Basic programs of Hadoop MapReduce

→ Driver Code

Program of Hadoop MapReduce

Driver Code

```
Configuration conf = new Configuration( );
Job job = new Job (conf, "My Word Count Program");
Job.setJarByClass (WordCount.class);
Job.setMapperClass (Map.class);
Job.setReducerClass (Reduce.class);
Job.setOutputKeyClass (Text.class);
Job.setOutputValueClass (IntWritable.class);
Job.setInputFormatClass (TextInputFormat.class);
Job.setOutputFormatClass (TextOutputFormat.class);
Path outputPath = new Path (args[0]);
FileInputFormat.addInputPath (job, newPath (args[0]);
FileOutputFormat.setOutputPath (job, new Path (args[1]);
```

Hadoop MapReduce

Driver Code

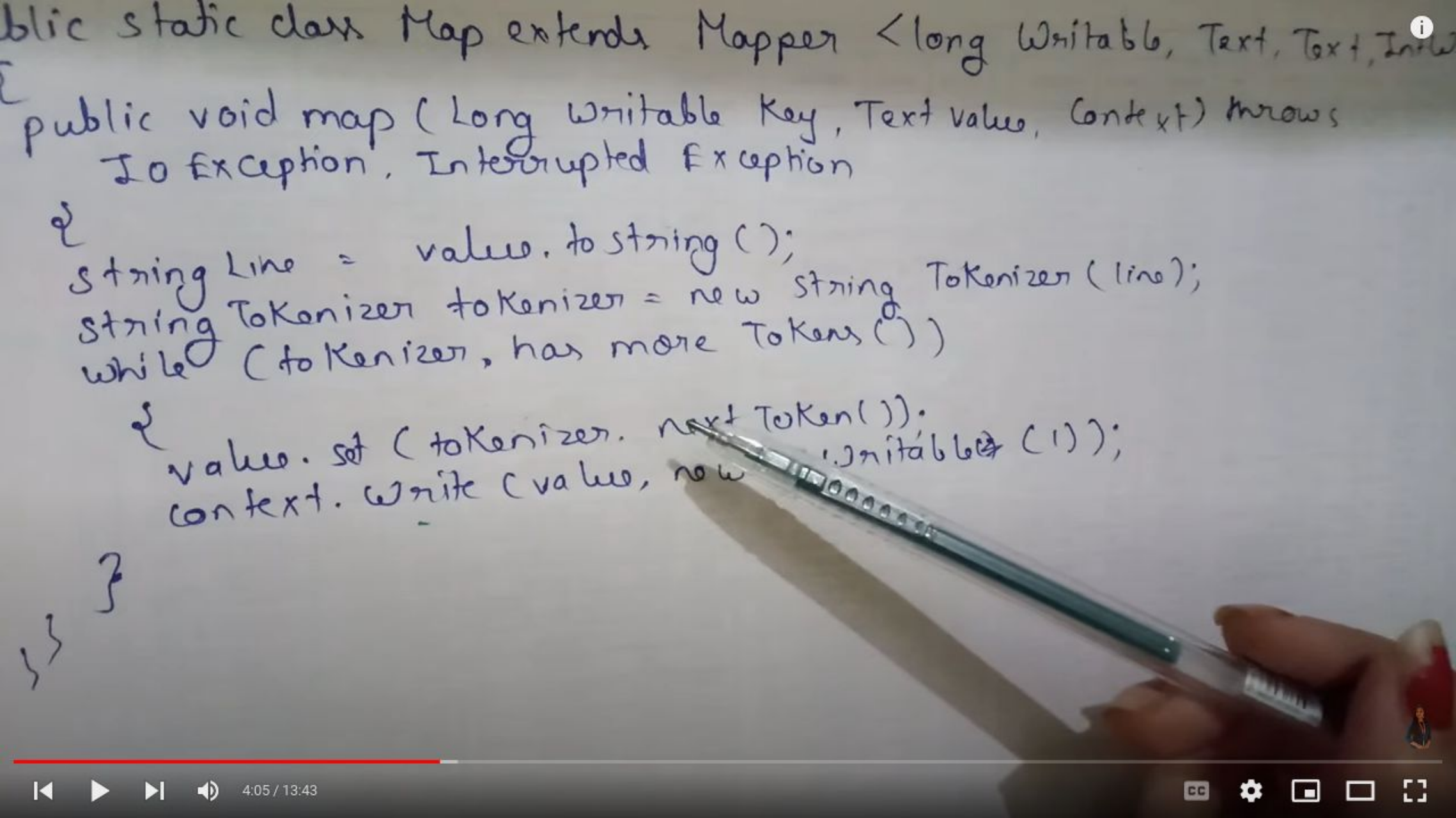
```
Configuration conf = new Configuration();
Job job = new Job(conf, "My Word Count Program");
Job.setJarByClass(WordCount.class);
Job.setMapperClass(Map.class);
Job.setReducerClass(Reduce.class);
Job.setOutputKeyClass(Text.class);
Job.setOutputValueClass(IntWritable.class);
Job.setInputFormatClass(TextOutputInput.class);
Job.setOutputFormatClass(TextOutputFormat.class);
Path outputPath = new Path(args[0]);
FileInputFormat.addInputPath(job, new Path(args[0]));
FileOutputFormat.setOutputPath(job, new Path(args[1]));
```

