



FREE eBook

LEARNING couchbase

Free unaffiliated eBook created from
Stack Overflow contributors.

#couchbase

Table of Contents

About.....	1
Chapter 1: Getting started with couchbase.....	2
Remarks.....	2
Examples.....	2
Installation or Setup.....	2
Chapter 2: C# SDK.....	3
Syntax.....	3
Examples.....	3
Connecting to a Bucket.....	3
Insert Document Sync.....	4
Adding the SDK to a project.....	5
Chapter 3: Connect to couchbase over SSL using SDK.....	6
Introduction.....	6
Examples.....	6
Secure connection to couchbase using java sdk with specific cipher suites and protocols.....	6
Chapter 4: Java SDK.....	8
Examples.....	8
Adding the SDK to a project.....	8
Connecting to a Bucket.....	8
Checking for document exists in DB.....	8
Creating document with TTL (Time To Live).....	9
Credits.....	10

About

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [couchbase](#)

It is an unofficial and free couchbase ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official couchbase.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapter 1: Getting started with couchbase

Remarks

This section provides an overview of what couchbase is, and why a developer might want to use it.

It should also mention any large subjects within couchbase, and link out to the related topics. Since the Documentation for couchbase is new, you may need to create initial versions of those related topics.

Examples

Installation or Setup

Detailed instructions on getting couchbase set up or installed.

Read **Getting started with couchbase online**: <https://riptutorial.com/couchbase/topic/1233/getting-started-with-couchbase>

Chapter 2: C# SDK

Syntax

- `new Cluster();` // Connect and bootstrap from the local cluster (localhost)
- `new Cluster(ClientConfiguration configuration)` // Use the specified custom configuration to connect to the cluster
- `new Cluster(string configurationSectionName)` // Use the specified configuration section from `app.config / web.config`
- `OpenBucket();` // Open the default bucket
- `OpenBucket(string bucketname);` // Open the specified bucket
- `OpenBucket(string bucketname, string password);` // Open the specified bucket with the provided password

Examples

Connecting to a Bucket

Configuring the connection programmatically:

```
var config = new ClientConfiguration
{
    Servers = new List<Uri> {
        new Uri("http://localhost:8091/pools")
    },
    BucketConfigs = new Dictionary<string, BucketConfiguration>
    {
        { "default", new BucketConfiguration
        {
            BucketName = "default",
            UseSsl = false,
            Password = "",
            DefaultOperationLifespan = 2000,
            PoolConfiguration = new PoolConfiguration
            {
                MaxSize = 10,
                MinSize = 5,
                SendTimeout = 12000
            }
        }
    }
};

var cluster = new Cluster(config);
var bucket = cluster.OpenBucket();
```

Configuring the connection in `web.config / app.config`:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
```

```

<configSections>
  <sectionGroup name="couchbaseClients">
    <section name="couchbase"
type="Couchbase.Configuration.Client.Providers.CouchbaseClientSection, Couchbase.NetClient" />
    </sectionGroup>
  </configSections>
  <couchbaseClients>
    <couchbase useSsl="false">
      <servers>
        <add uri="http://localhost:8091/pools"></add>
      </servers>
      <buckets>
        <add name="default" useSsl="false" password="">
          <connectionPool name="custom" maxSize="10" minSize="5" sendTimeout="12000"
/>
          </add>
        </buckets>
      </couchbase>
    </couchbaseClients>
  </configuration>

```

Using the config section:

```

var cluster = new Cluster("couchbaseClients/couchbase");
var bucket = cluster.OpenBucket();

```

Insert Document Sync

There are two basic ways in which you can Insert a document

1. Create a document, Then insert it

```

var bucket = cluster.OpenBucket("default");

var document = new Document<dynamic>
{
  Id = "doc_net",
  Content = new
  {
    name = "Roi",
    lastName = "Katz",
    someRandomField="Very important data!"
  },
  Expiry = 0, // TTL in ms
};

bucket.Insert(document);

```

2. Use a Serialized object and Newtonsoft JSON.net

```

public class MyDataObject
{
  public string Name { get; set; }
  public int LastName { get; set; }
  public string SomeRandomField { get; set; }
}

```

And Use it to insert your data

```
var dataObject = new MyDataObject();  
//...Fill up the object  
bucket.Insert("MyUniqueDocumentKey", dataObject, 10); // Insert a document with 10 seconds TTL  
- or you can use a TimeSpan
```

You can also sent persistence of replication factor while you insert the document.

▲ 10 of 10 ▼ **IOperationResult<T> Couchbase.Core.IBucket**
Inserts a document into the database for a given key. The **replicateTo** property specifies the durability requirement for replication.

Adding the SDK to a project

From the NuGet Package Manager console:

