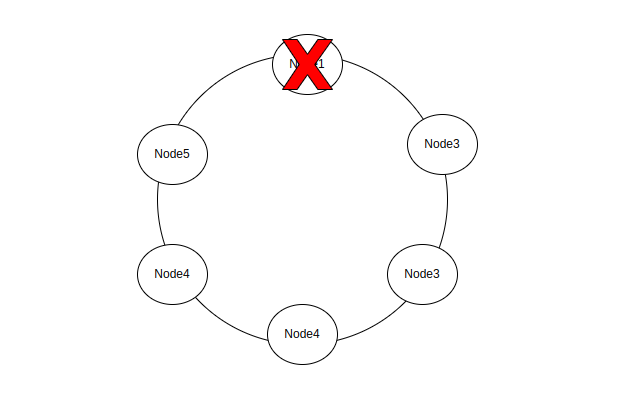
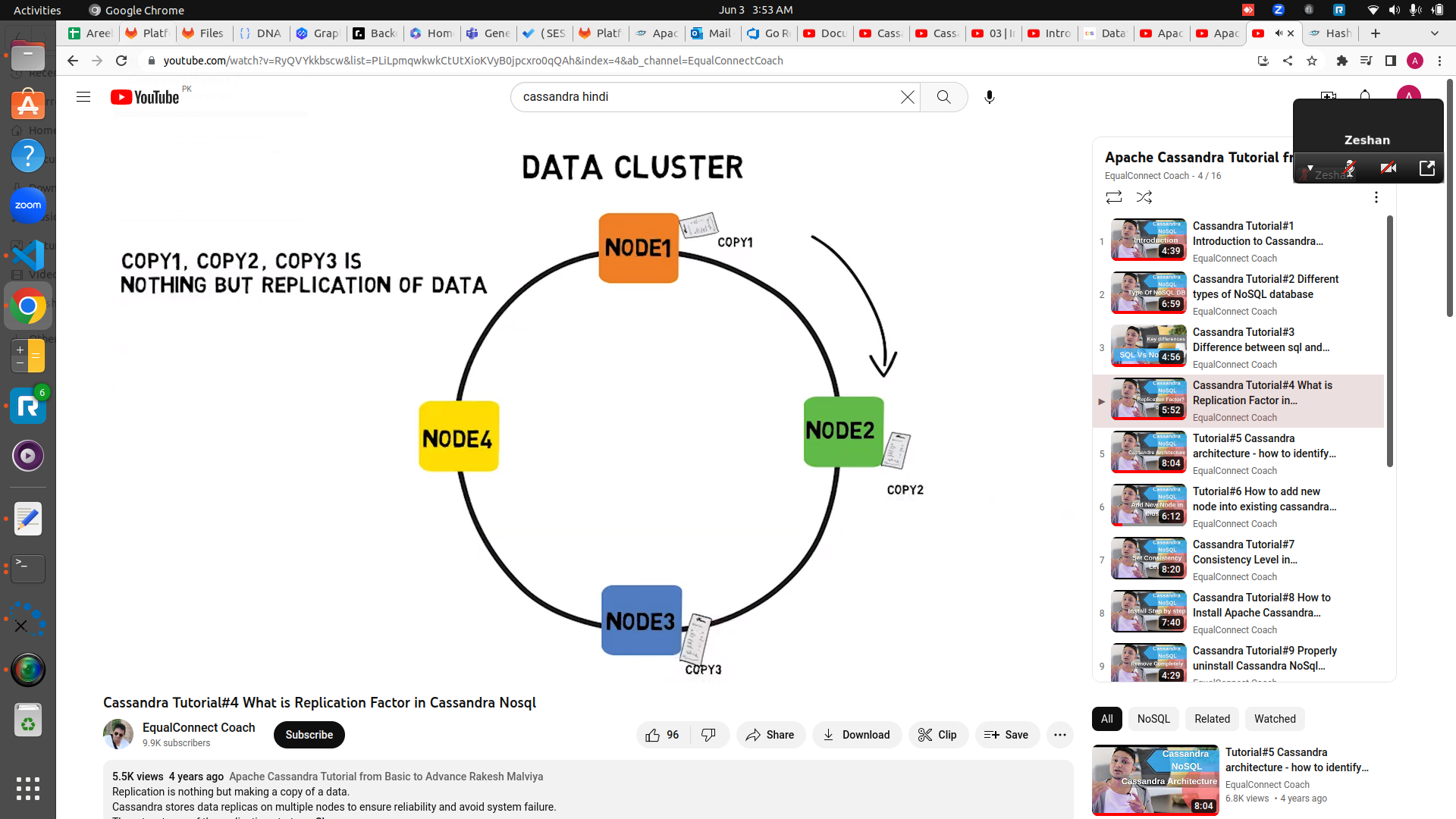
**How Replication work?**

