

SEQUIADB

Matyáš Brabec

PŘEHLED

financial-grade distribuovaná multi-model databáze

- High-performance
- Reliable
- Stable
- Infinite horizontally scalable

Podpora

- MySQL, MariaDB, PostgreSQL, SparkSQL
- JSON document database
- S3 object storage unstructured data

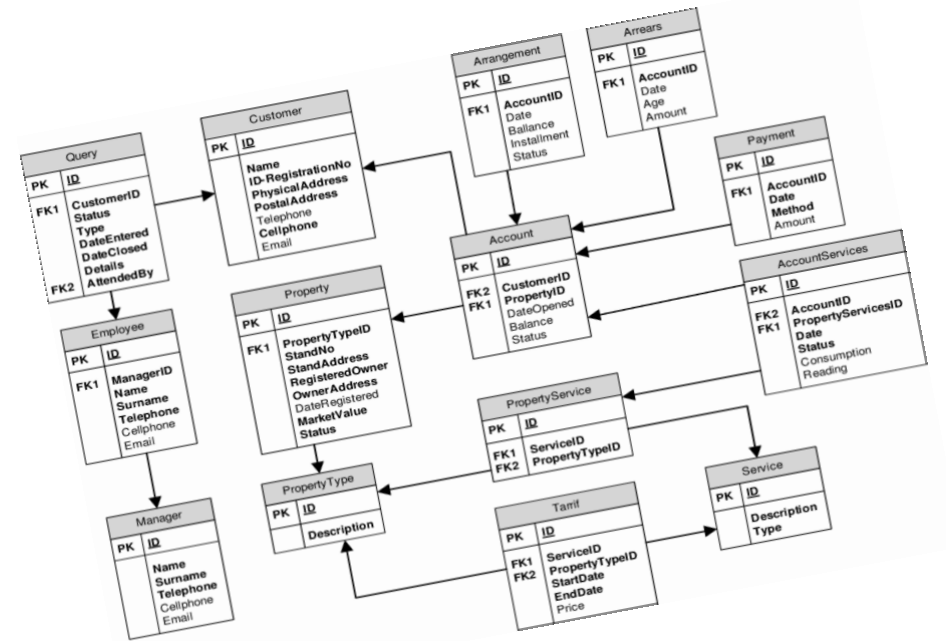


A screenshot of the SequoiaDB website homepage. The page features a dark blue background with a grid of white squares on the right side. The main heading is "湖仓一体, 金融级分布式数据库" (Lake and Warehouse Integrated, Financial-grade Distributed Database). Below the heading, it states "连续三年入选 Gartner 数据库报告" (Selected in Gartner Database Report for three consecutive years) and "在超过100家金融银行业的生产系统规模使用" (Used in production systems of over 100 financial institutions). There are two buttons: "立即下载" (Download Now) and "在线体验" (Experience Online). The top navigation bar includes links for "产品", "应用场景", "文档", "社区", "巨杉学", "新闻动态", and "关于". The bottom of the page features logos for several partner companies: PICC (中国人保财险), 广发银行/CGB, 中国民生银行 (China Min Sheng Bank), 广东农信 GDRC, and 中国银行 (Bank of China).

