Name – Nikhil Deshinge

Dockerfile for nginx 1.19

Write a Dockerfile to run nginx version 1.19 in a container. Choose a base image, considering security best practices, and aim for the image to pass a container image security test  
Your answer

FROM nginx:1.19-alpine

WORKDIR /etc/nginx

COPY nginx.conf /etc/nginx/nginx.conf

RUN addgroup -S nginx && adduser -S nginx -G nginx \

&& chown -R nginx:nginx /var/cache/nginx /var/run /var/log/nginx

EXPOSE 80

USER nginx

CMD ["nginx", "-g", "daemon off;"]

Kubernetes StatefulSet

Write a Kubernetes StatefulSet to deploy the nginx container from the previous question. Utilize persistent volume claims and define resource limits for optimal performance.  
Your answer

apiVersion: apps/v1

kind: StatefulSet

metadata:

name: nginx-statefulset

spec:

serviceName: nginx

replicas: 5

selector:

matchLabels:

app: nginx

template:

metadata:

labels:

app: nginx

spec:

securityContext:

fsGroup: 1000

containers:

- name: nginx

image: nginx:1.19-alpine

ports:

- containerPort: 80

volumeMounts:

- name: nginx-storage

mountPath: /usr/share/nginx/html

resources:

requests:

memory: "256Mi"

cpu: "250m"

limits:

memory: "512Mi"

cpu: "500m"

securityContext:

runAsUser: 1000

runAsGroup: 1000

allowPrivilegeEscalation: false

readinessProbe:

httpGet:

path: /

port: 80

initialDelaySeconds: 5

periodSeconds: 10

livenessProbe:

httpGet:

path: /

port: 80

initialDelaySeconds: 10

periodSeconds: 15

volumeClaimTemplates:

- metadata:

name: nginx-storage

spec:

accessModes: [ "ReadWriteOnce" ]

resources:

requests:

storage: 1Gi

---

apiVersion: v1

kind: Service

metadata:

name: nginx

spec:

clusterIP: None

selector:

app: nginx

ports:

- port: 80

targetPort: 80

Build a Deployment Pipeline

Set up a streamlined build and deployment pipeline for the nginx application using GitHub Actions or an equivalent CI/CD tool. Ensure the pipeline covers building the Docker image, running security checks, and deploying to a Kubernetes cluster.

Your answer

name: Build and Deploy Nginx to Kubernetes

on:

push:

branches:

- main

pull\_request:

branches:

- main

jobs:

build:

name: Build & Push Docker Image

runs-on: ubuntu-latest

steps:

- name: Checkout Code

uses: actions/checkout@v3

- name: Set up Docker Buildx

uses: docker/setup-buildx-action@v2

- name: Log in to Docker Hub

uses: docker/login-action@v2

with:

username: ${{ secrets.DOCKER\_USERNAME }}

password: ${{ secrets.DOCKER\_PASSWORD }}

- name: Build and Tag Docker Image

run: |

docker build -t ${{ secrets.DOCKER\_USERNAME }}/nginx:latest .

docker tag ${{ secrets.DOCKER\_USERNAME }}/nginx:latest ${{ secrets.DOCKER\_USERNAME }}/nginx:${{ github.sha }}

- name: Push Docker Image

run: |

docker push ${{ secrets.DOCKER\_USERNAME }}/nginx:latest

docker push ${{ secrets.DOCKER\_USERNAME }}/nginx:${{ github.sha }}

security-scan:

name: Scan Docker Image for Vulnerabilities

runs-on: ubuntu-latest

needs: build

steps:

- name: Install Trivy

run: |

curl -sfL https://raw.githubusercontent.com/aquasecurity/trivy/main/contrib/install.sh | sh

- name: Scan Docker Image

run: trivy image --exit-code 1 --severity HIGH,CRITICAL ${{ secrets.DOCKER\_USERNAME }}/nginx:latest

deploy:

name: Deploy to Kubernetes

runs-on: ubuntu-latest

needs: [build, security-scan]

if: github.ref == 'refs/heads/main'

steps:

- name: Checkout Code

uses: actions/checkout@v3

- name: Set up kubectl

uses: azure/setup-kubectl@v3

- name: Configure Kubernetes

run: |

echo "${{ secrets.K8S\_KUBECONFIG }}" | base64 --decode > kubeconfig.yaml

export KUBECONFIG=kubeconfig.yaml

- name: Deploy to Kubernetes

run: |

kubectl set image deployment/nginx nginx=${{ secrets.DOCKER\_USERNAME }}/nginx:latest --namespace=default

kubectl rollout status deployment/nginx --namespace=default

Text Manipulation Problem

Choose or create a text manipulation problem that involves using awk, sed, tr, and/or grep. Solve the problem, considering efficiency and readability.

Your answer

] 192.168.1.10 - - [06/Feb/2025:12:34:56 +0000] "GET /index.html HTTP/1.1" 200 512

10.0.0.5 - - [06/Feb/2025:12:35:02 +0000] "POST /login HTTP/1.1" 401 128

172.16.5.23 - - [06/Feb/2025:12:35:10 +0000] "GET /dashboard HTTP/1.1" 200 1024

awk '{print $1}' access.log | sort –u

awk '{print $1}' access.log | sort | uniq -c | sort –nr

grep 'HTTP/1.1" [45][0-9][0-9]' access.log

awk '{print $7}' access.log | sed 's/\?.\*//g' | sort | uniq

sed 's/HTTP\/1.1/HTTP\/2/g' access.log > access\_updated.log

Text Manipulation with an Object Orientated Programming Language

Solve the text manipulation problem from the previous question using any Object Orientated Programming language of your choice. Provide a clear and well-documented solution.

