```
pinstall.packages(“repmis”)
require(repmis)
require(rmr2)
rmr.options(backend = "local") #local or hadoop
url <-  "http://people.terry.uga.edu/rwatson/data/centralparktemps.txt"
t <- source_data(url, header=T, sep=',')
# save temperature in hdfs file
hdfs.temp <-  to.dfs(t$temperature)
# mapper for conversion to C
mapper <-  function(k,v) {
  key <-  round((v-32)*5/9,0)
  value <-  1
  keyval(key,value)
}
# Instead of function(k,v) one can also use (., v) because k is not being
used.


# reducer to count frequencies. Here the two arguments to function, k & v
correspond to key and value from the map function.
reducer <- function(k,v) {
 key <- k
 value = length(v)
 keyval(key,value)
}
out = mapreduce(
 input = hdfs.temp,
 map = mapper,
 reduce = reducer)
df2 = as.data.frame(from.dfs(out))
colnames(df2) = c('temperature', 'count')
df2[order(df2$temperature),]
```