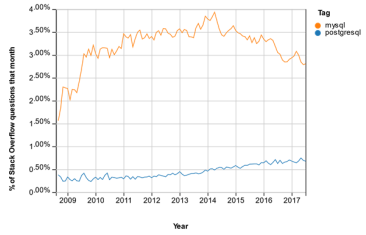
	PostgreSQL	MySQL
Known as	The world's most advanced open source database	The world's most popular open source database
Development	PostgreSQL is an open source project	MySQL is an open-source product
Pronunciation	post gress queue ell	my ess queue ell
Licensing	MIT-style license	GNU General Public License
Implementation programming language	C	C/C++
GUI tool	PgAdmin	MySQL Workbench
SQL compliant	PostgreSQL is largely SQL compliant.	Partly
ACID compliance	Yes	MySQL is ACID compliant only when it is used with InnoDB and NDB Cluster Storage engines.
SQL compliant	PostgreSQL is largely SQL compliant.	MySQL is partially SQL compliant. For example, it does not support check constraint.
Storage engine	Single storage engine	Multiple storage engines e.g., InnoDB and MyISAM, InMem, ...
Full-text search	Yes	Yes
Drop a temporary table	No TEMP or TEMPORARY keyword in DROP TABLE statement	MySQL supports the TEMP or TEMPORARY keyword in the DROP TABLE statement that allows you to remove the temporary table only.
DROP TABLE	Support CASCADE option to drop table's dependent objects e.g., tabl	Does not support CASCADE option
TRUNCATE TABLE	PostgreSQL TRUNCATE TABLE supports more features like CASCADE	MySQL TRUNCATE TABLE does not support CASCADE and transaction safe i.e., once data is deleted, it cannot be rolled back.
Auto increment Column	SERIAL	AUTO_INCREMENT
Analytic functions	Yes	No
Data types	Support many advanced types such as array, hstore, and user-define	SQL-standard types
Unsigned integer	No	Yes
Boolean type	Yes	Use TINYINT(1) internally for Boolean
IP address data type	Yes	No
Set default value for a column	Support both constant and function call	Must be a constant or CURRENT_TIMESTAMP for TIMESTAMP or DATETIME columns
EXPLAIN output	More detailed	Less detailed
Window Function	Yes	Yes. (since version 8.0)
Materialized views	Yes	No
CHECK constraint	Yes	No (MySQL ignores the CHECK constraint)
Table inheritance	Yes	No
Programming languages for stored procedures	Ruby, Perl, Python, TCL, PL/pgSQL, SQL, JavaScript, etc.	SQL:2003 syntax for stored procedures
FULL OUTER JOIN	Yes	No
INTERSECT	Yes	No
EXCEPT	Yes	No
Partial indexes	Yes	No
Bitmap indexes	Yes	No
Expression indexes	Yes	No
Covering indexes	Yes (since version 9.2)	Yes. MySQL supports covering indexes that allow data to be retrieved by scanning the index alone without touching the table data. This is advantageous in case of large tables with millions of rows.
Common table expression (CTE)	Yes	Yes. (since version 8.0)
Triggers	Support triggers that can fire on most types of command, except for	Limited to some commands
Partitioning	RANGE, LIST	RANGE, LIST, HASH, KEY, and composite partitioning using a combination of RANGE or LIST with HASH or KEY subpartitions
Task Schedule	pgAgent	Scheduled event
Connection Scalability	Each new connection is an OS process	Each new connection is an OS thread



	ORDBMS	An ORDBMS is primarily a relational database that supports some object oriented features. PostgreSQL or Postgres (not PostGres) supports table inheritance and function overloading. Both are features usually attributed to object oriented languages.									
	MVCC	One of the big selling points of Postgres is how it handles concurrency. The promise is simple: reads never block writes and vice versa. Postgres achieves this via a mechanism called Multi Version Concurrency Control. This technique is not unique to Postgres: there are several databases that implement some form of MVCC including Oracle, Berkeley DB, CouchDB and many more.									

