ROLL.NO: 210701314

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” },

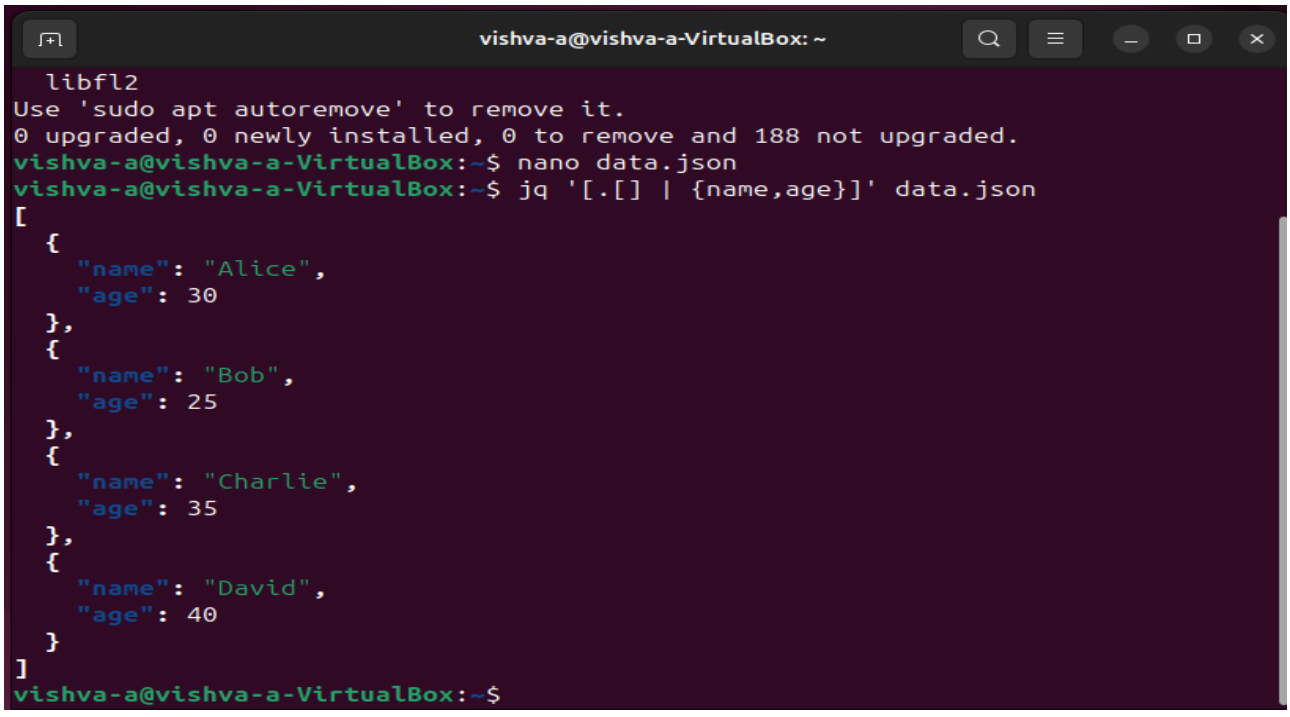
{“name”: “Bob”, “age”: 30, “city”: “Los Angeles” },

{“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.

