



brijpandeyji



# SQL & NoSQL

## Cheat Sheet

Your Go-To **Quick**  
Reference Guide  
for **SQL & NoSQL**



swipe >

# Querying Data

1

## SELECT

Retrieve data from one or more tables.

2

## DISTINCT

Select distinct values in a column.

3

## WHERE

Filter rows using a condition.

# Querying Data

4

## AND, OR, NOT

Combine multiple conditions.

5

## ORDER BY

Sort the result set by a column.

6

## LIMIT & OFFSET

Limit the number of rows and skip rows.

# Querying Data

7

## GROUP BY

Group rows with the same values in specified columns.

8

## HAVING

Filter the results of a GROUP BY.

# Joining Tables

1

## INNER JOIN

Combine rows from related tables based on a condition.

2

## LEFT JOIN

Return all rows from the left table and the matched rows from the right table.

3

## RIGHT JOIN

Return all rows from the right table and the matched rows from the left table.

# Joining Tables

4

## FULL JOIN

Return all rows when there's a match in either table.

5

## CROSS JOIN

Return the Cartesian product of two tables.

6

## SELF JOIN

Join a table to itself using aliases.

swipe >

# Aggregating & Analyzing Data

## 1 COUNT()

Count the number of rows.

2

## SUM()

Calculate the sum of a column.

3

## AVG()

Calculate the average of a column.

# Aggregating & Analyzing Data

1

## MIN() & MAX()

Find the minimum and maximum value in a column.

2

## GROUP\_CONCAT()

Concatenate values from a group.

swipe >

# SQL Functions & Expressions

1

## COALESCE()

Return the first non-null value from a list.

2

## NULLIF()

Return null if two expressions are equal.

3

## CASE

Perform conditional logic in SQL queries.

swipe >

# SQL Functions & Expressions

1

## CAST()

Convert a value to a specified data type.

2

## CONCAT()

Concatenate two or more strings.

# Working with Tables

1

## CREATE TABLE

Create a new table.

2

## ALTER TABLE

Modify an existing table.

3

## DROP TABLE

Remove a table.

# Working with Tables

4

## TRUNCATE TABLE

Remove all rows from a table without deleting the table structure.

5

## RENAME TABLE

Rename a table.

swipe >

# Constraints & Indexes

1

## PRIMARY KEY

Uniquely identify each row in a table.

2

## FOREIGN KEY

Ensure referential integrity between two tables.

3

## UNIQUE

Ensure unique values in a column.

# Constraints & Indexes

4

## CHECK

Ensure that all values in a column satisfy a condition.

5

## DEFAULT

Set a default value for a column.

6

## NOT NULL

Ensure a column cannot contain NULL values.

swipe >

# Constraints & Indexes

7

## CREATE INDEX

Create an index on a table.

8

## DROP INDEX

Remove an index.

swipe >

# Views, Stored Procedures & Triggers

1

## CREATE VIEW

Create a virtual table based on a SELECT statement.

2

## DROP VIEW

Remove a view.

3

## CREATE PROCEDURE

Create a stored procedure.

# Views, Stored Procedures & Triggers

4

## EXECUTE

Run a stored procedure.

5

## DROP PROCEDURE

Remove a stored procedure.

6

## CREATE TRIGGER

Create a trigger that executes a specified action when an event occurs.

swipe >

# Views, Stored Procedures & Triggers

7

## DROP TRIGGER

Remove a trigger.

# User Management

1

## CREATE USER

Create a new user.

2

## DROP USER

Remove a user.

3

## ALTER USER

Change the password of a user.

swipe >

# User Management

4

## GRANT

Give a user access to specific privileges.

5

## REVOKE

Remove a user's access to specific privileges.

brijpandeyji

# Database Specific Commands

## SQL Server

1

### IDENTITY

Auto-increment a column's value.

2

### TOP

Limit the number of rows returned.

swipe >

# Oracle

## 1 ROWNUM

Limit the number of rows returned.

## 2 SEQUENCE

Create a sequence for generating unique numbers.

## 3 NEXTVAL

Get the next value of a sequence.

# PostgreSQL

1

## SERIAL

Auto-increment a column's value.

2

## LIMIT & OFFSET

Limit the number of rows returned and skip rows.

# MySQL

1

## AUTO\_INCREMENT

Auto-increment a column's value.

2

## LIMIT & OFFSET

Limit the number of rows returned and skip rows.

3

## SHOW TABLES

List all tables in the current database

brijpandeyji

# MySQL

4

## DESCRIBE TABLE

Display a table's structure.

swipe >

brijpandeyji



# NoSQL Cheat Sheet

## NoSQL Database Types

1

### Key-Value

Stores data as key-value pairs (e.g., Redis, Riak).

