

Milvus Cheatsheet

Connection

pymilvus.connections.connect(host='localhost', port='19530')
Connect to a Milvus server using Python client

pymilvus.connections.connect(alias='default', uri='http://localhost:19530')
Connect using URI format

pymilvus.connections.disconnect('default')
Disconnect from Milvus server

pymilvus.connections.list_connections()
List all connections

pymilvus.connections.has_connection('default')
Check if a connection exists

Collection Management

pymilvus.Collection('collection_name')
Get a collection instance

pymilvus.utility.list_collections()
List all collections

pymilvus.has_collection('collection_name')
Check if a collection exists

collection.drop()
Drop a collection

collection.create_index('field_name', index_params)
Create an index

collection.has_index('field_name')
Check if a field is indexed

collection.drop_index('field_name')
Drop an index

Schema Definition

pymilvus.FieldSchema('id', pymilvus.DataType.INT64, is_primary=True)
Define a primary key field

pymilvus.FieldSchema('vector', pymilvus.DataType.FLOAT_VECTOR, dim=128)
Define a vector field

pymilvus.CollectionSchema(fields=[field1, field2, ...])
Create a schema with multiple fields

pymilvus.Collection('name', schema)
Create a collection with schema

Data Operations

collection.insert(data)
Insert data into collection

collection.delete('id in [1, 2, 3]')
Delete entities by condition

collection.load()
Load collection into memory

collection.release()
Release collection from memory

collection.flush()
Flush inserted data to storage

Search and Query

collection.search(vectors, 'field_name', search_params, limit=10, expr="")
Vector similarity search

collection.query('id > 100', output_fields=['field1', 'field2'])
Query entities by condition

collection.get_query_segment_info()
Get query segment information

Partitions

collection.create_partition('partition_name')
Create a partition

collection.has_partition('partition_name')
Check if a partition exists

collection.drop_partition('partition_name')
Drop a partition

collection.partitions
Get all partitions

Metrics and Monitoring

collection.num_entities
Get number of entities in collection

pymilvus.utility.get_server_version()
Get Milvus server version

pymilvus.utility.get_server_status()
Check server status