EX NO:6 **210701130**

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 tool.

PROCEDURE:

- Create a json file 'employees.json' and provide data in it.
- Open the command prompt.
- Navigate to the folder where employees.json is stored.
- Load and view the JSON data with jq.
- Use the jq commands for projection, aggregation, removal, counting, limiting, and sorting operations.

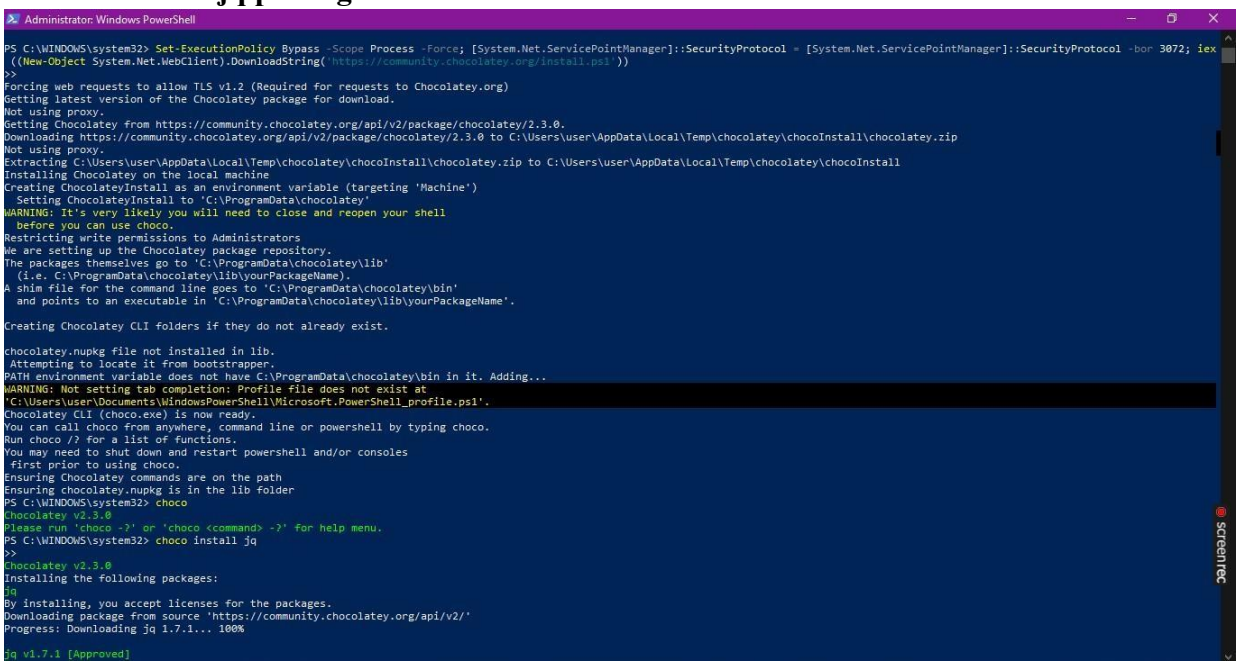
employees.json:

```
[
  {
    "id": 1,
    "name": "Alice Johnson",
    "department": "Engineering",
    "age": 29,
    "salary": 70000
  },
  {
    "id": 2,
    "name": "Bob Smith",
    "department": "Marketing",
    "age": 35,
    "salary": 55000
  },
  {
    "id": 3,
```

```
    "name": "Charlie Davis",
    "department": "Engineering",
    "age": 25,
    "salary": 60000
  },
  {
    "id": 4,
    "name": "Dana Lee",
    "department": "Human Resources",
    "age": 40,
    "salary": 65000
  },
  {
    "id": 5,
    "name": "Eve Martinez",
    "department": "Finance",
    "age": 45,
    "salary": 75000
  }
]
```


OUTPUT:

Installation of jq packages:



Running jq queries:

I. Projection:

jq ".[] | {name: .name, salary: .salary}" Desktop/employees.json

```
C:\Users\user>jq ".[] | {name: .name, salary: .salary}" Desktop/employees.json
{
  "name": "Alice Johnson",
  "salary": 70000
}
{
  "name": "Bob Smith",
  "salary": 55000
}
{
  "name": "Charlie Davis",
  "salary": 60000
}
{
  "name": "Dana Lee",
  "salary": 65000
}
{
  "name": "Eve Martinez",
  "salary": 75000
}
```

II. Aggregation:

jq "[.[] | .salary] | add" Desktop/employees.json

```
C:\Users\user>jq "[.[] | .salary] | add" Desktop/employees.json
325000
```

III. Remove:

jq "del(.[] | .age)" Desktop/employees.json

```
C:\Users\user>jq "del(.[] | .age)" Desktop/employees.json
[
  {
    "id": 1,
    "name": "Alice Johnson",
    "department": "Engineering",
    "salary": 70000
  },
  {
    "id": 2,
    "name": "Bob Smith",
    "department": "Marketing",
    "salary": 55000
  },
  {
    "id": 3,
    "name": "Charlie Davis",
    "department": "Engineering",
    "salary": 60000
  },
  {
    "id": 4,
    "name": "Dana Lee",
    "department": "Human Resources",
    "salary": 65000
  },
  {
    "id": 5,
    "name": "Eve Martinez",
    "department": "Finance",
    "salary": 75000
  }
]
```

IV. Count:

jq ". | length" Desktop/employees.json

```
C:\Users\user>jq ". | length" Desktop/employees.json
5
```

V. Limit:

jq ".[0:3]" Desktop/employees.json

```
C:\Users\user>jq ".[0:3]" Desktop/employees.json
[
  {
    "id": 1,
    "name": "Alice Johnson",
    "department": "Engineering",
    "age": 29,
    "salary": 70000
  },
  {
    "id": 2,
    "name": "Bob Smith",
    "department": "Marketing",
    "age": 35,
    "salary": 55000
  },
  {
    "id": 3,
    "name": "Charlie Davis",
    "department": "Engineering",
    "age": 25,
    "salary": 60000
  }
]
```

VI. Skip:

jq ".[2:]" Desktop/employees.json

```
C:\Users\user>jq ".[2:]" Desktop/employees.json
[
  {
    "id": 3,
    "name": "Charlie Davis",
    "department": "Engineering",
    "age": 25,
    "salary": 60000
  },
  {
    "id": 4,
    "name": "Dana Lee",
    "department": "Human Resources",
    "age": 40,
    "salary": 65000
  },
  {
    "id": 5,
    "name": "Eve Martinez",
    "department": "Finance",
    "age": 45,
    "salary": 75000
  }
]
```

VII. Sort:

```
jq "sort_by(.age)" Desktop/employees.json
```

```
C:\Users\user>jq "sort_by(.age)" Desktop/employees.json
[
  {
    "id": 3,
    "name": "Charlie Davis",
    "department": "Engineering",
    "age": 25,
    "salary": 60000
  },
  {
    "id": 1,
    "name": "Alice Johnson",
    "department": "Engineering",
    "age": 29,
    "salary": 70000
  },
  {
    "id": 2,
    "name": "Bob Smith",
    "department": "Marketing",
    "age": 35,
    "salary": 55000
  },
  {
    "id": 4,
    "name": "Dana Lee",
    "department": "Human Resources",
    "age": 40,
    "salary": 65000
  },
  {
    "id": 5,
    "name": "Eve Martinez",
    "department": "Finance",
    "age": 45,
    "salary": 75000
  }
]
```

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 tool is completed successfully.