

# First Map Reduce Job/Application



# Agenda

- **Introduction to MapReduce framework**
- **Our First MapReduce Job**

# Introduction

- **Divided in two phases**
  - Map phase
  - Reduce phase
- **Uses key-value pairs as input and output**
- **The implementer provides map and reduce functions**
- **Framework takes care of splitting, and distributing of Map and Reduce phases**

# Map Reduce

- **Job** – execution of map and reduce functions to accomplish a task
  - Equal to Java's main
- **Task** – single Mapper or Reducer
  - Performs work on a fragment of data

# First Map Reduce Job

- **WordCount Job**
  - Input is a text file
- **In this case deckofcards.txt**
  - Split text into tokens
  - For each first letter sum up all occurrences
- **Output to local result directory**

# WordCount Job

- **Configure the Job**
  - Specify Input, Output, Mapper, Reducer and Combiner
- **Implement Mapper**
  - Input is text – a line from deckofcards.txt
  - Tokenize the text and emit first character with a count of <token, 1>
- **Implement Reducer**
  - Sum up counts for each letter
  - Write out the result to result directory
- **Run the job**