



CERTIFICATE

## This is to certify that

Mr. **SONI SOHAM** **ALPESHKUMAR** with enrolment no.

**210305105530** has successfully completed his laboratory

experiments in the **BIG DATA ANALYSIS (203105348)** from the department of **COMPUTER SCIENCE & ENGINEERING** during the academic year 2024-2025.



Date of Submission:......................... Staff In charge:...........................

Head Of Department:......................................

**INDEX**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Sr. No** | **Experiment Title** | **Page No** | | **Date of Start** | **Date of Completion** | **Sign** | **Marks (out of 10)** |
| **From** | **To** |
| **1.** | **To understand the overall programming architecture using Map Reduce API.** | **4** | **5** | **15-06-24** | **22-06-24** |  |  |
| **2.** | **Write a program of Word Count in Map Reduce over HDFS.** | **6** | **9** | **22-06-24** | **29-06-24** |  |  |
| **3.** |  |  |  |  |  |  |  |
| **4.** |  |  |  |  |  |  |  |
| **5.** |  |  |  |  |  |  |  |
| **6.** |  |  |  |  |  |  |  |
| **7.** |  |  |  |  |  |  |  |
| **8.** |  |  |  |  |  |  |  |
| **9.** |  |  |  |  |  |  |  |
| **10.** |  |  |  |  |  |  |  |

**Practical-1**

**Aim: To understand the overall programming architecture using Map Reduce API.**

The MapReduce task is mainly divided into two phases Map Phase and Reduce Phase.

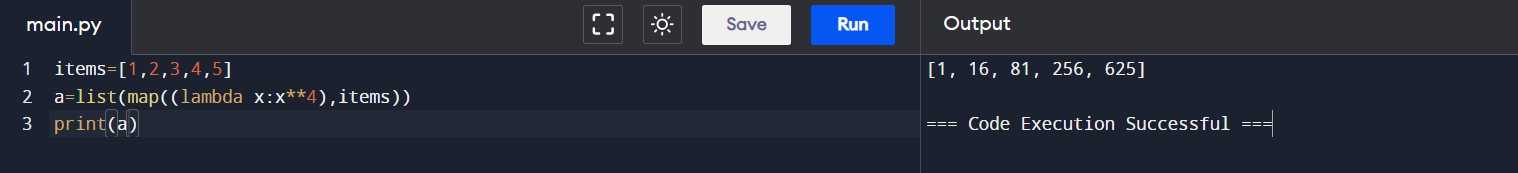
1. map(), filter() and reduce() in python.
2. These functions are most commonly used Lambda function.

**1. map():**

“A map function executes certain instructions or functionality provided to it on every item of an iterable.” The iterable could be a list, tuple, set, etc.

SYNTAX:

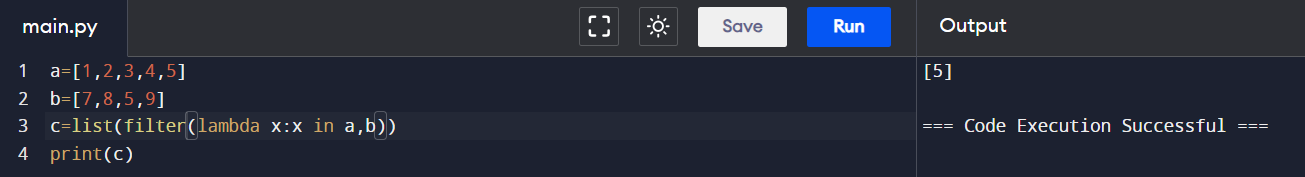
map(function, iterable)



**2. filter():**

“A filter function in python tests a specific user-defined condition for a function and returns an iterable for the elements and values that satisfy the condition.

