**Chương trình mapreduce (2 jobs)**

**- Mapper  
 + compare\_avg\_price\_mapper.py**

import sys

import json

import os

# map input file name (Hadoop env variable usually)

input\_file = os.environ.get('mapreduce\_map\_input\_file', '')

for line in sys.stdin:

try:

record = json.loads(line)

price = float(record.get('price', 0))

# emit with file tag

if 'books\_clean.json' in input\_file:

print(f"books\t{price}\t1")

elif 'tiki\_book\_data\_clean.json' in input\_file:

print(f"tiki\t{price}\t1")

except:

continue

**+ count\_rating\_mapper.py**

import sys

import json

for line in sys.stdin:

try:

record = json.loads(line)

rating = record.get('rating', None)

if rating is not None:

print(f"{rating}\t1")

except:

continue

* **Reducer**

**+ compare\_avg\_price\_reducer.py**

import sys

sum\_books = 0.0

count\_books = 0

sum\_tiki = 0.0

count\_tiki = 0

for line in sys.stdin:

try:

tag, price, cnt = line.strip().split('\t')

price = float(price)

cnt = int(cnt)

if tag == 'books':

sum\_books += price

count\_books += cnt

elif tag == 'tiki':

sum\_tiki += price

count\_tiki += cnt

except:

continue

avg\_books = sum\_books / count\_books if count\_books > 0 else 0

avg\_tiki = sum\_tiki / count\_tiki if count\_tiki > 0 else 0

print(f"Books\_Avg\_Price\t{avg\_books:.2f}")

print(f"Tiki\_Avg\_Price\t{avg\_tiki:.2f}")

**+ count\_rating\_reducer.py**

import sys

current\_rating = None

count = 0

for line in sys.stdin:

try:

rating, cnt = line.strip().split('\t')

cnt = int(cnt)

if current\_rating == rating:

count += cnt

else:

if current\_rating is not None:

print(f"{current\_rating}\t{count}")

current\_rating = rating

count = cnt

except:

continue

if current\_rating is not None:

print(f"{current\_rating}\t{count}")