MySQL and PostgreSQL have similar data types

MySQL	PostgreSQL
BIGINT	BIGINT
BINARY(n)	BYTEA
BIT	BOOLEAN
CHAR(n), CHARACTER(n)	CHAR(n), CHARACTER(n)
DATE	DATE
DATETIME	TIMESTAMP [WITHOUT TIME ZONE]
DECIMAL(p,s), DEC(p,s)	DECIMAL(p,s), DEC(p,s)
DOUBLE	DOUBLE PRECISION
FLOAT	REAL
INT, INTEGER	INT, INTEGER
MEDIUMINT	INTEGER
NUMERIC(p,s)	NUMERIC(p,s)
SMALLINT	SMALLINT
TINYBLOB, BLOB, MEDIUMBLOB, LONGBLOB	BYTEA
TINYINT	SMALLINT
TINYTEXT, TEXT, MEDIUMTEXT, LONGTEXT	TEXT
TIME	TIME [WITHOUT TIME ZONE]
TIMESTAMP	TIMESTAMP [WITHOUT TIME ZONE]
VARBINARY(n), VARBINARY(max)	BYTEA
VARCHAR(n)	VARCHAR(n)
VARCHAR(max)	TEXT
	UUID for storing Universally Unique Identifiers **Special Type**
	Array for storing array strings, numbers, etc.
	JSON stores JSON data
	hstore stores key-value pair
	TIMESTAMPTZ is a timezone-aware timestamp data type. It is the abbreviation for timestamp with the time zone.
	INTERVAL stores periods of time.
	XML
	box– a rectangular box.
	line – a set of points.
	point– a geometric pair of numbers.
	lseg– a line segment.
	polygon– a closed geometric.
	inet– an IP4 address.
	macaddr– a MAC address.

MySQL has an argument for integer-like columns called 'auto_increment', which increases the value of the field automatically each time when new row is inserted. PostgreSQL uses SERIAL type which will automatically generate and populate values into the SERIAL column.

MySQL	PostgreSQL
BIGINT AUTO_INCREMENT	BIGSERIAL
INTEGER AUTO_INCREMENT	SERIAL
SMALLINT AUTO_INCREMENT	SMALLSERIAL
TINYINT AUTO_INCREMENT	SMALLSERIAL

Unlike PostgreSQL, all MySQL integer types (tinyint, smallint, int, bigint) can have UNSIGNED attribute. Unsigned specification forces to take positive numbers only with larger upper range of acceptable values. Here is how MySQL unsigned types have to be mapped into PostgreSQL:

MySQL	PostgreSQL
BIGINT UNSIGNED	NUMERIC(20)
INT UNSIGNED	BIGINT
MEDIUMINT UNSIGNED	INTEGER
SMALLINT UNSIGNED	INTEGER
TINYINT UNSIGNED	INTEGER

	MySQL	**PostgreSQL**
B-tree indexes can be used for equality and range queries efficiently. They can operate against all datatypes, and can also be used to retrieve NULL values. B-trees are designed to work very well with caching, even when only partially cached.	X	X
Hash Indexes are only useful for equality comparisons	X (MyISAM only)	X
GINs are good for indexing array values as well as for implementing full-text search.		X
Generalized Search Tree (GiST) indexes can be used for operations beyond equality and range comparisons. They are used to index the geometric data types, as well as full-text search.		X

	PostgreSQL is an **Object** Relational Database Management System (ORDBMS) whereas MySQL is a community driven DBMS system.
	PostgreSQL is complete **ACID** compliant while MySQL is only ACID compliant when used with InnoDB and NDB.
	PostgreSQL support modern applications feature like **JSON**, **XML** etc. while MySQL only supports JSON.
	PostgreSQL performance well when executing complex queries
	PostgreSQL has a lot of data types which MySQL not support
	PostgreSQL has a lot of index types which MySQL not support

	https://www.postgresqltutorial.com/postgresql-vs-mysql/				
	Data Type				
	https://www.postgresqltutorial.com/postgresql-data-types/				
	https://www.slideshare.net/jkatz05/on-beyond-postgresql-data-types				
	https://www.convert-in.com/mysql-to-postgres-types-mapping.htm				
	Index				
	https://www.slideshare.net/jkatz05/explain-the-index-of-postgresql-indexes				