**Objective:**

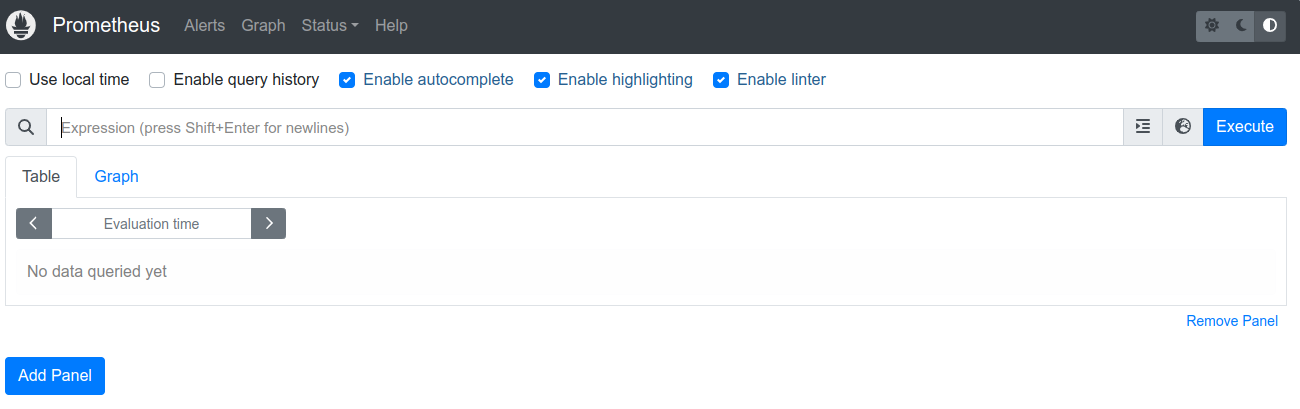
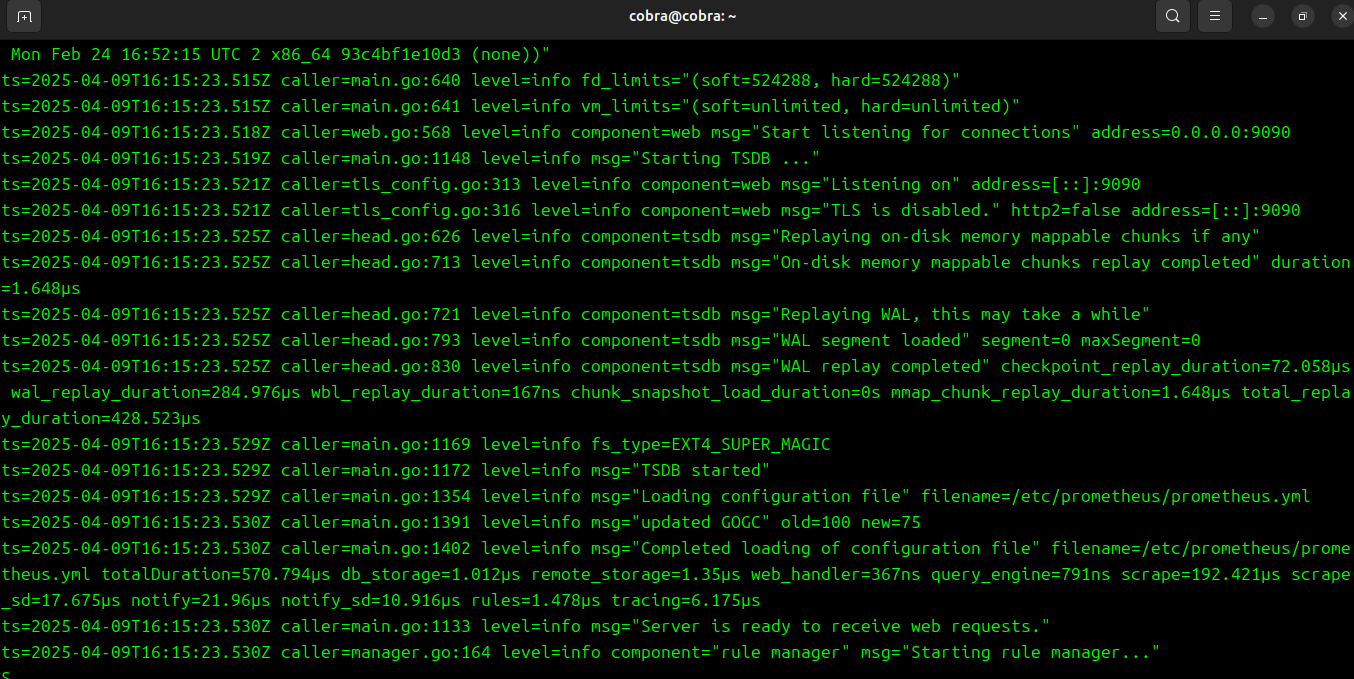
 To understand the fundamentals of Prometheus.