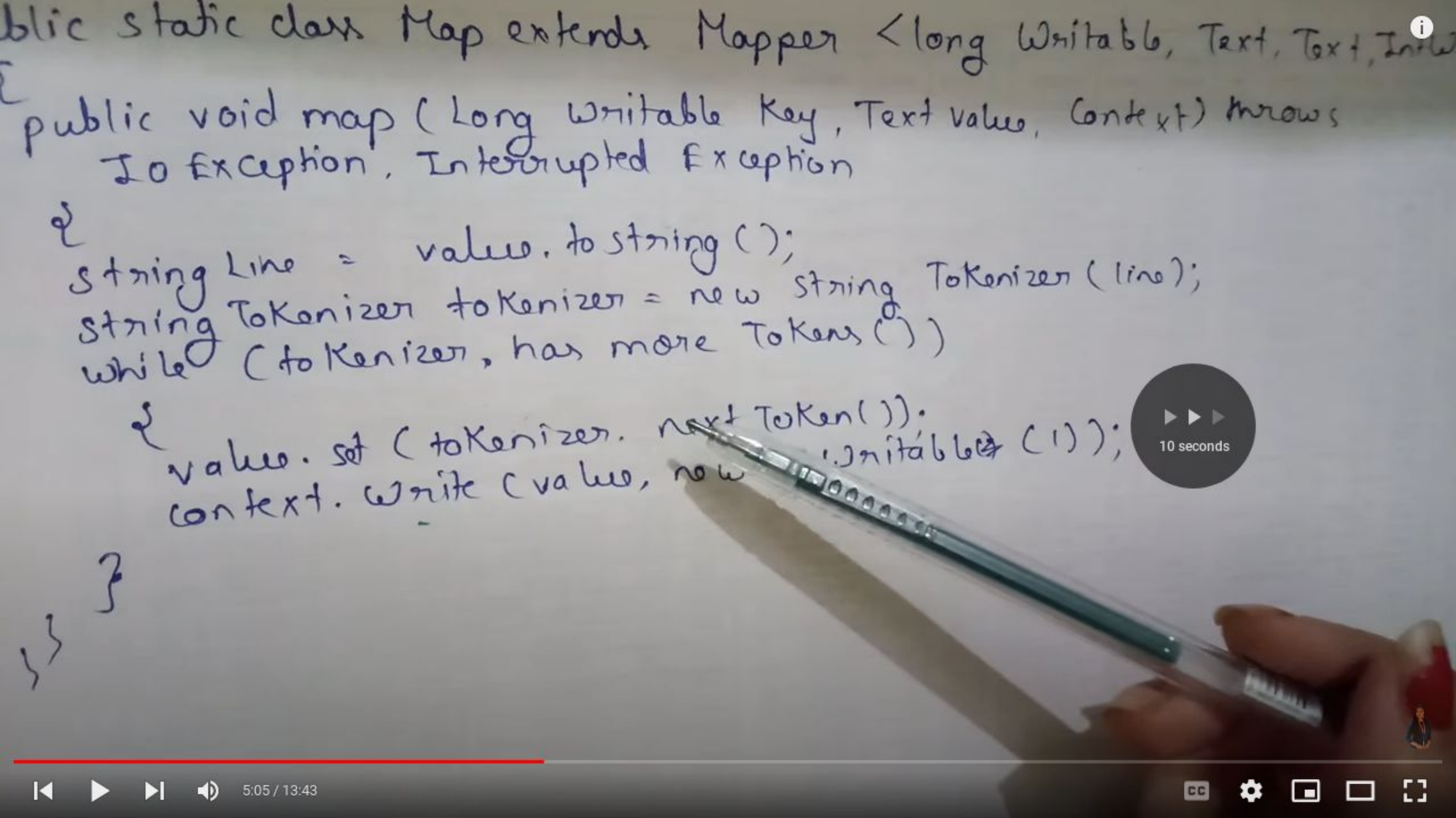
Hadoop MapReduce

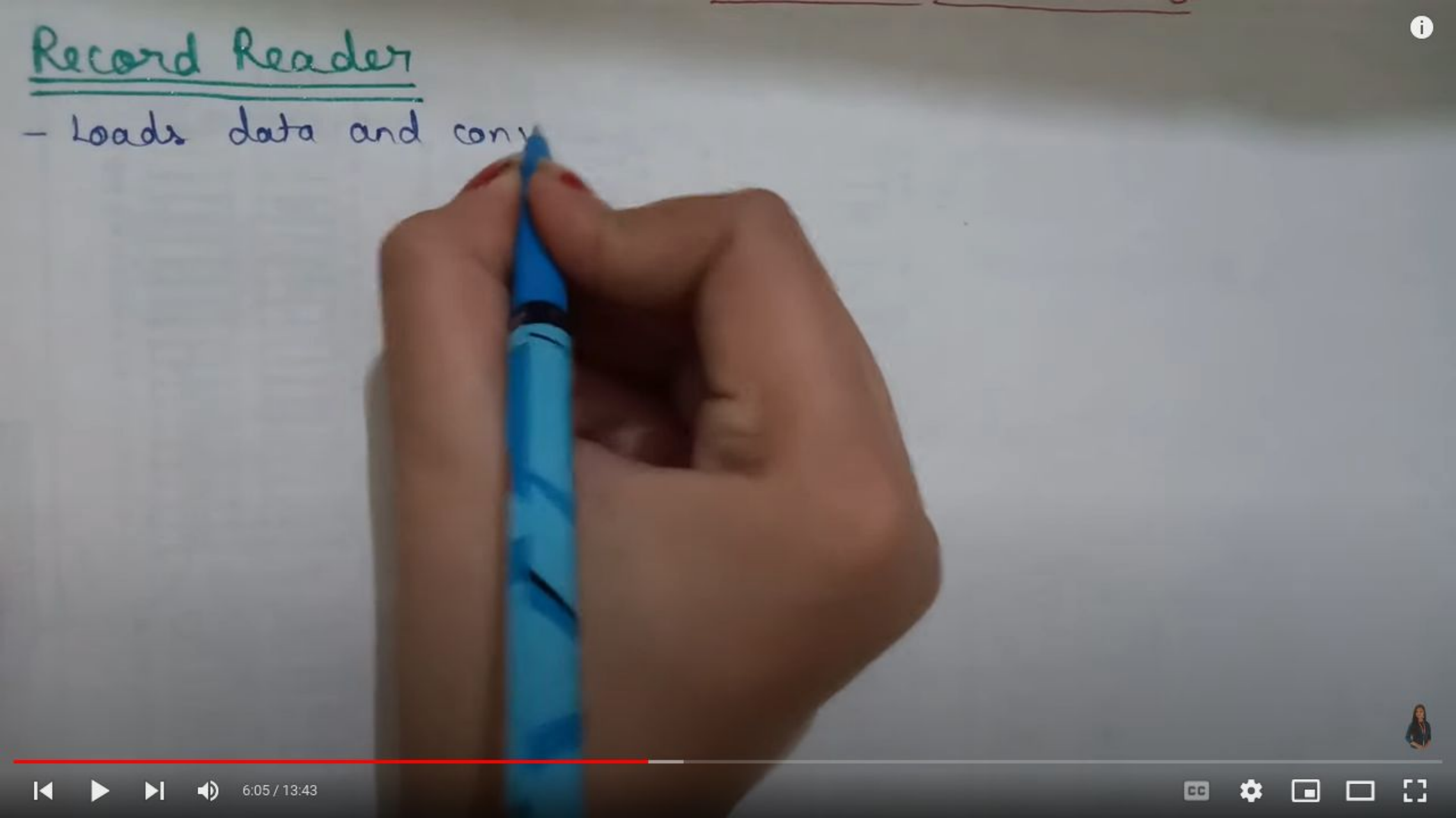
Driver Code

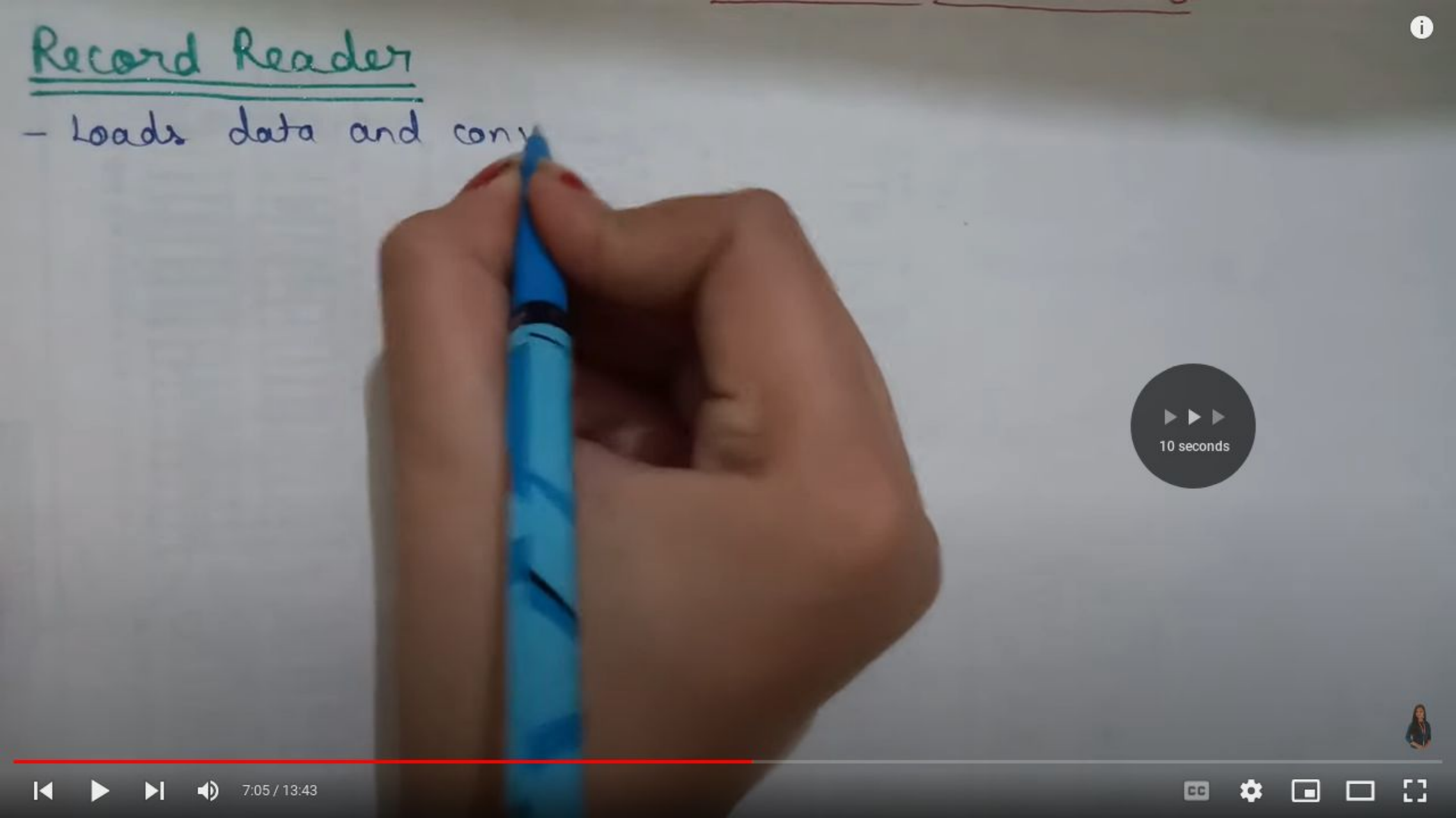
```
Configuration conf = new Configuration();
Job job = new Job(conf, "My Word Count Program");
Job.setJarByClass(WordCount.class);
Job.setMapperClass(Map.class);
Job.setReducerClass(Reduce.class);
Job.setOutputKeyClass(Text.class);
Job.setOutputValueClass(IntWritable.class);
Job.setInputFormatClass(TextOutputInput.class);
Job.setOutputFormatClass(TextOutputFormat.class);
Path outputPath = new Path(args[0]);
FileInputFormat.addInputPath(job, newPath(args[0]));
FileOutputFormat.setOutputPath(job, new Path(args[1]));
```

```
blic static class Map extends Mapper <long Writable, Text, Text, IntW
{
public void map (Long Writable Key, Text value, Context) throws
    IO Exception, Interrupted Exception
{
 String Line = value.to string ();
 String ToKenizer toKenizer = new string ToKenizer (line);
 while (toKenizer, has more ToKens ())
   {
    value. set (toKenizer. next ToKen()).
                                  .Writable (1));
    context. write (value, new
   }
}}
```

