

EXP.NO: 6 **IMPORT A JSON FILE FROM THE COMMAND LINE.**
APPLY THE FOLLOWING ACTIONS WITH THE DATA
PRESENT IN THE JSON FILE WHERE PROJECTION,
AGGREGATION, REMOVE, COUNT, LIMIT, SKIP AND SORT

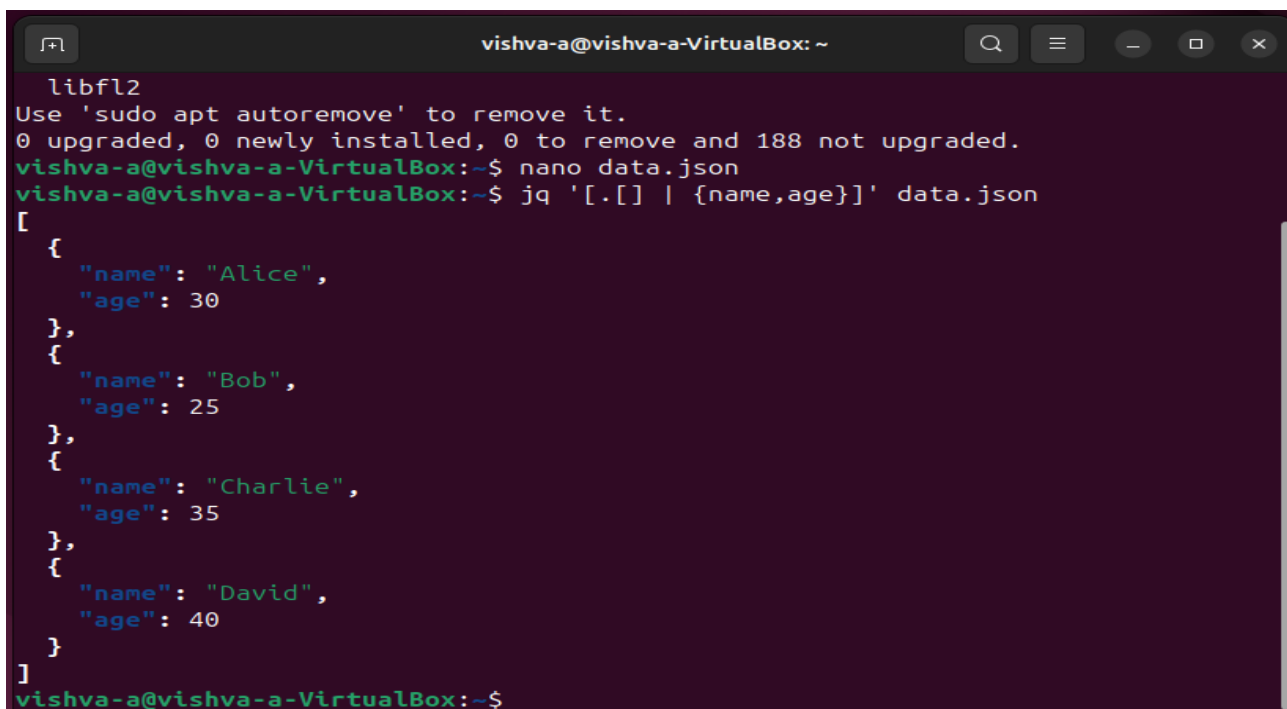
Step 1: Create a json file named “data.json” using the command

`$nano data.json`

Enter some data in the json file.

```
[  
  { "name": "Alice", "age": 25, "city": "New York" },  
  { "name": "Bob", "age": 30, "city": "Los Angeles" },  
  { "name": "Charlie", "age": 35, "city": "Chicago" },  
  { "name": "David", "age": 40, "city": "Houston" },  
]
```

Now, while projecting the data, we enter the command `$jq '[] | {name,age}[]' data.json` to display a specific values.



The screenshot shows a terminal window titled 'vishva-a@vishva-a-VirtualBox: ~'. It displays the output of a system update check, followed by the command `nano data.json` to create a JSON file. Then, the command `jq '[] | {name,age}[]' data.json` is executed, resulting in a JSON array of objects containing 'name' and 'age' for each person in the original file.

```
libfl2  
Use 'sudo apt autoremove' to remove it.  
0 upgraded, 0 newly installed, 0 to remove and 188 not upgraded.  
vishva-a@vishva-a-VirtualBox:~$ nano data.json  
vishva-a@vishva-a-VirtualBox:~$ jq '[] | {name,age}[]' data.json  
[  
  {  
    "name": "Alice",  
    "age": 30  
  },  
  {  
    "name": "Bob",  
    "age": 25  
  },  
  {  
    "name": "Charlie",  
    "age": 35  
  },  
  {  
    "name": "David",  
    "age": 40  
  }  
]  
vishva-a@vishva-a-VirtualBox:~$
```

Step 2: Using the command `$jq '[] | .age | add / length' data.json` we get the aggregate value of the data.