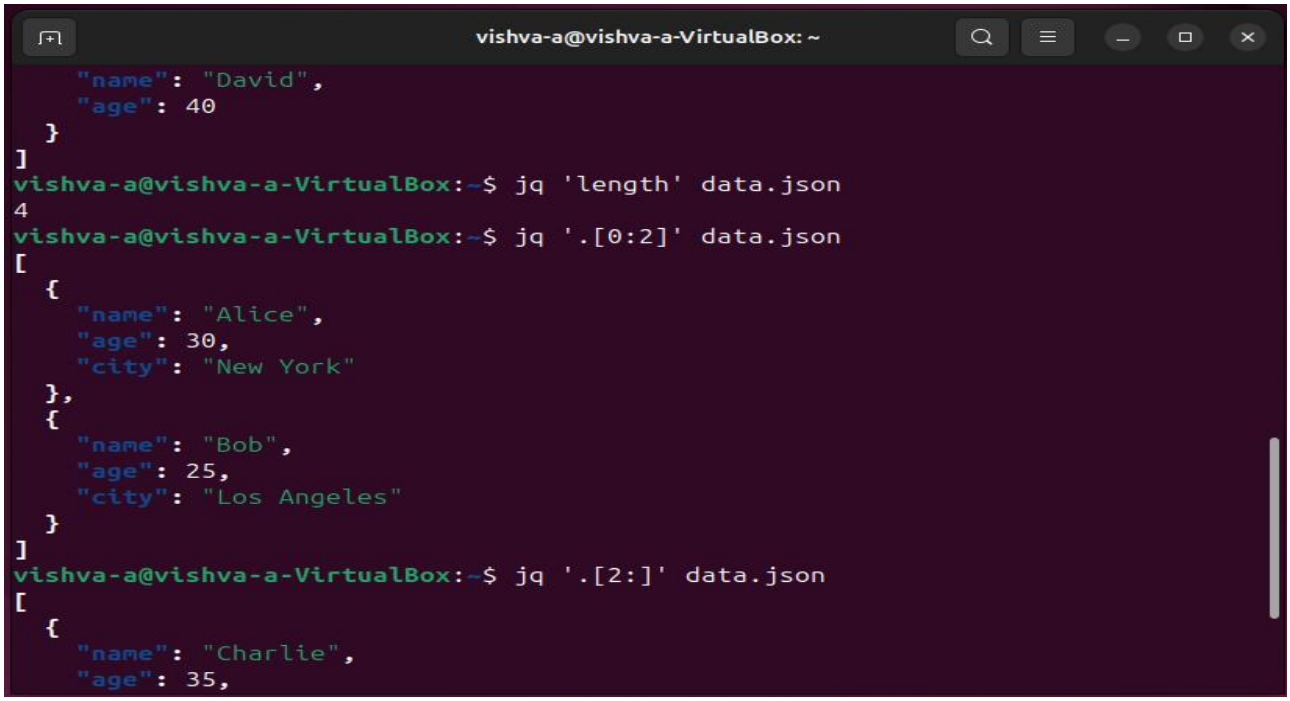


Step 2: Using the command $jq ‘[.[] | .age] | add / length’ data.json we get the aggregate value of the data.

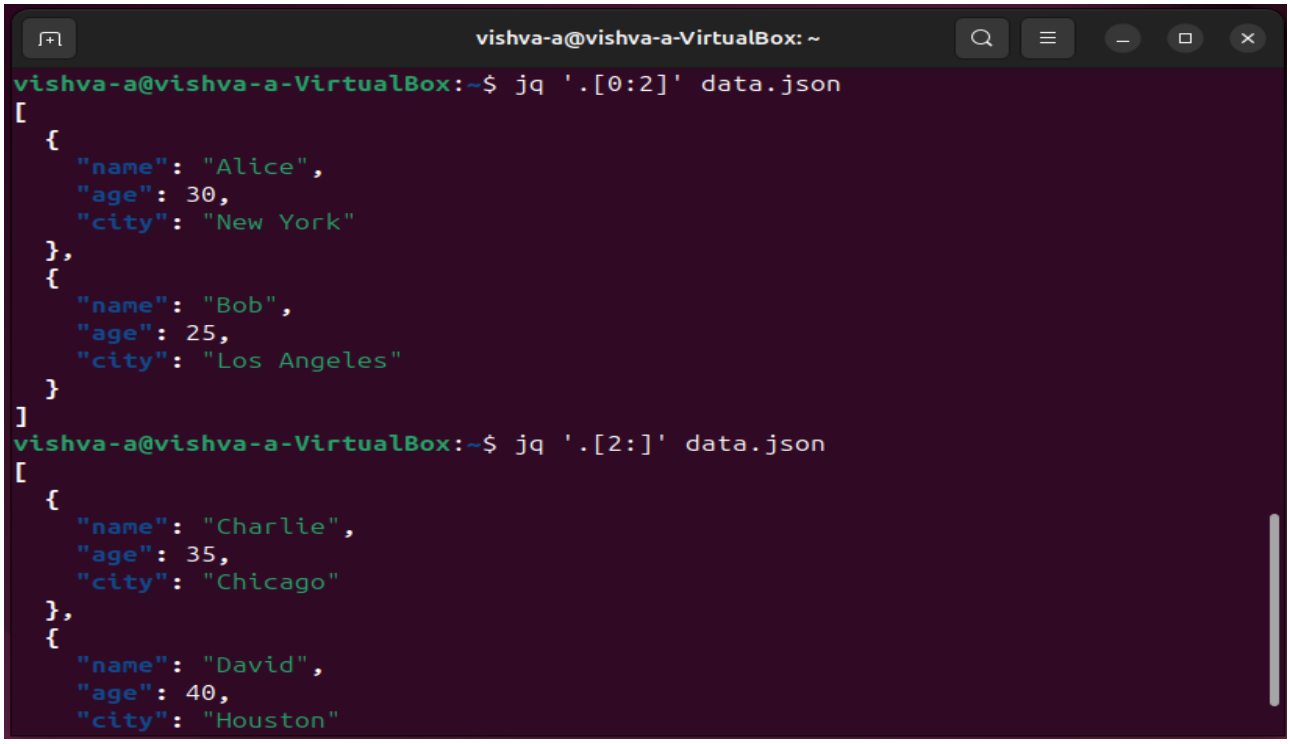
Step 3: Using the command $jq ‘map(del(.city))’ data.json we remove some specified values of the data.