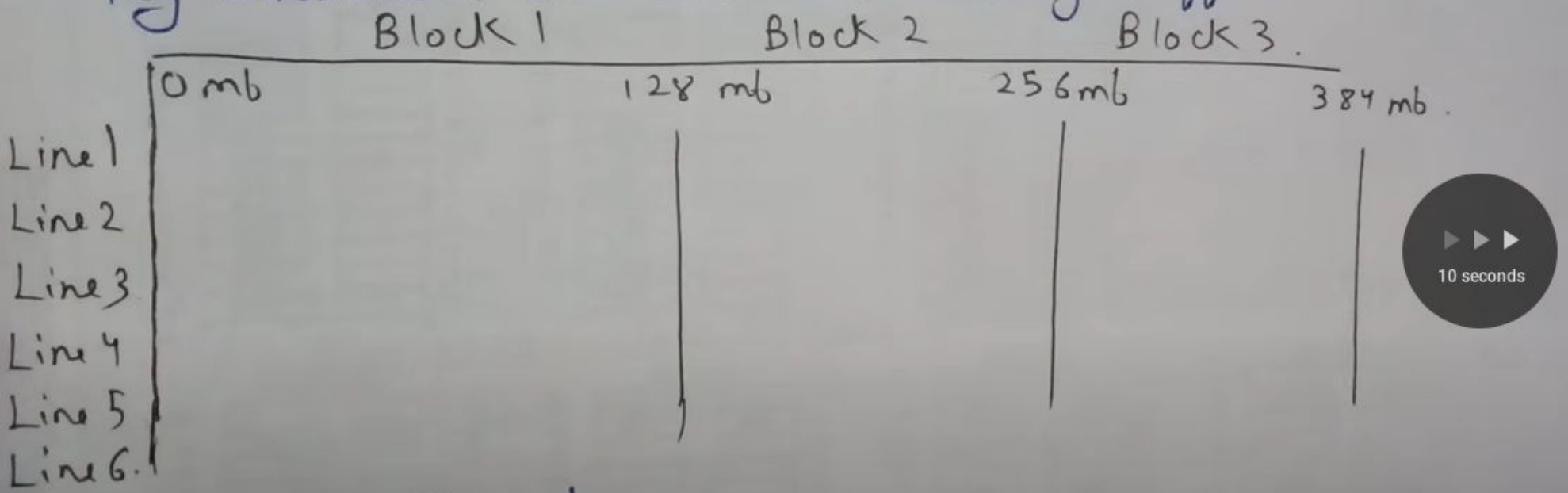



blic static class Map extends Mapper <long Writable, Text, Text, Int
{
public void map (Long Writable Key, Text value, Context) throws
IO Exception, Interrupted Exception
{
String Line = value. to string ();
String Tokenizer tokenizer = new string Tokenizer (line);
while (tokenizer, has more Tokens ())
{
value. set (tokenizer. next Token ()).
.Writable (1));
context. write (value, new
}
}}
10 seconds
5:05 / 13:43



