

1/

### **Mapper 3 :**

```
#!/usr/bin/env python
```

```
import sys
```

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

```
    purchases, time, store, item, cost, payment = line.strip().split("\t")
```

```
    print("{0}\t{1}".format(store, cost))
```

### **Reducer 3 :**

```
#!/usr/bin/env python
```

```
import sys
```

```
current_store = None
```

```
total_sales = 0
```

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

```
    store, cost = line.strip().split("\t")
```

```
    cost = float(cost)
```

```
    if store != current_store:
```

```
        if current_store is not None:
```

```
            print("{0}\t{1}".format(current_store, total_sales))
```

```
current_store = store
total_sales = 0
    total_sales += cost
```

```
if current_store is not None:
    print("{0}\t{1}".format(current_store, total_sales))
```

2/

#### **Mapper 4 :**

```
#!/usr/bin/env python
import sys

for line in sys.stdin:
    purchases, time, store, item, cost, payment = line.strip().split("\t")

    print("{0}\t{1}".format(item, cost))
```

#### **Reducer 4 :**

```
#!/usr/bin/env python

import sys

current_item = None
total_sales = 0

for line in sys.stdin:
```

```
item, cost = line.strip().split("\t")
cost = float(cost)
if item != current_item:

    if current_item is not None:
        print("{0}\t{1}".format(current_item, total_sales))
```

```
current_item = item
total_sales = 0
total_sales += cost
```

```
if current_item is not None:
    print("{0}\t{1}".format(current_item, total_sales))
3/
```

### **Mapper3:**

```
#!/usr/bin/env python
```

```
import sys
```

```
for line in sys.stdin:
    purchases, time, store, item, cost, payment = line.strip().split("\t")

    print("{0}\t{1}".format(store, cost))
```

### **Reducer 5 :**

```
#!/usr/bin/env python
```

```
import sys

total_sales_count = 0
total_sales_value = 0

current_item = None
total_sales = 0

for line in sys.stdin:
    item, cost = line.strip().split("\t")
    cost = float(cost)

    # Increment total sales count
    total_sales_count += 1

    # Accumulate total sales value
    total_sales_value += cost

    if item != current_item:
        if current_item is not None:
            # Imprimer le total des ventes pour l'article actuel
            print("{0}\t{1}".format(current_item, total_sales))

        current_item = item
        total_sales = 0

    total_sales += cost

if current_item is not None:
    print("{0}\t{1}".format(current_item, total_sales))
```

```
print("Nombre total des ventes : {0}".format(total_sales_count))  
print("Valeur totale des ventes : {0}".format(total_sales_value))
```

Remarque : on peut utiliser aussi la commande grep sous hadoop fs -cat