Protokoll zu Docker-Aufgaben

1. Finde auf Docker Hub das offizielle Node.js Image

Ein Bild, das Text, Screenshot, Schrift, Reihe enthält.

Automatisch generierte Beschreibung

1. Welche drei Hauptvarianten werden in der Dokumentation für das Image aufgeführt und was sind die Unterschiede zwischen ihnen?
   1. **node:<version>**: Dies ist das Standard-Node.js-Image, das auf Debian basiert.
   2. **node:<version>-alpine**: Dies ist die Alpine-basierte Variante des Node.js-Images.
   3. **node:<version>-slim**: Diese Variante basiert auf Debian und ist im Vergleich zur Standardvariante etwas schlanker.
2. Lade das aktuelle **node:alpine-Image** herunter

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

1. Wie groß ist es im Vergleich zum Standard-Node.js-Image?

Die alpine Version ist kleiner als die Standardversion, weil da eine schlankere Linux-Distribution

1. Benenne das heruntergeladene Image in small-node um

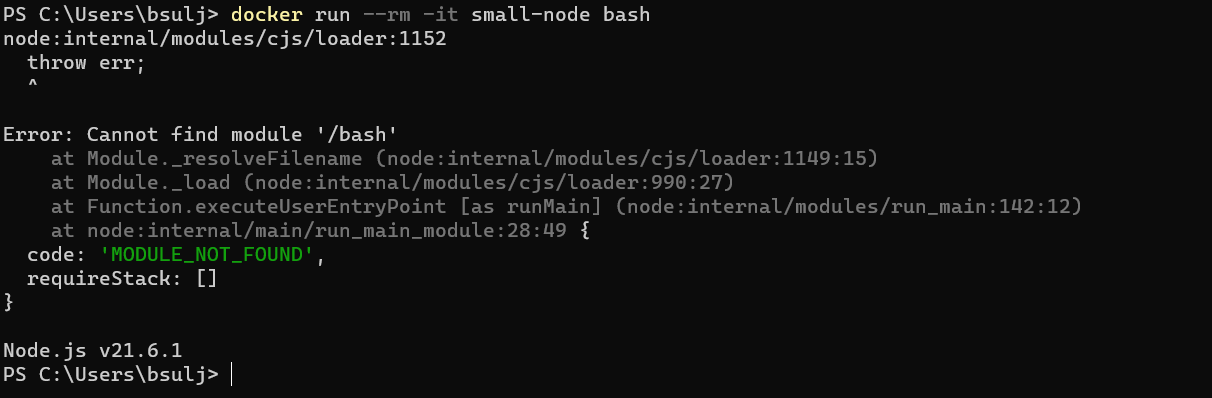


1. Überzeuge dich, dass bei dir jetzt ein Image namens small-node existiert

Ein Bild, das Text, Screenshot, Schrift, Zahl enthält.

Automatisch generierte Beschreibung

1. Erzeuge und starte einen Node.js-Container basierend auf dem small-node-Image, welcher automatisch wieder gelöscht werden soll. Starte dann per docker exec in diesem Container eine Bash



Bash kann nicht gestartet werden da, der Container gleich wieder gelöscht wird

1. Welche Fehlermeldung tritt dabei auf?

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

1. Lösche beide Node.js-Images, die auf Alpine basieren

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

Aufgabe 2:

1. Erzeuge einen neuen Node.js-Container namens test-container auf Grundlage des Standard-Node.js-Images und führe darin die bash aus

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

1. Dieser Container soll nach Beenden NICHT automatisch gelöscht werden

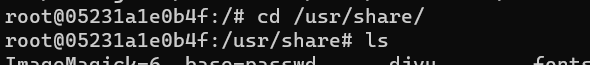
Damit das nicht passiert sollte der –rm Befehl nicht angeben werden

1. Steuere mit der bash im Container den Ordner etc an und finde die Version der zugrunde liegenden Debian-Distribution heraus

Ein Bild, das Text, Schrift, Screenshot enthält.

Automatisch generierte Beschreibung

1. Gehe dann zurück in den root-Ordner und von dort zu /usr/share



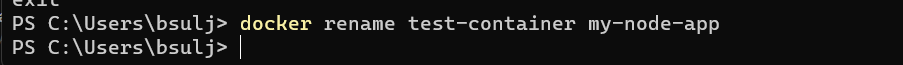
1. Findest du in dem Ordner Hinweise zu einer anderen Programmiersprache, die wir schon verwendet haben und die in diesem Container installiert ist?

Ein Bild, das Text, Screenshot, Schrift, Schwarz enthält.

Automatisch generierte Beschreibung

Aufgabe 3:

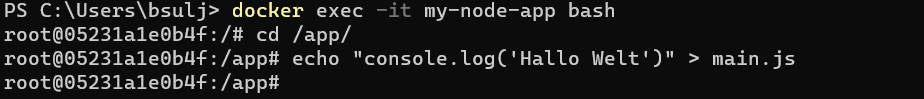
1. Benenne dann den Container test-container um in my-node-app



1. Erstelle dann dort dann den Ordner /app (Befehl: mkdir /app)



1. Wechsel anschließend in diesen Ordner und erstelle eine Datei "main.js" mit folgendem Inhalt console.log("Hallo Welt")



1. Führe anschließend dein Skript via node main.js aus

