

Cloudera Runtime 7.2.11

# Apache Phoenix SQL Command Reference

Date published: 2020-02-29

Date modified: 2021-09-09

The Cloudera logo is displayed in a bold, orange, sans-serif font. The word "CLOUDERA" is written in all caps, with a stylized 'E' that has three horizontal bars.

<https://docs.cloudera.com/>

# Legal Notice

© Cloudera Inc. 2026. All rights reserved.

The documentation is and contains Cloudera proprietary information protected by copyright and other intellectual property rights. No license under copyright or any other intellectual property right is granted herein.

Unless otherwise noted, scripts and sample code are licensed under the Apache License, Version 2.0.

Copyright information for Cloudera software may be found within the documentation accompanying each component in a particular release.

Cloudera software includes software from various open source or other third party projects, and may be released under the Apache Software License 2.0 (“ASLv2”), the Affero General Public License version 3 (AGPLv3), or other license terms. Other software included may be released under the terms of alternative open source licenses. Please review the license and notice files accompanying the software for additional licensing information.

Please visit the Cloudera software product page for more information on Cloudera software. For more information on Cloudera support services, please visit either the Support or Sales page. Feel free to contact us directly to discuss your specific needs.

Cloudera reserves the right to change any products at any time, and without notice. Cloudera assumes no responsibility nor liability arising from the use of products, except as expressly agreed to in writing by Cloudera.

Cloudera, Cloudera Altus, HUE, Impala, Cloudera Impala, and other Cloudera marks are registered or unregistered trademarks in the United States and other countries. All other trademarks are the property of their respective owners.

Disclaimer: EXCEPT AS EXPRESSLY PROVIDED IN A WRITTEN AGREEMENT WITH CLOUDERA, CLOUDERA DOES NOT MAKE NOR GIVE ANY REPRESENTATION, WARRANTY, NOR COVENANT OF ANY KIND, WHETHER EXPRESS OR IMPLIED, IN CONNECTION WITH CLOUDERA TECHNOLOGY OR RELATED SUPPORT PROVIDED IN CONNECTION THEREWITH. CLOUDERA DOES NOT WARRANT THAT CLOUDERA PRODUCTS NOR SOFTWARE WILL OPERATE UNINTERRUPTED NOR THAT IT WILL BE FREE FROM DEFECTS NOR ERRORS, THAT IT WILL PROTECT YOUR DATA FROM LOSS, CORRUPTION NOR UNAVAILABILITY, NOR THAT IT WILL MEET ALL OF CUSTOMER’S BUSINESS REQUIREMENTS. WITHOUT LIMITING THE FOREGOING, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, CLOUDERA EXPRESSLY DISCLAIMS ANY AND ALL IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, QUALITY, NON-INFRINGEMENT, TITLE, AND FITNESS FOR A PARTICULAR PURPOSE AND ANY REPRESENTATION, WARRANTY, OR COVENANT BASED ON COURSE OF DEALING OR USAGE IN TRADE.

# Contents

<b>Apache Phoenix SQL command reference.....</b>	<b>4</b>
--	----------

## Apache Phoenix SQL command reference

You can use these data definition language (DDL) and data manipulation language (DML) commands to perform database operations on Apache Phoenix tables.

**Table 1: Apache Phoenix commands**

Command	Description	Simple example
CREATE TABLE	Creates a new table	<pre>CREATE TABLE your_schema.TEST_TABLE ( id BIGINT not null primary key, date Date);</pre>
DROP TABLE	Drops an existing table	<pre>DROP TABLE your_schema.your_table;</pre>
CREATE VIEW	Creates a new view over an existing Apache HBase or Phoenix table	<pre>CREATE VIEW "hbase table" ( k VARCHAR primary key, "h" UNSIGNED_LONG) default_column_family='cf1';</pre>
DROP VIEW	Drops an existing view. When you drop a view, the underlying table data is not affected; the index data for the view is deleted	<pre>DROP VIEW your_view;</pre>
SELECT	Selects data from one or more tables	<pre>SELECT * FROM TEST_TABLE LIMIT 1000;</pre>
UPSERT VALUES	Inserts value if the value is not present in the table and updates the value in the table if the value is already present. The list of columns is optional and if not present, the data is mapped to the column in the order they are declared in the schema	<pre>UPSERT INTO TEST_TABLE VALUES( 'A', 'B', 4 );</pre>
UPSERT SELECT	Inserts value if the value is not already present and updates the rows in the table based on the results of running the next query	<pre>UPSERT INTO test.targetTable(cq1, cq2) SELECT cq3, cq4 FROM test.sourceTable WHERE cq5 &lt; 100</pre>