As hardware problems can occur or links can be down at any time during the data process, a solution is required to provide a backup when the problem has occurred. So data is replicated for assuring no single point of failure.

What if I lose a node do I lose my data? Let see how replication works to answer that



**Replication is nothing but just “MAKING COPY OF DATA”**



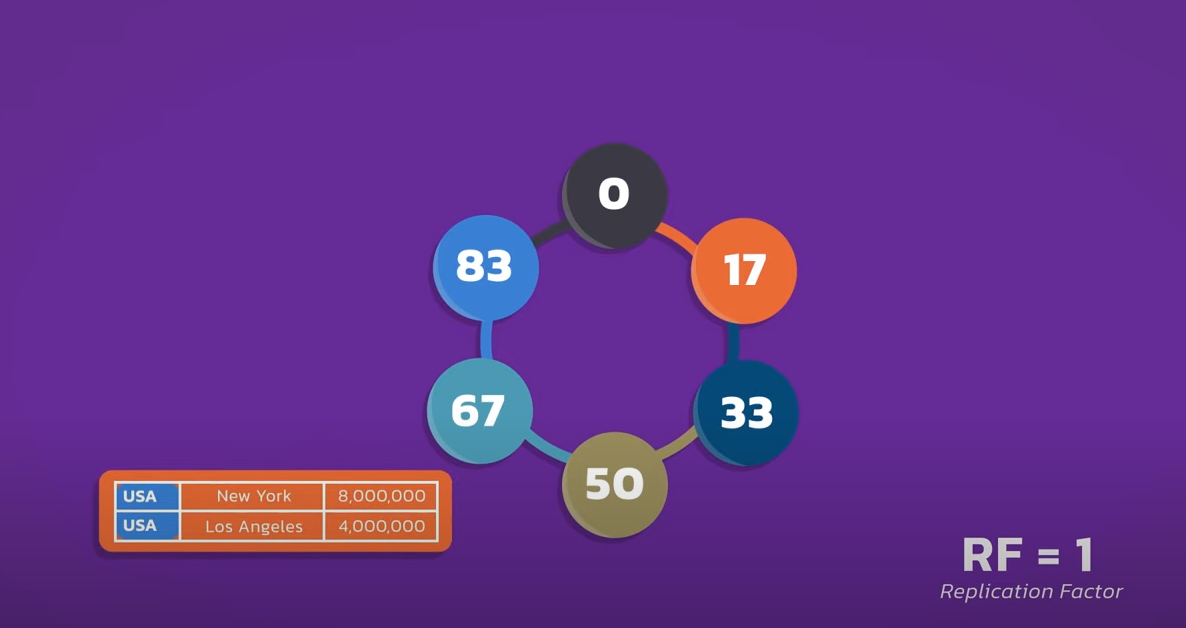
Cassandra places replicas of data on different nodes based on these two factors.

