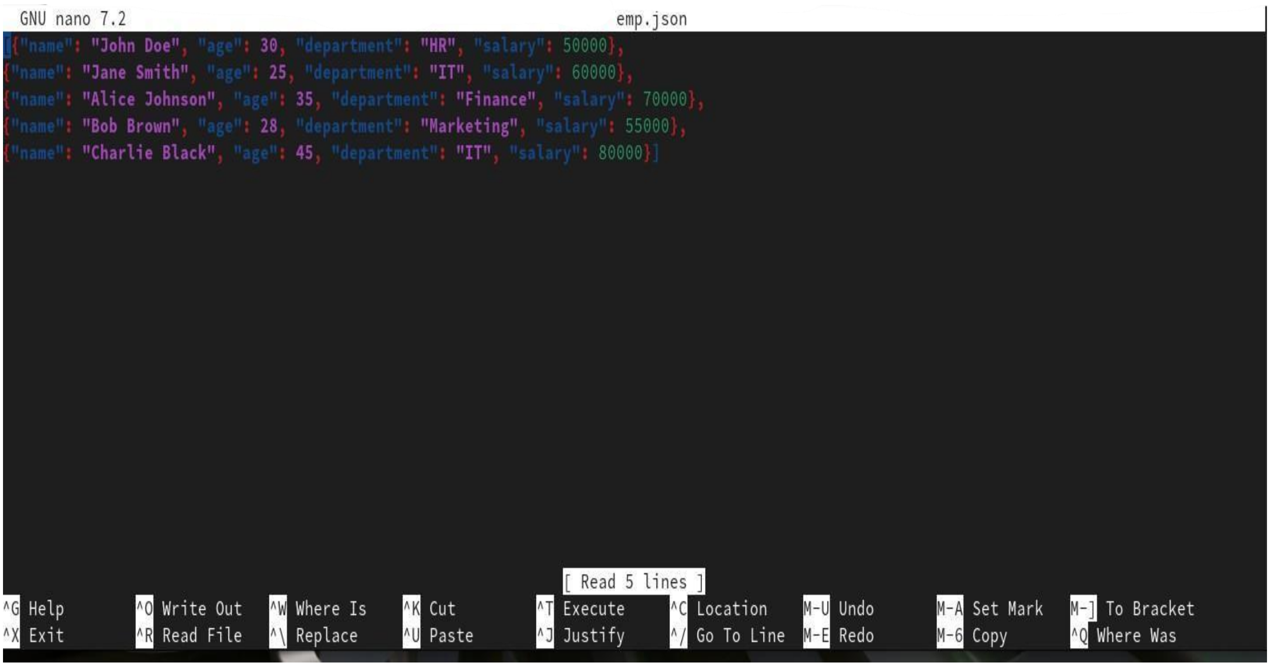
Exp. No : 6

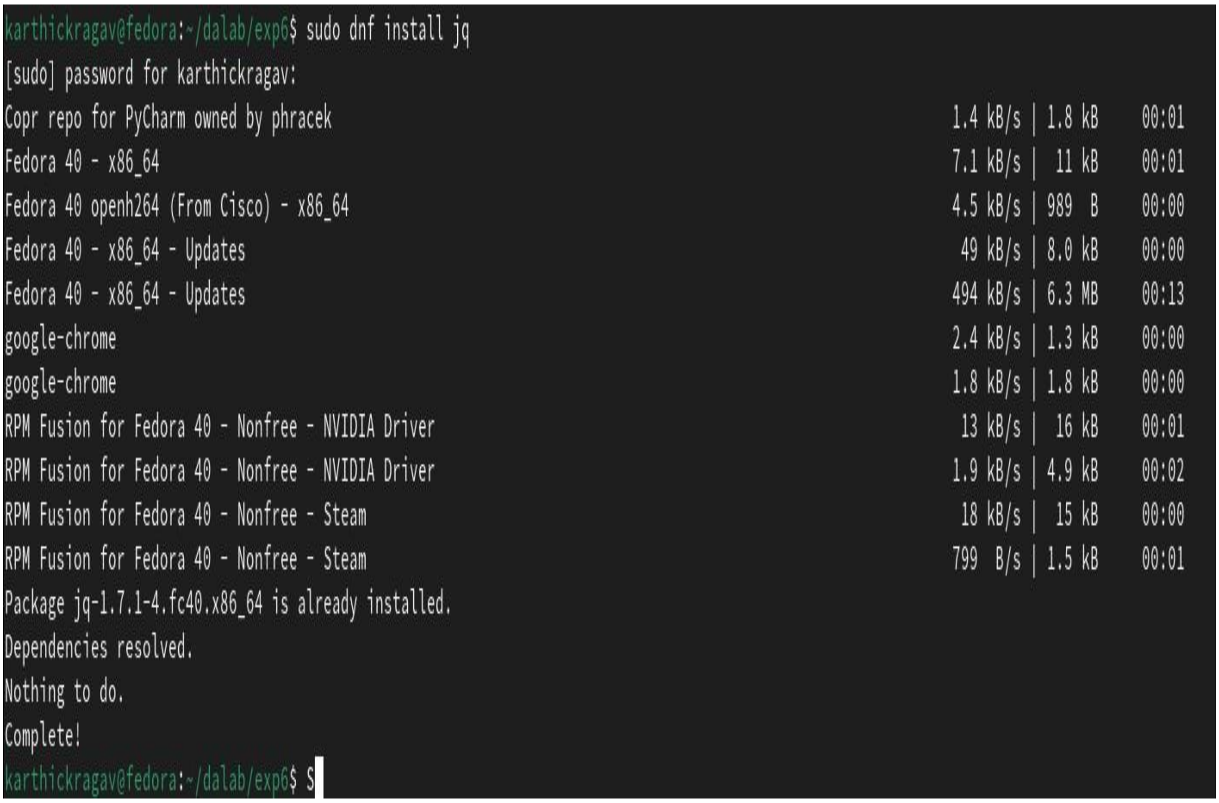
Handling JSON data using HDFS and Python

1. Create emp.json file



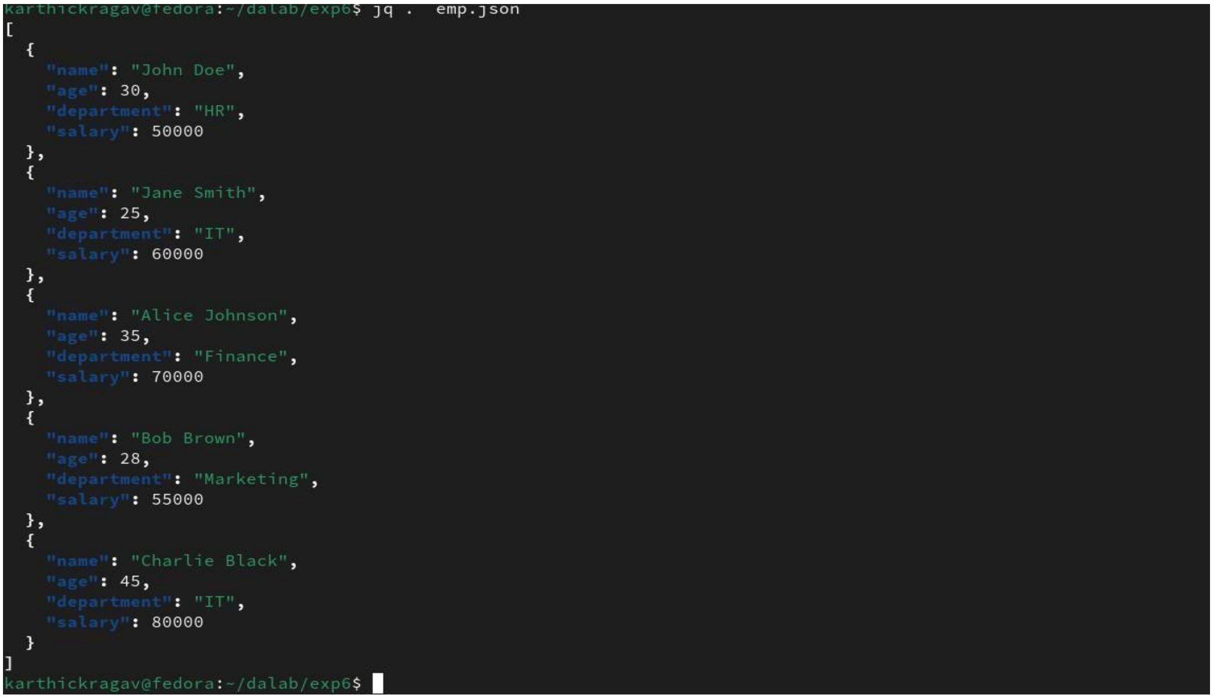
```
[{"name": "John Doe", "age": 30, "department": "HR", "salary": 50000},
{"name": "Jane Smith", "age": 25, "department": "IT", "salary": 60000},
{"name": "Alice Johnson", "age": 35, "department": "Finance", "salary": 70000},
{"name": "Bob Brown", "age": 28, "department": "Marketing", "salary": 55000},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

2. Install jq package



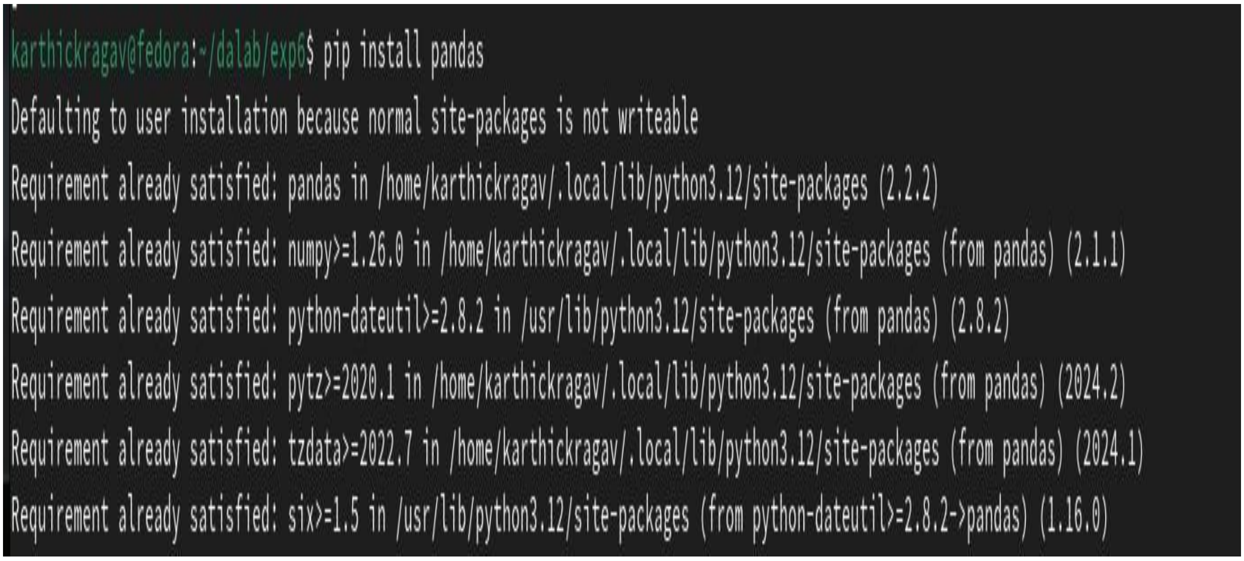
```
karthickragav@fedora:~/dalab/exp6$ sudo dnf install jq
[sudo] password for karthickragav:
Copr repo for PyCharm owned by phracek                                    1.4 kB/s | 1.8 kB     00:01
