

HBASE - DELETE DATA

http://www.tutorialspoint.com/hbase/hbase_delete_data.htm

Copyright © tutorialspoint.com

Deleting a Specific Cell in a Table

Using the **delete** command, you can delete a specific cell in a table. The syntax of **delete** command is as follows:

```
delete '<table name>', '<row>', '<column name >', '<time stamp>'
```

Example

Here is an example to delete a specific cell. Here we are deleting the salary.

```
hbase(main):006:0> delete 'emp', '1', 'personal data:city',  
1417521848375  
0 row(s) in 0.0060 seconds
```

Deleting All Cells in a Table

Using the “deleteall” command, you can delete all the cells in a row. Given below is the syntax of deleteall command.

```
deleteall '<table name>', '<row>',
```

Example

Here is an example of “deleteall” command, where we are deleting all the cells of row1 of emp table.

```
hbase(main):007:0> deleteall 'emp', '1'  
0 row(s) in 0.0240 seconds
```

Verify the table using the **scan** command. A snapshot of the table after deleting the table is given below.

```
hbase(main):022:0> scan 'emp'  
  
ROW          COLUMN &plus; CELL  
  
2 column = personal data:city, timestamp = 1417524574905, value = chennai  
2 column = personal data:name, timestamp = 1417524556125, value = ravi  
2 column = professional data:designation, timestamp = 1417524204, value = sr:engg  
2 column = professional data:salary, timestamp = 1417524604221, value = 30000  
3 column = personal data:city, timestamp = 1417524681780, value = delhi  
3 column = personal data:name, timestamp = 1417524672067, value = rajesh  
3 column = professional data:designation, timestamp = 1417523187, value = jr:engg  
3 column = professional data:salary, timestamp = 1417524702514, value = 25000
```

Deleting Data Using Java API

You can delete data from an HBase table using the **delete** method of the **HTable** class. Follow the steps given below to delete data from a table.

Step 1: Instantiate the Configuration Class

Configuration class adds HBase configuration files to its object. You can create a configuration object using the **create** method of the **HbaseConfiguration** class as shown below.

```
Configuration conf = HbaseConfiguration.create();
```

Step 2: Instantiate the HTable Class

You have a class called **HTable**, an implementation of Table in HBase. This class is used to communicate with a single HBase table. While instantiating this class, it accepts the configuration object and the table name as parameters. You can instantiate the HTable class as shown below.

```
HTable hTable = new HTable(conf, tableName);
```

Step 3: Instantiate the Delete Class

Instantiate the **Delete** class by passing the rowid of the row that is to be deleted, in byte array format. You can also pass timestamp and Rowlock to this constructor.

```
Delete delete = new Delete(toBytes("row1"));
```

Step 4: Select the Data to be Deleted

You can delete the data using the delete methods of the **Delete** class. This class has various delete methods. Choose the columns or column families to be deleted using those methods. Take a look at the following examples that show the usage of Delete class methods.

```
delete.deleteColumn(Bytes.toBytes("personal"), Bytes.toBytes("name"));
delete.deleteFamily(Bytes.toBytes("professional"));
```

Step 5: Delete the Data

Delete the selected data by passing the **delete** instance to the **delete** method of the **HTable** class as shown below.

```
table.delete(delete);
```

Step 6: Close the HTableInstance

After deleting the data, close the **HTable** Instance.

```
table.close();
```

Given below is the complete program to delete data from the HBase table.

```
import java.io.IOException;
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.hbase.HBaseConfiguration;
import org.apache.hadoop.hbase.client.Delete;
import org.apache.hadoop.hbase.client.HTable;
import org.apache.hadoop.hbase.util.Bytes;

public class DeleteData {

    public static void main(String[] args) throws IOException {

        // Instantiating Configuration class
        Configuration conf = HBaseConfiguration.create();
```

```
// Instantiating HTable class
HTable table = new HTable(conf, "employee");

// Instantiating Delete class
Delete delete = new Delete(Bytes.toBytes("row1"));
delete.deleteColumn(Bytes.toBytes("personal"), Bytes.toBytes("name"));
delete.deleteFamily(Bytes.toBytes("professional"));

// deleting the data
table.delete(delete);

// closing the HTable object
table.close();
System.out.println("data deleted.....");
}
}
```

Compile and execute the above program as shown below.

```
$javac Deletedata.java
$java DeleteData
```

The following should be the output:

```
data deleted
Loading [MathJax]/jax/output/HTML-CSS/jax.js
```