

Bind Mount, Dev Container & Ts Node Dev Cheat Sheet

Bind Mound Syntax For



For Git Bash

```
docker run -p 5000:5000 --name ts-container -w //app  
-v ts-docker-logs://app/logs -v "//$(pwd)"://app/ -v  
//app/node_modules --rm ts-docker
```



For Powershell

```
docker run -p 5000:5000 --name ts-container -w //app  
-v ts-docker-logs://app/logs -v "${PWD}"://app" -v //a  
pp/node_modules --rm ts-docker
```



For CMD

```
docker run -p 5000:5000 --name ts-container -w //app  
-v ts-docker-logs://app/logs -v "%cd%"://app/ -v //ap  
p/node_modules --rm ts-docker
```

ts-node-dev command for Docker Container



ts-node-dev --respawn --transpile-only --poll src/server.ts

1. First Open A .devcontainer folder in the root of the project
2. Inside the .devcontainer folder open a file named devcontainer.json and paste the following code
3. Change the json name, container names and file directories according to your project

```
{
  "name": "ts-container",
  "image": "node:20",
  "workspaceFolder": "/app",
  "mounts": [
    // Bind mount for your local project
    "source=/c/Projects/next-level/Docker/docker-with-typescript-backend,target=/app,type=bind",

    // Named volume for logs (similar to: -v ts-docker-log
    s://app/logs)
    "source=ts-docker-logs,target=/app/logs,type=volume",

    // Anonymous volume for node_modules (similar to: -v //ap
    p/node_modules)
    "target=/app/node_modules,type=volume"
  ],
  "runArgs": [
    "--name",
    "ts-container",
    "-p",
    "5000:5000",
  ]
}
```

```
    "--rm" // Automatically remove the container after exitin
g VS Code
],
"postCreateCommand": "npm install"
}
```