KLÍČOVÉ FÍČURY

Kompatibilní s relačními daty

- Data sharding je pro aplikace transparentní

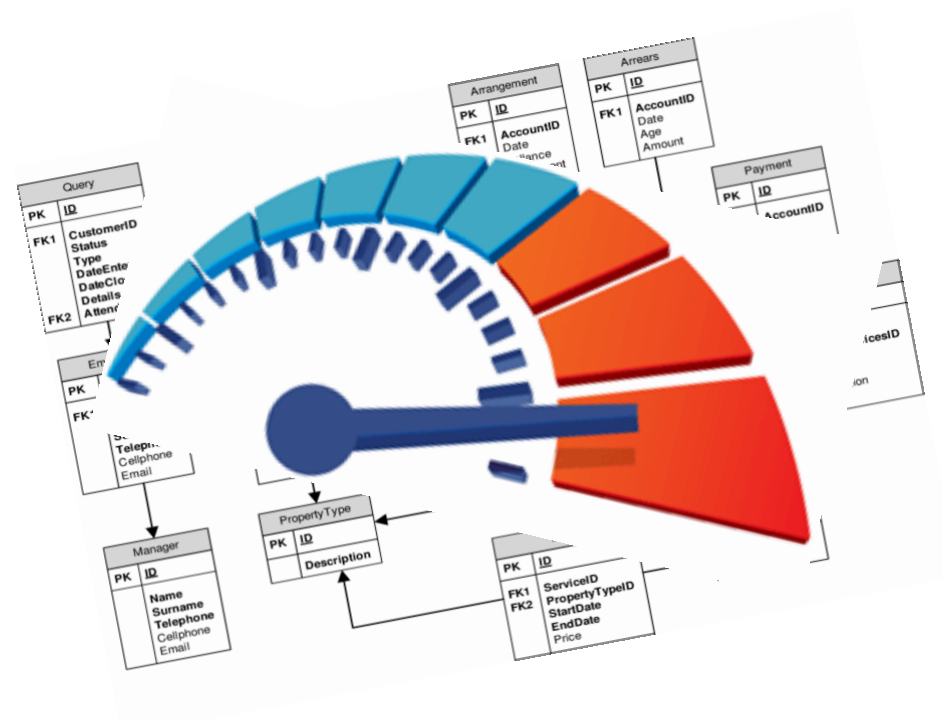


KLÍČOVÉ FÍČURY

Kompatibilní s relačními daty

- Data sharding je pro aplikace transparentní

Vysoký výkon