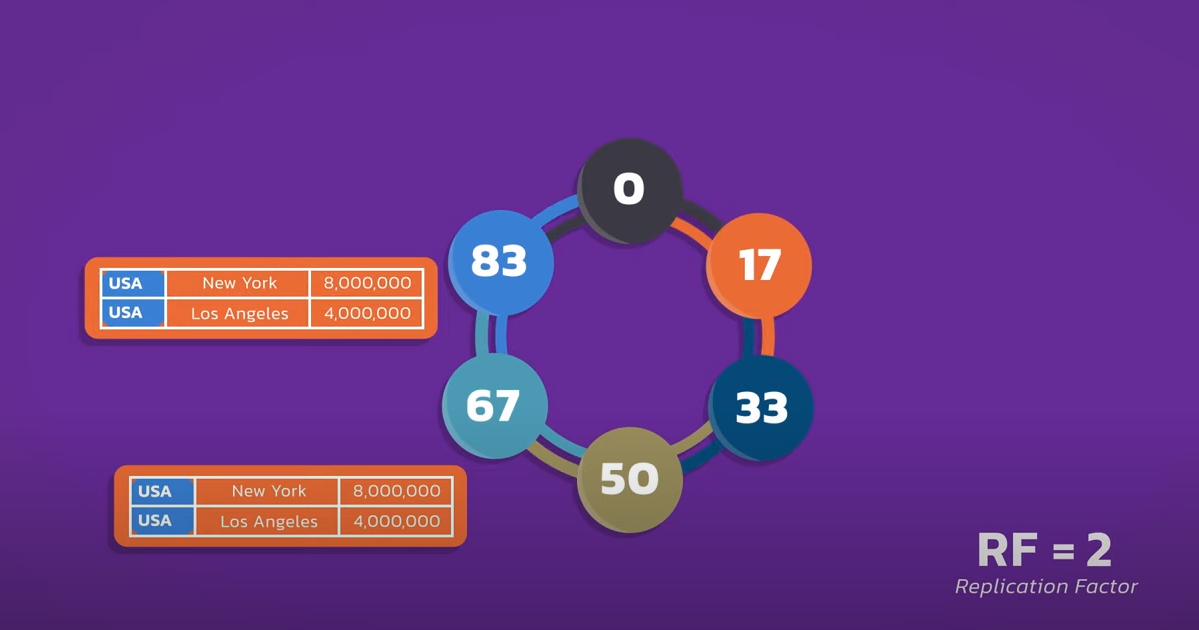
* Where to place the next replica is determined by the **Replication Strategy**.
* While the total number of replicas placed on different nodes is determined by the **Replication Factor**.**simply means number of copies you want to create of you data for example replication factor RF=3 then it means i want to make 3 copies of my data**