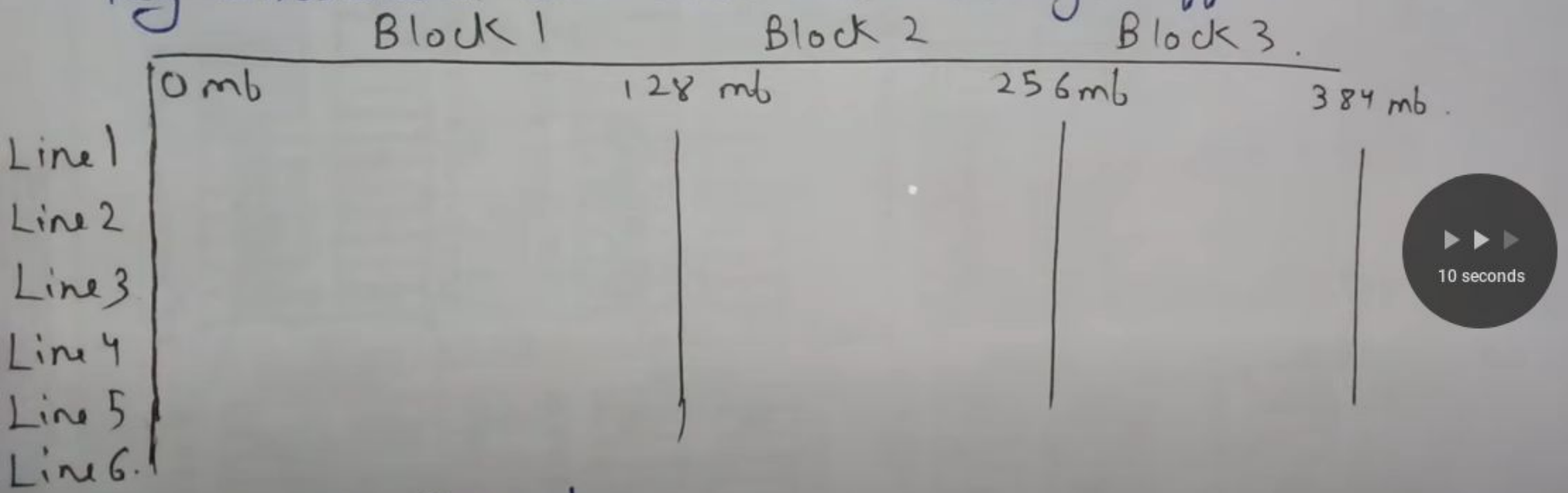

Record Reader
- Loads data and con
10 seconds
7:05 / 13:43

- Loads data and convert into Key Value pair.
- It's instance is defined by the Input format
- Text input format - Default.
- Key associated with each line is its byte offset.



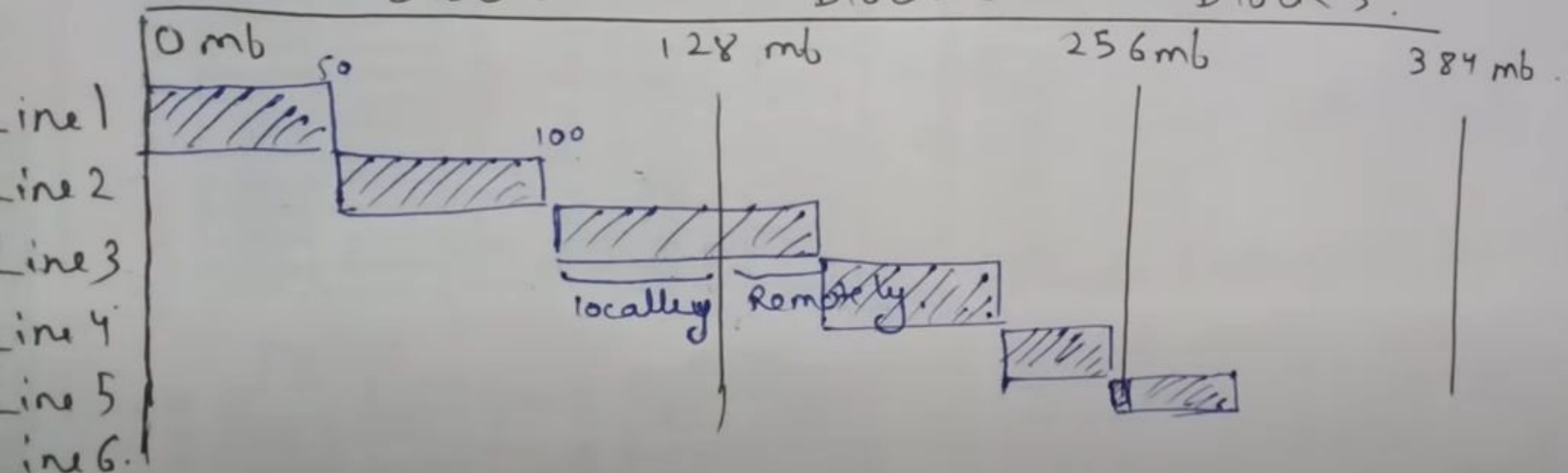
Data set = 300mb.
6 lines - 50 mb.

- Loads data and convert into Key Value pair.
- It's instance is defined by the Input format
- Text input format - Default.
- Key associated with each line is its byte offset.



