← Back **Properties of Key-value store**
Practice Quiz • 20 min

**Congratulations! You passed!**

**Grade received** 100% **To pass** 80% or higher

**Go to next item**

## Properties of Key-value store

**Total points** 10

**1.** What is a key-value store? **1 / 1 point**

● It is a system that stores values indexed by keys, can store structured and unstructured data.

○ It is a system that stores columns that supports referential integrity

✓ **Correct**
Well done! The values are stored as arrays of bytes and they are indexed by keys.

**2.** The key-value database Cassandra is more focused on availability and consistency than on fault tolerance. **1 / 1 point**

● False

○ True

✓ **Correct**
Right! Cassandra is more focused on availability and fault tolerance than on consistency. Therefore, it allows Eventual Consistency.

**3.** Cassandra as a key-value database is more recommended when... **1 / 1 point**

○ The application requires more availability and fault tolerance than consistency, when the application requires full support of ACID properties for performance and operational management advantages

● The application requires more availability and fault tolerance than consistency, when the application can sacrifice ACID properties for performance and operational management advantages

✓ **Correct**
Yes, Cassandra is suitable when the desired application does not require changes on data to be applied immediately, but it requires to be always available, even in front of a network partition.

**4.** In Cassandra as a key-value database, what are the main components of storage? **1 / 1 point**

○

○ Columns, values and timestamp

● Columns, super
columns, families of columns and keyspace

✓ **Correct**
Yes, its main components are Columns, super
columns, families of columns, and keyspace. Furthermore, a column is a tuple
that contains the name of the column, a value and a timestamp.

**5.** What are the elements of a supercolumn? **1 / 1 point**

○ Contains columns or grouped super-columns that use a
single, common RowKey

○ Families of columns or family of supercolumns

● An array of several columns

✓ **Correct**
Yes! A super column is composed of an array of several
columns. It is specified with a name and an ordered map of columns.

**6.** What are the elements of a column family? **1 / 1 point**

● Contains columns or grouped
super-columns that use a single, common RowKey

○ An array of several columns

○ Families of columns or family of supercolumns

✓ **Correct**
Well done, a family of columns contains columns or
grouped super-columns that use a single, common RowKey. It can be seen as
a set of key-value pairs.

**7.** What are the elements of a keyspace? **1 / 1 point**

○ Contains columns or grouped
super-columns that use a single, common RowKey

● Families
of columns or family of supercolumns

○ An array of several columns

✓ **Correct**
Well done! The keyspaces may have associated one or
more families of columns, although it is not always necessary that they have
families of columns. The keyspaces require that some attributes be defined,
such as user-defined names, replication strategies and others.

**8.** Comparing
Cassandra with relational databases, a column is analogous to... **1 / 1 point**

● An attribute name in a table

○ A
relational database without interrelations

○ A
tuple of a relation

✓ **Correct**
Yes! a Cassandra column is like a an attribute
name in a table.

**9.** Comparing Cassandra with relational databases, a keyspace is analogous to...

1 / 1 point

● A relational database without interrelations

○ An attribute name in a table

○ A tuple of a relation

✓ **Correct**
Yes! a Cassandra keyspace is like a relational database, but with no interrelations

**10.** Comparing Cassandra with relational databases, a column family is analogous to...

1 / 1 point

● A tuple of a relation

○ An attribute name in a table

○ A relational database without interrelations

✓ **Correct**
Yes! a Cassandra column family is analogous to a tuple of a relation.