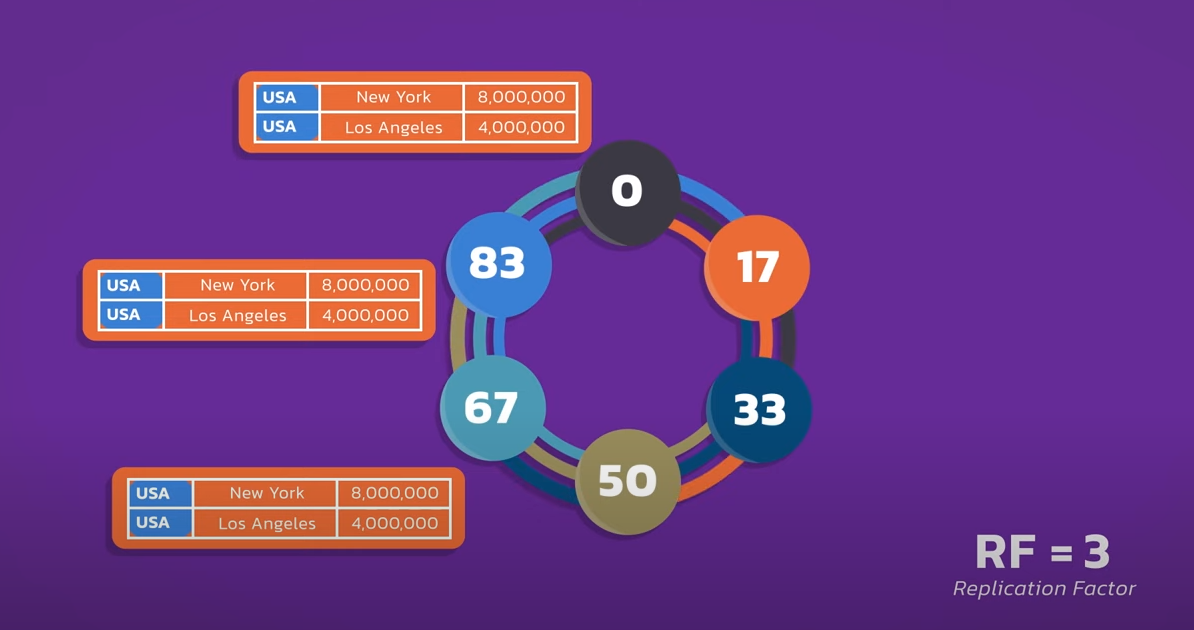
**Replication Factor**

The replication factor in Cassandra refers to the number of copies of each piece of data that are stored across the nodes in a cluster. When a new piece of data is written to a Cassandra cluster, it is automatically replicated to a specified number of nodes, based on the replication factor. For example, if the replication factor is set to 3, each piece of data will be stored on 3 different nodes in the cluster.

The replication factor can be set at the keyspace level, or at the individual table level. This means that you can have different replication factors for different tables in the same keyspace. The replication factor is set when the keyspace is created and can be modified at a later time.

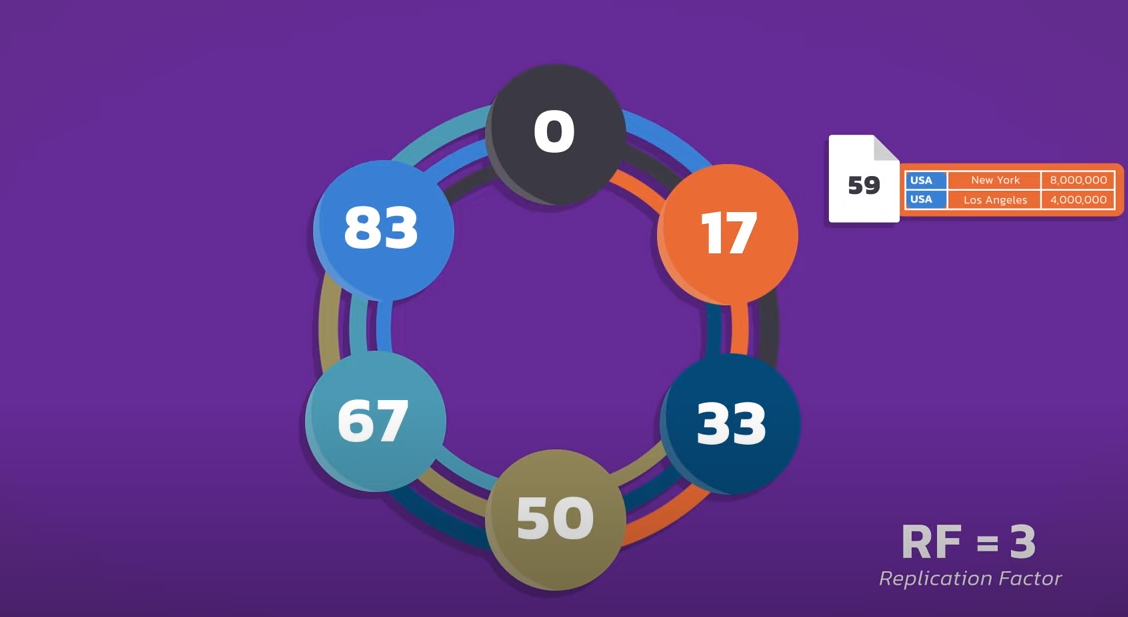
So here there is only single ring which represents replication factor is means there is only one copy of node 