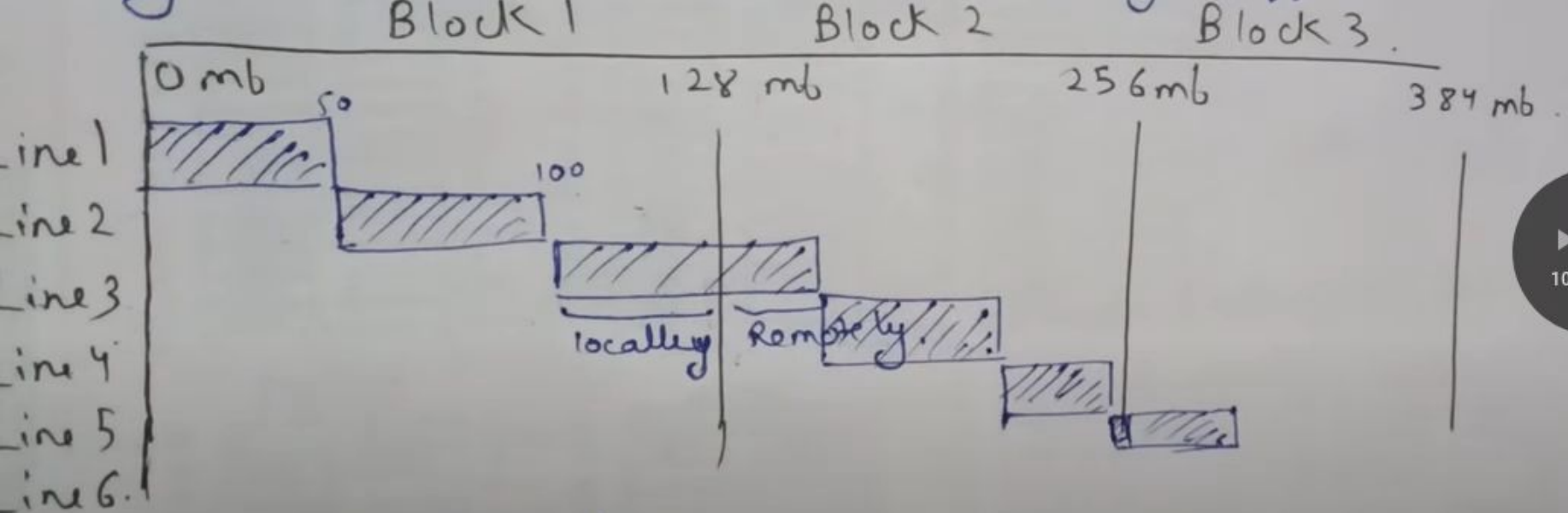
Data set = 300mb.
6 lines - 50 mb.

- Loads data and convert into Key Value pair.
- It's instance is defined by the Input format
- Text input format - Default.
- Key associated with each line is its byte offset.