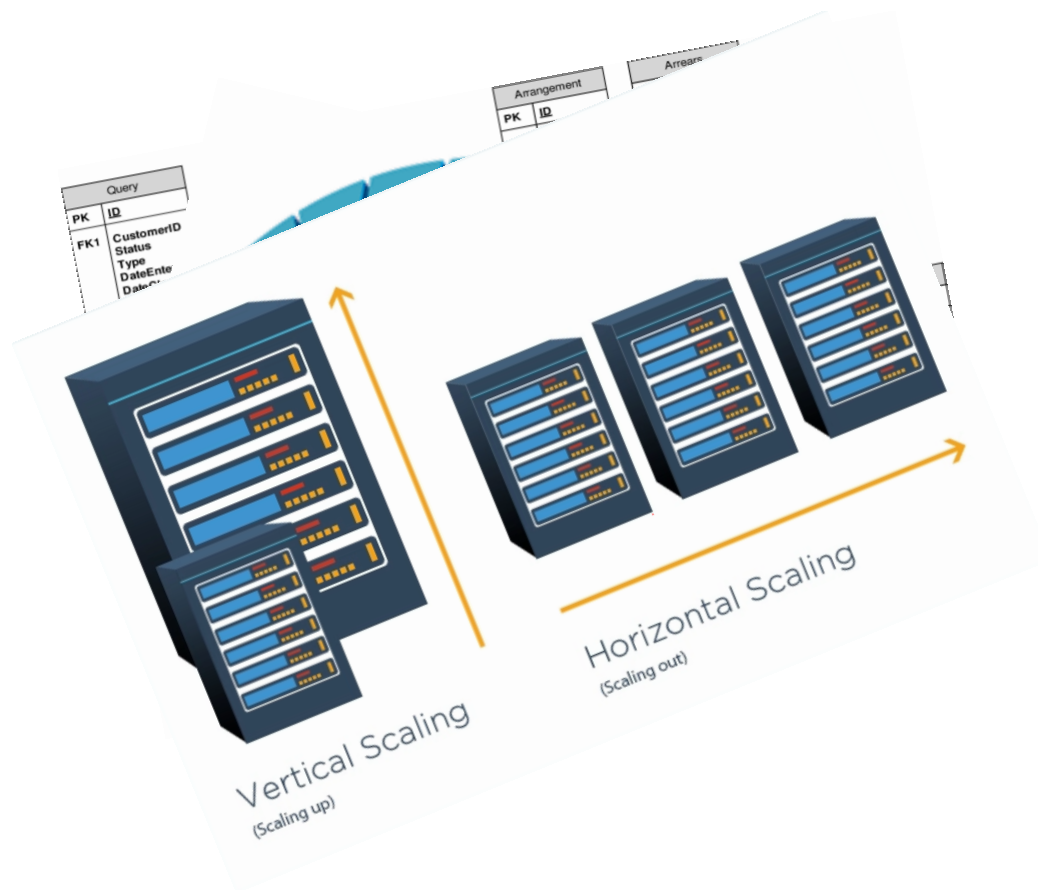
KLÍČOVÉ FÍČURY

Kompatibilní s relačními daty

- Data sharding je pro aplikace transparentní

Vysoký výkon

Neomezený horizontální scale



KLÍČOVÉ FÍČURY

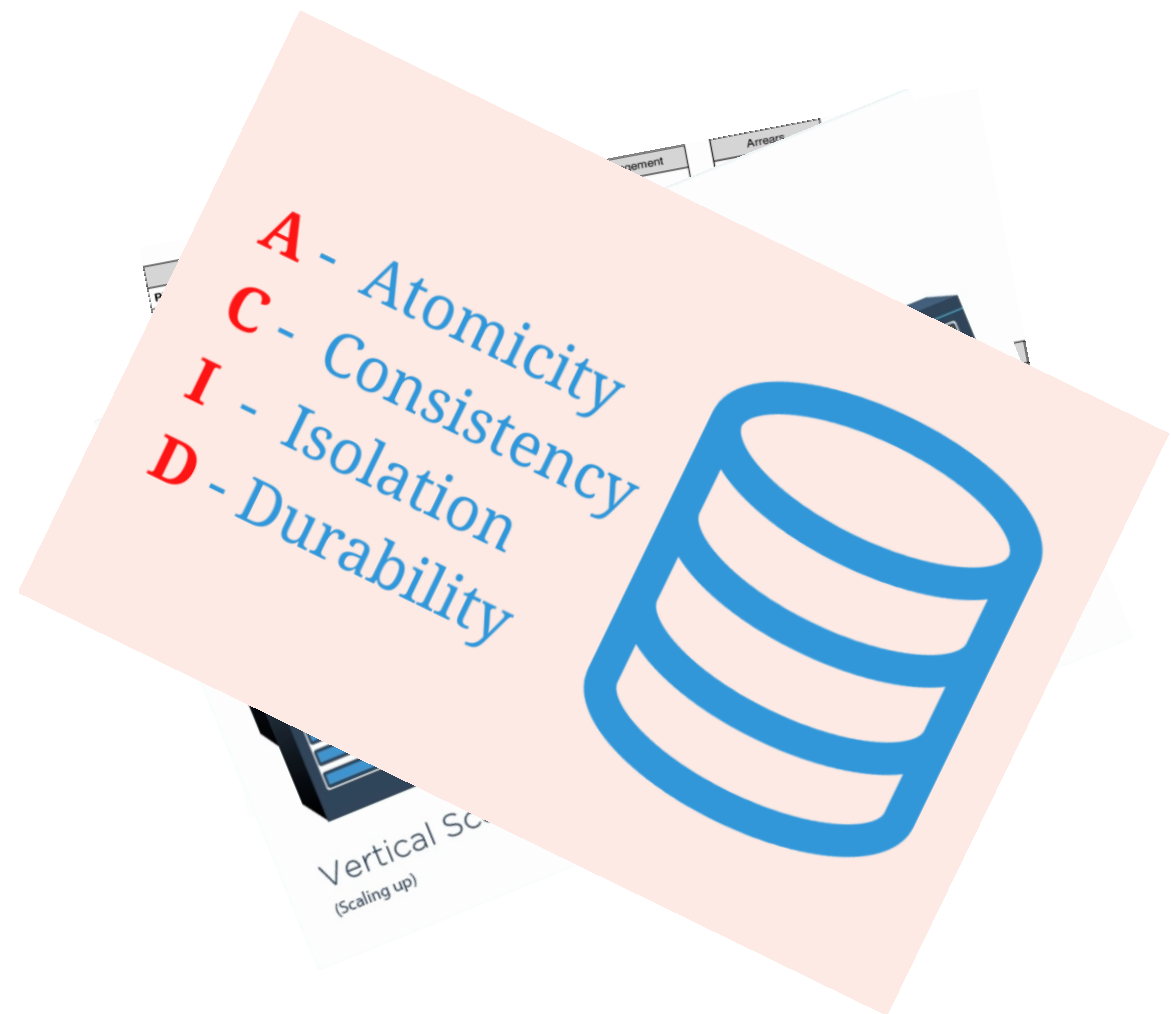
Kompatibilní s relačními daty

