

There are **3** supported endpoints in NEOH-Docker.

1. /start-process
2. /get-status
3. /get-result

Based on where it is deployed the host URL will change. For example

localhost/get-result  
127.0.0.1/get-result  
[www.server.com/get-result](http://www.server.com/get-result)

Refer “Sample” directory (in the NEOH-Docker repo) to understand the payload for the request.

The payload is requested through the “**POST**” method via endpoints. The Payload is wrapped as a JSON object in the body. No extra headers are required other than the default values.

## **/start-process**

The “Sample” directory (in the NEOH-Docker repo) provides a good starting point for the payload. Based on the dataset, the workflow is triggered.

This endpoint, on a successful request, returns a request id for the user to track the status and retrieve the results.

example:

```
{  
  "request_id": "PvIIYuepm0"  
}
```

## **/get-status**

Once the start-process workflow is triggered and has a successful response. Use the response to monitor updates on the request.

The payload for get-status is a list of request IDs. This is to get the status of multiple requests.

Example Payload:

```
{
  "request_id" :
  [
    "YYfVojoyDKJ",
    "PvIIYuepm0"
  ]
}
```

The response will look like

```
[
  {
    "status": "waiting",
    "message": "Job not started yet for request_id YYfVojoyDKJ",
    "request_id": "YYfVojoyDKJ",
    "type": "waiting",
    "creation_time": "n/a"
  },
  {
    "status": "waiting",
    "message": "Job not started yet for request_id PvIIYuepm0",
    "request_id": "PvIIYuepm0",
    "type": "waiting",
    "creation_time": "n/a"
  }
]
```

## /get-result

When the status and message on get-status response says successful and completed aggregation/processing. This endpoint is ready to be triggered. This end-point gives you the processed/aggregated values for the request made in the start-process.

The payload is a JSON object. Note: It is not a list of JSON objects. A request will return the result for one dataset at a time.

Example Payload:

```
{
  "request_id" : "PvIIYuepm0"
}
```

The Response will look like

```
{
  "status": "success",
  "message": "aggregate: All requested files successfully aggregated",
  "result": {
    "dataValues": [
      {
        "dataElement": "PvIIYuepm0",
        "period": "20190101",
        "orgUnit": "O6uvpzGd5pu",
        "value": 301.79597826086956
      },
      {
        "dataElement": "PvIIYuepm0",
        "period": "20190101",
        "orgUnit": "fdc6uOvgoji",
        "value": 303.5245112781955
      },
      {
        "dataElement": "PvIIYuepm0",
        "period": "20190101",
        "orgUnit": "Ic3eMKXaEfw",
        "value": 301.1334453781513
      },
      {
        "dataElement": "PvIIYuepm0",
```

```
"period": "20190101",
"orgUnit": "jUb8gELQAp",
"value": 300.445037037037
},
{
  "dataElement": "PvIIYuepm0",
  "period": "20190101",
  "orgUnit": "PMa2VCrupOd",
  "value": 303.0132
},
{
  "dataElement": "PvIIYuepm0",
  "period": "20190101",
  "orgUnit": "kJq2mPyFEHo",
  "value": 300.4309359605911
},
{
  "dataElement": "PvIIYuepm0",
  "period": "20190101",
  "orgUnit": "qhQAxPSTUXp",
  "value": 301.30705
},
{
  "dataElement": "PvIIYuepm0",
  "period": "20190101",
  "orgUnit": "Vth0fbpFcsO",
  "value": 301.77170454545455
},
{
  "dataElement": "PvIIYuepm0",
  "period": "20190101",
  "orgUnit": "jmIPBj66vD6",
  "value": 301.81300448430494
},
{
  "dataElement": "PvIIYuepm0",
  "period": "20190101",
  "orgUnit": "TEQlaapDQoK",
  "value": 302.69723076923077
},
{
  "dataElement": "PvIIYuepm0",
```

```
    "period": "20190101",
    "orgUnit": "bL4ooGhyHRQ",
    "value": 301.21918032786886
  },
  {
    "dataElement": "PvIIYuepm0",
    "period": "20190101",
    "orgUnit": "eIQbndfxQMb",
    "value": 303.3109615384615
  },
  {
    "dataElement": "PvIIYuepm0",
    "period": "20190101",
    "orgUnit": "at6UHUQatSo",
    "value": 303.45454545454544
  }
]
},
"creation_time": "03-23-2023T16:42:30Z",
"dataset": "temperature"
}
```