCC)
- Parallelization of read queries and building B-tree indexes
- Table partitioning
- All transaction isolation levels defined in the SQL standard, including Serializable
- Just-in-time (JIT) compilation of expressions
- **Reliability, Disaster Recovery**
  - Write-ahead Logging (WAL)
  - Replication: Asynchronous, Synchronous, Logical
  - Point-in-time-recovery (PITR), active standbys
  - Tablespaces
- **Security**
  - Authentication: GSSAPI, SSPI, LDAP, SCRAM-SHA-256, Certificate, and more
  - Robust access-control system
  - Column and row-level security
  - Multi-factor authentication with certificates and an additional method
- **Extensibility**
  - Stored functions and procedures
  - Procedural Languages: PL/PGSQL, Perl, Python (and many more)
  - SQL/JSON path expressions
  - Foreign data wrappers: connect to other databases or streams with a standard SQL interface
  - Customizable storage interface for tables
  - Many extensions that provide additional functionality, including PostGIS
- **Internationalisation, Text Search**
  - Support for international character sets, e.g. through ICU collations
  - Case-insensitive and accent-insensitive collations
  - Full-text search

There are many more features that you can discover in the PostgreSQL [documentation](#). Additionally, PostgreSQL is highly extensible: many features, such as indexes, have defined APIs so that you can build out with PostgreSQL to solve your challenges.

PostgreSQL has been proven to be highly scalable both in the sheer quantity of data it can manage and in the number of concurrent users it can accommodate. There are active PostgreSQL clusters in production environments that manage many terabytes of data, and specialized systems that manage petabytes.

## Any questions?

The first place to go to for any questions on PostgreSQL is its [world-renowned documentation](#) which discusses how to use the database software in-depth.

We also have many [mailing lists](#) where you can connect and participate in the [community](#), as well as many [events](#) and [local user groups](#) where you can connect with other users.

## Our users ❤️ us



We have normally had a fix within 24 hours. Frankly, the [mailing list] support has been amazing.

Kevin Grittner, [Wisconsin Court System](#)



## What's a database project without statistics?

30+ Years  
Development

650+ Contributors

52,000+ Commits

55+ Local User Groups

1,500,000+ Lines of C      650+ Events      Millions of Happy Users      ∞ Data Stored



[Policies](#) | [Code of Conduct](#) | [About PostgreSQL](#) | [Contact](#)

Copyright © 1996-2022 The PostgreSQL Global Development Group