- Data sharding je pro aplikace transparentní

Vysoký výkon

Neomezený horizontální scale

Distribuované transakce a ACID



KLÍČOVÉ FÍČURY

Kompatibilní s relačními daty

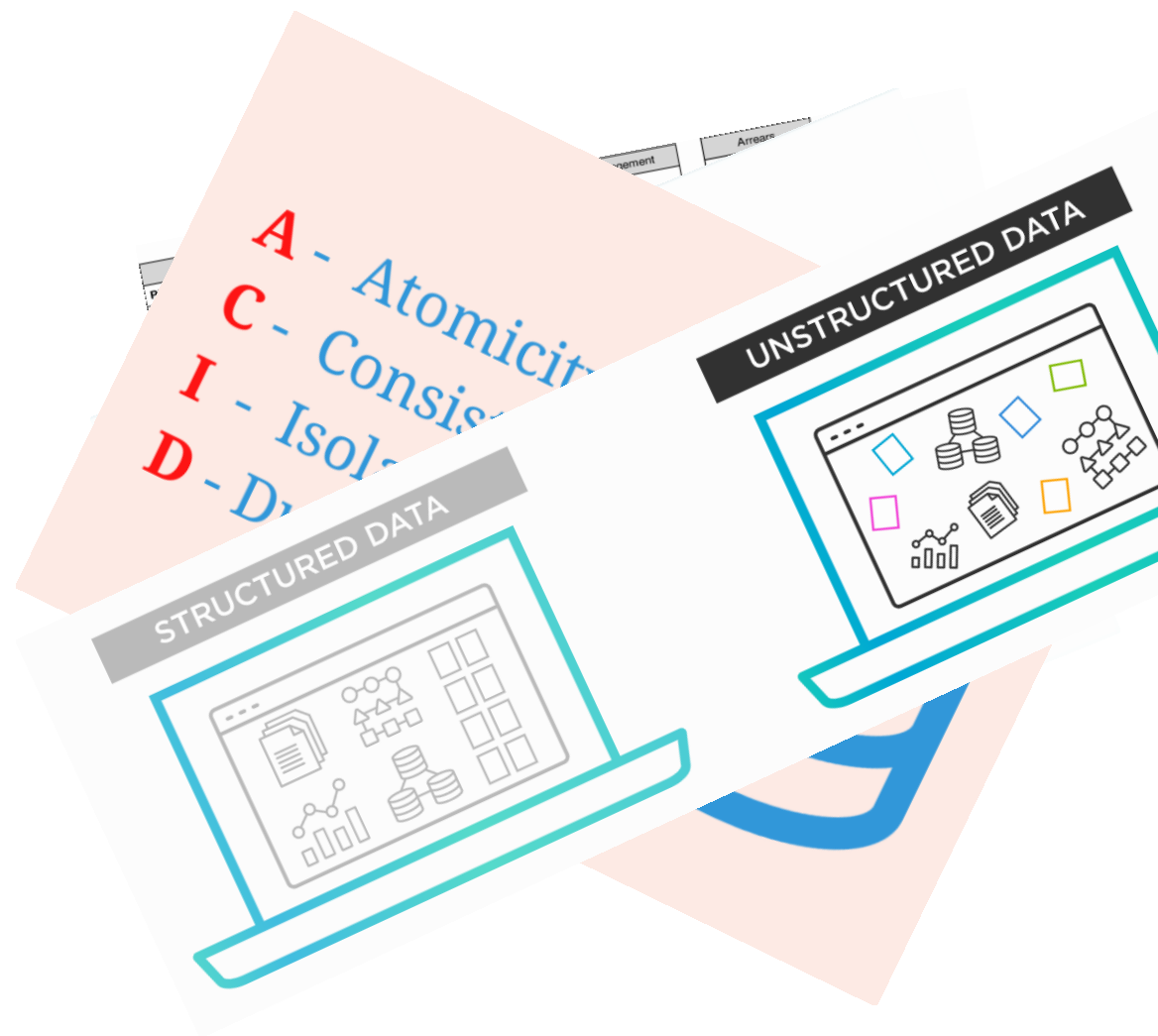
- Data sharding je pro aplikace transparentní

