

Business Problem – Assignment 2

Task 1: Demand-Supply Mismatch Analysis

Objective: Identify zones and regional zones with the highest mismatch between demand and supply.

Required Fields: zone, WH_regional_zone, product_wg_ton

--mapper.py :

```
#!/usr/bin/python3

"""mapper.py"""

import sys

import csv

for row in csv.reader(sys.stdin):

    print("%s\t%s\t%s"%(row[4],row[5],row[23]))
```

--reduce.py :

```
#!/usr/bin/python3

"""reducer.py"""

import sys

import csv

data = {}

for line in sys.stdin:

    zone, regional_zone, product_shipped = line.strip().split("\t")

    try:

        product_shipped = float(product_shipped)

    except ValueError:
```

```
continue
```

```
if zone in data:
```

```
    if regional_zone in data[zone]:
```

```
        data[zone][regional_zone] += product_shipped
```

```
    else:
```

```
        data[zone][regional_zone] = product_shipped
```

```
else:
```

```
    data[zone] = {regional_zone: product_shipped}
```

```
for zone in data:
```

```
    for regional_zone in data[zone]:
```

```
        print("%s\t%s\t%s" % (zone, regional_zone, data[zone][regional_zone]))
```

```
hadoop@hadoop-VirtualBox:~/mapreduce/task1$ hadoop fs -cat /assignment/output/part-00000
24/07/02 22:34:48 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform...
using builtin-java classes where applicable
East   Zone 3   33055.0
East   Zone 1   25139.0
East   Zone 5   16084.0
East   Zone 4   7090.0
East   Zone 6   31055.0
North  Zone 5   29086.0
North  Zone 6   26138.0
North  Zone 1   17112.0
North  Zone 2   21141.0
North  Zone 3   26127.0
North  Zone 4   24059.0
South  Zone 5   15121.0
South  Zone 2   18132.0
South  Zone 1   14070.0
South  Zone 4   12109.0
South  Zone 3   12067.0
South  Zone 6   10093.0
West   Zone 4   12127.0
West   Zone 2   17134.0
West   Zone 5   15125.0
West   Zone 6   17115.0
West   Zone 1   32134.0
West   Zone 3   8127.0
hadoop@hadoop-VirtualBox:~/mapreduce/task1$
```

Activate Windows
Go to Settings to activate Windows.

Task 2: Warehouse Refill Frequency Correlation

Objective: Determine the correlation between warehouse capacity and refill frequency.

Required Fields: WH_capacity_size, num_refill_req_l3m

--mapper.py:

```
#!/usr/bin/python3
```

```
"""mapper.py"""
```

```
import sys
```

```
import csv
```

```
for row in csv.reader(sys.stdin):
```

```
    print("%s\t%s"%(row[3],row[6]))
```

--reducer.py:

```
#!/usr/bin/python3
```

```
"""reducer.py"""
```

```
import sys
```

```
warehouse_data = {}
```

```
for line in sys.stdin:
```

```
    capacity, refill = line.strip().split("\t")
```

```
    try:
```

```
        refill = int(refill)
```

```
    except ValueError:
```

```
        continue
```

```
    if capacity in warehouse_data:
```

```
        warehouse_data[capacity].append(refill)
```

```
    else:
```

```
        warehouse_data[capacity] = [refill]
```

for warehouse in warehouse_data:

total_refill = sum(warehouse_data[warehouse])

count_refill = len(warehouse_data[warehouse])

print("%s\t%s\t%s" % (warehouse, total_refill, count_refill))

```
hadoop@hadoop-VirtualBox:~/mapreduce/task2$ hadoop fs -cat /assignment2/output/part-00000
24/07/02 22:52:51 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform...
using builtin-java classes where applicable
Large  41630  10169
Mid    41217  10020
Small  19379  4811
hadoop@hadoop-VirtualBox:~/mapreduce/task2$
```

Activate Windows
Go to Settings to activate Windows.

Task 3. Transport Issue Impact Analysis

Objective: Analyse the impact of transport issues on warehouse supply efficiency.

Required Fields: transport_issue_l1y, product_wg_ton

--mapper :

#!/usr/bin/python3

"""mapper.py"""

import sys

import csv

for row in csv.reader(sys.stdin):

print("%s\t%s"%(row[10],row[22]))

--reducer :

#!/usr/bin/python3

"""reducer.py"""

import sys

```

dict={}

for line in sys.stdin:

    transport, weight = line.strip().split("\t")

    try:

        weight = float(weight)

    except ValueError:

        continue

    if transport in dict:

        dict[transport]+=weight

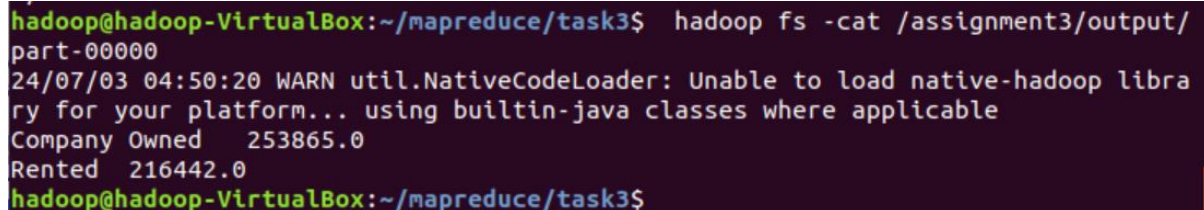
    else:

        dict[transport]=weight


for i in dict:

    print("%s\t%s"%(i, dict[i]))

```



```

hadoop@hadoop-VirtualBox:~/mapreduce/task3$ hadoop fs -cat /assignment3/output/part-00000
24/07/03 04:50:20 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Company Owned 253865.0
Rented 216442.0
hadoop@hadoop-VirtualBox:~/mapreduce/task3$

```

Task 4. Storage Issue Analysis

Objective: Evaluate the impact of storage issues on warehouse performance.

Required Fields: storage_issue_reported_l3m, product_wg_ton

--mapper :

#!/usr/bin/python3

"""mapper.py"""

import sys

import csv

```

for row in csv.reader(sys.stdin):

    print("%s\t%s"%(row[19],row[22]))

--reducer :

#!/usr/bin/python3

"""reducer.py"""

import sys

dict={}

for line in sys.stdin:

    storage, weight = line.strip().split("\t")

    try:

        weight = float(weight)

    except ValueError:

        continue

    if storage in dict:

        dict[storage].append(weight)

    else:

        dict[storage]=[weight]

for i in dict:

    print("%s\t%s\t%s"%(i, sum(dict[i]), sum(dict[i])/len(dict[i]))))

```

```

hadoop@hadoop-VirtualBox:~/mapreduce/task4$ hadoop fs -cat /assignment4/output/p
art-00000
24/07/03 04:30:10 WARN util.NativeCodeLoader: Unable to load native-hadoop libra
ry for your platform... using builtin-java classes where applicable
0      327281.0      18.789815133769665
1      143026.0      18.863888156159323
hadoop@hadoop-VirtualBox:~/mapreduce/task4$

```