

FIAP - FACULDADE DE INFORMÁTICA E ADMINISTRAÇÃO PAULISTA
DATA SCIENCE

Fábio Pereira de Lima – RM98803
Giovanna Cardoso Satorres – RM99944
Giullia Bianca Rocha Souza – RM552108
Gustavo Semenuk – RM550472
Mayara Prado Góes – RM98758

Sprint 2: Solução de Extração de Dados para o
Challenge MinSait

BIG DATA ARCHITECTURE & DATA INTEGRATION

São Paulo
2023

★ ALGORITMO DE MAP REDUCE

→ Código

```
1 from collections import Counter
2 import re
3
4 # Função de mapeamento
5 def mapper(line):
6
7     # Remove pontuações e transforma em minúsculo
8     line = re.sub(r'^\w\s', '', line).lower()
9
10    # Separa as palavras
11    words = line.split()
12
13    # Conta as ocorrências de cada palavra
14    return Counter(words)
15
16 # Função de redução
17 def reducer(counters):
18
19    # Combina os contadores de todas as linhas
20    return sum(counters, Counter())
21
22 # Lê o arquivo e aplica o MapReduce
23 with open('/Users/giulliabianca/Downloads/frase.txt') as f:
24     lines = f.readlines()
25
26 # Obtem a primeira linha do arquivo como a frase
27 frase = lines[0].strip()
28
29 # Mapeia cada linha para um contador de palavras
30 mapped = map(mapper, lines)
31
32 # Reduz todos os contadores em um único
33 reduced = reducer(mapped)
34
35 # Imprime o resultado
36 print(f"{frase}: {reduced}")
```

→ Resultado

```
0 cachorro late para a lua, mas a lua não responde. 0 cachorro fica triste quando a lua não responde.:  
Counter({'a': 3, 'lua': 3, 'o': 2, 'cachorro': 2, 'não': 2, 'responde': 2, 'late': 1, 'para': 1, 'mas': 1,  
'fica': 1, 'triste': 1, 'quando': 1})
```

```
Process finished with exit code 0
```