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

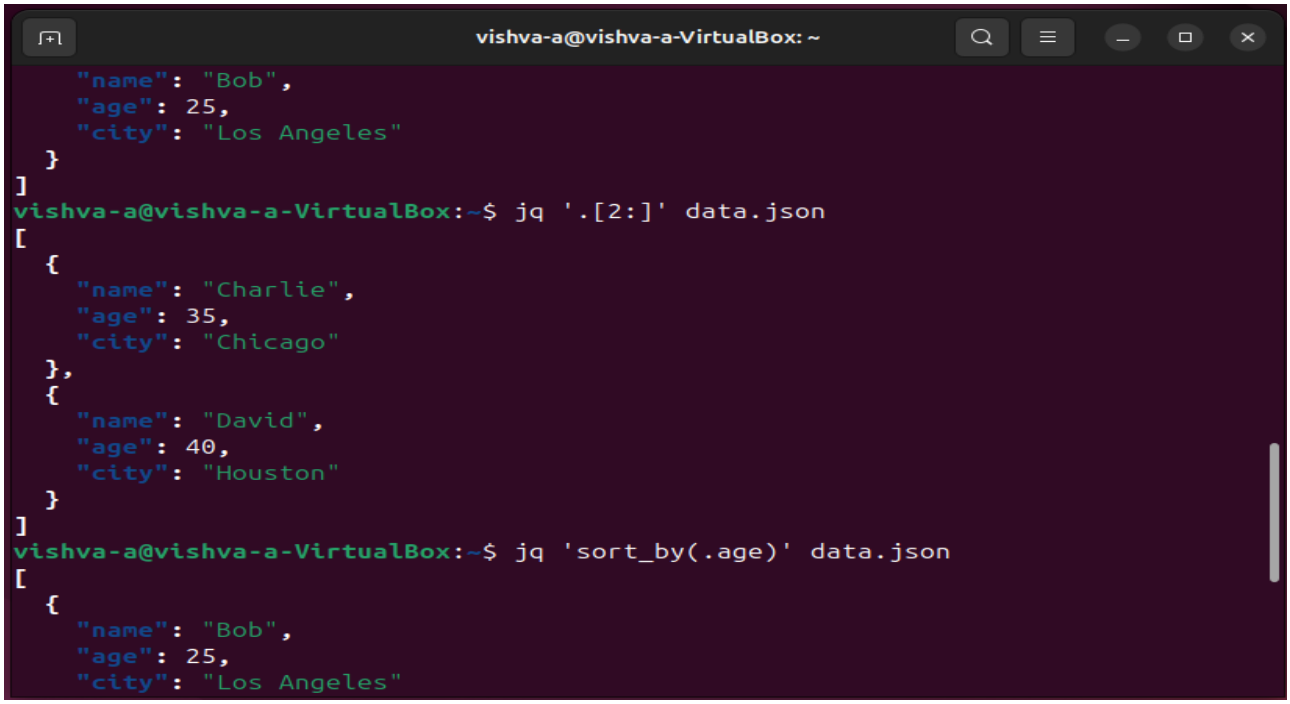
Step 5: Using the command $jq ‘.[0:2]’ data.json we assign the limit of the data to be displayed.

