1. docker ps command displayed the following docker images and containers:

* **Prometheussandbox-app\_one:** This image that contains the code and dependencies for the first application in project setup. It contains a container called “app\_one” that provides a service accessible on port 8000.
* **Wrouesnel/postgres\_exporter:** This docker image contains the PostgreSQL Exporter application and runs a container called “postgres\_exporter” for collecting and exporting metrics from PostgressSQL databases. It exposing metrics on port 9187/tcp that can be scraped by Prometheus or other monitoring systems.
* **Prometheussandbox-app\_two**: This image that contains the code and dependencies for the second application in project setup. It contains a container called “app\_two” that provides a service accessible on port 8001.
* **quay.io/prometheus/node-exporter**:This Docker image contains the Node Exporter, a tool for collecting system-level metrics. It has a container called “**prometheus\_node\_exporter**” running on port 9100 and used to monitor the host system and provide metrics to Prometheus.
* **postgres:13.3:** This image provides a PostgreSQL database with version 13.3. It has container called “postgres” that serves as a PostgreSQL database server accessible on port 5432.
* **prometheussandbox-grafana:** This image runs Grafana for visualization and dashboard creation. Its container “grafana” provides a Grafana interface accessible on port 3000/tcp for visualizing monitoring data.
* **prom/pushgateway:** This image runs the **prometheus\_push\_gateway** container for collecting metrics from short-lived jobs. That container is accessible on port 9091 for accessing job-related metrics.
* **prometheussandbox-prometheus**: Contains prometheus, a monitoring and alerting toolkit for collecting and storing metrics. It Runs Prometheus container that is accessible on port 9090 and works as central monitoring and alerting tool, that collects, stores, and provides access to various metrics for monitoring the entire system.

1. The docker-compose.yml file defines a multi-container application setup using Docker Compose. It specifies several services and their configurations, allowing you to easily deploy and manage them together. The services include:

* app\_one and app\_two are services built from Dockerfiles in their respective directories, exposing ports 8000 and 8001. Additionally, app\_two has an environment variable DOCKER\_NETWORK set to "push\_gateway:9091".
* prometheus, grafana, node\_exporter, and push\_gateway services are built from Dockerfiles in their respective directories and expose ports 9090, 3000, 9100, and 9091 respectively.
* postgres and postgres\_exporter are services using pre-built images from PostgreSQL and "wrouesnel/postgres\_exporter", respectively. They expose ports 5432 and 9187 respectively. postgres\_exporter depends on postgres.

1. The containers are attached to a network named "push\_gateway". This is evident from the app\_two service, which specifies the environment variable DOCKER\_NETWORK: push\_gateway:9091. Other services do not explicitly define a network, so they would also be attached to the default network created by Docker Compose.