(a)

The Three primary cloud service models include:

1. Infrastructure as a Service (also known as IaaS):

Users can get virtualized computing resources like servers,storage and networking.Also,users have full control over the operating system and applications.

For example,a development team can use it to provision virtual machines on platforms like Amazon EC2.Then they can create custom development and testing environments,install operating systems and software stacks needed for their projects.

1. Platform as a Service(aka.PaaS):

This service provides a managed platform where developers can build,test and deploy applications without worrying about infrastructure management.The platform handles scalability,middleware and runtime.

For example,we can use Google APP Engine to deploy a web application quickly without managing the underlying servers.

1. Software as a Service(aka.SaaS):

It offers ready-to-use applications over the Internet.Users consume the service without managing hardware ,operating systems and application updates.

For example , we can use Github for version control and collaboration in software development.

(b)

Docker is a platform for developing,delivering and running applications using containerization technology.A container is lightweight,self-contained,executable software package that contains everything needed to run an application such as code,runtime,libraries and system tools.

Below is scenario, a software development team is building a complex web application including different services such as a Python backend,a Node.js frontend and a PostgreSQL database.

Containerization helps consistency, isolation and simplified deployment.Docker solves this problems by packing the application and its dependencies into containers,ensuring the application to run identically on the developer’s machine,test servers,and production enviroments.Each service of the application can be placed in its own independent container.This isolates them from each other and the host system,preventing conflicts and making dependency management easier.The team can deploy the entire application by running a simple docker command,which launches all the necessary containers.This makes the deployment process faster,more reliable and more repeatable without manually installing and configuring each service on the server.

(c)

Enter the command below in powershell:

docker run -it --rm --name n8n -p 5678:5678 -v C:\Users\Max\.n8n:/home/node/.n8n docker.n8n.io/n8nio/n8n

**Explanation:**  
This command runs a Docker container named n8n, maps host port 5678 to container port 5678, mounts the host folder C:\Users\Max\.n8n into the container at /home/node/.n8n, and will auto-remove the container when it stops. The image used is docker.n8n.io/n8nio/n8n (official n8n image).

**docker run**:Start and run a container.

**-it**:-i keeps STDIN open (interactive); -t allocates a pseudo-TTY.

**--rm**:Automatically remove the container when it exits

**--name n8n**:Name the container n8n so you can manage it by name

**-p 5678:5678**  Port mapping hostPort:containerPort. This maps host port 5678 to container port 5678, so you can visit http://127.0.0.1:5678 in your browser.

**-v C:\Users\Max\.n8n:/home/node/.n8n**

Volume mount. Mount host path C:\Users\Max\.n8n into the container at /home/node/.n8n. n8n stores workflows/config in /home/node/.n8n; mounting makes that data persist on your Windows host