Fedora 40 - x86_64                                                        7.1 kB/s |  11 kB     00:01
Fedora 40 openh264 (From Cisco) - x86_64                                  4.5 kB/s | 989  B     00:00
Fedora 40 - x86_64 - Updates                                               49 kB/s | 8.0 kB     00:00
Fedora 40 - x86_64 - Updates                                              494 kB/s | 6.3 MB     00:13
google-chrome                                                             2.4 kB/s | 1.3 kB     00:00
google-chrome                                                             1.8 kB/s | 1.8 kB     00:00
RPM Fusion for Fedora 40 - Nonfree - NVIDIA Driver                         13 kB/s |  16 kB     00:01
RPM Fusion for Fedora 40 - Nonfree - NVIDIA Driver                        1.9 kB/s | 4.9 kB     00:02
RPM Fusion for Fedora 40 - Nonfree - Steam                                 18 kB/s |  15 kB     00:00
RPM Fusion for Fedora 40 - Nonfree - Steam                                799  B/s | 1.5 kB     00:01
Package jq-1.7.1-4.fc40.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
karthickragav@fedora:~/dalab/exp6$ $
```

3. Execute jq . emp.json command



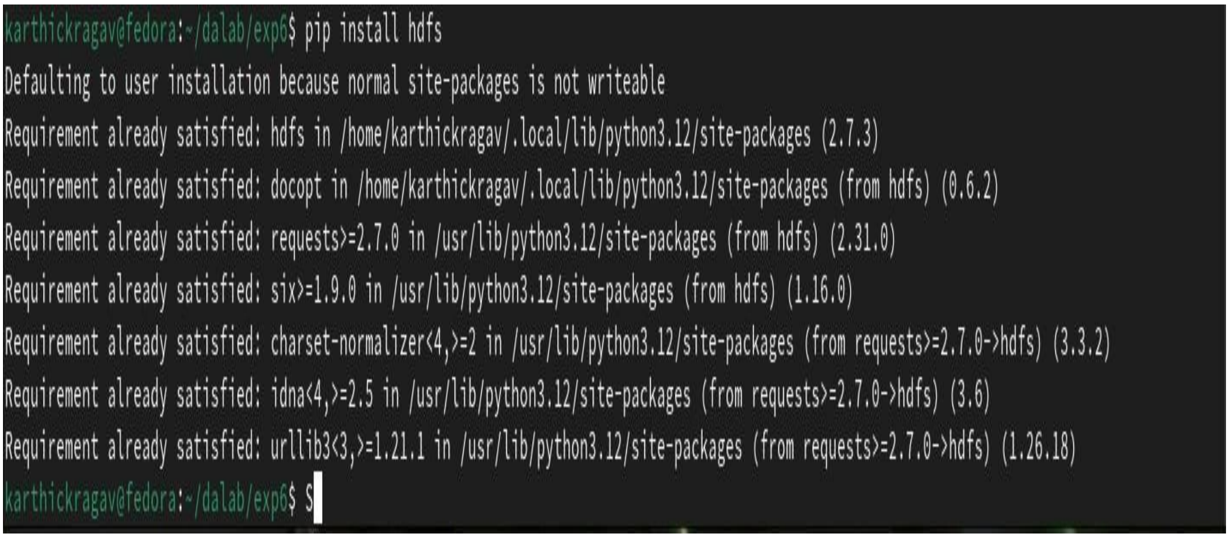
```
karthickragav@fedora:~/dalab/exp6$ jq .  emp.json
[
  {
    "name": "John Doe",
    "age": 30,
    "department": "HR",
    "salary": 50000
  },
  {
    "name": "Jane Smith",
    "age": 25,
    "department": "IT",
    "salary": 60000
  },
  {
    "name": "Alice Johnson",
    "age": 35,
    "department": "Finance",
    "salary": 70000
  },
  {
    "name": "Bob Brown",
