EX.NO:6 **210701165**

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 jq.

PROCEDURE:

1. Install jq from jq website.

2. Then open command prompt and run jq –version command to check whether jq is installed.

3. In command prompt navigate to the directory where json file is present

4. After navigating to the JSON file perform specific commands for projection, aggregation,remove, count, limit and sort.

OUTPUT:

```
C:\>jq ".[] | {name, age}" data.json
{
  "name": "John",
  "age": 30
}
{
  "name": "Jane",
  "age": 25
}
{
  "name": "Jim",
  "age": 40
}
```

```
C:\>jq "[.[] | .salary] | add" data.json
18000

C:\>
```

EX.NO:6 **210701165**

```
C:\>jq "map(del(.age))" data.json
[
  {
    "name": "John",
    "salary": 5000
  },
  {
    "name": "Jane",
    "salary": 6000
  },
  {
    "name": "Jim",
    "salary": 7000
  }
]
```

```
C:\>jq "length" data.json
3

C:\>
```

```
C:\>jq "limit(2; .[])" data.json
{
  "name": "John",
  "age": 30,
  "salary": 5000
}
{
  "name": "Jane",
  "age": 25,
  "salary": 6000
}
```

EX.NO:6 **210701165**

```
C:\>jq ".[1:]" data.json
[
  {
    "name": "Jane",
    "age": 25,
    "salary": 6000
  },
  {
    "name": "Jim",
    "age": 40,
    "salary": 7000
  }
]
```

```
C:\>jq "sort_by(.salary)" data.json
[
  {
    "name": "John",
    "age": 30,
    "salary": 5000
  },
  {
    "name": "Jane",
    "age": 25,
    "salary": 6000
  },
  {
    "name": "Jim",
    "age": 40,
    "salary": 7000
  }
]
```

EX.NO:6 **210701165**

```
C:\>jq "sort_by(.salary) | reverse" data.json
[
  {
    "name": "Jim",
    "age": 40,
    "salary": 7000
  },
  {
    "name": "Jane",
    "age": 25,
    "salary": 6000
  },
  {
    "name": "John",
    "age": 30,
    "salary": 5000
  }
]
```

```
C:\>jq "[.[] | .salary] | add / length" data.json
6000

C:\>
```

RESULT:

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