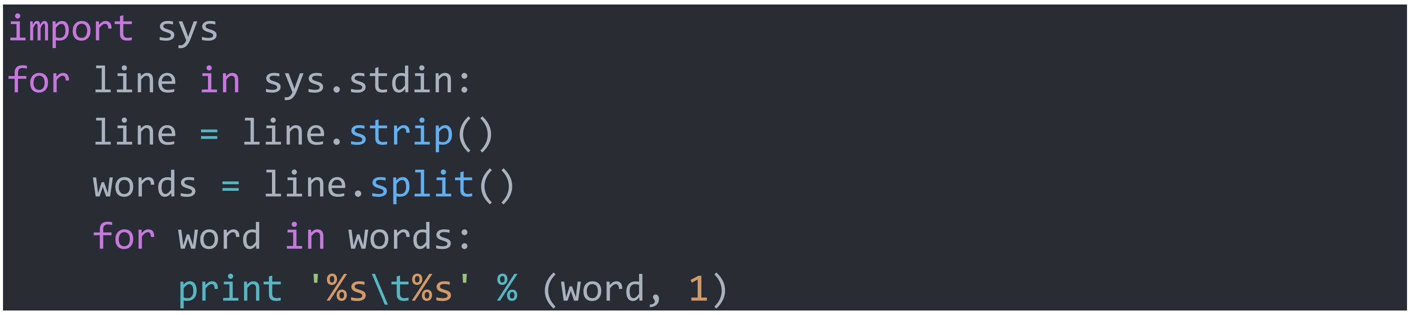
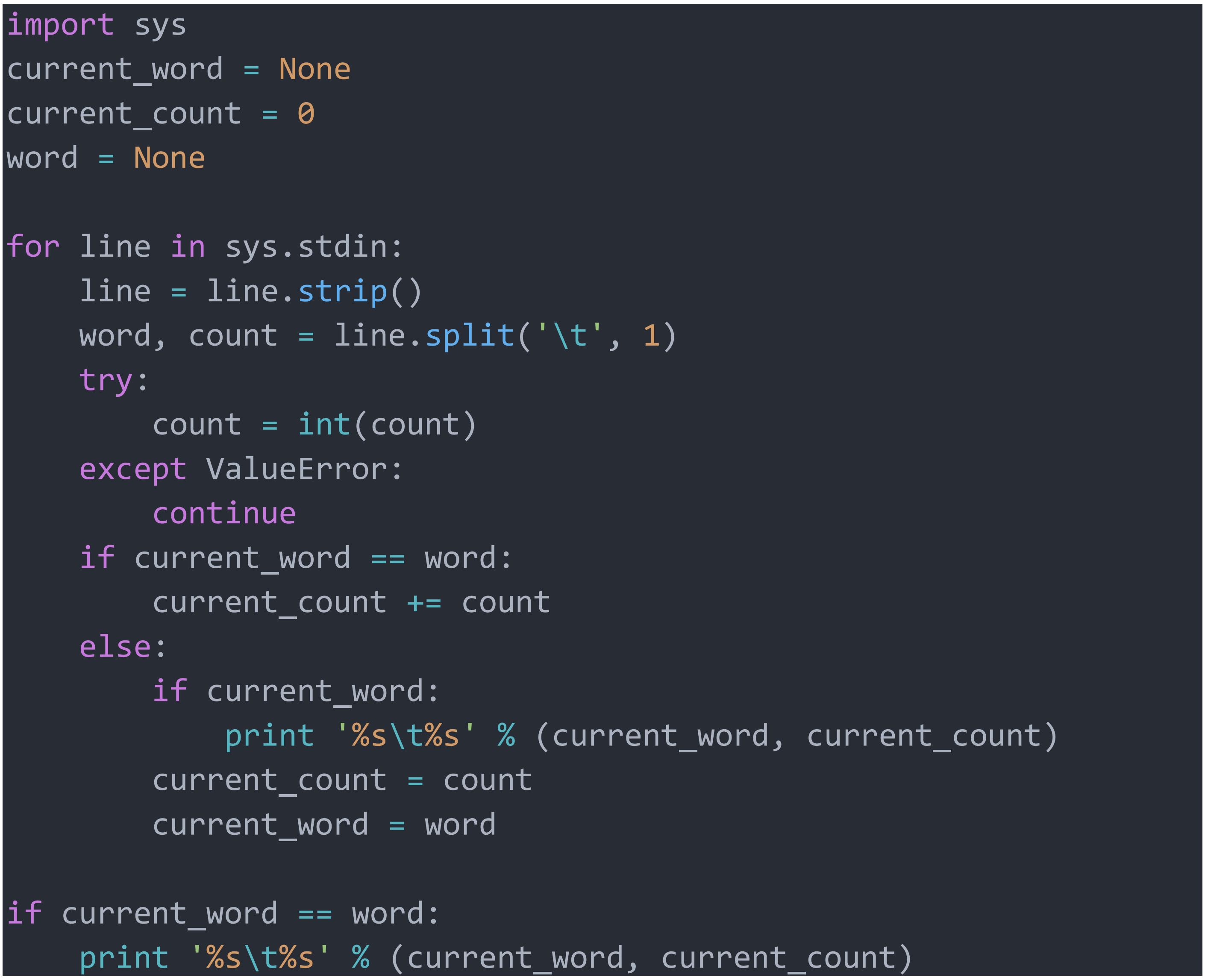
1. **Map Reduce Word Count Program**