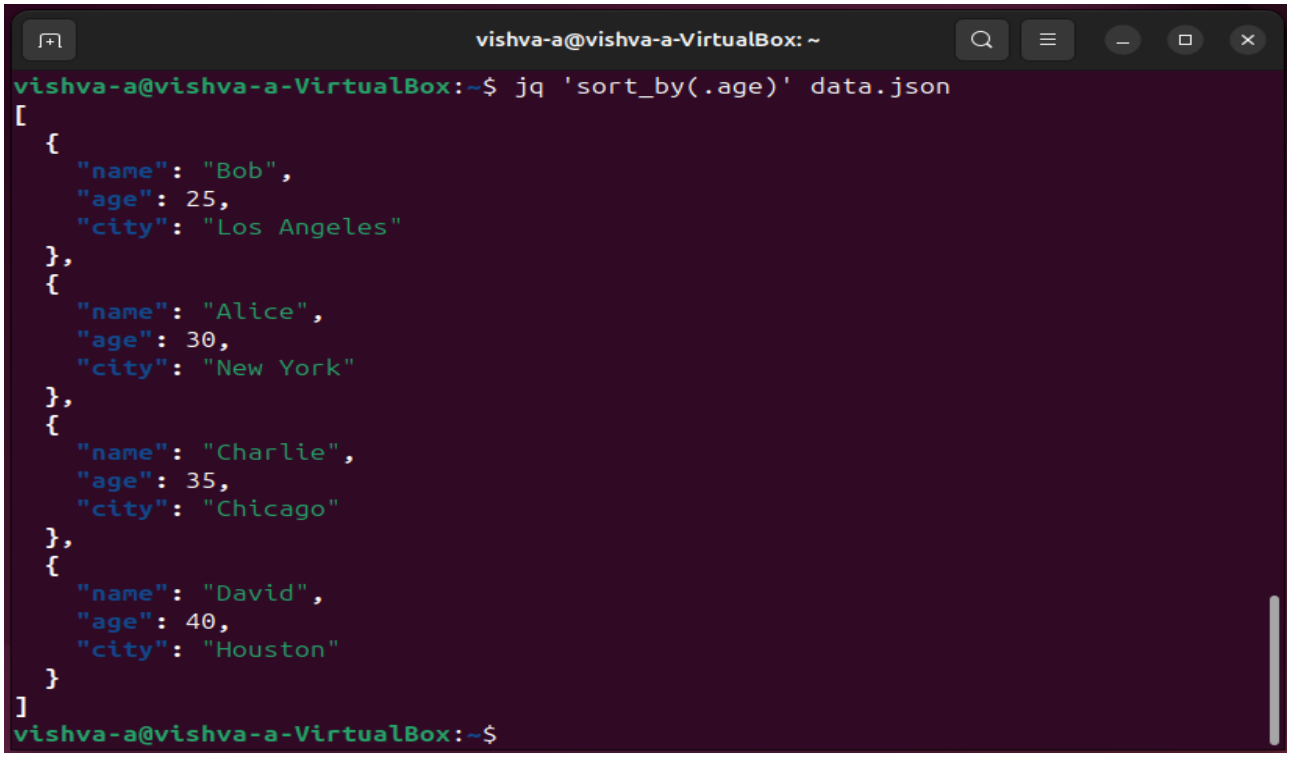


Step 6: Using the command $jq '.[2:]' data.json , it skips some of the data and display the remaining data.



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 '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:~$