210701141
K.MADHUMITHA

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

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.

emp.json

```
[
  {
   "id": 1,
   "name": "Madhumitha",
   "department": "Engineering",
   "age": 29,
   "salary": 70000
  },
  {
   "id": 2,
   "name": "Deepak",
   "department": "Marketing",
   "age": 35,
   "salary": 55000
  },
  {
   "id": 3,
```

```
  "name": "Priya ",
  "department": "Engineering",
  "age": 25,
  "salary": 60000
 },
 {
  "id": 4,
  "name": "Meghna",
  "department": "Human Resources",
  "age": 40,
  "salary": 65000
 },
 {
  "id": 5,
  "name": "Mrithula",
  "department": "Finance",
  "age": 45,
  "salary": 75000
 }
]
```


OUTPUT:

INSTALLATION OF JQ PACKAGE:

```
C:\Windows\System32>choco
Chocolatey v2.2.2
Please run 'choco -?' or 'choco <command> -?' for help menu.

C:\Windows\System32>choco install jq
Chocolatey v2.2.2
Installing the following packages:
jq
By installing, you accept licenses for the packages.
Progress: Downloading jq 1.7.1... 100%

jq v1.7.1 [Approved]
jq package files install completed. Performing other installation steps.
The package jq wants to run 'chocolateyinstall.ps1'.
Note: If you don't run this script, the installation will fail.
Note: To confirm automatically next time, use '-y' or consider:
choco feature enable -n allowGlobalConfirmation
Do you want to run the script?([Y]es/[A]ll - yes to all/[N]o/[P]rint): yes

Downloading jq 64 bit
  from 'https://github.com/jqlang/jq/releases/download/jq-1.7.1/jq-windows-amd64.exe'
Progress: 100% - Completed download of C:\ProgramData\chocolatey\lib\jq\tools\jq.exe (962 KB).
Download of jq.exe (962 KB) completed.
Hashes match.
C:\ProgramData\chocolatey\lib\jq\tools\jq.exe
 ShimGen has successfully created a shim for jq.exe
 The install of jq was successful.
  Software install location not explicitly set, it could be in package or
  default install location of installer.

Chocolatey installed 1/1 packages.
 See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).
```

210701141
K.MADHUMITHA

Running jq queries:

I. Projection:
jq ".[] | {name: .name, salary: .salary}" Desktop/emp.json

```
D:\madhumitha\DA>jq ".[] | {name: .name,salary: .salary}" emp.json
{
  "name": "Madhumitha",
  "salary": 70000
}
{
  "name": "Deepak",
  "salary": 55000
}
{
  "name": "Priya ",
  "salary": 60000
}
{
  "name": "Meghna",
  "salary": 65000
}
{
  "name": "Mrithula",
  "salary": 75000
}
```

II. Aggregation:
jq "[.[] | .salary] | add" Desktop/emp.json

```
D:\madhumitha\DA>jq "[.[]|.salary]|add" emp.json
325000
```

III. REMOVE:

jq "del(.[] | .age)" emp.json

```
D:\madhumitha\DA>jq "del(.[] |.age)" emp.json
[
  {
    "id": 1,
    "name": "Madhumitha",
    "department": "Engineering",
    "salary": 70000
  },
  {
    "id": 2,
    "name": "Deepak",
    "department": "Marketing",
    "salary": 55000
  },
  {
    "id": 3,
    "name": "Priya ",
    "department": "Engineering",
    "salary": 60000
  },
  {
    "id": 4,
    "name": "Meghna",
    "department": "Human Resources",
    "salary": 65000
  },
  {
    "id": 5,
    "name": "Mrithula",
    "department": "Finance",
    "salary": 75000
  }
]
```

IV. COUNT:
jq ". | length" emp.json

```
D:\madhumitha\DA>jq ".|length" emp.json
5
```

V. LIMIT:
jq ".[0:3]" emp.json

```
D:\madhumitha\DA>jq ".[0:3]" emp.json
[
  {
    "id": 1,
    "name": "Madhumitha",
    "department": "Engineering",
    "age": 29,
    "salary": 70000
  },
  {
    "id": 2,
    "name": "Deepak",
    "department": "Marketing",
    "age": 35,
    "salary": 55000
  },
  {
    "id": 3,
    "name": "Priya ",
    "department": "Engineering",
    "age": 25,
    "salary": 60000
  }
]
```

VI. SKIP

jq ".[2:]"emp.json

```
D:\madhumitha\DA>jq ".[3:]" emp.json
[
  {
    "id": 4,
    "name": "Meghna",
    "department": "Human Resources",
    "age": 40,
    "salary": 65000
  },
  {
    "id": 5,
    "name": "Mrithula",
    "department": "Finance",
    "age": 45,
    "salary": 75000
  }
]
```

VII. SORT:

jq " sort_by(.name)" emp.json

```
D:\madhumitha\DA>jq " sort_by(.name)" emp.json
[
  {
    "id": 2,
    "name": "Deepak",
    "department": "Marketing",
    "age": 35,
    "salary": 55000
  },
  {
    "id": 1,
    "name": "Madhumitha",
    "department": "Engineering",
    "age": 29,
    "salary": 70000
  },
  {
    "id": 4,
    "name": "Meghna",
    "department": "Human Resources",
    "age": 40,
    "salary": 65000
  },
  {
    "id": 5,
    "name": "Mrithula",
    "department": "Finance",
    "age": 45,
    "salary": 75000
  },
  {
    "id": 3,
    "name": "Priya ",
    "department": "Engineering",
    "age": 25,
    "salary": 60000
  }
]
```

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.