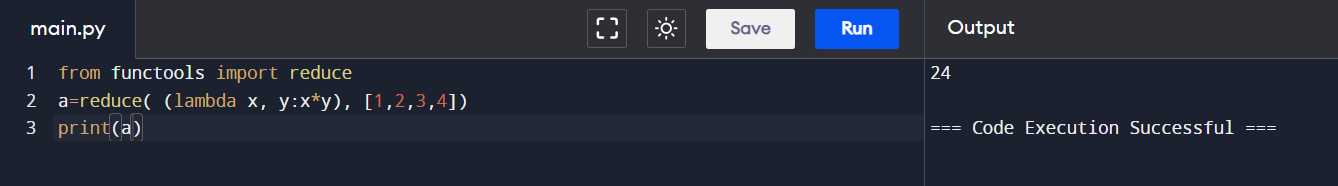
SYNTAX:

filter(function, iterable)

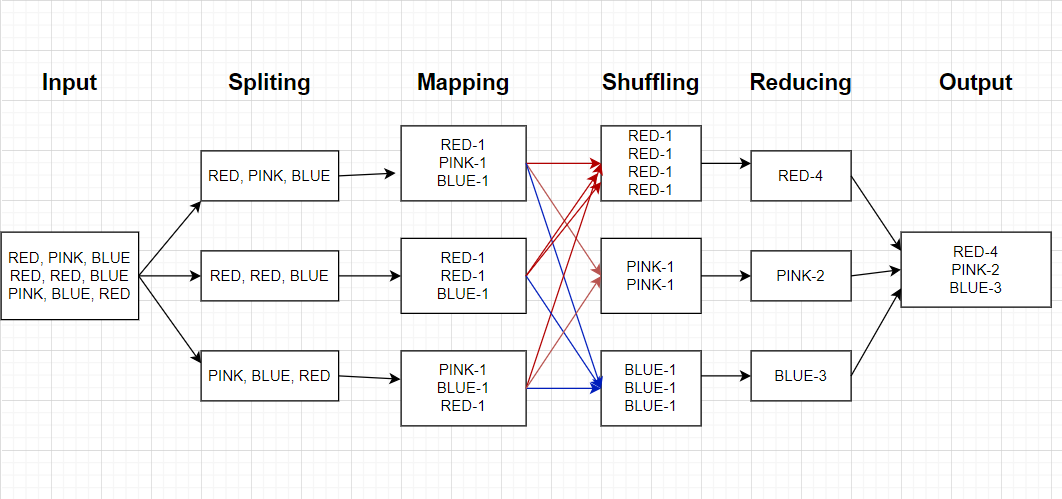
**3. reduce():**

“Reduce functions apply a function to every item of an iterable and gives back a single value as a resultant.”

SYNTAX:  
filter(function, iterable)



* **Map Reduce Architecture:**



**Practical-2**

**Aim: Write a program of Word Count in Map Reduce over HDFS.**

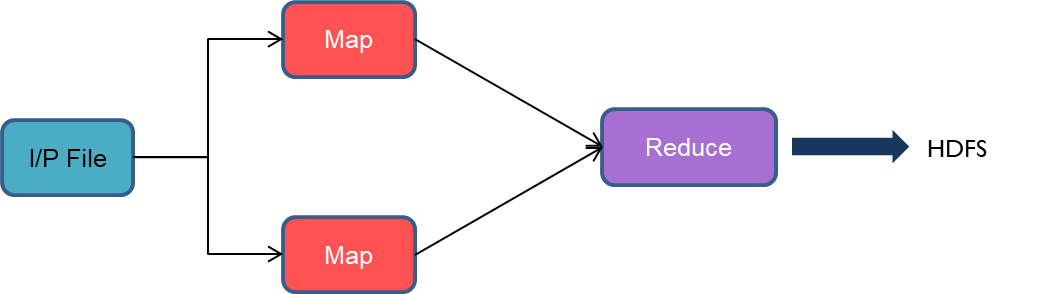
**Description:**

MapReduce is a framework for processing large datasets using a large number of computers (nodes), collectively referred to as a cluster. Processing can occur on data stored in a file system (HDFS).A method for distributing computation across multiple nodes.Each node processes the data that is stored at that node.

