



Workshop 2 – Using Couchbase



- Using Query Workbench / cbq
- Using the REST API
- Using the SDK (.NET / Java / Node)



Couchbase
DEVELOPER COMMUNITY



- <https://github.com/couchbaselabs/workshop>
- If you find a typo, mistake, or spot an improvement, please send a pull request!

<http://localhost:8091>



Couchbase Enterprise Edition

[Documentation](#) • [Support](#) • [About](#) • [Sign Out](#)

[Overview](#)

[Server Nodes](#)

[Data Buckets](#)

[Query](#)

[Indexes](#)

[XDCR](#)

[Security](#)

[Log](#)

[Settings](#)

Execute



1/1



Clear History

Save Query

1 `SELECT d.* FROM `default` d`



Bucket Analysis



Fully Queryable Buckets

- `crm`
- `default`

Queryable on Indexed Fields

Non-Indexed Buckets

JSON

Table

Tree

Results

Save JSON

Status: success

Elapsed: 14.99ms

Execution: 14.93ms

Result Count: 1

Result Size: 109

```
1 [
2   {
3     "email": "baz@qux.gov",
4     "firstname": "Foo",
5     "lastname": "Bar"
6   }
7 ]
```

- Command Line N1QL tool
- **Windows:** C:\Program Files\Couchbase\Server\bin
- **OS X:** /Applications/Couchbase
Server.app/Contents/Resources/couchbase-core/bin
- **Linux:** /opt/couchbase/bin/cbq
- **Docker:** docker exec -it db /bin/bash



- REST API documentation:
<http://developer.couchbase.com/documentation/server/4.5/rest-api/rest-intro.html>
- Cluster details: <http://localhost:8091/pools/default>



- Cluster
- Nodes
- Server Group
- Buckets
- Indexes
- Views
- XDCR
- Compaction
- Log
- User/Security
- N1QL



- Create a primary index on default bucket

```
CREATE PRIMARY INDEX on `default`
```

- Add document(s) to default bucket
 - Use “Create Document” button in Documents view
- SELECT documents

```
SELECT d.* FROM `default` d
```




[Overview](#)

[Server Nodes](#)

[Data Buckets](#)

[Query](#)

[Indexes](#)

[XDCR](#)

[Security](#)

[Log](#)

[Settings](#)

Execute



1/1



Clear History

Save Query

1 `SELECT d.* FROM `default` d`



Bucket Analysis



Fully Queryable Buckets
 > [crm](#)
 > [default](#)
Queryable on Indexed Fields
Non-Indexed Buckets

JSON

Table

Tree

Results

Save JSON

Status: success

Elapsed: 14.99ms

Execution: 14.93ms

Result Count: 1

Result Size: 109

```
1 [
2   {
3     "email": "baz@qux.gov",
4     "firstname": "Foo",
5     "lastname": "Bar"
6   }
7 ]
```



Cluster details: <http://localhost:8091/pools/default>



```
{
  "storageTotals": {
    "ram": {
      "total": 2061664256,
      "quotaTotal": 268435456,
      "quotaUsed": 209715200,
      "used": 1674903552,
      "usedByData": 119026504,
      "quotaUsedPeraNode": 209715200,
      "quotaTotalPerNode": 268435456
    },
    "hdd": {
      "total": 59151257600,
      "quotaTotal": 59151257600,
      "used": 1183025152,
      "usedByData": 130504611,
      "free": 57968232448
    }
  },
  "ftsMemoryQuota": 256,
  "indexMemoryQuota": 512,
  "memoryQuota": 256,
  "name": "default",
  "alerts": [],
  // etc....
}
```

Exercise 3: Using SDK in “Hello World”



- <https://github.com/couchbaselabs/workshop/tree/master/connect2016/developer/o2>
- dotnet
- java
- node
- Insert document(s)
- Select documents with N1QL
- Fill in the blanks: look for “TODO”

- How to install
 - NuGet: Install-Package CouchbaseNetClient
- How to connect to a Couchbase cluster

```
//connecting to a Couchbase cluster
var config = new ClientConfiguration();
config.Services = new List<Uri> { "couchbase://localhost" };
ClusterHelper.Initialize(config);

//opening a bucket in the cluster
_bucket = ClusterHelper.GetBucket("travel-sample", "password");

//preparing N1ql
var myQuery = QueryRequest.Create("SELECT * FROM `travel-sample`");

//creating and saving a Document
var document = new Document<MyType> { Id = "123", Content = new MyType() };
_bucket.Insert(document);
```

- How to install
 - npm install couchbase
- How to connect to a Couchbase cluster

```
//including the Node.js dependency
var Couchbase = require("couchbase");

//connecting to a Couchbase cluster
var cluster = new Couchbase.Cluster("couchbase://localhost:8091");

//opening a bucket in the cluster
var myBucket = cluster.openBucket("travel-sample", "password");

//preparing N1ql
var myQuery = Couchbase.N1qlQuery();

//creating and saving a Document
var document = { firstname: "Matt", lastname: "Groves" };
myBucket.insert("my-key", document, function(error, result) {});
```

- How to install
 - Edit the pom.xml file
- How to connect to a Couchbase cluster

```
// connecting to a cluster
Cluster cluster = CouchbaseCluster.create("localhost");

// opening a bucket in the cluster
Bucket bucket = cluster.openBucket("default");

// preparing N1QL
N1qlQuery query = N1qlQuery.parameterized("SELECT * FROM `travel-sample`");

// creating and saving a document
JsonObject person = JsonObject.create()
    .put("firstName", "John")
    .put("lastName", "Doe");
bucket.upsert(JsonDocument.create(key, person));
```



Questions