

## Ex 6 JSON FILE OPERATIONS

### Aim:

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

### Procedure:

1. To allow PowerShell to run scripts that would otherwise be blocked by the default execution policy use the command  
**Set-ExecutionPolicy Bypass -Scope Process -Force;**
2. To install Chocolatey use command  
**[System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://community.chocolatey.org/install.ps1'))**

```
PS C:\Windows\system32> Set-ExecutionPolicy Bypass -Scope Process -Force;
[System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://community.chocolatey.org/install.ps1'))
Forcing web requests to allow TLS v1.2 (Required for requests to Chocolatey.org)
Getting latest version of the Chocolatey package for download.
Not using proxy.
Getting Chocolatey from https://community.chocolatey.org/api/v2/package/chocolatey/2.3.0.
Downloading https://community.chocolatey.org/api/v2/package/chocolatey/2.3.0 to C:\Users\jawah\AppData\Local\Temp\chocolatey\chocoInstall\chocolatey.zip
Not using proxy.
Extracting C:\Users\jawah\AppData\Local\Temp\chocolatey\chocoInstall\chocolatey.zip to C:\Users\jawah\AppData\Local\Temp\chocolatey\chocoInstall
Installing Chocolatey on the local machine
Creating ChocolateyInstall as an environment variable (targeting 'Machine')
Setting ChocolateyInstall to 'C:\ProgramData\chocolatey'
WARNING: It's very likely you will need to close and reopen your shell
before you can use choco.
Restricting write permissions to Administrators
We are setting up the Chocolatey package repository.
The packages themselves go to 'C:\ProgramData\chocolatey\lib'
(i.e. C:\ProgramData\chocolatey\lib\yourPackageName).
A shim file for the command line goes to 'C:\ProgramData\chocolatey\bin'
and points to an executable in 'C:\ProgramData\chocolatey\lib\yourPackageName'.
Creating Chocolatey CLI folders if they do not already exist.
chocolatey.nupkg file not installed in lib.
Attempting to locate it from bootstrapper.
PATH environment variable does not have C:\ProgramData\chocolatey\bin in it. Adding...
WARNING: Not setting tab completion: Profile file does not exist at
'C:\Users\jawah\OneDrive\Documents\WindowsPowerShell\Microsoft.PowerShell_profile.ps1'.
Chocolatey CLI (choco.exe) is now ready.
You can call choco from anywhere, command line or powershell by typing choco.
Run choco /? for a list of functions.
You may need to shut down and restart powershell and/or consoles
first prior to using choco.
Ensuring Chocolatey commands are on the path
Ensuring chocolatey.nupkg is in the lib folder
```

3. To find the version of chocolatey  
**choco --version**

```
PS C:\Windows\system32> choco --version
2.3.0
```

4. To install jq  
**choco install jq**

```
PS C:\Windows\system32> choco install jq
Chocolatey v2.3.0
Installing the following packages:
jq
By installing, you accept licenses for the packages.
Downloading package from source 'https://community.chocolatey.org/api/v2/'
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): y

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).
```

## JSON COMMANDS:

### 1. To project

**jq '.[] | {name: .name, salary: .salary}'**

**C:\Users\jawah\OneDrive\Desktop\LathikaDA\employees.json**

output:

```
>> jq '.[] | {name: .name, salary: .salary}' C:\Users\jawah\OneDrive\Desktop\LathikaDA\employees.json
{
  "name": "Diya",
  "salary": 70000
},
{
  "name": "lathika",
  "salary": 55000
},
{
  "name": "Sanvi",
  "salary": 60000
},
{
  "name": "Ram",
  "salary": 65000
},
{
  "name": "Isha",
  "salary": 75000
}
```

### 2. Aggregation: Calculate the total salary.

**jq '[.[] | .salary] | add' C:\Users\jawah\OneDrive\Desktop\LathikaDA\employees.json**

```
PS C:\Windows\system32> jq '[.[] | .salary] | add' C:\Users\jawah\OneDrive\Desktop\LathikaDA\employees.json
325000
```

### 3. Count: Count the number of employees.

**jq '. | length' C:\Users\jawah\OneDrive\Desktop\LathikaDA\employees.json**

```
PS C:\Windows\system32> jq '. | length' C:\Users\jawah\OneDrive\Desktop\LathikaDA\employees.json
5
```

4. Filter: Get employees from the Engineering department.

```
jq '[] | select(department == "Finance")'
```

C:\Users\jawah\OneDrive\Desktop\LathikaDA\employees.json

```
{
  "id": 1,
  "name": "Diya",
  "department": "Engineering",
  "age": 20,
  "salary": 70000
}
{
  "id": 3,
  "name": "Sanvi",
  "department": "Engineering",
  "age": 21,
  "salary": 60000
}
```

5. Sort: Sort employees by age.

```
jq 'sort_by(.age)' C:\Users\jawah\OneDrive\Desktop\LathikaDA\employees.json
```

```
PS C:\Windows\system32> jq 'sort_by(.age)' C:\Users\jawah\OneDrive\Desktop\LathikaDA\employees.json
[
  {
    "id": 1,
    "name": "Diya",
    "department": "Engineering",
    "age": 20,
    "salary": 70000
  },
  {
    "id": 2,
    "name": "lathika",
    "department": "Marketing",
    "age": 21,
    "salary": 55000
  },
  {
    "id": 3,
    "name": "Sanvi",
    "department": "Engineering",
    "age": 21,
    "salary": 60000
  },
  {
    "id": 4,
    "name": "Ram",
    "department": "Human Resources",
    "age": 21,
    "salary": 65000
  },
  {
    "id": 5,
    "name": "Isha",
    "department": "Finance",
    "age": 21,
    "salary": 75000
  }
]
```

## RESULT:

Thus the operations using json file is completed successfully.