Creating an sample python app to generate logs.

Step1)

Created a python script that generates sample logs .

import logging

# Configure the logging module to print messages to the console

logging.basicConfig(level=logging.INFO)

t=1

# Log an informational message

while(t==1):

    logging.info("helloworld")

step2)

To deploy the above python script into pod we need to build an image of it.

To build the docker image

Write a dockerfile

# Use an official Python runtime as a parent image

FROM python:3.9

# Set the working directory to /app

WORKDIR /app

# Copy the current directory contents into the container at /app

COPY . /app

# Install any needed packages specified in requirements.txt

# (Replace requirements.txt with the actual requirements file if needed)

# RUN pip install --no-cache-dir -r requirements.txt

# Make port 80 available to the world outside this container

# (Update the port if your script runs on a different port)

EXPOSE 80

# Run sample.py when the container launches

CMD ["python", "./sample.py"]

To run it : docker build -t pythonscript .

Step 3)

After the image is created you need to push it in to the docker hub.

To push docker image.

docker tag pythonscript:latest syam01/pythonscript:latest

docker push syam01/pythonscript:latest

step4) deploy the Kubernetes pod

deployment.yaml file:

apiVersion: apps/v1

kind: Deployment

metadata:

  name: sampledeployment

spec:

  replicas: 1

  selector:

    matchLabels:

      app: sample

  template:

    metadata:

      labels:

        app: sample

    spec:

      containers:

      - name: pythonscript

        image: syam01/pythonscript:latest

        ports:

        - containerPort: 80

To deploy

kubectl apply -f deployment.yaml

the pod got deployed and it will generate the log we can visualize the logs in the Grafana .