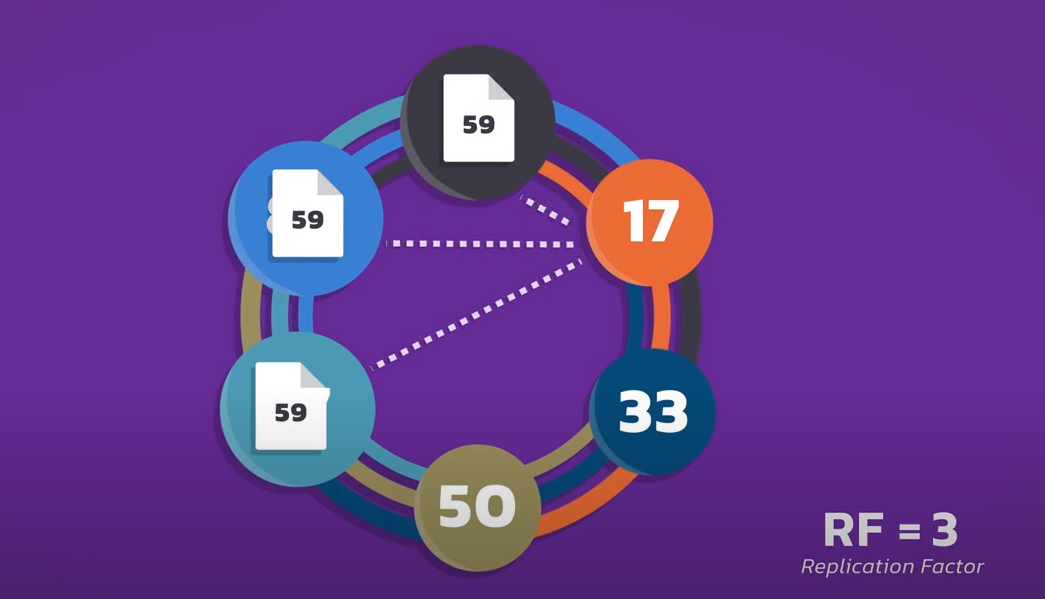
If we increase replication factor to two now we have 2 rings and 2 node which contains the copy 

If we increase replication factor to three now we have 3 rings and 3 copies of node 

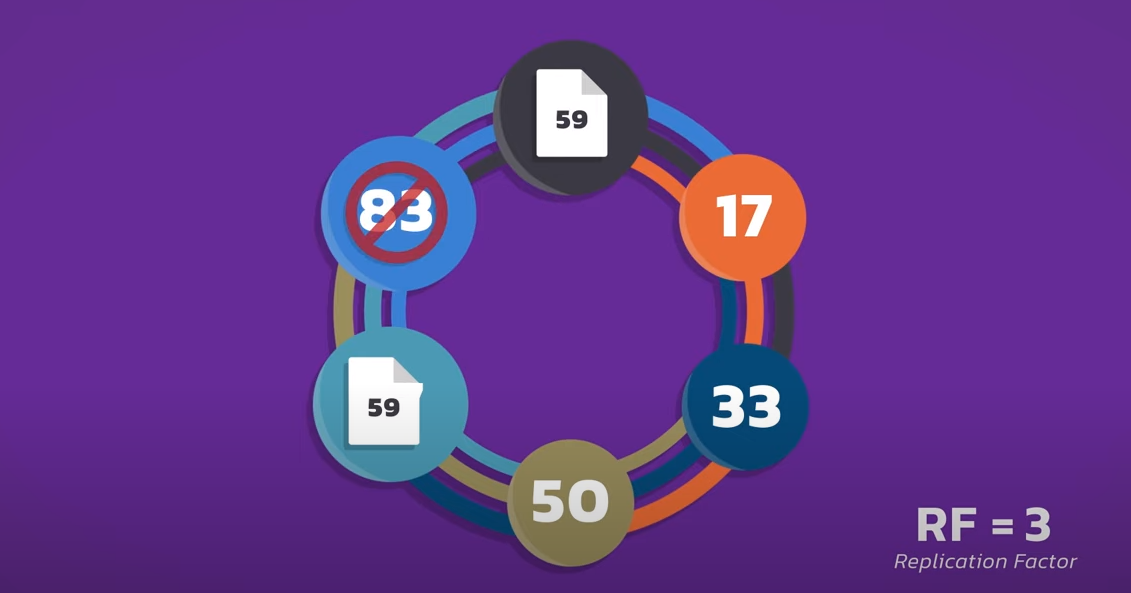
Replication factor of 3 is standard in cassandra as it is a good balance of **availability**, **performance** and **consistency**