Your answer

import re

from collections import Counter

class LogAnalyzer:

def \_\_init\_\_(self, log\_file):

"""Initialize the LogAnalyzer with a log file path."""

self.log\_file = log\_file

self.logs = self.\_read\_logs()

def \_read\_logs(self):

"""Read the log file and return lines as a list."""

try:

with open(self.log\_file, 'r') as file:

return file.readlines()

except FileNotFoundError:

print(f"Error: {self.log\_file} not found.")

return []

def get\_unique\_ips(self):

"""Extract and return a set of unique IP addresses."""

ips = {line.split()[0] for line in self.logs}

return sorted(ips)

def count\_requests\_per\_ip(self):

"""Count the number of requests per IP address."""

ip\_counts = Counter(line.split()[0] for line in self.logs)

return dict(sorted(ip\_counts.items(), key=lambda x: x[1], reverse=True))

def get\_error\_requests(self):

"""Extract requests with HTTP 4xx or 5xx response codes."""

error\_requests = [line for line in self.logs if re.search(r'HTTP/1.1" [45]\d{2}', line)]

return error\_requests

def extract\_urls(self):

"""Extract and return unique requested URLs without query parameters."""

urls = {re.split(r'\?| ', line.split()[6])[0] for line in self.logs}

return sorted(urls)

def replace\_http\_version(self, old\_version="HTTP/1.1", new\_version="HTTP/2"):

"""Replace HTTP version in logs and return updated log content."""

updated\_logs = [line.replace(old\_version, new\_version) for line in self.logs]

return updated\_logs

def save\_updated\_logs(self, output\_file="access\_updated.log"):

"""Save updated logs with HTTP/2 to a new file."""

updated\_logs = self.replace\_http\_version()

with open(output\_file, 'w') as file:

file.writelines(updated\_logs)

print(f"Updated logs saved to {output\_file}")

# Eg:

if \_\_name\_\_ == "\_\_main\_\_":

log\_analyzer = LogAnalyzer("access.log")

print("Unique IPs:", log\_analyzer.get\_unique\_ips())

print("Request Counts per IP:", log\_analyzer.count\_requests\_per\_ip())

print("Error Requests:", log\_analyzer.get\_error\_requests())

print("Unique URLs:", log\_analyzer.extract\_urls())

log\_analyzer.save\_updated\_logs()

Sum of Even Fibonacci Numbers

Write a program in a Object Orientated Programming language of your choice to calculate the sum of the first 100 even-valued Fibonacci numbers. Consider efficiency and demonstrate good coding practices.

Your answer

def sum\_even\_fibonacci(n):

even\_fib1, even\_fib2 = 2, 8

sum\_even\_fibs = even\_fib1 + even\_fib2

for \_ in range(n - 2): # Already counted first two

next\_even\_fib = 4 \* even\_fib2 + even\_fib1

sum\_even\_fibs += next\_even\_fib

even\_fib1, even\_fib2 = even\_fib2, next\_even\_fib

return sum\_even\_fibs

result = sum\_even\_fibonacci(100)

print("Sum of the first 100 even Fibonacci numbers:", result)

Intersection of Sorted Arrays

Write a function in a Object Orientated Programming language of your choice that takes two sorted arrays of integers as input and returns an array containing numbers common to both arrays without duplicates.

Your answer

] def intersection\_of\_sorted\_arrays(arr1, arr2):

i, j = 0, 0

result = []

while i < len(arr1) and j < len(arr2):

if arr1[i] < arr2[j]:

i += 1

elif arr1[i] > arr2[j]:

j += 1

else:

if not result or result[-1] != arr1[i]:

result.append(arr1[i])

i += 1

j += 1

return result

# Eg

arr1 = [1, 2, 4, 5, 9]

arr2 = [2, 4, 6, 8]

result = intersection\_of\_sorted\_arrays(arr1, arr2)

print("Intersection of the arrays:", result)

Decimal Digit Transformation

Write a function in an Object Orientated Programming language of your choice that, when passed a decimal digit X, calculates and returns the value of X + XX + XXX + XXXX. For example, if X is 3, the function should return 3702 (3 + 33 + 333 + 3333). Ensure the function handles valid inputs and provides meaningful error messages for invalid inputs.

Your answer

def decimal\_digit\_transformation(X):

if not isinstance(X, int):

raise ValueError("Input must be an integer.")

if X < 0 or X > 9:

raise ValueError("Input must be a decimal digit (0-9).")

X\_str = str(X) # Convert X to string for concatenation

result = X + int(X\_str \* 2) + int(X\_str \* 3) + int(X\_str \* 4)

return result

# Example

try:

X = 3

result = decimal\_digit\_transformation(X)

print(f"The result of {X} + {X}{X} + {X}{X}{X} + {X}{X}{X}{X} is: {result}")

except ValueError as e:

print(e)