Step 3: Using the command `$jq 'map(del(.city))' data.json` we remove some specified values of the data.

```
vishva-a@vishva-a-VirtualBox: ~  
]   
vishva-a@vishva-a-VirtualBox:~$ jq '[.[] | .age] | add / length' data.json  
32.5  
vishva-a@vishva-a-VirtualBox:~$ jq 'map(del(.city))' data.json  
[  
  {  
    "name": "Alice",  
    "age": 30  
  },  
  {  
    "name": "Bob",  
    "age": 25  
  },  
  {  
    "name": "Charlie",  
    "age": 35  
  },  
  {  
    "name": "David",  
    "age": 40  
  }  
]  
vishva-a@vishva-a-VirtualBox:~$
```

Step 4: Using the command `$jq 'length' data.json` we get the total count of the data.

Step 5: Using the command `$jq '.[0:2]' data.json` we assign the limit of the data to be displayed.

```
vishva-a@vishva-a-VirtualBox: ~  
  {  
    "name": "David",  
    "age": 40  
  }  
]  
vishva-a@vishva-a-VirtualBox:~$ jq 'length' data.json  
4  
vishva-a@vishva-a-VirtualBox:~$ jq '.[0:2]' data.json  
[  
  {  
    "name": "Alice",  
    "age": 30,  
    "city": "New York"  
  },  
  {  
    "name": "Bob",  
    "age": 25,  
    "city": "Los Angeles"  
  }  
]  
vishva-a@vishva-a-VirtualBox:~$ jq '.[2:]' data.json  
[  
  {  
    "name": "Charlie",  
    "age": 35,  
    "city": "Chicago"  
  },  
  {  
    "name": "David",  
    "age": 40,  
    "city": "New York"  
  }  
]  
vishva-a@vishva-a-VirtualBox:~$
```

Step 6: Using the command `$jq '[2:]' data.json`, it skips some of the data and display the remaining data.

```
vishva-a@vishva-a-VirtualBox: ~  
vishva-a@vishva-a-VirtualBox:~$ jq '[0:2]' data.json  
[  
  {  
    "name": "Alice",  
    "age": 30,  
    "city": "New York"  
  },  
  {  
    "name": "Bob",  
    "age": 25,  
    "city": "Los Angeles"  
  }  
]  
vishva-a@vishva-a-VirtualBox:~$ jq '[2:]' data.json  
[  
  {  
    "name": "Charlie",  
    "age": 35,  
    "city": "Chicago"  
  },  
  {  
    "name": "David",  
    "age": 40,  
    "city": "Houston"  
  }  
]
```

Step 7: Using the command `$jq 'sort_by(.age)' data.json`, it sorts the data in a specified criteria..

```
vishva-a@vishva-a-VirtualBox: ~  
vishva-a@vishva-a-VirtualBox:~$ jq '[2:]' data.json  
[  
  {  
    "name": "Charlie",  
    "age": 35,  
    "city": "Chicago"  
  },  
  {  
    "name": "David",  
    "age": 40,  
    "city": "Houston"  
  }  
]  
vishva-a@vishva-a-VirtualBox:~$ jq 'sort_by(.age)' data.json  
[  
  {  
    "name": "Bob",  
    "age": 25,  
    "city": "Los Angeles"  
  }  
]
```

```
vishva-a@vishva-a-VirtualBox: ~  
vishva-a@vishva-a-VirtualBox:~$ jq 'sort_by(.age)' data.json  
[  
  {  
    "name": "Bob",  
    "age": 25,  
    "city": "Los Angeles"  
  },  
  {  
    "name": "Alice",  
    "age": 30,  
    "city": "New York"  
  },  
  {  
    "name": "Charlie",  
    "age": 35,  
    "city": "Chicago"  
  },  
  {  
    "name": "David",  
    "age": 40,  
    "city": "Houston"  
  }  
]  
vishva-a@vishva-a-VirtualBox:~$
```