Vysoký výkon

Neomezený horizontální scale

Distribuované transakce a ACID

Podpora strukturovaných i nestrukturovaných dat



KLÍČOVÉ FÍČURY

Kompatibilní s relačními daty

- Data sharding je pro aplikace transparentní

Vysoký výkon

Neomezený horizontální scale

Distribuované transakce a ACID

Podpora strukturovaných i nestrukturovaných dat

Financial-grade security features



DOKUMENTACE



产品概述

SequoiaDB 巨杉数据库是一款金融级分布式关系型数据库，主要面对高并发联机交易型场景提供高性能、可靠稳定以及无限水平扩展的数据库服务。

用户可以在 SequoiaDB 巨杉数据库中创建多种类型的数据库实例，以满足上层不同应用程序各自的需求。

SequoiaDB 巨杉数据库支持 MySQL、MariaDB、PostgreSQL 和 SparkSQL 四种关系型数据库实例、JSON 文档类数据库实例、以及 S3 对象存储的非结构化数据实例。

关键特性

SequoiaDB 巨杉数据库可以为用户带来如下价值：

- 完全兼容传统关系型数据，数据分片对应用程序完全透明
- 高性能与无限水平弹性扩展能力
- 分布式事务与 ACID 能力
- 同时支持结构化、半结构化与非结构化数据
- 金融级安全特性，多数据中心间容灾做到 RPO=0
- 混合负载，同时运行联机交易与批处理任务且互不干扰
- 多租户能力，云环境下支持多种级别的物理与逻辑隔离