 To install and configure Prometheus for monitoring.

 To monitor software performance metrics using Prometheus.

 To visualize data using Prometheus's web UI and/or Grafana (optional)

**Step 1: Install Prometheus**



**Step 2: Install Node Exporter (for system metrics)**

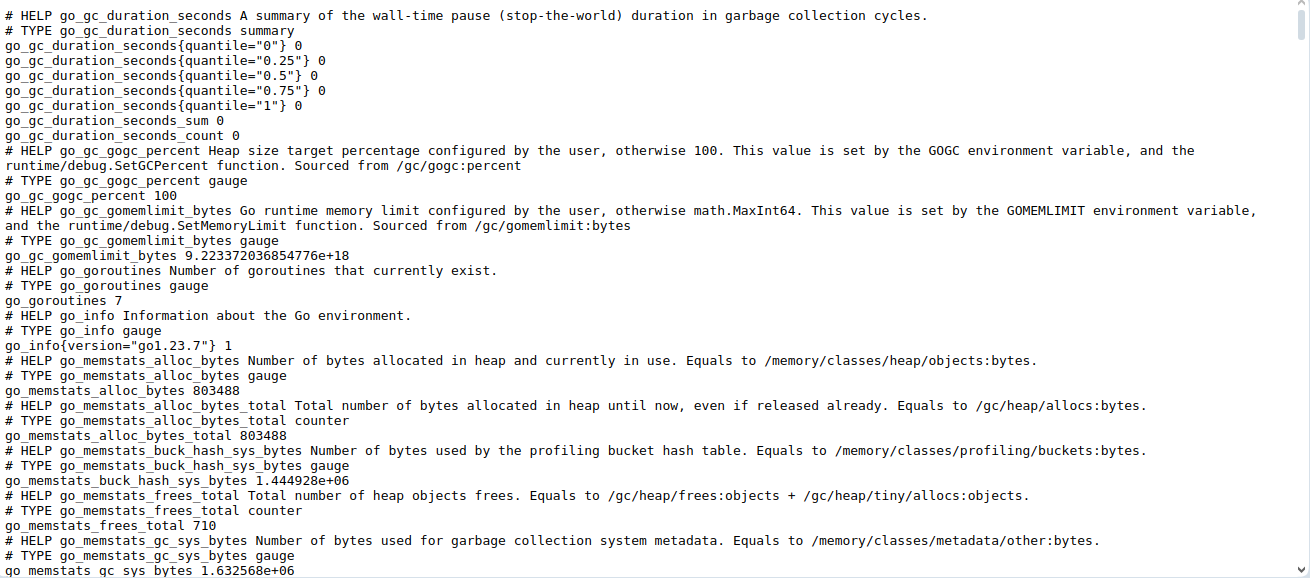
**Using Docker:**

docker run -d -p 9100:9100 prom/node-exporter

**Manually:**

1. Download from https://prometheus.io/download/#node\_exporter
2. Run:

./node\_exporter



**Step 3: Configure Prometheus to Scrape Node Exporter**

Edit prometheus.yml and add:

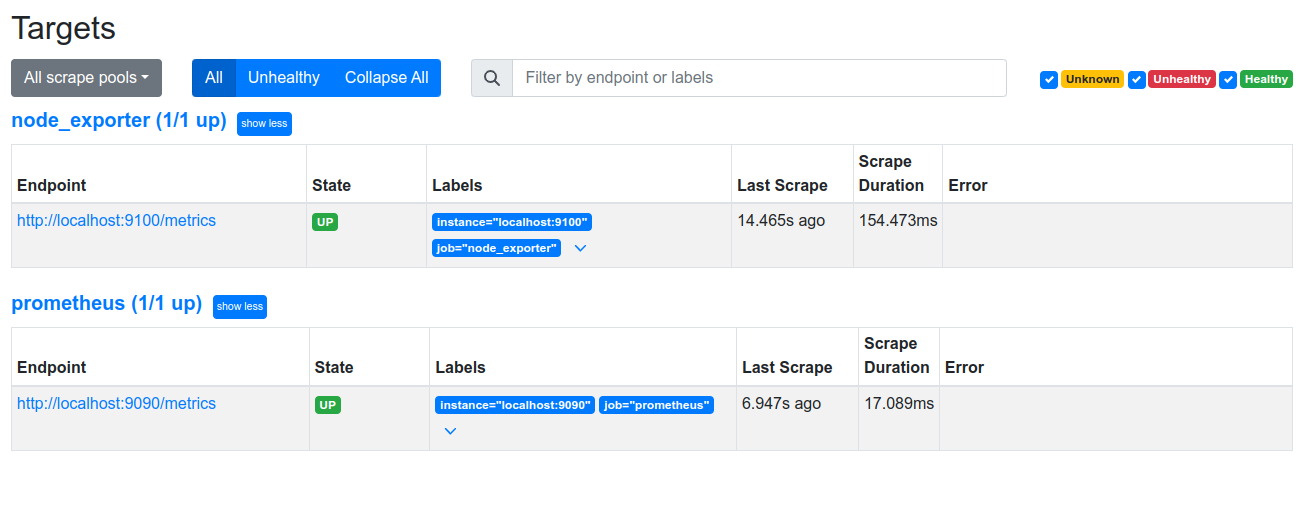
scrape\_configs:

- job\_name: 'node\_exporter'

static\_configs:

- targets: ['localhost:9100']

Restart Prometheus after saving the config.

