

**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**

**AIM:**

To import a JSON file from the command line and apply the following actions with the data present in the JSON file where, projection, aggregation, remove, count, limit, skip and sort using MongoDB.

**PROCEDURE:**

1. Install and Set path jq to apply actions on the JSON file.
2. After performing step 1 check for jq version to confirm the installation.

```
PS C:\WINDOWS\system32> jq --version
jq-1.7.1
```

3. Give following commands of jq to perform projection, aggregation, remove, count, limit, skip and sort using jq.

- Projection (Select Specific Fields): Display only the name and salary fields:  
`jq '.[[] | {name, salary}]' "C:\Users\asus\Documents\DA\DA6\employees.json"`
- Count the Number of Employees:  
`jq 'length' "C:\Users\asus\Documents\DA\DA6\employees.json"`
- Remove (Filter Out Specific Records): Remove the employee with id: 2:  
`jq 'map(select(.id != 2))' "C:\Users\asus\Documents\DA\DA6\employees.json"`
- Limit (Limit the Output): Display only the first 3 employees:  
`jq '.[0:3]' "C:\Users\asus\Documents\DA\DA6\employees.json"`
- Skip (Skip a Number of Records): Skip the first 2 employees and display the rest:  
`jq '.[2:]' "C:\Users\asus\Documents\DA\DA6\employees.json"`
- Sort (Sort by Salary): Sort employees by salary in descending order:  
`jq 'sort_by(.salary) | reverse' "C:\Users\asus\Documents\DA\DA6\employees.json"`

- Aggregation (Calculate Average Salary): Calculate the average salary of employees:

```
jq '[.[] | .salary] | add / length' "C:\Users\asus\Documents\DA\DA6\employees.json"
```

## OUTPUT:

```
PS C:\WINDOWS\system32> jq '.[] | {name, salary}' "C:\Users\asus\Documents\DA\DA6\employees.json"
>>
[
  {
    "name": "Joshin",
    "salary": 70000
  },
  {
    "name": "Madhan",
    "salary": 55000
  },
  {
    "name": "Kanna",
    "salary": 60000
  },
  {
    "name": "Jegan",
    "salary": 65000
  },
  {
    "name": "Babu",
    "salary": 75000
  }
]
PS C:\WINDOWS\system32> jq 'length' "C:\Users\asus\Documents\DA\DA6\employees.json"
>>
5
PS C:\WINDOWS\system32> jq --version
>>
jq-1.7.1
```

```
PS C:\WINDOWS\system32> jq 'map(select(.id != 2))' "C:\Users\asus\Documents\DA\DA6\employees.json"
>>
[
  {
    "id": 1,
    "name": "Joshin",
    "department": "ECE",
    "age": 20,
    "salary": 70000
  },
  {
    "id": 3,
    "name": "Kanna",
    "department": "CSE",
    "age": 21,
    "salary": 60000
  },
  {
    "id": 4,
    "name": "Jegan",
    "department": "EEE",
    "age": 21,
    "salary": 65000
  },
  {
    "id": 5,
    "name": "Babu",
    "department": "ECE",
    "age": 21,
    "salary": 75000
  }
]
```

```
PS C:\WINDOWS\system32> jq '[0:3]' "C:\Users\asus\Documents\DA\DA6\employees.json"
>>
[
  {
    "id": 1,
    "name": "Joshin",
    "department": "ECE",
    "age": 20,
    "salary": 70000
  },
  {
    "id": 2,
    "name": "Madhan",
    "department": "CSE",
    "age": 21,
    "salary": 55000
  },
  {
    "id": 3,
    "name": "Kanna",
    "department": "CSE",
    "age": 21,
    "salary": 60000
  }
]
```

```
PS C:\WINDOWS\system32> jq '[2:]' "C:\Users\asus\Documents\DA\DA6\employees.json"
>>
[
  {
    "id": 3,
    "name": "Kanna",
    "department": "CSE",
    "age": 21,
    "salary": 60000
  },
  {
    "id": 4,
    "name": "Jegan",
    "department": "EEE",
    "age": 21,
    "salary": 65000
  },
  {
    "id": 5,
    "name": "Babu",
    "department": "ECE",
    "age": 21,
    "salary": 75000
  }
]
```

```
PS C:\WINDOWS\system32> jq 'sort_by(.salary) | reverse' "C:\Users\asus\Documents\DA\DA6\employees.json"
>>
[
  {
    "id": 5,
    "name": "Babu",
    "department": "ECE",
    "age": 21,
    "salary": 75000
  },
  {
    "id": 1,
    "name": "Joshin",
    "department": "ECE",
    "age": 20,
    "salary": 70000
  },
  {
    "id": 4,
    "name": "Jegan",
    "department": "EEE",
    "age": 21,
    "salary": 65000
  },
  {
    "id": 3,
    "name": "Kanna",
    "department": "CSE",
    "age": 21,
    "salary": 60000
  },
]
```

```
PS C:\WINDOWS\system32> jq '[.[] | .salary] | add / length' "C:\Users\asus\Documents\DA\DA6\employees.json"
>>
65000
```

## RESULT:

Thus to import a JSON file from the command line and apply the following actions with the data present in the JSON file where, projection, aggregation, remove, count, limit, skip and sort using jq is completed successfully.