

(a) Describe the three primary cloud service models in cloud computing infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). Provide specific examples of how each model can be applied in the context of software development.

Infrastructure as a Service (IaaS) provides the fundamental building blocks like virtual servers and networks. Developers use this for complete control, such as provisioning raw compute power on Amazon EC2 to host a custom-built application and its database, managing the entire software stack themselves.

Platform as a Service (PaaS) offers a managed environment for building and deploying applications. Developers focus solely on writing code, while the platform, like Heroku or Google App Engine, automatically handles infrastructure, runtime, and scaling which dramatically increases development speed.

Software as a Service (SaaS) delivers finished applications over the internet. In development, teams use these as tools, such as GitHub for source code management or Slack for communication, without any concern for maintaining the underlying software or infrastructure.

(b) What is Docker? Describe a scenario where you would use containerization technologies such as Docker in software development. How does containerization contribute to the development and deployment process of software in this scenario?

Docker is a platform that uses containerization technology to package an application and all its dependencies into a standardized, lightweight, portable unit called a container. This container can run consistently and reliably in any environment that has Docker installed. I may use Docker when building a complex application in a development team, like a Python web app that uses a Redis cache and a PostgreSQL database. Containerization ensures absolute consistency from development to production, eliminates environment-specific bugs ("it works on my machine"), and simplifies deployment by using the same container everywhere. This streamlines collaboration and is a foundation for modern CI/CD pipelines

(c) Deploy n8n (n8n.io) with Docker and capture a screenshot of <http://127.0.0.1:5678>. Please explain the docker command in detail.

(Windows - powershell)

1、 docker pull n8nio/n8n:

docker pull:

Instructs Docker to download a specific image from a registry

Docker contacts Docker Hub and downloads the latest version of the n8nio/n8n image, which stores the image locally on my machine

```
PS C:\Windows\system32> docker pull n8nio/n8n
Using default tag: latest
latest: Pulling from n8nio/n8n
3d64802e5816: Pull complete
016c0e952111: Pull complete
23fe6350486d: Pull complete
87ab795ed18b: Pull complete
4f4fb700ef54: Pull complete
9b94a1e882c2: Pull complete
52719e552fdf: Pull complete
9824c27679d3: Pull complete
3efaf331e66e: Pull complete
6a088b2daae0: Pull complete
4307f681e63e: Pull complete
c4a8c5e8e683: Pull complete
Digest: sha256:c5fe3ff0b79f7831dc21f9c709bdb7eee4fff4453a28ce84c8e9fa5b9f562686
Status: Downloaded newer image for n8nio/n8n:latest
docker.io/n8nio/n8n:latest
```

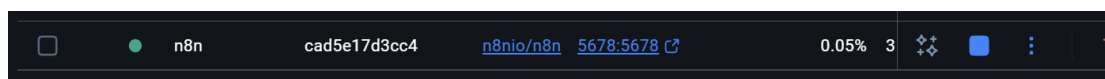
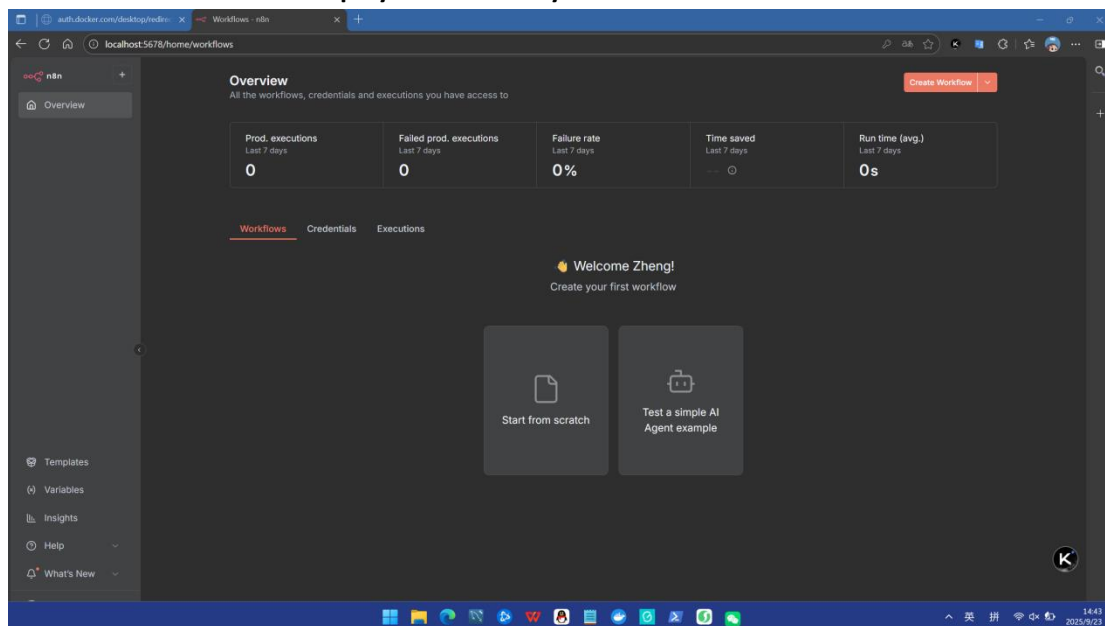
2、docker run -d --rm -p 5678:5678 --name n8n n8nio/n8n

This command creates and starts a new container from the downloaded image

```
PS C:\Windows\system32> docker run -d --rm -p 5678:5678 --name n8n n8nio/n8n
cad5e17d3cc4cceb0c8d20910214a85185b23f68da6a8cbd2af08af6ae294a
```

3、open web page : <http://localhost:5678>

Test if the container is deployed successfully or not



ps:Deploy log in docker