用户案例

当前已经有超过 **50 家银行机构与上百家企业级用户** 在生产环境大规模使用 SequoiaDB 巨杉数据库取代传统数据库。

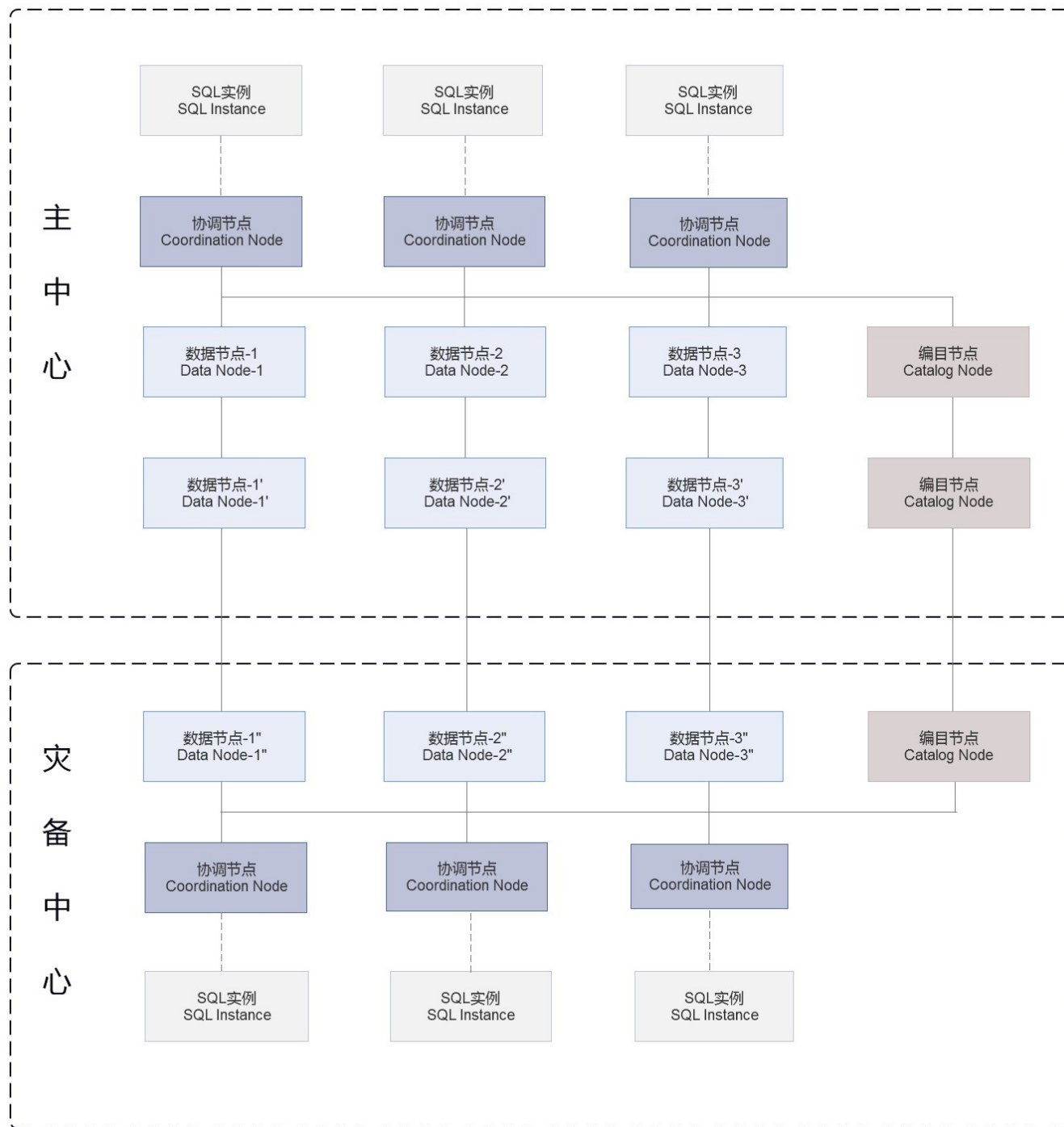
STRUKTURA SYSTÉMU

SQL instance

Coordination node

Cataloging node

Data node



DEMO

ZÁKLADNÍ OPERACE

update

Collection space

Collection

```
db.sample.employee.update({$inc: {age: 1}})
```

```
db.sample.employee.update({$inc: {age: 1}}, {age: {$gt: 20}})
```

remove

```
db.sample.employee.remove()
```

```
db.sample.employee.remove({age: {$gte: 20}})
```

init databáze v JS

insert

```
var db = new Sdb( "sdbserver1", 11810 )
```

```
db.sample.employee.insert( { _id: 10, name: "Kvetoslav", age: 20 } )
```

```
db.sample.employee.insert( [
  { _id: 10, name: "Kvetoslav", age: 20 },
  { _id: 20, name: "Borislav", age: 42 } ]
)
```

find

```
db.sample.employee.find( { age: { $gt: 15 }, name: "Kvetoslav" } )
{
  "_id": {
    "$oid": "5c6f660ce700db6048677154"
  },
  "name": "Kvetoslav",
  "age": 20
}
Return 1 row(s).
```

```
db.sample.employee.find()
{
  "_id": {
    "$oid": "5c6f660ce700db6048677154"
  },
  "name": "Kvetoslav",
  "age": 20
}
Return 1 row(s).
```