2

### Document

Stores data as documents, typically in JSON format (e.g., MongoDB, Couchbase).

swipe >

# NoSQL Database Types

3

## Column-Family

Stores data in columns grouped together as column families (e.g., Cassandra, HBase).

4

## Graph

Stores data as nodes and edges in a graph (e.g., Neo4j, Amazon Neptune)

# CRUD Operations

1

## CREATE

Add a new item to the database.

2

## READ

Retrieve an item or items from the database.

3

## UPDATE

Modify an existing item in the database.

swipe >

# CRUD Operations

4

## DELETE

Remove an item from the database

swipe >

brijpandeyji

# MongoDB Specific

1

**db.collection.insertOne()**

Insert a single document.

2

**db.collection.insertMany()**

Insert multiple documents.

3

**db.collection.find()**

Query documents.

swipe >

# MongoDB Specific

4

**db.collection.findOne()**

Query a single document.

5

**db.collection.updateOne()**

Update a single document.

6

**db.collection.updateMany()**

Update multiple documents.

# MongoDB Specific

7

**db.collection.deleteOne()**

Delete a single document.

8

**db.collection.deleteMany()**

Delete multiple documents.

9

**db.collection.createIndex()**

Create an index.

brijpandeyji

# Cassandra Specific

1

## CREATE KEYSPACE

Create a new keyspace.

2

## CREATE TABLE

Create a new table.

3

## INSERT INTO

Insert data into a table.

swipe >

# Cassandra Specific

**SELECT**

Retrieve data from a table.

4

5

**UPDATE**

Update data in a table.

6

**DELETE**

Remove data from a table.

swipe >

# Cassandra Specific

7

## ALTER TABLE

Modify an existing table.

8

## DROP TABLE

Delete a table.

brijpandeyji

# Neo4j Specific

**CREATE (n)**

Create a new node.

1

2

**CREATE (a)-  
[r:REL\_TYPE]->(b)**

Create a new relationship  
between nodes.

3

**MATCH (n)**

Query nodes.

swipe >

# Neo4j Specific

4

**MATCH (a)-  
[r:REL\_TYPE]->(b)**

Query relationships.

5

**SET n.property = value**

Update a node's property

6

**SET r.property =  
value**

Update a relationship's  
property.

swipe >

# Neo4j Specific

7

**DETACH DELETE n**

Delete a node and its relationships.

8

**DELETE r**

Delete a relationship.

# Redis Specific

1

## SET Key Value

Set the value of a key.

2

## GET Key

Get the value of a key.

3

## DEL Key

Delete a key.

# Redis Specific

4

## EXISTS Key

Check if a key exists.

5

## INCR Key

Increment the integer value of a key.

6

## DECR Key

Decrement the integer value of a key.

# Redis Specific

7

## LPUSH Key Value

Prepend a value to a list.

8

## RPUSH Key Value

Append a value to a list.

9

## LPOP Key

Remove and return the first element of a list.

# Redis Specific

10

## RPOP Key

Remove and return the last element of a list.

11

## SADD Key Value

Add a value to a set.

12

## SREM Key Value

Remove a value from a set.

# Redis Specific

13

## **SMEMBERS Key**

Get all members of a set.

14

## **HSET Key Field Value**

Set the value of a field in a hash.

15

## **HGET Key Field**

Get the value of a field in a hash.

swipe >

# Redis Specific

16

## HDEL Key Field

Delete a field from a hash.

17

## HGETALL Key

Get all fields and values of a hash.

18

## EXPIRE Key Seconds

Set a key's time to live in seconds.

swipe >

# Redis Specific

19

## TTL Key

Get the remaining time to live of a key.

swipe >

# Couchbase Specific

1

## CREATE BUCKET

Create a new bucket.

2

## INSERT

Insert a document into a bucket.

3

## SELECT

Query documents using N1QL (SQL-like query language).

swipe >

# Couchbase Specific

4

## UPDATE

Update a document.

5

## DELETE

Remove a document from a bucket.

6

## CREATE INDEX

Create an index for efficient querying.

swipe >

# Couchbase Specific

7

## DROP INDEX

Remove an index.

8

## UPsert

Insert or update a document.

brijpandeyji [in](#) [tw](#)

# Amazon DynamoDB Specific

## 1 CreateTable

Create a new table.

## 2 DeleteTable

Remove a table.

## 3 PutItem

Insert an item into a table.

swipe >

# Amazon DynamoDB Specific

4

## UpdateItem

Modify an item in a table.

5

## GetItem

Retrieve an item from a table.

6

## DeleteItem

Remove an item from a table.

swipe >

# Amazon DynamoDB Specific

7

## Query

Query items based on a condition.

8

## Scan

Scan a table and retrieve items.

swipe >



brijpandeyji  

## For More Interesting Content



**Brij Kishore Pandey**



**Follow Me On  
LinkedIn**

<https://www.linkedin.com/in/brijpandeyji/>