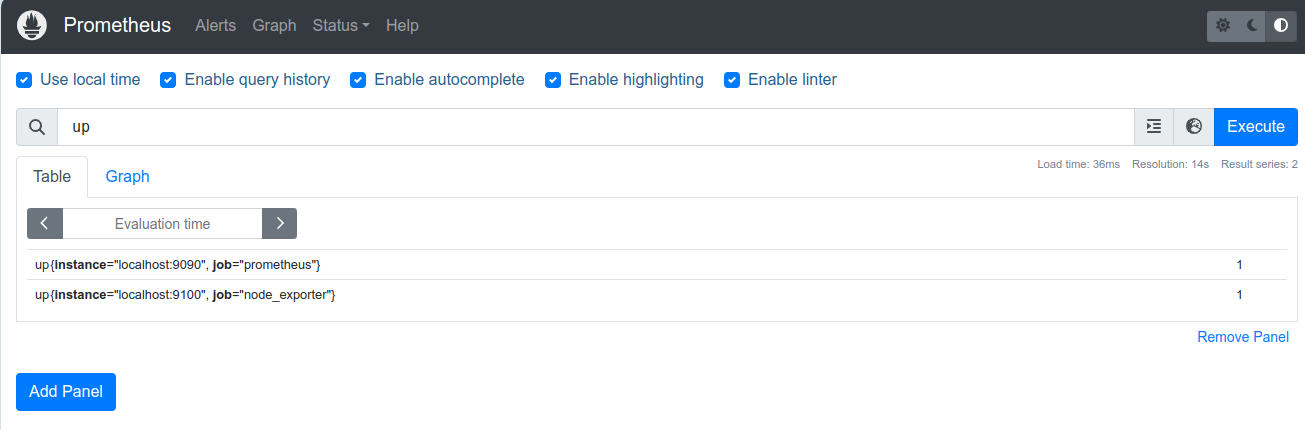


**Step 4: Access Prometheus Dashboard**

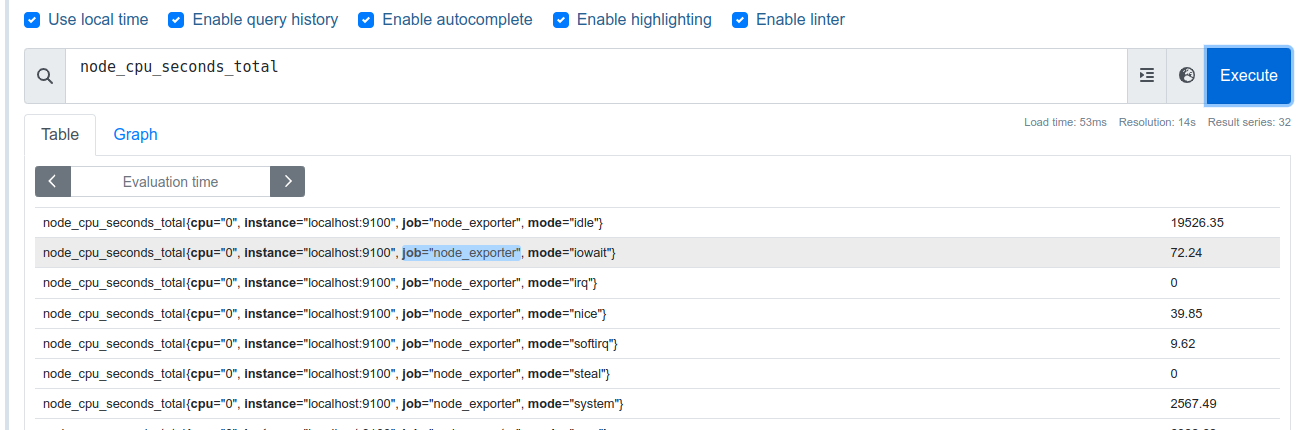
Open your browser and go to:  
 http://localhost:9090

Try some **PromQL** queries like:

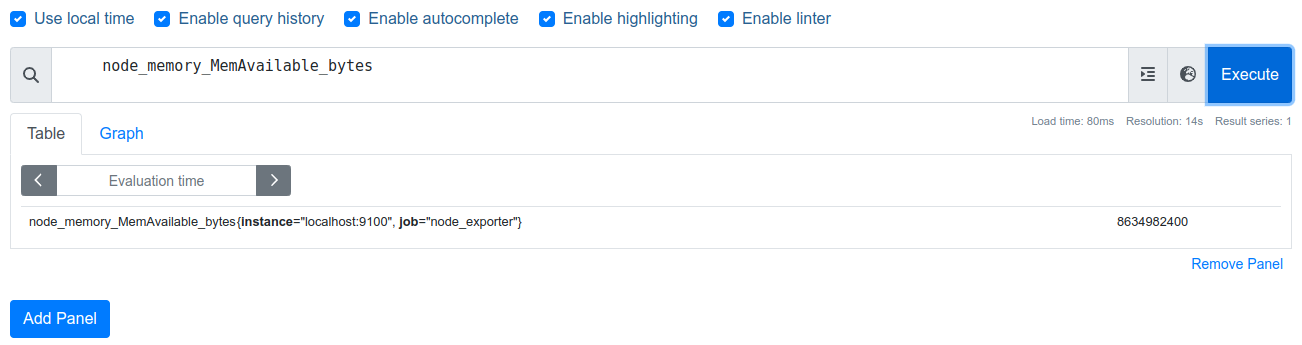
* up



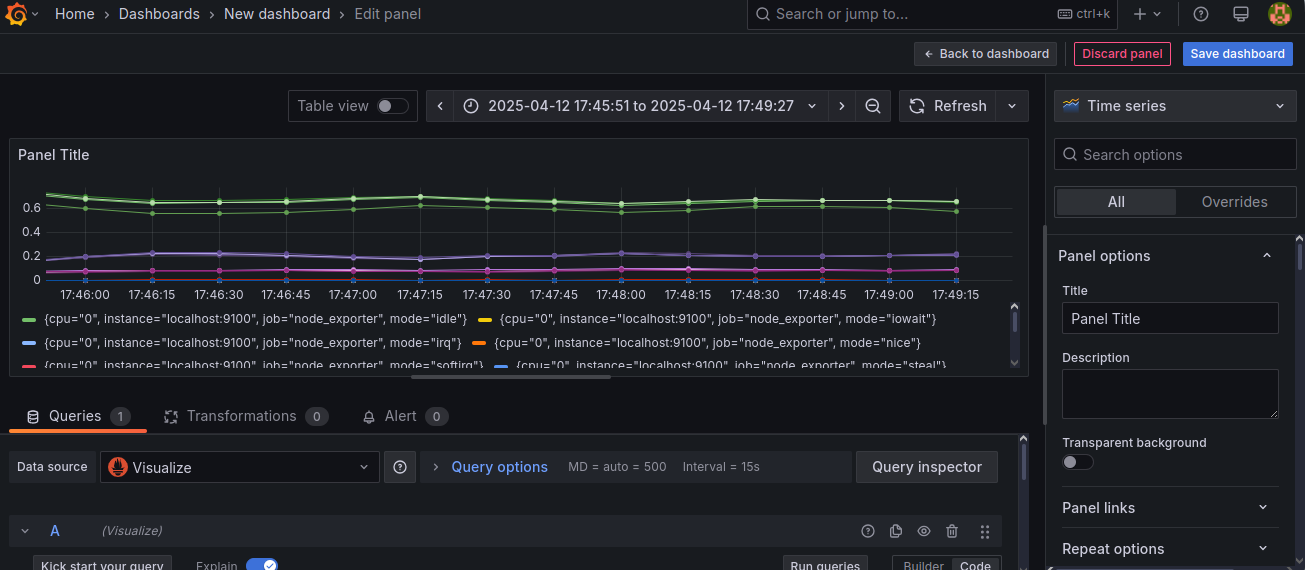
* node\_cpu\_seconds\_total



* node\_memory\_MemAvailable\_bytes



**Step 5 (Optional): Integrate with Grafana**



**5. Observations:**

| **Metric Name** | **Value** | **Time Stamp** | **Description** |
| --- | --- | --- | --- |
| node\_cpu\_seconds\_total | ...7465.13 | ...Now | CPU usage |
| node\_memory\_Active\_byte | ...9413283840 | .1681321234 | Active memory |
| up | 1 | 2025-04-12 17:49:15 | Target status |

**6. Result**:

Prometheus was successfully installed and configured to monitor system performance. System metrics were retrieved and visualized using Prometheus's built-in web UI.

**7. Viva Questions:**

# Viva Questions – Answered:

* Q: What is Prometheus used for?

A: Prometheus is used for monitoring and alerting. It collects, stores, and visualizes time-series metrics data from targets like servers or containers.

* Q: What is a time-series database?

A: A time-series database stores data points indexed by time. It is ideal for tracking changes over time, such as CPU load or memory usage.

* Q: Explain PromQL with an example.

A: PromQL is Prometheus’s query language. Example: `node\_cpu\_seconds\_total` – returns total CPU usage in seconds.

* Q: What is the role of exporters in Prometheus?

A: Exporters expose system/application metrics in a format Prometheus can scrape. E.g., Node Exporter exposes system-level metrics.

* Q: How do you configure a new target in Prometheus?

A: In the `prometheus.yml` file under `scrape\_configs`, you define a `job\_name` and list of `targets` with their host and port.