SQL ZAJÍMAVOSTI



```
1 // database
2 { id: 1, whiskyName: "octomore 12.1", owners: ["Zdibrich", "Milena", "Kvetoslav"] }
3 { id: 2, whiskyName: "ardbeg 10", owners: ["Zdibrich", "Milena"] }
4
5 // query
6 db.exec( "select * from mojeSekvoj.whisky split by owners" )
7 { "id": 1, "whiskyName": "octomore 12.1", "owners": "Zdibrich" }
8 { "id": 1, "whiskyName": "octomore 12.1", "owners": "Milena" }
9 { "id": 1, "whiskyName": "octomore 12.1", "owners": "Kvetoslav" }
10 { "id": 2, "whiskyName": "ardbeg 10", "owners": "Zdibrich" }
11 { "id": 2, "whiskyName": "ardbeg 10", "owners": "Milena" }
12 Return 5 row(s).
13 Takes 0.5281s.
```

AGGREGATE

Data v DB

```
1  {
2    no: 1000,
3    score: 80,
4    interest: ["Drinking", "Crying"],
5    major: "Computer Science",
6    dep: "Chinese department of democracy",
7    info:
8    {
9      name: "Zdibrich",
10     age: 25,
11     gender: "male"
12   }
13 }
```

AGGREGATE

Dotaz

- Průměrný věk skupin zaměstnanců dle jejich titulu, kteří mají alespoň jedno hobby
 - Setříváno dle průměrného věku a poté podle titulu
 - Vypsán 3.-5. záznam

AGGREGATE

Data v DB

```
1 {
2   no: 1000,
3   score: 80,
4   interest: ["Drinking", "Crying"],
5   major: "Computer Science",
6   dep: "Chinese department of democracy",
7   info:
8     {
9       name: "Zdibrich",
10      age: 25,
11      gender: "male"
12    }
13 }
```

```
1 db.sample.employee.aggregate(
2   { $match: { interest: { $exists: 1 } } },
3   { $group: { _id: "$major",
4               avg_age: { $avg: "$info.age" },
5               major: { $first: "$major" } } },
6   { $sort: { avg_age: -1, major: -1 } },
7   { $skip: 2 },
8   { $limit: 3 }
9 )
```

Dotaz

Výsledek

```
1 {
2   "avg_age": 25,
3   "major": "Computer Science"
4 }
5 {
6   "avg_age": 22,
7   "major": "Obraceni Hranolek v Mekaci"
8 }
9 {
10  "avg_age": 22,
11  "major": "Oprava traktoru"
12 }
```


A DALŠÍ ...

aggregate()
alter()
attachCL()
count()
createAutoIncrement()
createIdIndex()
createIndex()
createLobID()
deleteLob()
detachCL()
disableCompression()
disableSharding()
dropAutoIncrement()
dropIdIndex()

remove()
setAttributes()
split()
splitAsync()
truncate()
truncateLob()
update()
upsert()

getLob()
getLobDetail()
insert()
listIndexes()
listLobs()
putLob()
remove()
setAttributes()
split()

dropIndex()
enableCompression()
enableSharding()
find()
findOne()
getDetail()
getIndex()
getIndexStat()

arrayAccess()
close()
count()
current()
explain()
flags()
getQueryMeta()
hint()
limit()

limit()
next()
query()
remove()
size()
skip()
sort()
toArray()
update()



DĚKUJI ZA POZORNOST