```
Install-Package CouchbaseNetClient
```

Read C# SDK online: <https://riptutorial.com/couchbase/topic/5079/csharp-sdk>

Chapter 3: Connect to couchbase over SSL using SDK

Introduction

In this post I am giving an example on how to connect to Couchbase over SSL/TLS to establish a secure connection to protect data on wire.

Hopefully, you have enabled SSL on Couchbase side. For Information on enabling SSL on Couchbase side you can refer to <http://docs.couchbase.com/developer/dotnet-2.0/configuring-ssl.html> .

In example, I am setting required cipher suite and enabled protocols.

Examples

Secure connection to couchbase using java sdk with specific cipher suites and protocols

```
import com.couchbase.client.core.endpoint.SSLEngineFactory
import com.couchbase.client.java.env.DefaultCouchbaseEnvironment
import com.couchbase.client.java.CouchbaseCluster

object CouchbaseConnection extends App {

  //Create default environment object.
  //Set the keystone file path(download keystone from couch base cluster) and keystore password

  val env = DefaultCouchbaseEnvironment
    .builder()
    .sslEnabled(true)
    .sslKeystoreFile("./conf/couchbase.keystore") //
    .sslKeystorePassword("pR8PHe452353546474778r4reThUfu45678523422")
    .build();

  //Get all SSL configuration for the default environment

  val sslEngineFactory = new SSLEngineFactory(env)
  val sslEngine:SSLEngine = sslEngineFactory.get()

  //Set the list of enabled ciphers and transport protocols
  sslEngine.setEnabledCipherSuites(Array("TLS_RSA_WITH_AES_256_CBC_SHA"))
  sslEngine.setEnabledProtocols(Array("TLSv1.2"))

  val cluster = CouchbaseCluster.create(env, "127.0.0.1")
  // Open a bucket person
  val bucket = cluster.openBucket("person", "test123")
}
```

Read [Connect to couchbase over SSL using SDK online](#):

<https://riptutorial.com/couchbase/topic/8664/connect-to-couchbase-over-ssl-using-sdk>

Chapter 4: Java SDK

Examples

Adding the SDK to a project

Add the following dependency to your pom.xml file:

```
<dependency>
  <groupId>com.couchbase.client</groupId>
  <artifactId>java-client</artifactId>
  <version>2.3.1</version>
</dependency>
```

You can check the [maven repository](#) for the latest version.

If you want to use the Spring OEM use this Gradle dependency:

```
compile(group: 'org.springframework.data', name: 'spring-data-couchbase', version:
'2.1.6.RELEASE')
```

Replace the version with your desired release. Note that this is NOT the version of Couchbase but the version of the Spring connector.

Connecting to a Bucket

```
String bucketName = "default";
String bucketPassword = "";
List<String> nodes = Arrays.asList("127.0.0.1"); // IP or hostname of one or more nodes in the
cluster

Cluster cluster = CouchbaseCluster.create(nodes);
Bucket bucket = cluster.openBucket(bucketName, bucketPassword);
```

Checking for document exists in DB

```
String bucketName = "bucket";
List<String> nodes = Arrays.asList("node1", "node2"); // IP or hostname of one or more nodes in
the cluster

Cluster cluster = CouchbaseCluster.create(nodes);
Bucket bucket = cluster.openBucket(bucketName);

//check for a document by its ID

String id="bucket_collection_user_123456";//document id
boolean exists=bucket.exists(id);
if(exists){
  System.out.println("Docuemnt exists");
}
```

```
bucket.close();
cluster.disconnect();
```

Creating document with TTL (Time To Live)

TTL value can be used to decide for how long the document needs to be there in the bucket. By default TTL value is 0, which means it will be there for indefinite time period.

```
String bucketName = "bucket";
List<String> nodes = Arrays.asList("node1", "node2"); // IP or hostname of one or more nodes in
the cluster

Cluster cluster = CouchbaseCluster.create(nodes);
Bucket bucket = cluster.openBucket(bucketName);
//create the document with id 123 and TTL 1seconds
bucket.insert(JsonDocument.create("123", JsonObject.empty(), 1)); //if TTL is 0 document will
be there in the DB for indefinite time
//do other stuffs
//to update the TTL
bucket.upsert(JsonDocument.create("123", JsonObject.empty())); //no TTL value is provided
bucket.close();
cluster.disconnect();
```

Read Java SDK online: <https://riptutorial.com/couchbase/topic/5976/java-sdk>

Credits

S. No	Chapters	Contributors
1	Getting started with couchbase	Community
2	C# SDK	David Ostrovsky , Roi Katz
3	Connect to couchbase over SSL using SDK	mastrgamr , Umesh Pawar
4	Java SDK	David Ostrovsky , Girish Bhat M , Robin Ellerkmann