

This chapter describes the java client API for HBase that is used to perform **CRUD** operations on HBase tables. HBase is written in Java and has a Java Native API. Therefore it provides programmatic access to Data Manipulation Language *DML*.

## Class HBase Configuration

Adds HBase configuration files to a Configuration. This class belongs to the **org.apache.hadoop.hbase** package.

## Methods and description

S.No.	Methods and Description
-------	-------------------------

- |   |  |
|---|--|
| 1 | <b>static org.apache.hadoop.conf.Configuration create</b><br>This method creates a Configuration with HBase resources. |
|---|--|

## Class HTable

HTable is an HBase internal class that represents an HBase table. It is an implementation of table that is used to communicate with a single HBase table. This class belongs to the **org.apache.hadoop.hbase.client** class.

## Constructors

S.No.	Constructors and Description
-------	------------------------------

- |   |   |
|---|---|
| 1 | <b>HTable</b>   |
| 2 | <b>HTable</b> <i>TableNametableName, ClusterConnectionconnection, ExecutorServicepool</i><br>Using this constructor, you can create an object to access an HBase table. |

## Methods and description

S.No.	Methods and Description
-------	-------------------------

- |   |  |
|---|--|
| 1 | <b>void close</b><br>Releases all the resources of the HTable.             |
| 2 | <b>void delete</b> <i>Deletedelete</i><br>Deletes the specified cells/row. |

3

**boolean exists***Getget*

Using this method, you can test the existence of columns in the table, as specified by Get.

4

**Result get***Getget*

Retrieves certain cells from a given row.

5

**org.apache.hadoop.conf.Configuration getConfiguration**

Returns the Configuration object used by this instance.

6

**TableName getName**

Returns the table name instance of this table.

7

**HTableDescriptor getTableDescriptor**

Returns the table descriptor for this table.

8

**byte[] getTableName**

Returns the name of this table.

9

**void put***Putput*

Using this method, you can insert data into the table.

## Class Put

This class is used to perform Put operations for a single row. It belongs to the **org.apache.hadoop.hbase.client** package.

## Constructors

S.No.	Constructors and Description
1	<b>Put</b> <i>byte[]row</i> Using this constructor, you can create a Put operation for the specified row.
2	<b>Put</b> <i>byte[]rowArray, introwOffset, introwLength</i> Using this constructor, you can make a copy of the passed-in row key to keep local.
3	<b>Put</b> <i>byte[]rowArray, introwOffset, introwLength, longts</i>

Using this constructor, you can make a copy of the passed-in row key to keep local.

4

### **Put***byte[]row, longts*

Using this constructor, we can create a Put operation for the specified row, using a given timestamp.

## Methods

S.No.	Methods and Description
1	<b>Put</b> <i>addbyte[]family, byte[]qualifier, byte[]value</i> Adds the specified column and value to this Put operation.
2	<b>Put</b> <i>addbyte[]family, byte[]qualifier, longts, byte[]value</i> Adds the specified column and value, with the specified timestamp as its version to this Put operation.
3	<b>Put</b> <i>addbyte[]family, ByteBufferqualifier, longts, ByteBuffervalue</i> Adds the specified column and value, with the specified timestamp as its version to this Put operation.
4	<b>Put</b> <i>addbyte[]family, ByteBufferqualifier, longts, ByteBuffervalue</i> Adds the specified column and value, with the specified timestamp as its version to this Put operation.

1

### **Put** *addbyte[]family, byte[]qualifier, byte[]value*

Adds the specified column and value to this Put operation.

2

### **Put** *addbyte[]family, byte[]qualifier, longts, byte[]value*

Adds the specified column and value, with the specified timestamp as its version to this Put operation.

3

### **Put** *addbyte[]family, ByteBufferqualifier, longts, ByteBuffervalue*

Adds the specified column and value, with the specified timestamp as its version to this Put operation.

4

### **Put** *addbyte[]family, ByteBufferqualifier, longts, ByteBuffervalue*

Adds the specified column and value, with the specified timestamp as its version to this Put operation.

## Class Get

This class is used to perform Get operations on a single row. This class belongs to the **org.apache.hadoop.hbase.client** package.

## Constructor

S.No.	Constructor and Description
1	<b>Get</b> <i>byte[]row</i> Using this constructor, you can create a Get operation for the specified row.
2	<b>Get</b> <i>Getget</i>

1

### **Get***byte[]row*

Using this constructor, you can create a Get operation for the specified row.

2

### **Get***Getget*

## Methods

## S.No. Methods and Description

- 1  
**Get addColumn***byte[]family, byte[]qualifier*  
Retrieves the column from the specific family with the specified qualifier.
- 2  
**Get addFamily***byte[]family*  
Retrieves all columns from the specified family.

## Class Delete

This class is used to perform Delete operations on a single row. To delete an entire row, instantiate a Delete object with the row to delete. This class belongs to the **org.apache.hadoop.hbase.client** package.

## Constructor

### S.No. Constructor and Description

- 1  
**Delete***byte[]row*  
Creates a Delete operation for the specified row.
- 2  
**Delete***byte[]rowArray, introwOffset, introwLength*  
Creates a Delete operation for the specified row and timestamp.
- 3  
**Delete***byte[]rowArray, introwOffset, introwLength, longts*  
Creates a Delete operation for the specified row and timestamp.
- 4  
**Delete***byte[]row, longtimestamp*  
Creates a Delete operation for the specified row and timestamp.

## Methods

### S.No. Methods and Description

- 1  
**Delete addColumn***byte[]family, byte[]qualifier*  
Deletes the latest version of the specified column.
- 2  
**Delete addColumns***byte[]family, byte[]qualifier, longtimestamp*  
Deletes all versions of the specified column with a timestamp less than or equal to the specified timestamp.

3

### **Delete addFamily***byte[]family*

Deletes all versions of all columns of the specified family.

4

### **Delete addFamily***byte[]family, longtimestamp*

Deletes all columns of the specified family with a timestamp less than or equal to the specified timestamp.

## **Class Result**

This class is used to get a single row result of a Get or a Scan query.

## **Constructors**

S.No.	Constructors
-------	--------------

1

### **Result**

Using this constructor, you can create an empty Result with no KeyValue payload; returns null if you call raw Cells.

## **Methods**

S.No.	Methods and Description
-------	-------------------------

1

### **byte[] getValue***byte[]family, byte[]qualifier*

This method is used to get the latest version of the specified column.

2

### **byte[] getRow**

This method is used to retrieve the row key that corresponds to the row from which this Result was created.