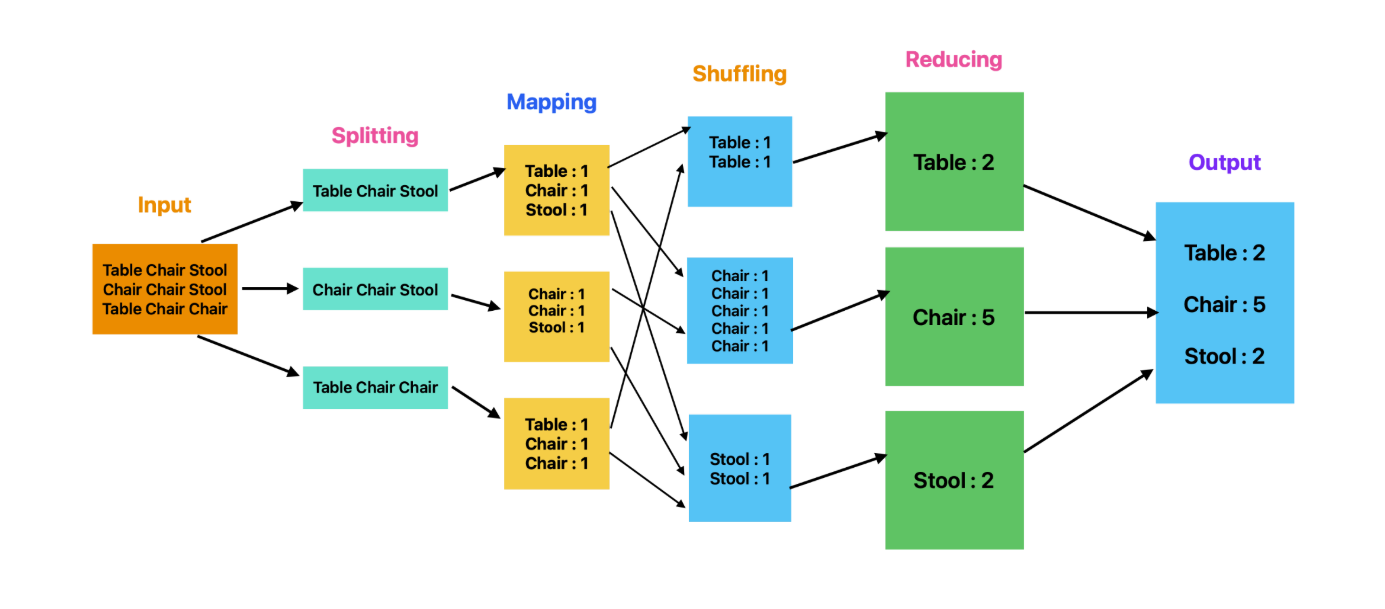
Consists of two main phases

Mapper Phase

Reduce phase



Input data set is split into independent blocks – processed in parallel. Each input split is converted in Key Value pairs. Mapper logic processes each key value pair and produces and intermediate key value pairs based on the implementation logic. Resultant key value pairs can be of different type from that of input key value pairs. The output of Mapper is passed to the reducer. Output of Mapper function is the input for Reducer. Reducer sorts the intermediate key value pairs. Applies reducer logic upon the key value pairs and produces the output in desired format.Output is stored in HDFS



**Execution Step:**

**Code:1**

countOfWords **=** len("Geeksforgeeks is best Computer Science Portal".split())

**print**("Count of Words in the given Sentence:", countOfWords)

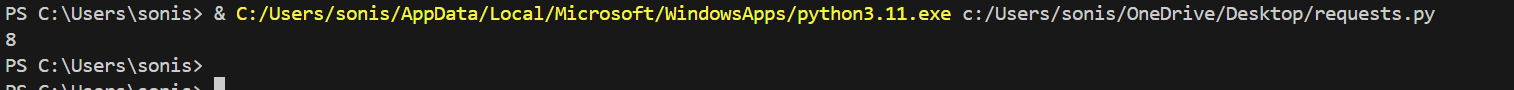
**Output:**

****

**Code:2**

print(len("Soham Soni is a student of parul university".split()))

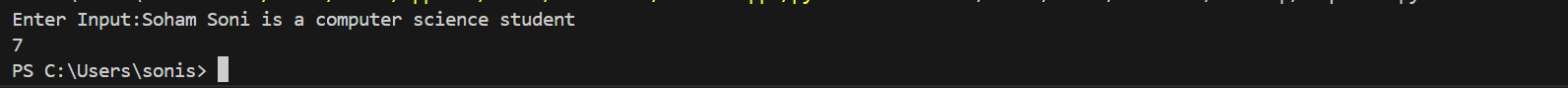
**Output:2**



**Code:3**

print(len(input("Enter Input:").split()))

**Output:3**



**Code:4**

from collections import Counter

def word\_count(file\_path):

    with open(file\_path, 'r') as file:

        text = file.read()

    words = text.split()

    word\_counts = Counter(words)

    for word, count in word\_counts.items():