    "age": 28,
    "department": "Marketing",
    "salary": 55000
  },
  {
    "name": "Charlie Black",
    "age": 45,
    "department": "IT",
    "salary": 80000
  }
]
karthickragav@fedora:~/dalab/exp6$ 
```

4. pip install pandas



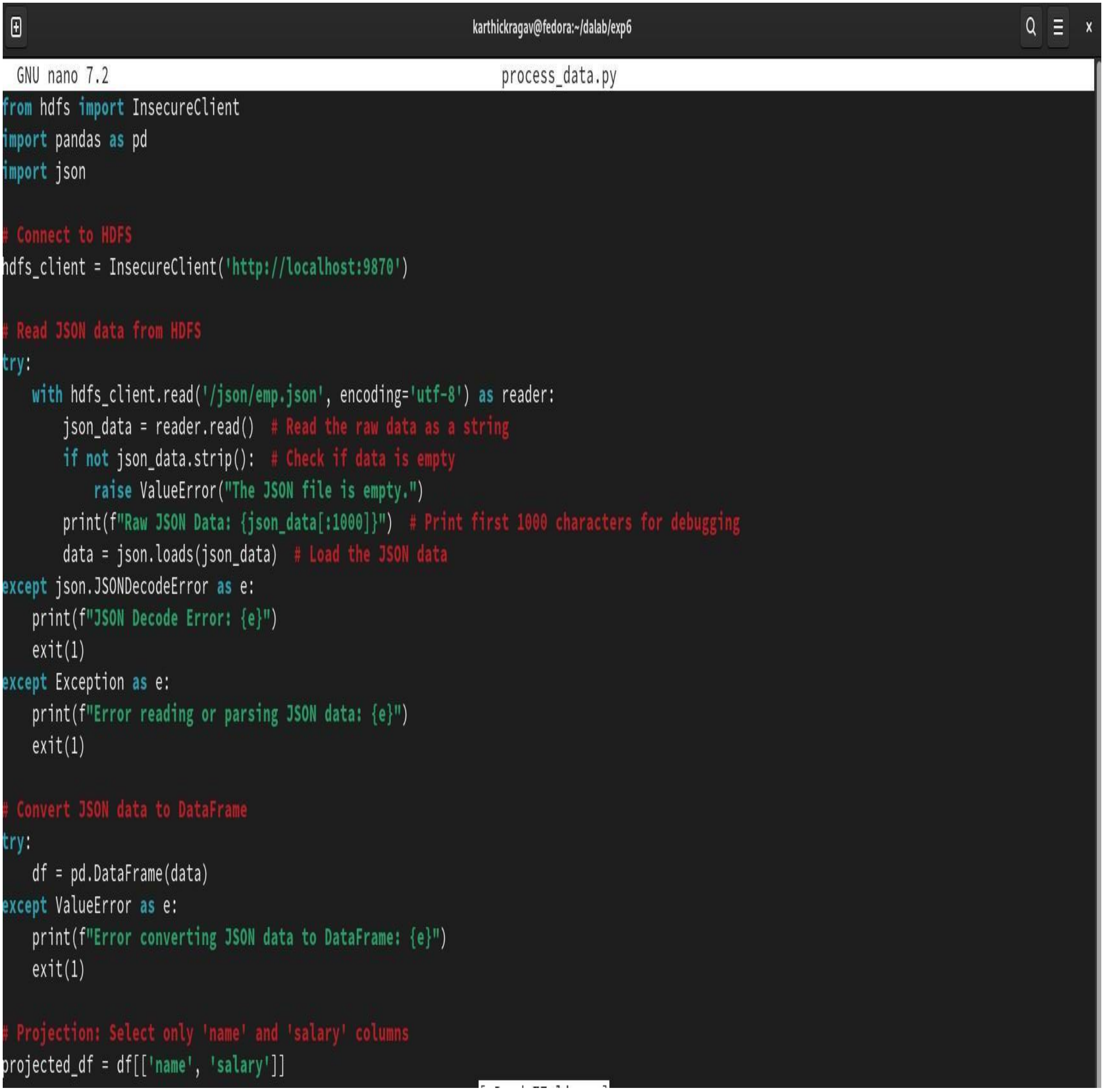
```
karthickragav@fedora:~/dalab/exp6$ pip install pandas
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: pandas in /home/karthickragav/.local/lib/python3.12/site-packages (2.2.2)
Requirement already satisfied: numpy>=1.26.0 in /home/karthickragav/.local/lib/python3.12/site-packages (from pandas) (2.1.1)
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 /home/karthickragav/.local/lib/python3.12/site-packages (from pandas) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in /home/karthickragav/.local/lib/python3.12/site-packages (from pandas) (2024.1)
