

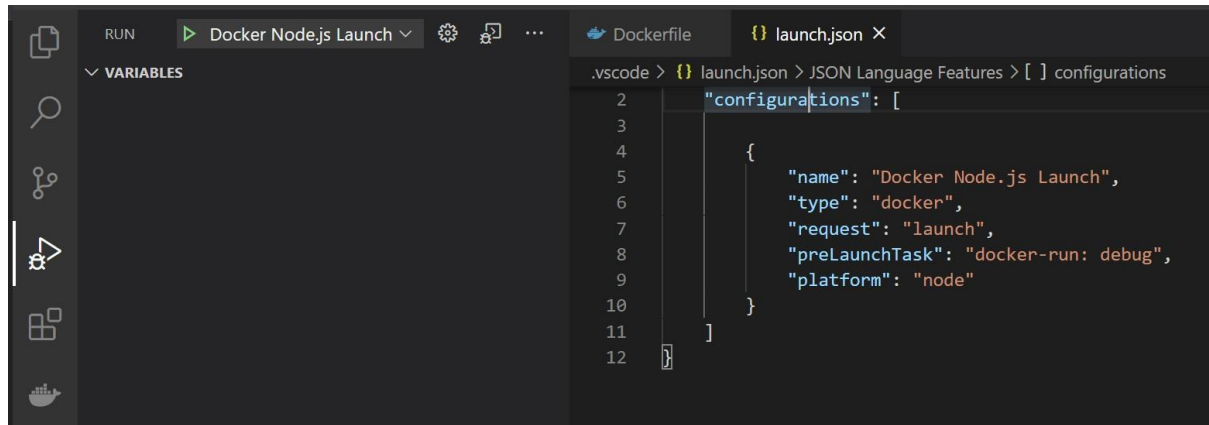
**Name:** KHUSHI POLSHETTIWAR

**Class:** TE IT Div: 2

**Roll No.:** I3236

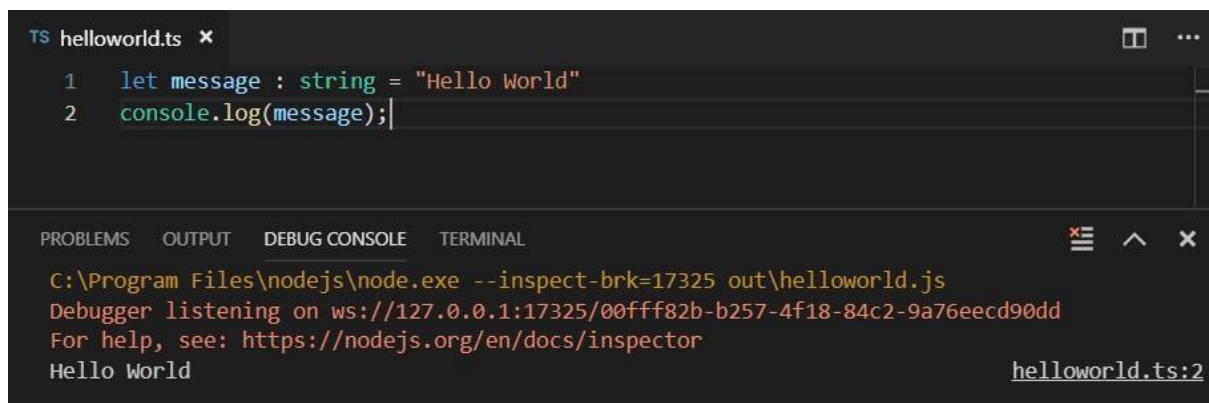
**Assignment:** 2B

**Problem Statement:** Create Docker Container Environment (NVIDIA Docker or any other)



The screenshot shows the Visual Studio Code interface with the 'launch.json' file open. The file contains a configuration for a Docker container named 'Docker Node.js Launch'. The configuration is as follows:

```
.vscode > {} launch.json > JSON Language Features > [ ] configurations
2      "configurations": [
3
4      {
5          "name": "Docker Node.js Launch",
6          "type": "docker",
7          "request": "launch",
8          "preLaunchTask": "docker-run: debug",
9          "platform": "node"
10     }
11 ]
12 ]
```