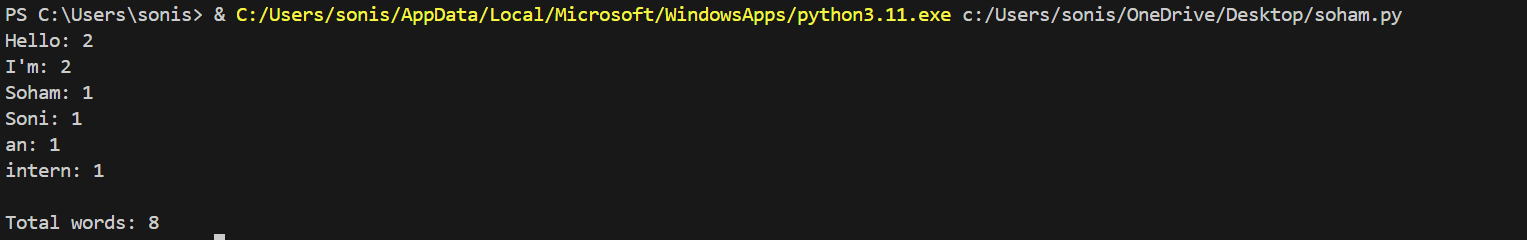
        print(f"{word}: {count}")

    total\_words = sum(word\_counts.values())

    print(f"\nTotal words: {total\_words}")

file\_path = r"C:\Users\sonis\OneDrive\Desktop\soham.txt"

word\_count(file\_path)

**Output:4**  
  


**Code:5**

import urllib.request

from collections import Counter

# URL of the text file

story\_url = 'https://sixty-north.com/c/t.txt'

# Fetch the text from the URL

with urllib.request.urlopen(story\_url) as response:

    story\_text = response.read().decode('utf-8')

# Split the text into words and convert them to lowercase

words = story\_text.lower().split()

# Count the occurrences of each word

word\_counts = Counter(words)

# Print the word counts

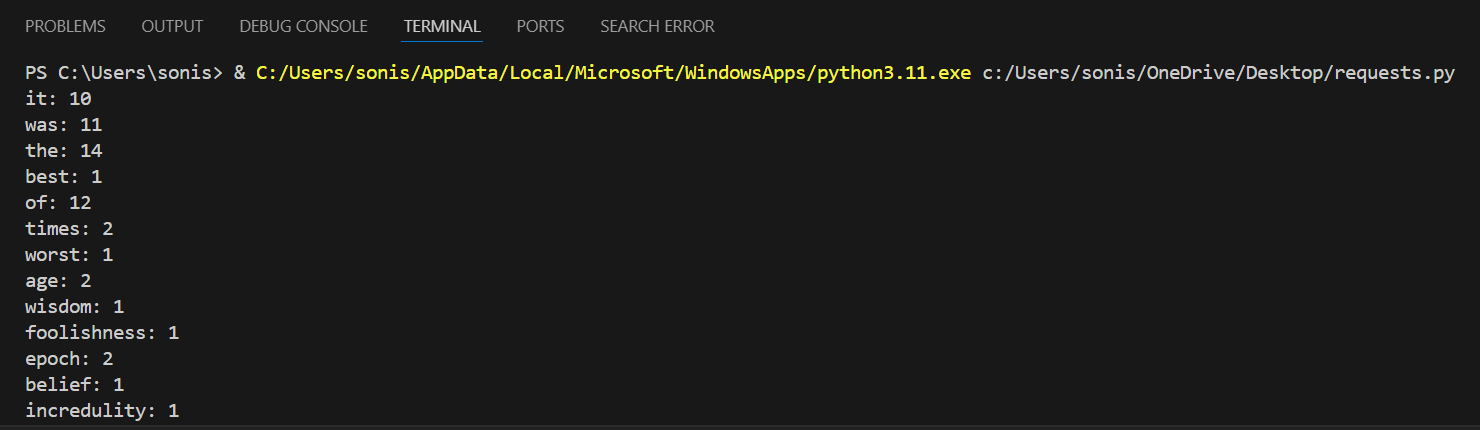
for word, count in word\_counts.items():

    print(f"{word}: {count}")

# Calculate total words

total\_words = sum(word\_counts.values())

print(f"\nTotal words: {total\_words}")

**Output:5  
  
**