If we want to write some data, when the request comes in, any node is chosen to do the task and the node which is chosen is called **Coordinator node.** Any node can be coordinator



In the case of replication factor of 3 there are 3 node which contains the copy of our node on which we have to write data so what happens is the coordinator node forward the right to all 3 node and each node stores the copy 

If any node is down in his process cassandra store called hint once the node backs

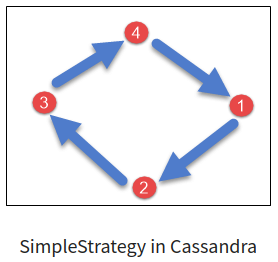
up the right is automatically played back to hal the node 

### There are two kinds of replication strategies in Cassandra.

### **SimpleStrategy in Cassandra**

**SimpleStrategy** is used when you have just one data center. SimpleStrategy places the first replica on the node selected by the partitioner. After that, remaining replicas are placed in clockwise direction in the Node ring.

Here is the pictorial representation of the SimpleStrategy:



Let’s consider taking an example, strategy\_demo is a keyspace name in which class is SimpleStrategy and replication\_factor is 2 which simply means there are two redundant copies of each row in a single data center.

**Creating a keyspace:**

CREATE KEYSPACE cluster1 WITH

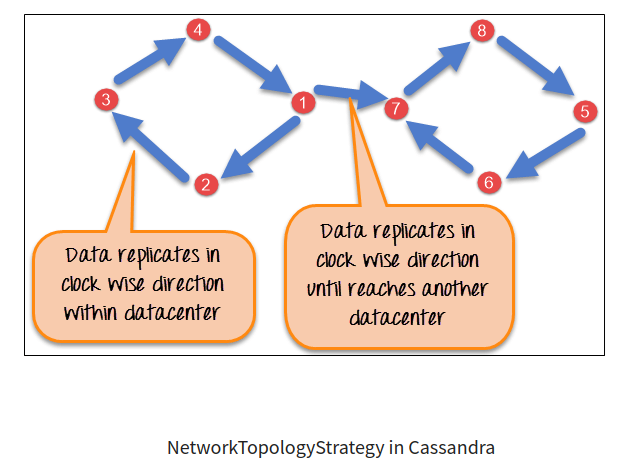
replication = {'class': 'SimpleStrategy',

'replication\_factor' : 2};

**NetworkTopologyStrategy** is used when you have more than two data centers. In NetworkTopologyStrategy, replicas are set for each data center separately. NetworkTopologyStrategy places replicas in the clockwise direction in the ring until it reaches the first node in another rack. This strategy tries to place replicas on different racks in the same data center.

This is due to the reason that sometimes failure or problem can occur in the rack. Then replicas on other nodes can provide data.

Here is the pictorial representation of the Network topology strategy:



Let’s consider an example, cluster1 is a keyspace name in which NetworkTopologyStrategy is a replication strategy and there are two data centers one is east with RF( Replication Factor) = 2 and second is west with RF( Replication Factor) = 3.

**Creating a keyspace:**

CREATE KEYSPACE cluster1 WITH

replication = {'class': 'NetworkTopologyStrategy',

'east' : 2, 'west' : 3};