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



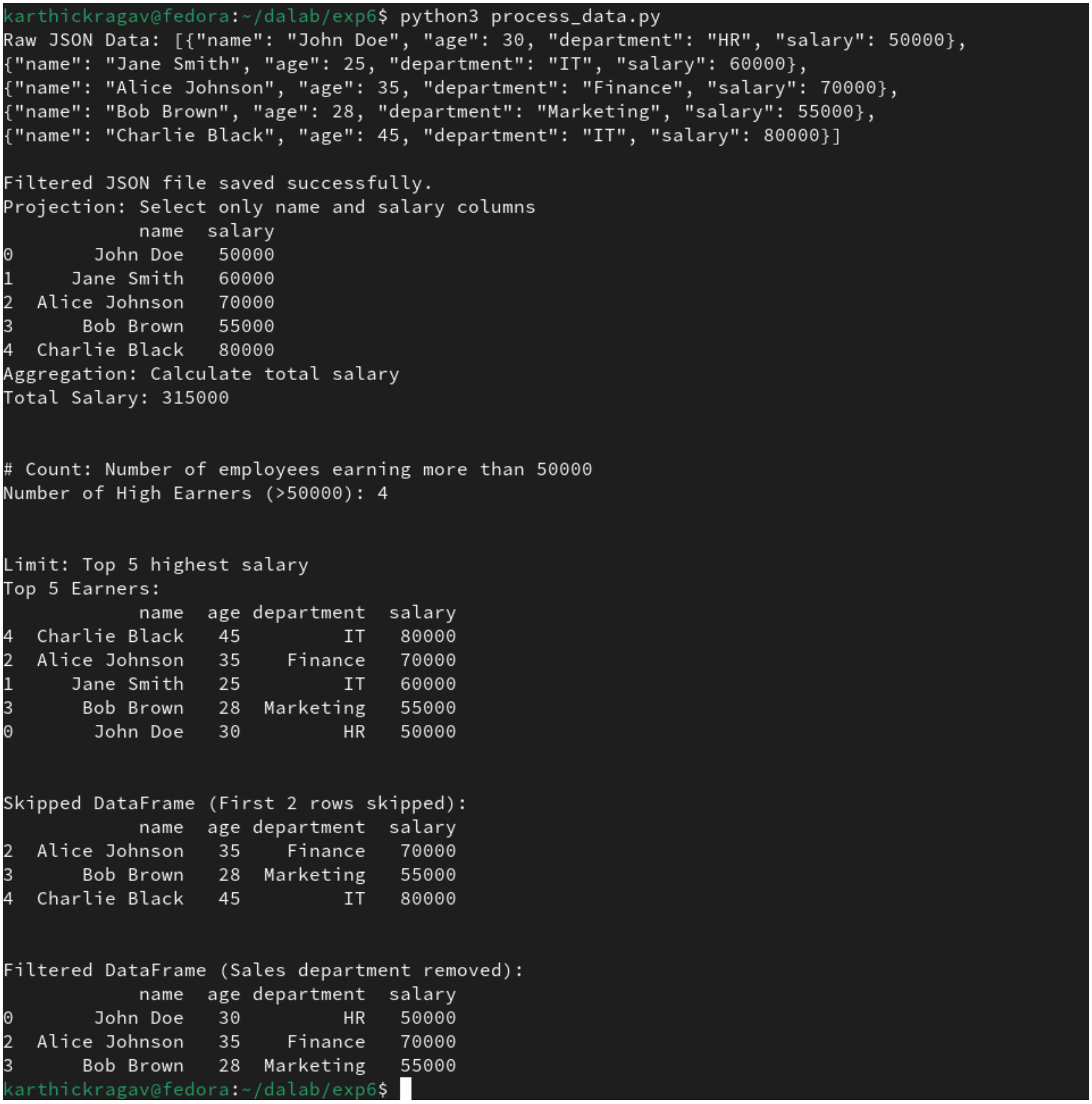
```
karthickragav@fedora:~/dalab/exp6$ pip install hdfs
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: hdfs in /home/karthickragav/.local/lib/python3.12/site-packages (2.7.3)
Requirement already satisfied: docopt in /home/karthickragav/.local/lib/python3.12/site-packages (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) (3.6)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/lib/python3.12/site-packages (from requests>=2.7.0->hdfs) (1.26.18)
karthickragav@fedora:~/dalab/exp6$
```

Create process_data.py



```
GNU nano 7.2                                          process_data.py
from hdfs import InsecureClient
import pandas as pd
import json

# Connect to HDFS
hdfs_client = InsecureClient('http://localhost:9870')

# Read JSON data from HDFS
try:
    with hdfs_client.read('/json/emp.json', encoding='utf-8') as reader:
        json_data = reader.read()  # Read the raw data as a string
        if not json_data.strip():  # Check if data is empty
            raise ValueError("The JSON file is empty.")
        print(f"Raw JSON Data: {json_data[:1000]}")  # Print first 1000 characters for debugging
        data = json.loads(json_data)  # Load the 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 only 'name' and 'salary' columns
projected_df = df[['name', 'salary']]
```

Output:



```
karthickragav@fedora:~/dalab/exp6$ python3 process_data.py
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
karthickragav@fedora:~/dalab/exp6$ 
```