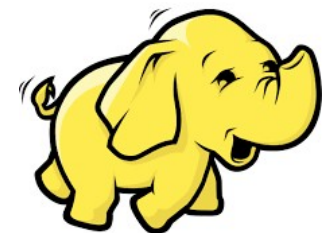


MapReduce API

Tushar B. Kute,
<http://tusharkute.com>



Mapper Class

- In MapReduce, the role of the Mapper class is to map the input key-value pairs to a set of intermediate key-value pairs.
- It transforms the input records into intermediate records.
- These intermediate records associated with a given output key and passed to Reducer for the final output.

Mapper Class Method

- `void cleanup(Context context)`
 - This method called only once at the end of the task.
- `void map(KEYIN key, VALUEIN value, Context context)`
 - This method can be called only once for each key-value in the input split.
- `void run(Context context)`
 - This method can be override to control the execution of the Mapper.
- `void setup(Context context)`
 - This method called only once at the beginning of the task.

Reducer Class

- In MapReduce, the role of the Reducer class is to reduce the set of intermediate values.
- Its implementations can access the Configuration for the job via the `JobContext.getConfiguration()` method.

Reducer Class Methods

- `void cleanup(Context context)`
 - This method called only once at the end of the task.
- `void map(KEYIN key, Iterable<VALUEIN> values, Context context)`
 - This method called only once for each key.
- `void run(Context context)`
 - This method can be used to control the tasks of the Reducer.
- `void setup(Context context)`
 - This method called only once at the beginning of the task.

Job Class

- The Job class is used to configure the job and submits it.
- It also controls the execution and query the state.
- Once the job is submitted, the set method throws IllegalStateException.

Job Class – Methods

- Counters getCounters()
 - This method is used to get the counters for the job.
- long getFinishTime()
 - This method is used to get the finish time for the job.
- Job getInstance()
 - This method is used to generate a new Job without any cluster.
- Job getInstance(Configuration conf)
 - This method is used to generate a new Job without any cluster and provided configuration.

Job Class – Methods

- Job getInstance(Configuration conf, String jobName)
 - This method is used to generate a new Job without any cluster and provided configuration and job name.
- String getJobFile()
 - This method is used to get the path of the submitted job configuration.
- String getJobName()
 - This method is used to get the user-specified job name.
- JobPriority getPriority()
 - This method is used to get the scheduling function of the job.

Job Class – Methods

- `void setJarByClass(Class<?> c)`
 - This method is used to set the jar by providing the class name with .class extension.
- `void setJobName(String name)`
 - This method is used to set the user-specified job name.
- `void setMapOutputKeyClass(Class<?> class)`
 - This method is used to set the key class for the map output data.
- `void setMapOutputValueClass(Class<?> class)`
 - This method is used to set the value class for the map output data.

Job Class – Methods

- `void setMapperClass(Class<? extends Mapper> class)`
 - This method is used to set the Mapper for the job.
- `void setNumReduceTasks(int tasks)`
 - This method is used to set the number of reduce tasks for the job
- `void setReducerClass(Class<? extends Reducer> class)`
 - This method is used to set the Reducer for the job.

Writable Classes – Hadoop Data Types

- Hadoop provides classes that wrap the Java primitive types and implement the WritableComparable and Writable Interfaces.
- They are provided in the org.apache.hadoop.io package.
- All the Writable wrapper classes have a get() and a set() method for retrieving and storing the wrapped value.

Primitive Writable Classes

- These are Writable Wrappers for Java primitive data types and they hold a single primitive value that can be set either at construction or via a setter method.
- All these primitive writable wrappers have get() and set() methods to read or write the wrapped value. Below is the list of primitive writable data types available in Hadoop.
 - BooleanWritable
 - ByteWritable
 - IntWritable
 - FloatWritable
 - LongWritable
 - DoubleWritable

Thank you

This presentation is created using LibreOffice Impress 5.1.6.2, can be used freely as per GNU General Public License



@mitu_skillologies



/mITuSkillologies



@mitu_group



/company/mitu-
skillologies



MITUSkillologies

Web Resources

<https://mitu.co.in>
<http://tusharkute.com>

contact@mitu.co.in
tushar@tusharkute.com