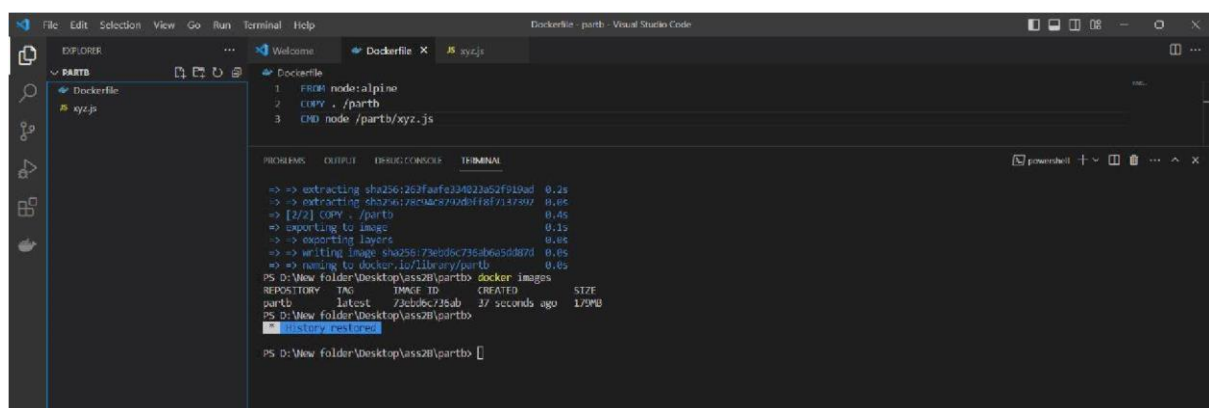


The screenshot shows the Visual Studio Code interface with a TypeScript file named 'helloworld.ts' and its execution output in the terminal. The code in 'helloworld.ts' is:

```
1 let message : string = "Hello World"
2 console.log(message);
```

The terminal output shows the command used to run the code:

```
C:\Program Files\nodejs\node.exe --inspect-brk=17325 out\helloworld.js
Debugger listening on ws://127.0.0.1:17325/00fff82b-b257-4f18-84c2-9a76eecd90dd
For help, see: https://nodejs.org/en/docs/inspector
Hello World
```



The screenshot shows the Visual Studio Code interface with a Dockerfile and its execution output in the terminal. The Dockerfile content is:

```
1 FROM node:alpine
2 COPY . /partb
3 CMD node /partb/xyz.js
```

The terminal output shows the command used to build the Docker image:

```
PS D:\New folder\Desktop\ass2b\partb> docker build -t partb .
=> extracting sha256:462fafe234022a52f919ad 0.2s
=> extracting sha256:78c3a0c9f4d0f16f7157592 0.0s
=> [2/2] COPY . /partb 0.4s
=> exporting to image 0.1s
=> exporting layers 0.0s
=> writing image sha256:72e08c73e8b6a5d87d 0.0s
=> naming to docker.io/library/partb 0.0s
PS D:\New folder\Desktop\ass2b\partb> docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
partb latest 72e08c73e8b6a5d87d 37 seconds ago 1.9MB
PS D:\New folder\Desktop\ass2b\partb> docker run --rm partb
PS D:\New folder\Desktop\ass2b\partb>
```

Containers

Images

Volumes

Dev Environments BETA

Learning Center

Extensions •

Add Extensions

<

partb:latest

73ebd4c736ab

CREATED

less than a minute ago

SIZE

179.19 MB

Recommended fixes

▼

Run

▼

Advanced image analysis is provided by Docker Scout (Early Access)

Upgrade to continue to get access to guided vulnerability remediation and additional software supply chain features: [Read about Docker Scout](#)

Image hierarchy

FROM alpine:3, 3.17, 3.17.3, latest

FROM node:20-alpine, 20-alpine3.17, 20.0-alpine, 20.0-alpine3...

ALL partb:latest

Layers (11)

0

ADD file:9a4f77dfaba7fd2aa781b6e4ef0e7486ad...

7.05 MB

1

CMD ["/bin/sh"]

0 B

2

ENV NODE\_VERSION=20.0.0

0 B

3

addgroup -g 1000 node && adduser -u 1000 -G no...

164.37 MB

4

ENV YARN\_VERSION=1.22.19

0 B

5

apk add --no-cache --virtual build-deps-yarn curl ...

7.77 MB

6

COPY file:4d192565a7220e135cab6c77fbc1c732...

388 B

7

ENTRYPOINT [ "docker-entrypoint.sh" ]

0 B

Vulnerabilities (0)

Packages (230)

Command

[Give feedback](#)

Package or CVE name

Fixable packages

Reset filters

Package

Vulnerabilities

No vulnerabilities were added with this layer

0-0 of 0