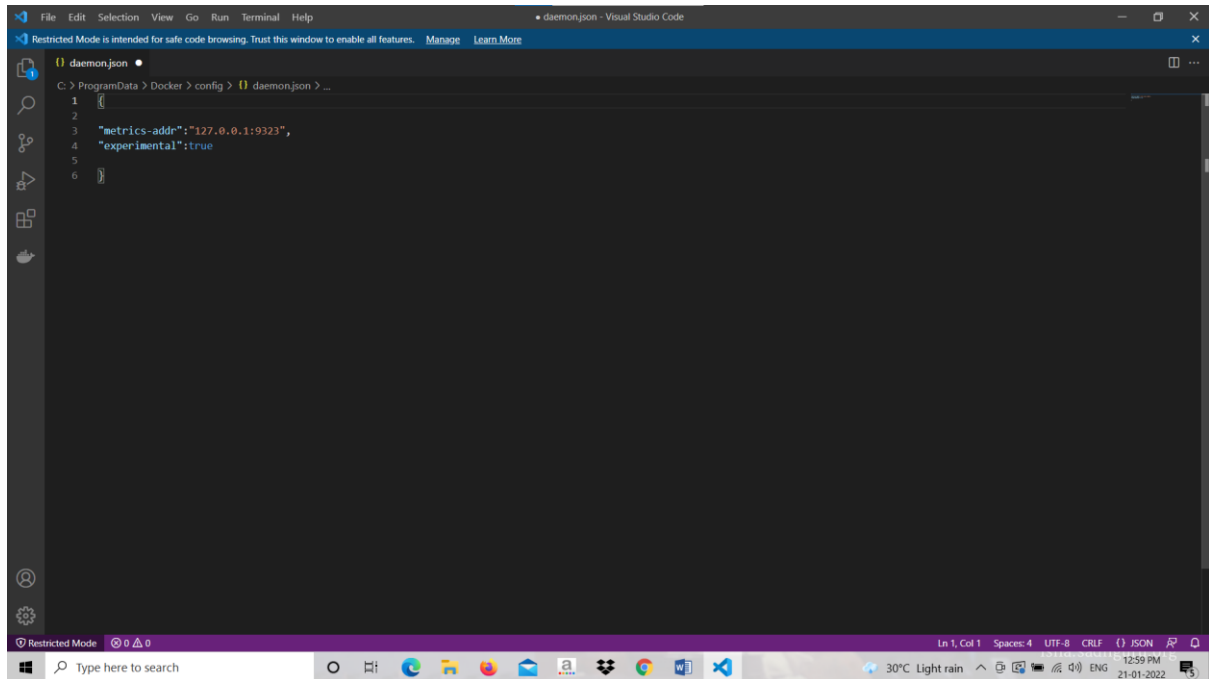


Configuring Docker using Prometheus.

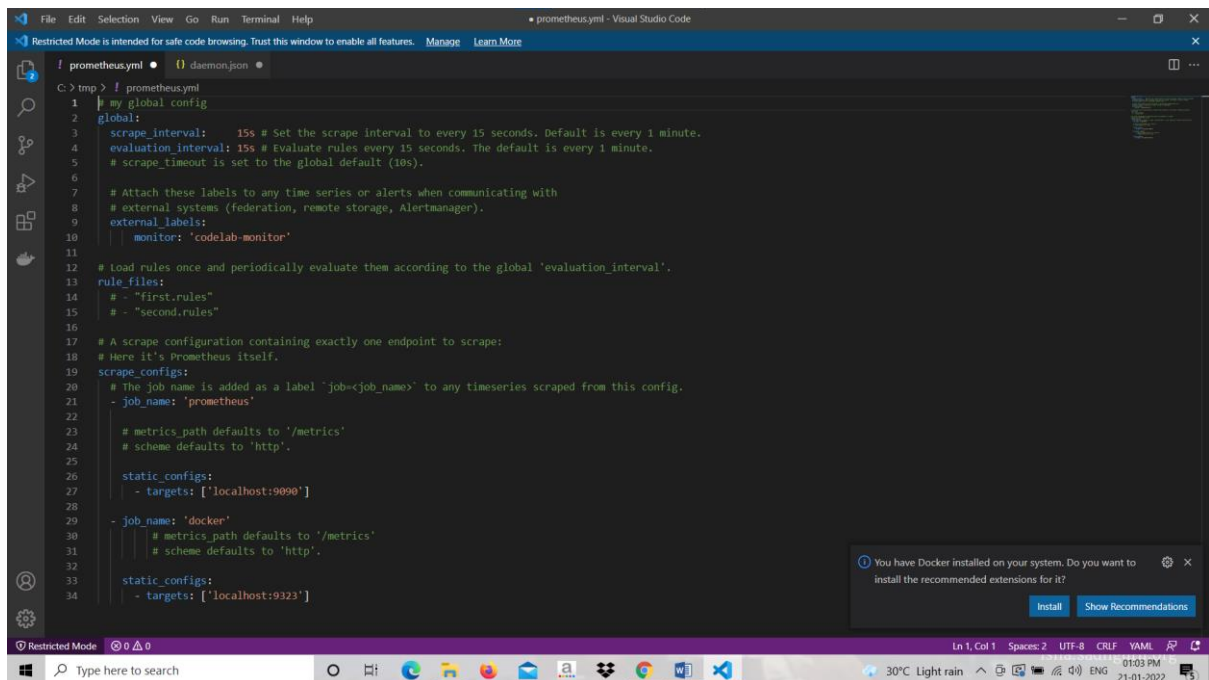
Specifying metrics-address for configuring docker daemon as a Prometheus target. For that created a daemon.json file.

