

**Exp. No : 6****Handling JSON data using HDFS and Python****1. Create emp.json file**

```
hayagriv@fedora:~$ hadoop fs -cat /exp6/emp.json
[
  {
    "name": "Jane",
    "age": 30,
    "department": "HR",
    "Salary": 50000
  },
  {
    "name": "Bob",
    "age": 25,
    "department": "Marketing",
    "Salary": 60000
  },
  {
    "name": "Charlie",
    "age": 32,
    "department": "IT",
    "Salary": 70000
  },
  {
    "name": "Mark",
    "age": 28,
    "department": "Finance",
    "Salary": 55000
  },
  {
    "name": "Chris",
    "age": 38,
    "department": "IT",
    "Salary": 80000
  }
]
hayagriv@fedora:~$
```

## 2. Install jq package

```
hayagriv@fedora:~$ sudo dnf install jq
[sudo] password for hayagriv:
Last metadata expiration check: 1:20:19 ago on Sat 19 Oct 2024 07:19:44 AM UTC.
Package jq-1.7.1-7.fc40.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
hayagriv@fedora:~$ S
```

### 3. Execute jq . emp.json command

```
hayagriv@fedora:~$ hadoop fs -cat /exp6/emp.json
```

```
[
  {
    "name": "Jane",
    "age": 30,
    "department": "HR",
    "Salary": 50000
  },
  {
    "name": "Bob",
    "age": 25,
    "department": "Marketing",
    "Salary": 60000
  },
  {
    "name": "Charlie",
    "age": 32,
    "department": "IT",
    "Salary": 70000
  },
  {
    "name": "Mark",
    "age": 28,
    "department": "Finance",
    "Salary": 55000
  },
  {
    "name": "Chris",
    "age": 38,
    "department": "IT",
    "Salary": 80000
  }
]
```

```
hayagriv@fedora:~$
```

#### 4. pip install pandas

```
hayagriv@fedora:~$ pip install pandas
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: pandas in ~/.local/lib/python3.12/site-packages (2.2.3)
Requirement already satisfied: numpy>=1.26.0 in ~/.local/lib/python3.12/site-packages (from pandas) (2.1.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/lib/python3.12/site-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in ~/.local/lib/python3.12/site-packages (from pandas) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in ~/.local/lib/python3.12/site-packages (from pandas) (2024.2)
Requirement already satisfied: six>=1.5 in /usr/lib/python3.12/site-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)
```

#### 5. pip install hdfs

```
hayagriv@fedora:~$ pip install hdfs
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: hdfs in ~/.local/lib/python3.12/site-packages (2.7.3)
Requirement already satisfied: docopt in ~/.local/lib/python3.12/site-packages (from hdfs) (0.6.2)
Requirement already satisfied: requests>=2.7.0 in /usr/lib/python3.12/site-packages (from hdfs) (2.31.0)
Requirement already satisfied: six>=1.9.0 in /usr/lib/python3.12/site-packages (from hdfs) (1.16.0)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/lib/python3.12/site-packages (from requests>=2.7.0->hdfs) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/lib/python3.12/site-packages (from requests>=2.7.0->hdfs) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/lib/python3.12/site-packages (from requests>=2.7.0->hdfs) (1.26.20)
hayagriv@fedora:~$
```

## Create process\_data.py

```
GNU nano 7.2 process_data.py
hayagriv@fedora:~ — nano process_data.py

from hdfs import InsecureClient
import pandas as pd
import json

#Connect to HDFS
hdfs_client = InsecureClient("http://localhost:9870", user="hdfs")

#Read JSON data from HDFS
try:
    with hdfs_client.read("/exp6/emp.json", encoding="utf-8") as reader:
        json_data = reader.read()
        if not json_data.strip():
            raise ValueError("The JSON file is empty.")
        print(f"Raw JSON Data: {json_data[:1000]}")
        data = json.loads(json_data)
except json.JSONDecodeError as e:
    print(f"JSON Decode Error: {e}")
    exit(1)
except Exception as e:
    print(f"Error reading or parsing JSON data: {e}")
    exit(1)

#Convert JSON data to DataFrame
try:
    df = pd.DataFrame(data)
except ValueError as e:
    print(f"Error converting JSON data to DataFrame: {e}")
    exit(1)

#Projection: Select onlt 'name' and 'salary' columns
projected_df = df[['name', 'Salary']]

#Aggregation: Calculation total salary
total_salary = df['Salary'].sum()
```

**Output:**

```
Raw JSON Data: [{"name": "John Doe", "age": 30, "department": "HR", "salary": 50000},
{"name": "Jane Smith", "age": 25, "department": "IT", "salary": 60000},
{"name": "Alice Johnson", "age": 35, "department": "Finance", "salary": 70000},
{"name": "Bob Brown", "age": 28, "department": "Marketing", "salary": 55000},
{"name": "Charlie Black", "age": 45, "department": "IT", "salary": 80000}]
```

Filtered JSON file saved successfully.

Projection: Select only name and salary columns

	name	salary
0	John Doe	50000
1	Jane Smith	60000
2	Alice Johnson	70000
3	Bob Brown	55000
4	Charlie Black	80000

Aggregation: Calculate total salary

Total Salary: 315000

# Count: Number of employees earning more than 50000

Number of High Earners (>50000): 4

Limit: Top 5 highest salary

Top 5 Earners:

	name	age	department	salary
4	Charlie Black	45	IT	80000
2	Alice Johnson	35	Finance	70000
1	Jane Smith	25	IT	60000
3	Bob Brown	28	Marketing	55000
0	John Doe	30	HR	50000

Skipped DataFrame (First 2 rows skipped):

	name	age	department	salary
2	Alice Johnson	35	Finance	70000
3	Bob Brown	28	Marketing	55000
4	Charlie Black	45	IT	80000

Filtered DataFrame (Sales department removed):

	name	age	department	salary
0	John Doe	30	HR	50000
2	Alice Johnson	35	Finance	70000
3	Bob Brown	28	Marketing	55000