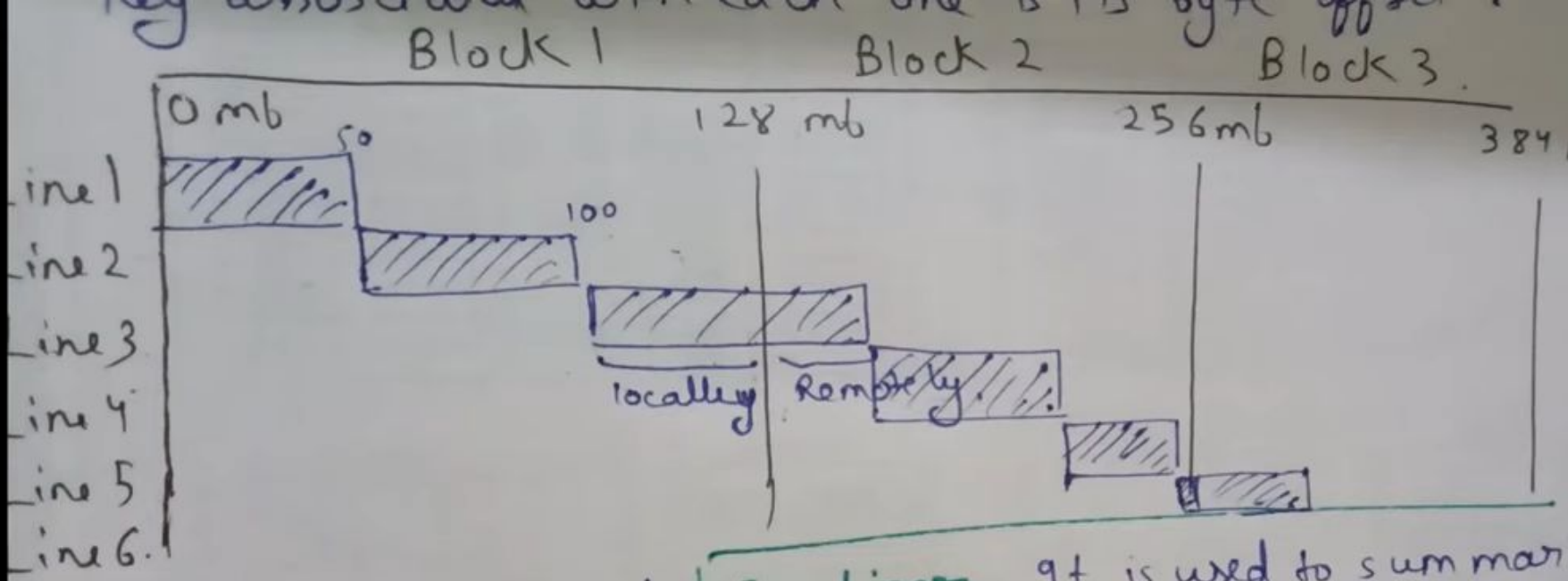


Data set = 300 mb.
6 lines - 50 mb.

- Loads data and convert into Key Value pair.
- It's instance is defined by the Input format
- Text input format - Default.
- Key associated with each line is its byte offset.



Data set = 300 mb.
6 lines - 50 mb.



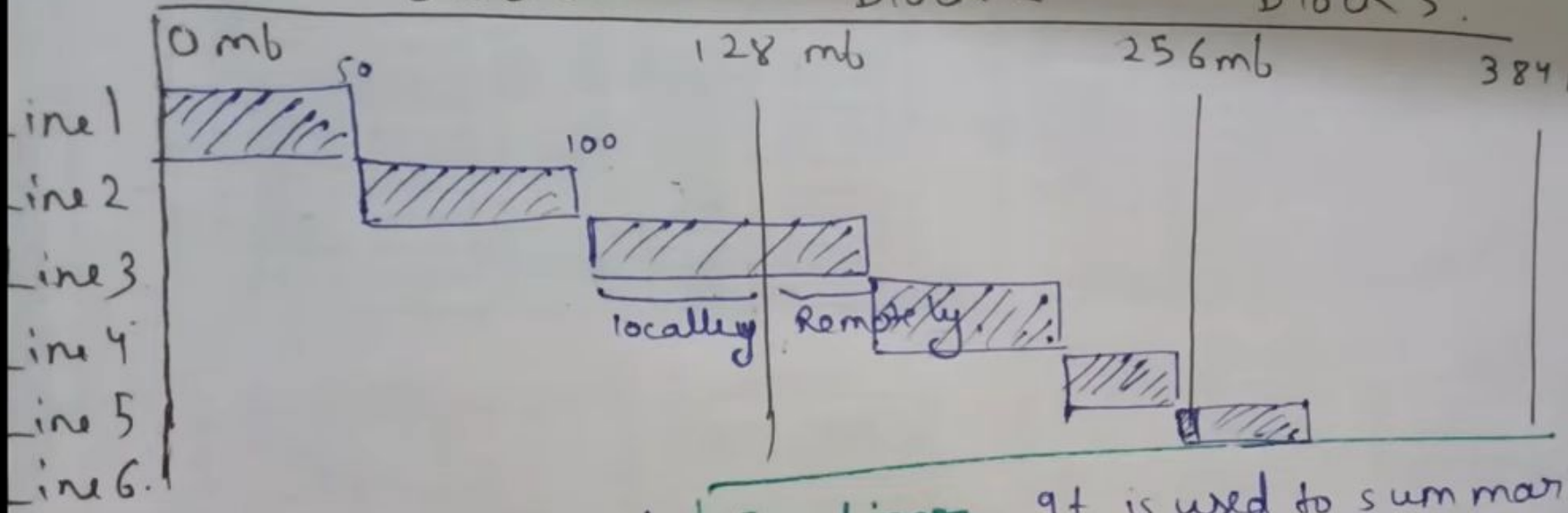
Data set = 300 mb.
6 lines – 50 mb.

Combiner - It is used to summarize the map output records with the same Key.
(semi reducer)

Partitioner - controls the partitioning of the Keys of the Inter

10 seconds

12:05 / 13:43



Data set = 300 mb.
6 lines – 50 mb.

Combiner - It is used to summarize the map output records with the same Key.
(semi reducer)

Partitioner - controls the partitioning of the Keys of the Inter

10 seconds

13:05 / 13:43