mapper.py



```
import sys
for line in sys.stdin:
    line = line.strip()
    words = line.split()
    for word in words:
        print '%s\t%s' % (word, 1)
```

reducer.py



```
import sys
current_word = None
current_count = 0
word = None

for line in sys.stdin:
    line = line.strip()
    word, count = line.split('\t', 1)
    try:
        count = int(count)
    except ValueError:
        continue
    if current_word == word:
        current_count += count
    else:
        if current_word:
            print '%s\t%s' % (current_word, current_count)
        current_count = count
        current_word = word

if current_word == word:
    print '%s\t%s' % (current_word, current_count)
```

2. Map Reduce Temperature NCDC dataset

temp-mapper.py

```python
import re
import sys
for line in sys.stdin:
    val = line.strip()
    (year, temp, q) = (val[15:19], val[87:92], val[92:93])
    if (temp != "+9999" and re.match("[01459]", q)):
        print "%st%s" % (year, temp)
```

temp-reducer.py

```python
import sys
(last_key, max_val) = (None, 0)
for line in sys.stdin:
    (key, val) = line.strip().split("\t")
    if last_key and last_key != key:
        print "%st%s" % (last_key, max_val)
        (last_key, max_val) = (key, int(val))
    else:
        (last_key, max_val) = (key, max(max_val, int(val)))
if last_key:
    print "%st%s" % (last_key, max_val)
```

3. Map Reduce Stock Analysis

mapper.py

```python
import sys

for line in sys.stdin:
    line = line.strip()
    data = line.split(",")
    stock, price = data[1], data[6]
    print("%s\t%s"%(stock,price))
```

reducer.py

```python
import sys

max_price = 9999
max_stock = None

for line in sys.stdin:
    line = line.strip()
    stock,price = line.split("\t",1)

    if max_stock and max_stock!=stock:
        if max_price > price:
            max_price = price
            max_stock = stock

    else:
        max_stock, max_price = stock,max(max_price,price)

if max_stock:
    if max_price > price:
            max_price = price
            max_stock = stock

print("%s\t%s"%(max_stock,max_price))
```

4. Pig Wordcount

input2 = LOAD '/data2.txt' AS (line:chararray);

words = FOREACH input2 GENERATE FLATTEN(TOKENIZE(line)) AS word;

grpd = GROUP words BY word;

cntd = FOREACH grpd GENERATE group AS word, COUNT(words) AS count;

DUMP cntd;

pig -x local wc.pig

pig -x mapreduce wc.pig

5. Pig Maxtemprecords = LOAD 'sample.txt'

```
AS (year:chararray, temperature:int, quality:int);

filtered_records = FILTER records BY temperature != 9999 AND (quality == 0 OR quality == 1 OR quality == 4 OR quality == 5 OR quality == 9);

grouped_records = GROUP filtered_records BY year;

max_temp = FOREACH grouped_records GENERATE group,

MAX(filtered_records.temperature);

DUMP max_temp;
```

6. Pig Number of products in each country

```
salesTable = LOAD '/SalesJan2009.csv' USING PigStorage(',') AS

(Transaction_date:chararray,Product:chararray,Price:chararray,Payment_Type:chararray,Name:chararray,
City:chararray,State:chararray,Country:chararray,Account_Created:chararray,Last_Login:chararray,Latitude:chararray,Longitude:chararray);

GroupByCountry = GROUP salesTable BY Country;

CountByCountry = FOREACH GroupByCountry GENERATE
CONCAT((chararray)$0,CONCAT(':',(chararray)COUNT($1)));

STORE CountByCountry INTO 'pig_output_sales' USING PigStorage('\t');
```