In C:\Programdata\docker\config\daemon.json



```
1 {
2   "metrics-addr": "127.0.0.1:9323",
3   "experimental": true
4 }
```

Created Prometheus.yml file in C:\tmp\prometheus.yml (Windows)



```
1 # my global config
2 global:
3   scrape_interval: 15s # Set the scrape interval to every 15 seconds. Default is every 1 minute.
4   evaluation_interval: 15s # evaluate rules every 15 seconds. The default is every 1 minute.
5   # scrape_timeout is set to the global default (10s).
6
7   # Attach these labels to any time series or alerts when communicating with
8   # external systems (federation, remote storage, Alertmanager).
9   external_labels:
10    monitor: 'codelab-monitor'
11
12 # load rules once and periodically evaluate them according to the global 'evaluation_interval'.
13 rule_files:
14   # - "first.rules"
15   # - "second.rules"
16
17 # A scrape configuration containing exactly one endpoint to scrape:
18 # Here it's Prometheus itself.
19 scrape_configs:
20   # The job name is added as a label 'job=<job_name>' to any timeseries scraped from this config.
21   - job_name: 'prometheus'
22
23     # metrics_path defaults to '/metrics'
24     # scheme defaults to 'http'.
25
26     static_configs:
27       - targets: ['localhost:9090']
28
29   - job_name: 'docker'
30
31     # metrics_path defaults to '/metrics'
32     # scheme defaults to 'http'.
33
34     static_configs:
35       - targets: ['localhost:9323']
```

Starting a single-replica Prometheus service using this configuration

```
Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/powershell

PS C:\Users\PARVATHI_AJITH>
PS C:\Users\PARVATHI_AJITH> cd ..
PS C:\Users> cd ..
PS C:\> docker service create --replicas 1 --name my-prometheus `
> --mount type=bind,source=C:\tmp\prometheus.yml,destination=/etc/prometheus/prometheus.yml `
> --publish published=9090,target=9090,protocol=tcp `
> prom/prometheus
$?
overall progress: 1 out of 1 tasks
1/1: running [=====]
verify: Service converged
PS C:\>
```

Verifying that the docker target is listed at <http://localhost:9090/targets/>.

Prometheus Time Series Collector

localhost:9090/targets

Prometheus Alerts Graph Status Help Classic UI

Targets

All Unhealthy Collapse All

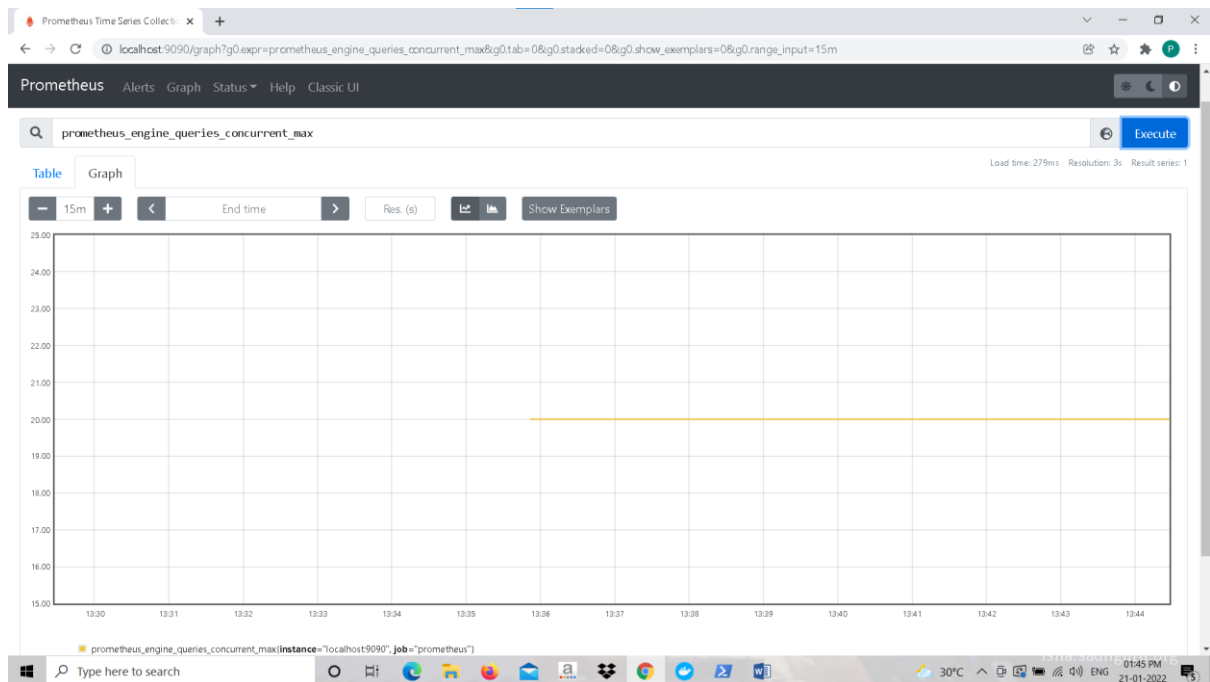
docker (0/1 up) [show less](#)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9323/metrics	DOWN	instance="localhost:9323" job="docker"	6.86s ago	1.541ms	Get "http://localhost:9323/metrics": dial tcp 127.0.0.1:9323: connect: connection refused

prometheus (1/1 up) [show less](#)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9090/metrics	UP	instance="localhost:9090" job="prometheus"	249.000ms ago	7.199ms	

Creating a graph



Creating some network actions by starting a service with 5 tasks that just ping docker non-stop.

```
P C:/> docker service create `
    --replicas 5 `
    --name ping_service `
    Alpine ping docker.com
```

Now after a few minutes when we reload the graph, we will get a diff graph with those network actions on it.

To stop the service:

docker service remove ping_service command is used

